Separating Abstraction from Implementation in Communication Network Design

Ramón Cáceres

Computer Science Division
University of California
Berkeley, California 94720

ABSTRACT

Datagrams and virtual circuits are not disjoint conceptual models for data communication, but rather inhabitants of a wide design space containing many other viable networking solutions. Many design choices often closely associated with these two communication styles can be decoupled from the datagram and virtual circuit abstractions, and combined to form new and effective network implementations. This paper examines several key elements of network architecture. For each element, it shows how certain characteristics often thought to differentiate datagrams and virtual circuits are independent of these two concepts and form a multi-valued spectrum of design choices. This discussion is motivated by the current drive to design a new generation of high-speed wide-area networks, and the observation that this effort would benefit from a more systematic evaluation of existing and future network design alternatives.

Introduction

Datagrams and virtual circuits are widely used data communication paradigms. The two terms have been defined many ways. A useful analogy is drawn in [Tan81] and [Wec80], where datagram service is compared to the common postal service, and virtual circuit service is compared to ordinary telephone service. We define them in a similar fashion as follows: Like letters flowing through the postal system, datagrams are individually addressed and delivered items of information. No guarantees are made regarding the ordering or reliability of datagrams. Like voice conversations in the telephone system, virtual circuits must be explicitly set up and torn down before and after information is delivered. Virtual circuit setup consists of a user host requesting a connection to a specified destination address, and the network acknowledging that the connection has been established. In addition, virtual circuit information is delivered in the same order in which it was sent, although not necessarily free of errors.

In the remainder of this paper, this is our exclusive definition for datagrams and virtual circuits. It consists of the least common denominator of the many properties that are

This work was supported in part by AT&T Bell Laboratories and the International Computer Science Institute.

often implied when the terms datagram and virtual circuit are used. Any properties not explicitly included in the above definition, such as reliable data delivery and guaranteed performance, are not considered part of the datagram and virtual circuit abstractions. In this restricted sense, we also consider the concept of connectionless communication equivalent to the datagram abstraction, and connection-oriented communication equivalent to the virtual circuit abstraction.

We urge that this minimal definition of these terms be used more widely. Many times the datagram versus virtual circuit argument arises only because unnecessary functionality is bundled with the two concepts, not because of any fundamental quality of either concept. The networking literature often refers to datagrams and virtual circuits as general network architectures, when in fact it is referring to specific network implementations of the datagram and virtual circuit abstractions. Examples of such implementations are the DARPA Internet in the case of datagrams and TYMNET in the case of virtual circuits [QuH86]. It is not enough to refer to an architecture as forming a datagram or virtual circuit network – detailed characteristics of a network design, such as buffer and bandwidth allocation policies, should be separately described when presenting a network.

A common perception found in the literature is that designing a network entails a binary choice between two opposite approaches, datagrams and virtual circuits. This view is pictured in Figure 1. It is the thesis of this work that regarding network design as a set of often independent choices within multi-valued ranges of properties more closely approaches reality. Network architectures exhibit a large variety of characteristics, such as different routing policies and congestion control strategies, which the single terms datagram or virtual circuit can not and should not be used to encompass. Instead, we propose that the space of characteristics be considered a multi-valued *spectrum* of design choices, with existing implementations of datagrams and virtual circuit networks merely instances of possible designs near each extreme of the spectrum. This outlook is shown in Figure 2.

We believe that the network design outlook presented in this paper will help address the needs of future networks. Current work on high-speed wide-area networks is leading to new styles of communication. Under these new conditions, the network will share with the client hosts more responsibility for managing individual data streams than was previously true [Lei88]. These new solutions do not fall into the traditional datagram and virtual circuit categories, but somewhere between them in the network design space. Maintaining a clean separation between abstraction and implementation allows an objective comparison of the prevailing paradigms, as well as a systematic evaluation of new proposals.

In the following section of this paper, we point out the pitfalls inherent in imprecise terminology, particularly in the case of the words datagram and virtual circuit. We then define the scope of this paper and other terms to be used throughout. Subsequently, we describe several network properties that historically have been used to differentiate datagrams and virtual circuits and, within each property, identify the range over which solutions may operate. We consider the following properties: the degree of state information maintained by the network, the presence or absence of connection establishment and teardown procedures, buffer and bandwidth allocation strategies, routing policies,

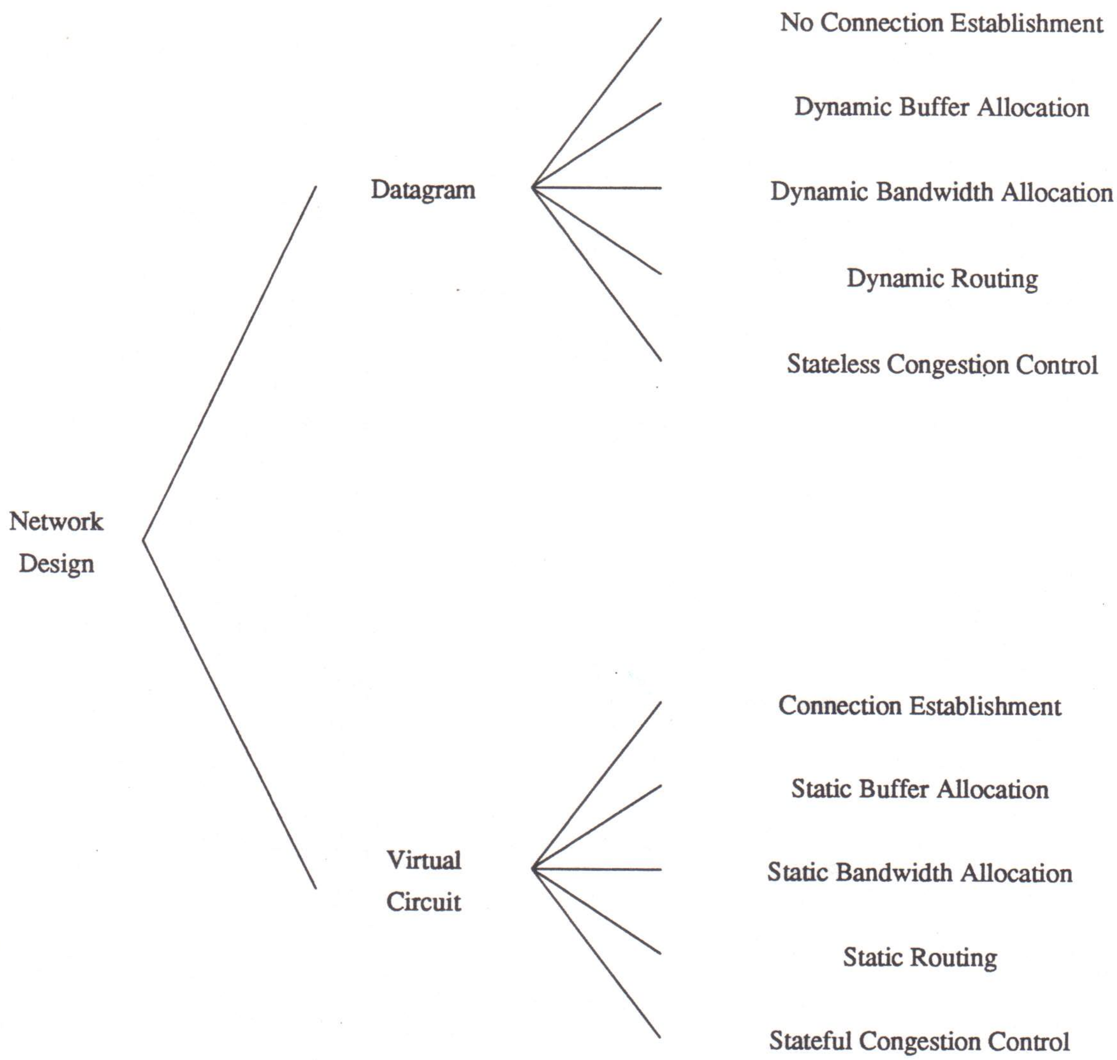

Figure 1 - A Common View of Network Design

and congestion control schemes. Finally, we point out why we feel the spectrum model will be useful in the design of the next generation of long haul networks.

1. The Terminology Problem

Many misconceptions regarding datagrams and virtual circuits are simply due to shortcomings in current terminology. Ambiguities regularly emerge in technical fields when established terms come to be used in different ways by different people. Such overloading has certainly occurred with the terms *datagram* and *virtual circuit.* For instance, establishing a connection for the purpose of sequencing data should not imply static buffer allocation, static bandwidth allocation, static routing, or a host of other functions within the network. However, many people seem to bundle this functionality with the concept of a virtual circuit. Misunderstandings inevitably arise when one person bases arguments on one definition of virtual circuit, and another person counters the arguments based on another definition.

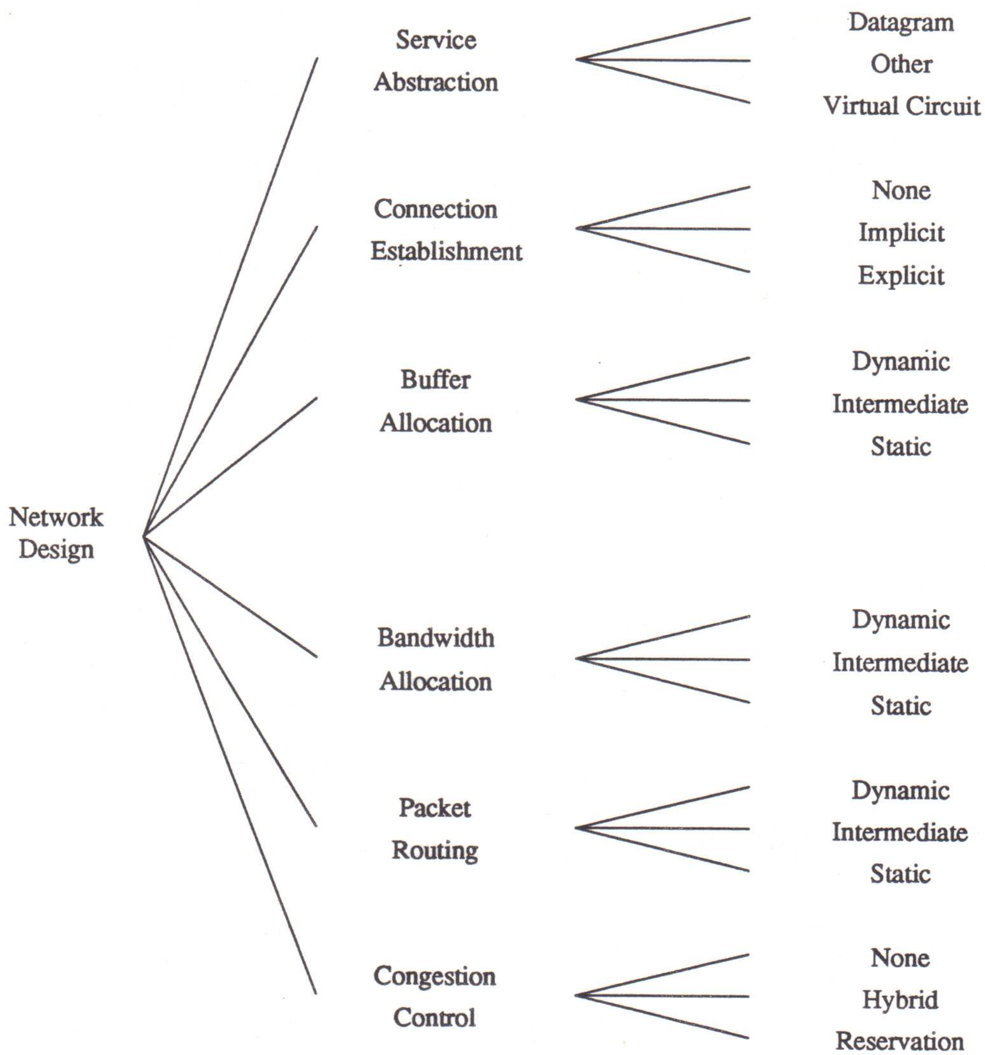

Figure 2 - The Spectrum of Network Designs

Unfortunately, once words start losing their meaning this way, the phenomenon is irreversible. The only way back to unambiguous communication seems to be the creation of new, more precise terminology. The terms datagram and virtual circuit have come to stand for such a wide range of properties that it may be impossible to use them further in a precise manner. Until better terminology comes into use, fully defining the terms to be used preceding a discussion of these issues is absolutely necessary.

However, the issue goes beyond the terminology problem. It is useful to examine the elements of network design and, whenever possible, cleanly separate each element from the superimposed datagram and virtual circuit abstractions. Disassociating network implementation characteristics, such as buffer allocation policies, from the network service abstractions, such as datagrams or virtual circuits, will go a long way towards an objective evaluation of alternative future network designs.

2. Scope of the Discussion

Throughout this paper, we concentrate our treatment of network architecture on the network and lower layers of the International Standards Organization (ISO) Open Systems Interconnect (OSI) model. The transport and higher-layer protocols are treated only as clients of the network. In other words, we are referring to the core communication subnetwork, not including user hosts.

This distinction is important because the interface the network offers its clients, that is, the interface the network layer offers the transport layer, is crucial to the effectiveness of the network. At the network layer interface, designers have chosen to offer both datagram and virtual circuit services. For example, the DARPA Internet presents a datagram interface to its users, while Datakit [Fra83] presents a virtual circuit interface. Granted, hosts may choose to superimpose other higher level abstractions above that provided by the network, but missing network functionality is in many cases impossible to correct by higher layers of software. Thus, there exist cases of a virtual circuit abstraction superimposed on a datagram service, such as when the Transmission Control Protocol is used over the Internet Protocol [NNN82]. There are also examples of a datagram abstraction being provided over a virtual circuit service, such as when the Internet Protocol is used over the X.25 protocol suite [LLL81] [CoK83]. However, no amount of software will allow certain real time performance guarantees to be offered to higher layers unless the underlying network makes the same guarantees.

In the remainder of this paper, we speak of a *datagram network* as one that presents a datagram abstraction to its client hosts, and of a *virtual circuit network* as one that presents a virtual circuit abstraction. Specifically, we are referring only to the network to transport layer interface. We imply nothing about implementation choices made within the network, or about the many possible abstractions that a client host may superimpose on the network's offered service though the use of suitable higher level protocols.

In addition, we are deliberately vague when speaking of the *endpoints* of data traffic. This is because different networks, and different resource allocation entities within those networks, deal with different kind of endpoints. Some consider a client host as a single endpoint, while others treat individual transport or higher-layer protocol participants as endpoints. We do not distinguish between them, and use the term *conversation* to denote any logically cohesive data stream between two traffic endpoints.

Concerning the terms *flow control* and *congestion control,* we adopt a common convention [JaR88] [DKS89] and regard them as separate concepts. Flow control insures that a particular data stream receiver is not overrun by its transmitter. As such, it is an end-to-end function usually assigned to the transport and higher layers. Congestion control insures that the network as a whole is not overrun by the combined effect of all its traffic sources. It can be treated in the transport layer, in the network layer, or in a combination of both. Following our decision to treat only network layer issues, we include congestion control, but not flow control, among the network characteristics discussed in the following section.

3. The Spectrum of Network Designs

3.1. Stateless versus Stateful Networks

Before beginning our breakdown of network characteristics, we note an underlying theme common to much of the discussion that follows. This theme concerns the question of stateless versus stateful networks.

At the network and lower layers of the OSI model, different networks maintain varying degrees of state information regarding the transport and higher-layer communication flowing through it. The type and quantity of state information can vary from zero information to detailed records of buffer allocation, bandwidth allocation, routing choices, and traffic history. For example, the ARPAnet and NSFnet backbones of the DARPA Internet save no state concerning Transmission Control Protocol or User Datagram Protocol [NNN82] traffic. In contrast, public data networks based on the X.25 family of protocols, such as TYMNET and Telenet [QuH86], keep explicit information regarding each individual user virtual circuit.

Although this range of possibilities does not convey enough detail to be useful in categorizing actual networks, it is is representative of what we mean by a multi-valued spectrum of network architectures. As other network characteristics are presented, we shall see that in many cases the available design choices involve the question of how much state information the network maintains.

3.2. Connection Establishment and Teardown

The most visible difference between datagrams and virtual circuits is the lack or presence of connection establishment and teardown procedures. The matter appears to be quite simple: datagram networks accept and deliver individual packets, while virtual circuit networks require the establishment of a connection before data can be transmitted.

Even here, however, a middle ground exists. In the place of explicit connection procedures, gateways and internal nodes in an otherwise connectionless network may use traffic history to implicitly make many connection-oriented decisions regarding data. For example, after some number of packets between a particular source and destination have been detected inside some time interval, switching nodes may consider a connection to be implicitly established. Similarly, for connection teardown, if no traffic is detected between that source and destination within some period of time, the connection is implicitly terminated. A network may decide to treat a packet stream in this fashion for congestion control, accounting, network management, and other purposes,

Such implicit connections, sometimes termed soft connections [Lei88], have already appeared in the literature. An example is the fair queueing congestion control scheme [DKS89], which calls for gateways and internal nodes in a datagram network to identify each conversation flowing through that node. This information allows the network to isolate the sources of congestion and feed back control information only to misbehaved hosts.

3.3. Packet Routing

Packet routing is another area where certain design choices have been strongly associated with either datagrams or virtual circuits. In datagram networks, packets are typically routed individually, that is, a routing decision is made for each packet at each intermediate node. In contrast, a virtual circuit network usually maintains per-connection routing information. The route is determined at circuit establishment time, and all packets for that circuit are sent along the same route.

The situation again appears rather clear cut, but a range of choices does exist. For instance, the amount of per-connection routing information kept by the network can vary from zero in a fully dynamic routing scheme, to caches of recently used source-destination routes, to explicit per-connection routing data in a static routing scheme. Source routing schemes add to this flexibility by including a complete hop-by-hop path with each item of information sent. The routing decision can thus be made independently for each packet by the client hosts, but once a packet is in transit there is no simple way to alter its route. Many of these routing schemes can be used in both connectionless and connection-oriented networks.

An example of an intermediate solution is the network design proposal made by the Xpress Transfer Protocol definition [PPP88]. This network maintains caches of routing information indexed by global route tags, and uses them to reduce routing decision overhead for all but the first packet flowing between each source-destination pair. Examples of source routing networks are Blazenet [HaC88] and the Purdue FLOWS proposal [CoY88].

3.4. Buffer Allocation

Datagram networks typically have not differentiated between packet streams for the purposes of buffer allocation. They traditionally have allocated buffers in gateways and internal nodes from a single buffer pool, usually following a first-come first-serve policy, with no regard to which packet stream has made the request for memory. In contrast, some virtual circuit networks have made use of explicit virtual circuit identifiers to statically allocate a certain number of buffers to each connection. In reality, there is nothing to force either a datagram or a virtual circuit network to operate with the respective policies above. A datagram network may introduce more discriminating buffer allocation policies than a fully dynamic one, and similarly a virtual circuit network may choose to disregard virtual circuit information for buffer allocation.

Recent work aimed at fair allocation of resources in datagram networks calls for an intermediate position in this matter. An example is the fair queueing congestion control scheme [DKS89]. Using information about the resource usage by active conversations, these strategies guarantee a fair share of the available buffer space to all network clients using neither fully static or fully dynamic allocation policies.

3.5. Bandwidth Allocation

Datagram networks have traditionally not performed any explicit bandwidth allocation. In contrast, physical circuit switching networks reserve bandwidth for each conversation. Analysis shows that using the full transmission line bandwidth on demand is superior to statically dividing this bandwidth into sub-channels [HaC88]. However, this result does not preclude the network from attempting to allocate the available bandwidth by less direct means, whether it be for fairness, performance guarantees, or other purposes. This goal can be pursued by both datagram and virtual circuit networks.

As with buffer allocation, there exist intermediate designs that fall between the first-in first-out packet queueing of traditional datagram networks and the time-division or frequency-division multiplexing of circuit-switched networks. For example, Datakit [Fra83] statistically multiplexes the available line bandwidth through a mixture of queueing policies at the output links of switching nodes, without physically dividing the available line bandwidth into channels. Ideally, a bit-by-bit round robin queueing policy would guarantee fair bandwidth sharing between conversations. Fair queueing [DKS89] approximates bit-by-bit round robin with more practical measures, while parametrized message channels [And88] [Fer88] use deadline scheduling of packet transmissions in order to guarantee real time data delivery. The former scheme guarantees a certain percentage of the available bandwidth, while the latter guarantees an absolute amount of bandwidth.

3.6. Congestion Control

The issue of congestion control has been gaining importance in recent times as major networks reach saturation and run the risk of congestion collapse. The problem has been treated in the context of certain types of virtual circuit networks, where it lends itself to analytical modeling [Mor89]. Analytical and simulation modeling have also been used to evaluate schemes in datagram networks [Jac88] [JaR88] [DKS89]. Traditionally, the problem has been treated differently for datagram networks than for virtual circuit networks [Pou81]. The reason commonly given for this separation is that congestion control is easier in virtual circuit networks than in datagram networks, since virtual circuit networks are assumed to statically reserve resources during connection establishment. They are expected to deny service to new connection requests in order to avoid congestion.

We have already seen, however, that the situation is not so clearly defined. On the one hand, the completely stateless networks that are the logical extension of a pure datagram approach may prove incapable of effectively avoiding or controlling congestion, especially when the fairness issue is considered. On the other hand, virtual circuit networks need not statically allocate any particular resource. Performing congestion control through the simple denial of service to new connections is not an effective solution under these conditions, since congestion can be brought on by even a small number of misbehaved virtual circuits.

Solutions falling outside the traditional datagram and virtual circuit categories can be found in the congestion control literature, for example, stream tables and packet trails [Pou81]. Such hybrid schemes aim to allow a datagram network to identify natural packet streams flowing between source and destination endpoints in the network. Differentiating between conversations allows them to throttle only those data streams most responsible for the congestion, without affecting well-behaved sources. This element of fairness is an important component of any congestion control scheme, and has been addressed more directly by work on a binary feedback scheme [JaR88] and fair queueing [DKS89]. Both these approaches can also be considered hybrids of the traditional datagram and virtual circuit approaches.

3.7. Some Network Design Classifications

Figure 2 presents the taxonomy of network designs that emerges from the previous discussion. The multiple-valued spectrums of choices available to the network designer is clear from this figure. Existing network implementations, as well as current design proposals, already make use of some of the flexibility inherent in this design space. However, much of this flexibility is hidden under the labels datagram and virtual circuit.

Figure 3 shows classifications of some sample network designs using this taxonomy. The designs examined are the DARPA Internet [NNN82], AT&T Bell Laboratories' Datakit network [Fra83], Stanford University's Blazenet proposal [HaC88], the suggestions for network design made by Protocol Engines Incorporated's Xpress Transfer Protocol specification (XTP) [PPP88], the Massachusetts Institute of Technology's flows-based proposal (MIT flows) [Zha87], Purdue University's FLOWS proposal (Purdue FLOWS) [CoY88], and finally the University of California at Berkeley's parametrized message channel proposal (Berkeley channels) [And88] [Fer88].

The range of choices made by these network designs is evident in this table, and again suggests that the single terms datagram and virtual circuit are not sufficient to describe anything more than the network layer abstraction a network presents its clients. In fact, the table restricts itself to somewhat comprehensive designs for existing or proposed network architectures. Other, more restricted proposals demonstrate further flexibility, such as the implicit connection establishment procedures called for by several of the congestion control schemes outlined earlier.

Network Design	Service Abstraction	Connection Establishment	Buffer Allocation	Bandwidth Allocation	Packet Routing	Congestion Control
Internet	Datagram	None	Dynamic	Dynamic	Dynamic	Source Quench
Blazenet	Datagram	None	None	Dynamic	Source	Back Pressure
Datakit	Virtual Circuit	Explicit	Static	Dynamic	Static	Resource Reservation
XTP	Virtual Circuit	Explicit	Dynamic	Dynamic	Cached	(Not Available)
MIT flows	MIT flows	Explicit	Intermediate	Intermediate	(Not Available)	Resource Reservation
Purdue FLOWS	Purdue FLOWS	Explicit	Intermediate	Intermediate	Source	Resource Reservation
Berkeley channels	Berkeley channels	Explicit	Intermediate	Intermediate	Static	Resource Reservation

Figure 3 - Some Network Design Classifications

4. Relevance to Future Networks

Existing long haul networks are rapidly becoming saturated due to increasing traffic loads. This trend, together with the emergence of viable optical fiber communication technology, motivates the development of a new generation of high-speed wide-area networks. The improvement by several orders of magnitude in the speed of transmission lines introduces many challenges into the design of such a network, since conventional store-and-forward packet switching nodes are not expected to match this increase in speed. This mismatch leads to a performance bottleneck at the switching nodes and not at the lines, the reverse of the current situation. Even if new technologies like fully optical packet switches [HaC88] eliminate some of these problems, many factors combine to make next-generation network design a non-trivial problem. These factors include the sheer volume of data, the increasing number of different types of traffic, and the complexity of the interconnect.

Thus, new solutions will need to be formulated for many of the central problems of network design, such as congestion control, routing, and network management. It seems clear that solutions falling outside the traditional datagram and circuit designs will be better suited to the new conditions and will find application in future networks.

An important area where future networks will differ from current ones is in their support for multiple service classes, specifically in the area of performance guarantees. The number of different types of network traffic is increasing, from the traditional set of interactive and bulk transfer traffic, to a much richer set that also includes real time voice, video, and high fidelity sound applications. Each type of traffic exhibits different properties and demands different performance characteristics from the network, such as specific values of throughput, delay, and other parameters.

Offering different degrees of service, especially of the real time variety, will require new facilities inside the network. Recent work in this area points to many features of traditional virtual circuit networks as being necessary for effective operation under the new conditions. For example, the Purdue FLOWS scheme uses global flow identifiers, a form of virtual circuit number, to keep track of the performance parameters associated with each flow. With Berkeley channels, explicit connection establishment procedures call for systematic resource allocation to insure that performance guarantees are not violated. A fixed route is chosen for each channel during establishment and used for the lifetime of the conversation.

Nevertheless, the functionality of traditional virtual circuit networks is not sufficient. The real-time parameter negotiations that go on with Berkeley channels, Purdue FLOWS, and the MIT flows proposals are an intrinsic part of the network to transport layer interface, and have no parallel in the traditional datagram or virtual circuit worlds. The network service models that emerge from these proposals are richer than either the datagram or virtual circuit abstraction, and should be considered new abstractions.

Whatever the names of these new communication models, flows or channels, it is important to separate the abstract interface presented by the network layer from the implementation details hidden within the network. The service abstraction should contain only the minimal intrinsic properties visible to the network client, leaving irrelevant details hidden withing the network. In this light, the approach to network classification presented in this paper remains a valid and useful tool for evaluating future network designs and comparing them to existing solutions.

Conclusion

This paper argues for a clean separation between the service abstraction a network presents its clients and the underlying implementation details. In particular, it presents network design as the process of choosing design points from largely independent, multi-valued ranges of properties. The paper contrasts this spectrum model with making a binary choice between the often ill-defined concepts of datagrams and virtual circuits. Reevaluating the traditional datagram and virtual circuit approaches in this way exposes the full range of design possibilities to the network architect, without the inhibiting influence of unnecessarily bundled functionality.

Looking towards the future, the coming of high-speed wide-area networks will require new solutions to many of the central problems of network design. We believe the search for such solutions would benefit from the outlook presented in this paper. Namely, putting aside taboos regarding the datagram and virtual circuit styles of communication, choosing the most appropriate design points from each spectrum, and creating richer service abstractions if necessary, will yield solutions that are better suited to the new conditions than the traditional datagram and virtual circuit approaches.

Acknowledgments

I would like to thank Domenico Ferrari, Srinivasan Keshav, and the other members of the Subnet group at Berkeley for many stimulating discussions and their comments on earlier drafts of this paper. Mark Sullivan was also very helpful in improving this work.

References

[And88] D. P. Anderson, A Software Architecture for Network Communication, *Proceedings of the 8th International Conference on Distributed Computer Systems*, June 1988.

[CoK83] D. Comer and J. Korb, CSNET Protocol Software: The IP-to-X.25 Interface, *Proceedings of the ACM SIGCOMM 1983 Symposium*, March 1983.

[CoY88] D. Comer and R. Yavatkar, FLOWS: Performance Guarantees in Best Effort Delivery Systems, *Purdue University Computer Science Department Technical Report 791*, West Lafayette, Indiana, July 1988.

[DKS89] A. Demers, S. Keshav and S. Shenker, Analysis and Simulation of a Fair Queueing Algorithm, *submitted to the ACM SIGCOMM 1989 Symposium*, April 1989.

[Fer88] D. Ferrari, Guaranteeing Performance for Real-Time Communication in Wide-Area Networks, *U. C. Berkeley Computer Science Division Technical Report 88/485*, Berkeley, California, December 1988.

[Fra83] A. G. Fraser, Towards a Universal Data Transport System, *IEEE Journal on Selected Areas in Communication SAC-1*, 5 (November 1983).

[HaC88] Z. Haas and D. Cheriton, A Case for Packet Switching in High-Performance Wide-Area Networks, *Proceedings of the ACM SIGCOMM 1988*

Symposium, 1988.

[Jac88] V. Jacobson, Congestion Avoidance and Control, *Proceedings of the ACM SIGCOMM 1988 Symposium*, 1988.

[JaR88] R. Jain and K. K. Ramakrishnan, Congestion Avoidance in Computer Networks with a Connectionless Network Layer: Concepts, Goals, and Methodology, *Proceedings of the 1988 Computer Communication Conference* , 1988.

[Lei88] B. Leiner, Critical Issues in High Bandwidth Networking, *Network Information Center RFC 1077*, Menlo Park, California, November 1988.

[LLL81] *The X.25 Protocol and Seven Other Key CCITT Recommendations*, Lifetime Learning Publications, Belmont, California, 1981.

[Mor89] S. P. Morgan, Queueing Disciplines and Passive Congestion Control in Byte-Stream Networks, *to appear in the Proceedings of Infocom 1989*, 1989.

[NNN82] *Internet Protocol Transition Workbook*, Network Information Center, SRI International, Menlo Park, California, March 1982.

[Pou81] L. Pouzin, Methods, Tools, and Observations on Flow Control in Packet-Switched Data Networks, *IEEE Transactions on Communications COM-29*, 4 (April 1981).

[PPP88] *XTP Protocol Definition, Revision 3.3*, Protocol Engines, Inc., Santa Barbara, California, December 1988.

[QuH86] J. Quarterman and J. Hoskins, Notable Computer Networks, *Communications of the ACM 29*, 10 (October 1986).

[Tan81] A. Tanenbaum, *Computer Networks*, Prentice-Hall, Englewood Cliffs, New Jersey, 1981.

[Wec80] S. Wecker, Computer Network Architectures, *IEEE Computer 23*, 12 (December 1980).

[Zha87] L. Zhang, Designing a New Architecture for Packet Switching Communication Networks, *IEEE Communications Magazine 25*, 9 (September 1987).