Trends in High-Speed Data Networks[1]

Manoel A. Rodrigues
OTL Odebrecht Automação e Telecomunicações Ltda.
Rua do Rócio, 351
04552 São Paulo, SP

In this paper we present a discussion of the trends in high-speed data networking. We describe some of the factors that have been instrumental in the development of today's needs for high-speed interconnection. Based on the experience with the Internet, we find that a successful (high-speed) data networking solution should be offered as a set of services that are valuable to the user, instead of relegating the burden of creating value out of the technology to the user. In light of that, we describe some of the services that have been developed to address the specific needs of LAN applications, like Frame Relay (FR) and Switched Multi-megabit Data Service (SMDS), as well as some of the enabling technologies, like Distributed Queue Dual Bus (DQDB) and Asynchronous Transfer Mode (ATM). We also discuss the B-ISDN efforts towards providing a general purpose platform for supporting (current and future) applications having high-speed interconnection needs. We finalize the paper with a discussion of technology challenges for supporting applications with widely different high-speed requirements.

Keywords: ATM,BISDN, Congestion control, Frame relay, IEEE 802.6, SMDS.

1 Introduction

There are two major factors driving the trends in high-speed data networking today: applications and technologies. On one hand, the existing LAN applications have created a need for high-speed interconnection. On the other hand, new technology in communications, transmission and VLSI has presented us with viable alternatives for creating broader solutions for high-speed interconnection, e.g., extending the connectivity of LAN applications to wider geographic areas. Furthermore, these new technologies may be capable of supporting a wide range of new applications (e.g., multi-media), some of which may find significant demand in the future.

We now face a series of questions that need to be answered in anticipation of future wide-area high-speed interconnection needs. For instance, should we tailor the solution towards applications that already exist (e.g., LAN applications), or should we steer the solution towards creating new needs? Can we actually address all potential needs at once? The questions range from technology choices, through service needs, to cost effectiveness of architectures. To make educated decisions, it is important to

[1] Artigo convidado; palestra sobre este tema foi apresentada pelo autor no 9º Simpósio Brasileiro de Telecomunicações, São Paulo, 1991.

understand both how information systems have evolved and the tradeoffs among upcoming technologies.

In this paper we describe some of the factors that were instrumental in the development of current needs for high-speed interconnection. This includes: (1) the evolution of the computing model; (2) the merits of a successful network interconnect service (i.e., Internet); (3) two services that have been developed to address the specific needs of LAN applications (i.e., Frame Relay (FR) and Switched Multi-megabit Data Service (SMDS)); and (4) two enabling technologies (i.e., Distributed Queue Dual Bus (DQDB) and Asynchronous Transfer Mode (ATM)). We also discuss the B-ISDN efforts to provide a general purpose platform for both current and future applications having high-speed interconnection needs. Finally, we discuss technical challenges in designing a general purpose high-speed platform that supports applications having significantly different requirements and traffic characteristics. The issues that are challenging are those that make a high-speed system behave fundamentally differently from a similar lower speed system.

2 Computing Model Evolution

The traditional computing model is that of centralized resources. In this model, users access a mainframe computer through dumb terminals. The mainframe computer performs all the processing and controls all the resources. This model was successful for a long time. Its success can be attributed to its cost effectiveness in an environment where computing resources are expensive. In fact, until recently, computing cost was the overwhelming factor in the cost effectiveness of a computing solution.

The environment has changed significantly over the past twenty years with a dramatic decrease in costs of computing hardware (e.g., memory, magnetic disks and processing power) and with an increase in performance. The development of PCs and workstations followed immediately from this new environment. However, cost and performance alone were not sufficient to change the model of computing. Other major factors in the evolution of the computing model were the creation of open systems and the development of simple operating systems with friendly user interfaces. However, it was not until the widespread development of software for open systems and the development of operating systems with effective networking capabilities that a decentralized model of computing became preferred to a strictly centralized one.

Although the centralized resource model may still be, in concept, the most resource-efficient model, the decrease in costs and the availability of inexpensive PC and workstation software has changed the cost benefit curve significantly towards a more distributed system. The new computing model attempts to reap the benefits of both centralized and distributed systems: the client-server model. Tasks that consist of presentation (e.g., word processing and graphics) or simple data manipulation (e.g., spreadsheets) are more efficiently programmed and processed at each client machine.

Tasks that require consistency among large amounts of data (e.g., database services) or that require system administration efforts (e.g., host name services) are more efficiently performed by the network. In the new model, even the tasks that are perceived by the user as being centralized may be distributed among several different servers across the network.

While in the previous model the resources are directly connected to the mainframe, in the client-server model the resources are accessible through the network. An analogy can be made between the two models, where the network now is an extension of the computer bus. Thus, with the high-speed requirements of a computer bus, high-speed networking has become a significant component in the efficient operation of the new computing model.

From the point of view of how technology fits in the real world, the evolution of the computing and networking models is part of a single process. It has to do with the "re-engineering" of how corporations operate. The modern corporation, to be competitive, has to find ways to reduce the size of its hierarchical structure. This can only be achieved if the decision power is delegated further down in the hierarchy, i. e., decisions need to be decentralized. However, good decisions require not only processing power (e.g., PCs), but, more importantly, prompt access to up-to-date information (i.e., telecommunications). Thus, powerful computers on the desktop and powerful data communication capabilities are not just technological niceties, but are well-integrated in the process of making society more productive.

3 The Development of the Internet

The evolution from a standalone PC into a client-server environment happened rapidly. This development was only possible because of the research in data networking that has taken place in the last twenty years. Probably the most influential project in data networking was the ARPANET [1]. The Defense Advanced Research Projects Agency (DARPA) funded the ARPANET project with the objective of creating a packet-switch data network that was robust to hostile environments. This approach led to fundamental contributions to distributed networking, including the creation of TCP/IP (a suite of protocols designed around a "best-effort" internetwork protocol), and the concept of shortest-path routing.

A series of strategic decisions contributed significantly to the success of TCP/IP as the most widely used data networking protocol. First, very useful applications (i. e., remote login, file transfer, mail transfer) were made integral parts of TCP/IP. This protocol was then made an open standard by the U.S. Department of Defense (DOD). A central authority, the Network Information Center (NIC), was created to assign unique IP network addresses so that isolated networks could be interconnected at a later time without having conflicting addresses. Later, TCP/IP was incorporated into the Berkeley Software Distribution (BSD) UNIX, becoming widely available at universities and research institutions. Any institution was allowed to become part of this mesh of interconnected networks, which received the name of Internet. A standards organization, the Internet Engineering Task Force (IETF), was created to support further developments of Internet protocols. Several other supporting protocols and services (e.g., ARP, RIP, OSPF, DNS, MIB) were created to automate network configuration and management. Finally, the National Sciences Foundation (NSF) established access networks, centered around its supercomputer centers, to facilitate the access to the Internet.

There is much to be learned from the success of the Internet when compared to other fine networking efforts like X.25. It seems that the concept of a data network as a data transport technology is not an adequate one. The experience gained with the Internet shows that data networking should be viewed as a set of valuable services, and that data transport is only useful if the services bring some value to the user. The development process of a data network service is not very efficient when the burden of creating value out of a data transport is relegated to the user.

One of the pioneering aspects of the TCP/IP protocol was the creation of protocol layers. The concept of layers has two major benefits: (1)it defines in detail (i.e., layer by layer) the responsibilities of the protocol; (2) it makes the development of code simpler by breaking the protocol into modules and clearly describing the interfaces between the modules. **Figure 1** depicts the Internet and the OSI reference models.

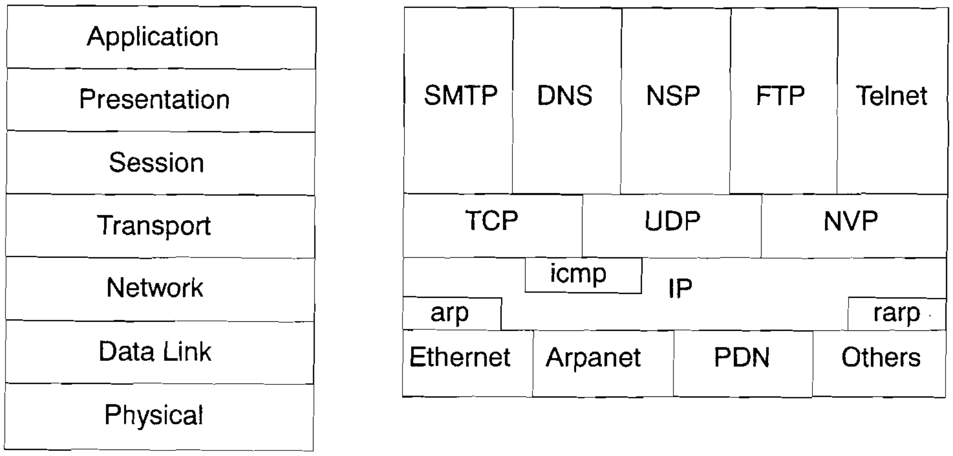


Figure 1 - The Internet and OSI reference models

TCP/IP's inter networking protocol (IP) is referred to as a connectionless protocol. Among others, an important characteristic of this kind of protocol is that the task of routing packets through the network does not require the knowledge of an end-to-end "session". The network relegates all the functionalities that require the knowledge of a "session" (e.g., reliable delivery of packets) to an upper layer (i.e., end-to-end transport protocol). This approach has two major advantages over that of connection-oriented (e.g., virtual-circuit) protocols. The first advantage is that routing decisions can be made on a node-by-node basis, asynchronously from the initiation or termination of a session. The second advantage is that completely distributed routing algorithms, not tied to events at higher layers, have become feasible. In fact, one can argue that this advantage has contributed to the increased popularity of permanent virtual circuit (PVC) networks (e.g., Frame Relay) relative to switched virtual circuits (SVC) networks (e.g., X.25) although representing a step towards routing inefficiency. All that is required for routing in a datagram network (i.e., a network that uses a connectionless internetworking protocol) is a function (routing table) on each router that associates an incoming packet with an outgoing port, and a routing algorithm that fills in the routing table entries such that the ensemble of routers operate in a coordinated way. **Figure 2** depicts the elements of a distributed routing architecture.

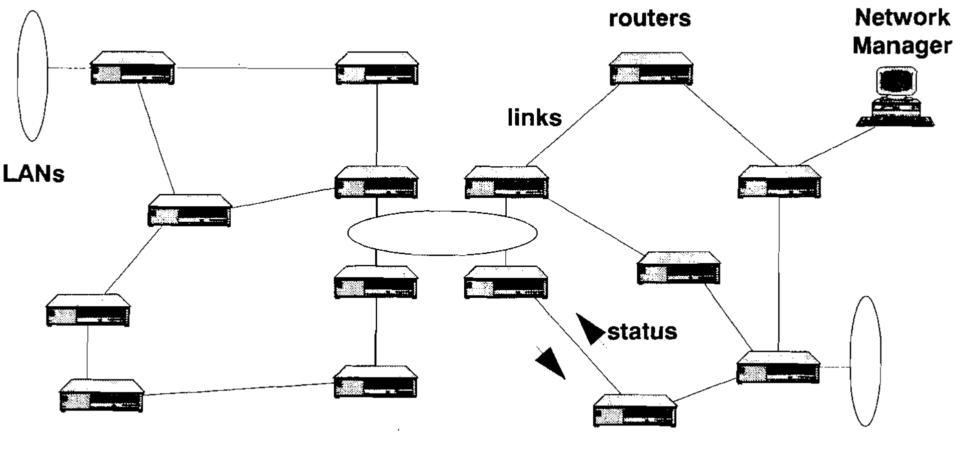
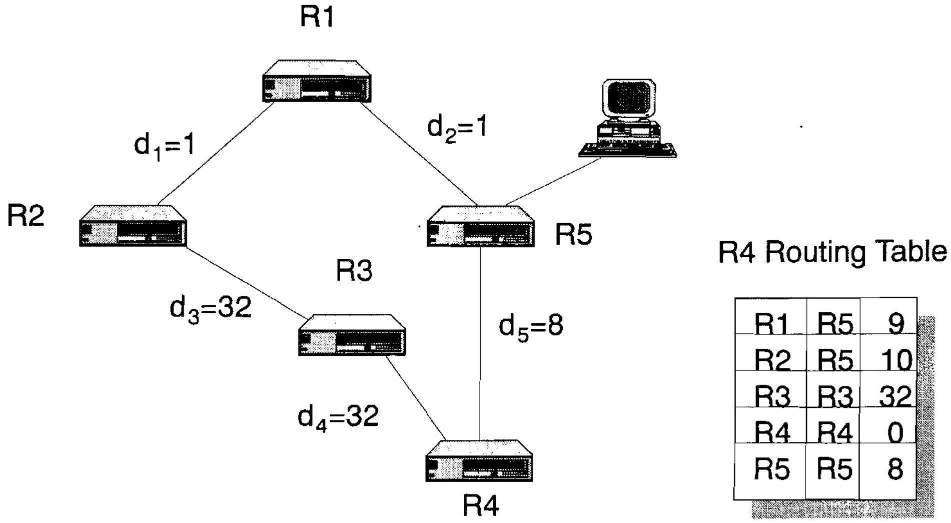


Figure 2 - Elements of a Distributed Routing Architecture

Some mechanism is necessary to fill in the routing table entries (e.g., it could be manual); when automated, it is referred to as a routing algorithm. The concept of shortest-path routing as a distributed routing algorithm for data networks is another outcome of the ARPANET project. Shortest-path has become the preferred solution for routing algorithm in data networks today. Most data networking routing standards (e.g RIP, OSPF, IS-IS) are based on shortest-path routing. Shortest-path routing simply requires a routing table where all routes (i.e., routing table entries) are shortest paths with respect to a pre-assigned distance metric. This concept allows for computationally efficient distributed implementations and an effective re-routing mechanism upon node and link failure. Furthermore, it provides a set of controls that are easy to manage.

In a shortest-path environment, a distance metric (i.e., $D=\{d_l\}$) is assigned to each link (i.e., $L=\{l\}$) in the network by the network manager. The distance metrics are then disseminated among all routers (i.e., N={n}) in the network and each router calculates the shortest paths to all destinations in the network. The resulting shortest paths determine the routing table entries. **Figure 3** depicts an example of a network using a shortest path routing algorithm; the number associated with each link corresponds to the distance metric, and the first, second and third columns of the routing table correspond to the destination router, the next hop router and the shortest distance, respectively.



R1	R5	9
R2	R5	10
R3	R3	32
R4	R4	0
R5	R5	8

Figure 3 - A network using shortest-path routing

There are two basic types of protocols that disseminate the routing information through a network of routers: link-state and distance-vector protocols. In link-state protocols, the routers exchange among themselves information about the topology of the network, including information about which links are currently up or down and the distance metric associated to each link. See [Moy91] and [IS091] for detailed information on a distributed link-state protocols. After receiving complete information on network topology and on link-distance metrics, each router then calculates the shortest paths to all other routers in the network (i.e., based on Dijkstra's algorithm). Thus, the routing table of all routers in the network have entries that are consistent and they all synthesize the shortest path routes. In distance-vector protocols, the ensemble of routers in the network jointly runs the Bellman-Ford algorithm. Routers exchange with their neighbors information about the distance to every other node in the network. Each router then applies the triangle inequality between (1) its distance to each destination and, (2) its distance to each neighbor plus each neighbor's distance to each destination, always selecting the shortest path. The routing in all routers in the network eventually converges to the shortest path routing.

Routing algorithms other than shortest-path usually require direct intervention of a centralized network manager to assign the routing table entries. The main advantages of shortest-path routing algorithms over other types of routing algorithms are: ease of management and effectiveness upon failures. It is easier to manage since the network manager only has to assign |L| (i.e., the number of links in the network) values as opposed to |N| |N| values (i.e., the number of routers times the number of entries on each routing table). It is more robust to configuration errors since if the network manager makes a mistake while assigning distance metrics, what can happen is that routing in the network may not be as optimal as it could be; on the other hand, if the network manager makes a mistake while assigning routing table entries, it could have very disruptive effects on network operation (e.g., looping). It is more robust upon network failures since it does not depend on centralized intervention to change routing tables entries; upon a network component failure (e.g., a link or a node failure), the appropriate distance metrics are set to infinity, this information is disseminated, and new shortest paths are automatically calculated to avoid the failed components.

4 High-Speed Interconnect Technologies and Services

There are three different fronts in high-speed networking. Each one of these fronts has different assumptions about the problem that high-speed networking is meant to solve. Consequently, each front has a clearly different focus and a different strategic approach towards high-speed networking. Despite having different immediate objectives and different architectural approaches, these fronts seem to agree that ATM is the technology of choice due to its flexibility and its potential of supporting a large class of services.

One of these fronts has its roots in data networking and can be exemplified by commercial initiatives like Frame Relay (FR), Switched Multi-megabit Data Service (SMDS) and some versions of Asynchronous Transfer Mode (ATM). These initiatives perceive the main role of the technology as that of providing wide-area interconnection for existing LAN applications. They focus on extending LANs to the wide area, while making the architecture flexible enough to support other services. For instance, voice as well as video traffic are considered to be data applications with particularly stringent delay variability requirements. Thus, offering voice or video services would be contingent on developing efficient techniques to carry real-time traffic over an evolving data network architecture.

Another of these fronts has its roots in the voice world and can be exemplified by most ATM efforts in Europe. The most distinctive characteristic of this approach is that very small buffers are used in their ATM switches. On one hand, this characteristic solves in a simple way any problem related to delay variability in the transport of voice traffic. On the other hand, not having enough buffers in the network may result in one of two things: it may either reduce the ability of a chosen architecture to support a wide range of high-speed applications or it may force the network to operate under light or highly controlled traffic conditions. The preferred solution in this front has been to rely on access control techniques to keep the incoming traffic well-behaved so that the queue occupancies are small.

The third of these fronts has its roots in the research community and can be exemplified by research initiatives like the NREN and the CNRI Testbeds. These initiatives perceive their role as that of nurturing new applications for high-speed data networking. These initiatives are, in principle, a continuation of the Internet research efforts, directed, however, towards issues that may be present in data networks of the future. These initiatives are unique since they combine the efforts of industry, academia and the government in an orchestrated effort; they combine high-speed transport networks, built by the industry, interconnecting research labs and universities with the NSF supercomputing centers, with research grants from the NSF to support the development of new supercomputing applications.

The prospects resulting from the efficient way society has been applying its efforts towards the study and development of high-speed networking are encouraging. The lessons learned from the success of the Internet (e.g., that technology is only useful if it provides services that bring value to the user) are making an impact in most of today's efforts in high-speed data networking. However, before we can make sense out of the myriad of choices that are presented to the users, equipment providers and service providers, it is necessary to understand the technologies and services that will make it all happen. Next we discuss some of the existing high-speed interconnect technologies and services.

4.1 High-Speed Interconnect Technologies

4.1.1 Asynchronous Transfer Mode

In ATM, all information is carried in fixed-sized slots called cells. These cells are 53 bytes long, with 5 bytes of header and 48 bytes of payload. The header formats at the User-Network Interface (UNI), which is the interface seen by a user, and at the Network-Node Interface (NNI), which is the interface between two network elements, are shown in **Figure 4**. The 4-bit Generic Flow Control (GFC) field is mainly used for Media Access Control (MAC), so that multiple access at the UNI be allowed. The 8-bit (12-bit at the NNI) Virtual Path Identifier (VPI) field provides enhanced routing capabilities (e.g., denotes membership in a virtual network). The 16-bit Virtual Circuit Identifier (VCI) field provides basic routing capabilities. The 2-bit Payload Type (PT) field identifies the type of information carried by a cell (e.g., user data or Operation And Management (OAM)). The 1-bit Cell Loss Priority (CLP or C) field identifies cells that are to be discarded first, when necessary (e.g., upon congestion). The 1-bit Reserved (R) field is currently unused.

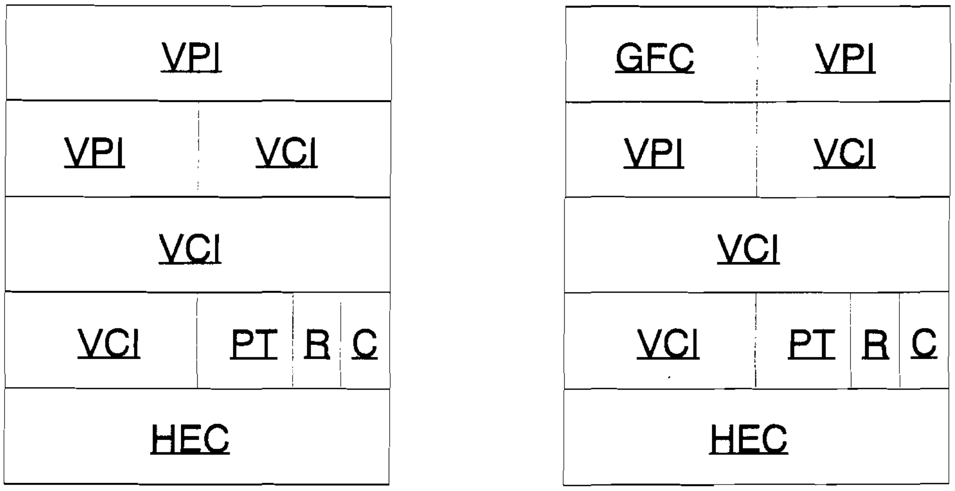


Figure 4 - ATM Cell Format at the B-ISDN NNI and UNI Interfaces

The conceptual difference between the ATM approach and a conventional channelized approach is depicted in **Figure 5**.

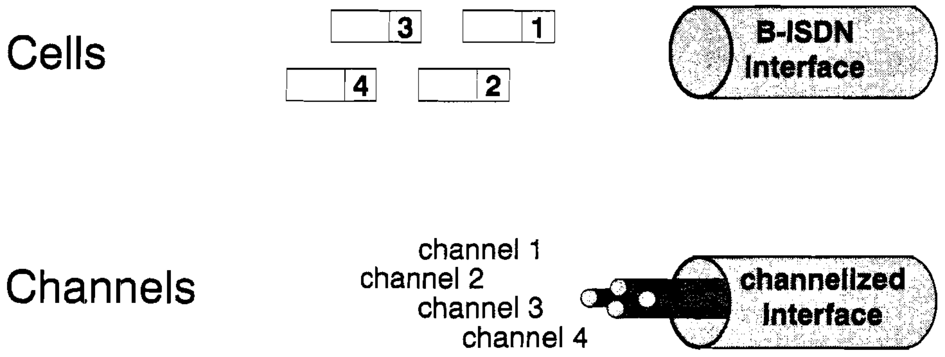


Figure 5 - B-ISDN Concept

The ATM structure incorporates three fundamental features: (1) a transmission format that allows for powerful multiplexing capabilities, (2) switching concepts that allow for extremely high switching rates, and (3) a data slicing and control technique that allows for extremely flexible service disciplines. The first feature attracts to ATM the enthusiasm of those who provide transmission services, since it creates an infrastructure alternative to the hierarchical structure of the current DACs networks that makes it much easier to offer more flexible services with a wider range of circuit speeds. The second feature attracts to ATM the enthusiasm of the switched service providers and equipment vendors who have been looking for ways to create and support new applications. The third feature attracts to ATM the enthusiasm of those responsible for the design, the engineering and the operation of network services, since it allows for the use of a vast number of algorithms and methods in support of their specific objectives.

4.1.2 Distributed Queue Dual Bus

Distributed Queue Dual Bus (DQDB) is a Media Access Control (MAC) protocol for the access of multiple nodes to a bi-directional Bus. This protocol has been adopted by the IEEE 802.6 working group as the Metropolitan Area Network (MAN) Standard.

The IEEE 802.6 working group had two major goals in mind when it decided to adopt DQDB as the MAN Standard: (1) to create the first multiple access network protocol based on ATM technology, and (2) to position the IEEE 802.6 Standard as the multiple-access protocol standard to a B-ISDN interface. The working group was successful in its first goal but not as successful in its second goal. Although the differences between the B-ISDN and IEEE 802.6 formats are small, there are enough differences that make them incompatible. Nevertheless, the interfacing between them is relatively simple, positioning DQDB as a good migration technology to B-ISDN.

DQDB was conceived to provide simultaneously three types of services: connectionless, connection-oriented and isochronous. Bellcore decided that the DQDB Standard would be the appropriate platform to define a high-speed connectionless data service, which received the name of Switched Multimegabit Data Service (SMDS). The DQDB connectionless service standard became the priority of the working group, being finalized before the other services.

The IEEE 802.6 protocol is independent of the underlying physical medium. It can operate on copper as well as on fiber. It can operate at arbitrary speeds when supporting either connectionless or connection-oriented services, or with a minimum speed of 3.392 Mbps (a 53-byte cell per 125, μs frame) when supporting isochronous services. This protocol can also use different transmission systems, including Plesiochronous Digital Hierarchy (PDH) and Synchronous Digital Hierarchy (SDH).

DQDB consists of two unidirectional buses, one in each direction of transmission, with several nodes connected to both buses (Figure 6). The head node on each bus generates empty slots that traverse all the nodes until they reach the tail node. Isochronous slots, also referred to as pre-arbitrated slots, are marked as such by the head station with an appropriate VCI value, and reserved for exclusive use by the nodes that requested isochronous service. The access by the network nodes to all other slots, also called queue-arbitrated slots, is controlled by the MAC protocol. The operating principle of the MAC protocol is as follows:

1. When a node has a segment to transmit in a given direction (i.e., downstream), it sends a request in the direction opposite to transmission (i.e., upstream). A node can only have one outstanding request at a time.

2. Each node continually estimates the number of nodes downstream (i.e., its position in the distributed queue) that are awaiting an empty slot (i.e., by subtracting the number of requests going upstream from the number of empty slots going downstream).

3. When a given node has a segment to transmit, which it has received from an upper layer, it records its position in the distributed queue at that time and sends a request upstream.

4. Meanwhile, the given node continues to estimate the number of nodes downstream that are awaiting an empty slot.

5. Before transmitting that segment, it lets a certain number of empty slots pass by, corresponding to the number of stations ahead of it in the distributed queue at the time it sent the request.

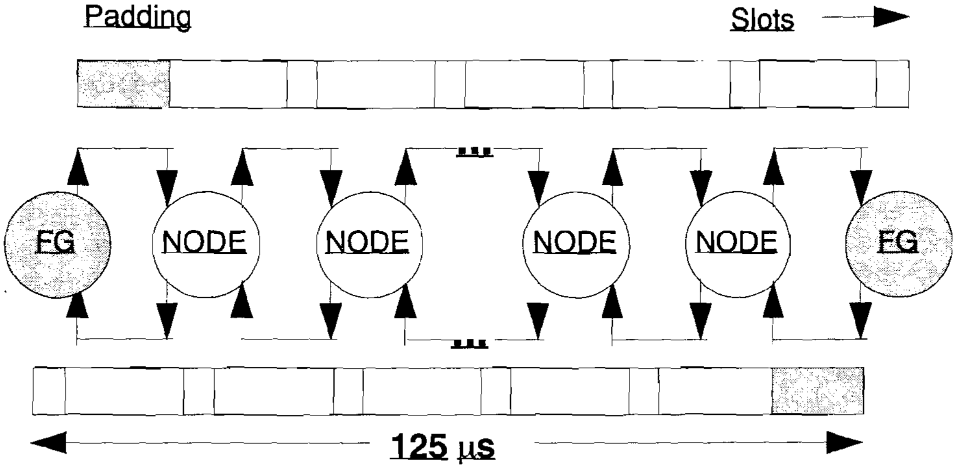


Figure 6 - DQBD Topology

The DQDB protocol allows for automatic reconfiguration in case of failure, providing self-healing capabilities. This is achieved by making the head and tail nodes the same node (i.e., looped configuration). Upon failure (e.g., fiber cut or node failure) the head/tail node becomes just a regular node, while the nodes around the failure become the head and tail nodes (i.e., as depicted **Figure** 7).

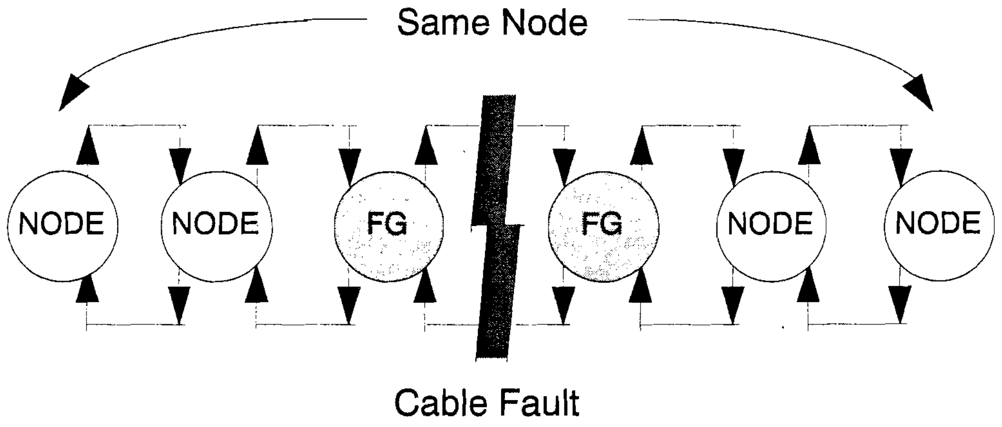


Figure 7 - Fault tolerance

The DQDB protocol was designed to be a low-latency MAN. That is achieved by every node having a passive optical interface to the fiber. As slots traverse the node's optical interface, a local copy of its contents is made to the node's buffer (e.g., bit-by-bit). The node can then process the local copy of the segment, while the actual segment follows its path through the fiber without being delayed.

The ATM structure of DQDB allows for the efficient spatial reuse of the bus bandwidth. Spatial reuse can lead to large potential gains, especially when traffic is localized among neighboring nodes [Rod90], as depicted in **Figure 8**.

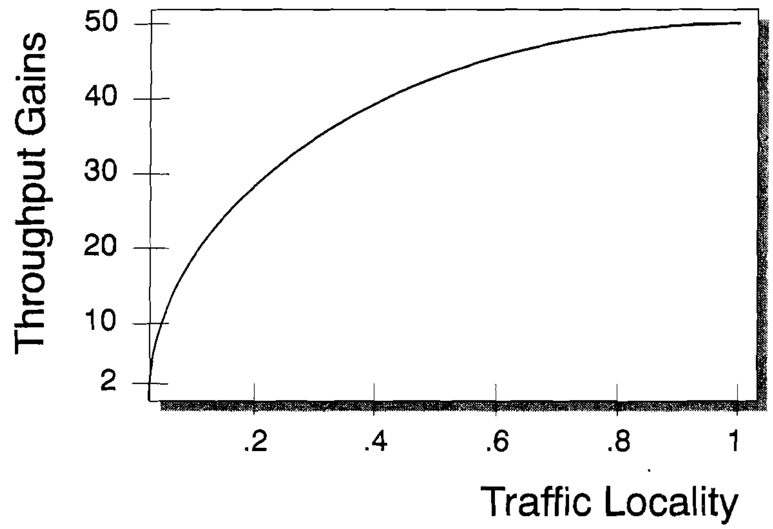


Figure 8 - Throughput gains for levels of traffic locality

Spatial reuse, as depicted in **Figure 9**, can be achieved in two different ways: (1) by transforming the DQDB nodes into store-and-forward nodes (DR), or (2) by using Erasure Nodes (EN).

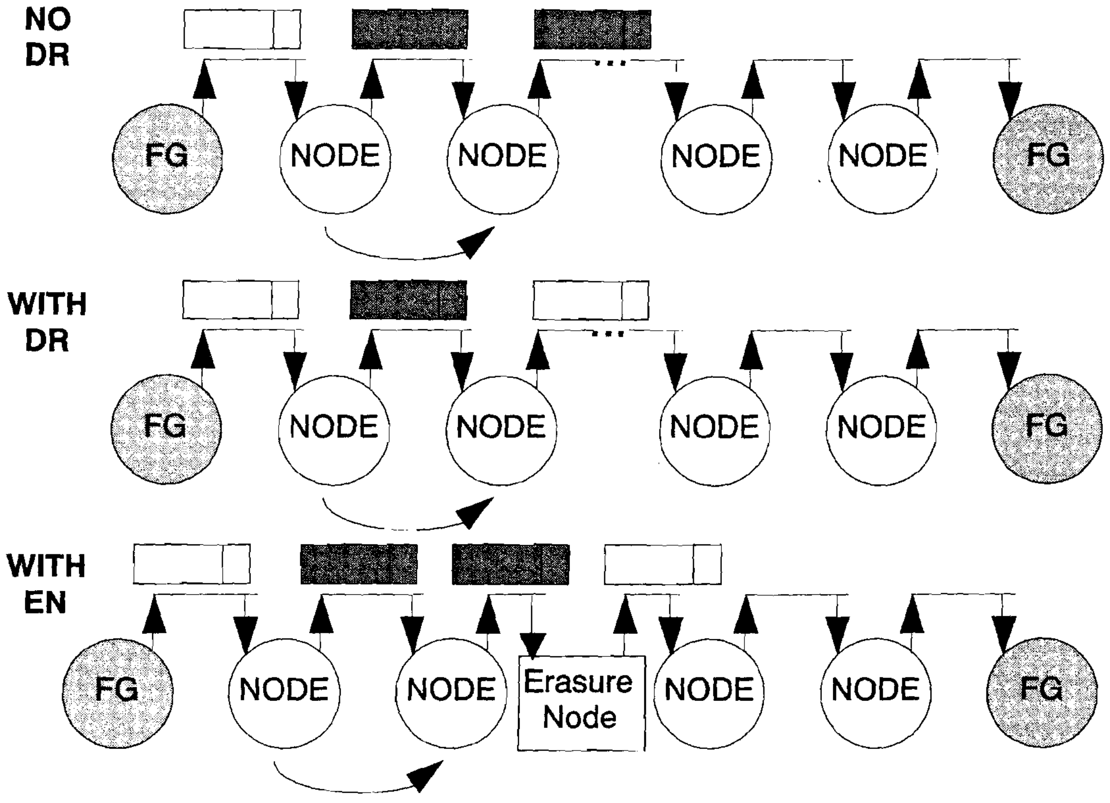


Figure 9 - Destination release (DR) and erasure nodes (EN)

A store-and-forward node stores and processes each segment before forwarding it to the next node. Segments (non-multicast) that are destined to it are then copied locally and an empty slot is forwarded in its place. When a store-and-forward solution is used for improving bandwidth efficiency, the low-latency characteristic of DQDB is somewhat lost. A solution using erasure nodes allows the regular nodes to maintain their original passive optical interface, thereby retaining the low latency characteristics of DQDB. As slots traverse the optical interface of a node, a local copy of its contents is made to the buffer. If that particular node is that segment's destination, it sets the Previous Segment Read (PSR) bit in the header of the following segment. Erasure nodes, placed appropriately along the bus, erase the marked segments. Note that each erasure node has to store-and-forward a little more than a slot. However, since there would be only a few erasure nodes in a DQDB network, it would not increase significantly the overall latency of the bus.

4.2 High Speed Interconnect Services

4.2.1 Frame Relay

Frame Relay (FR) is a DTE-DCE interface specification. It is based on LAPD (i.e., Q.921), the ISDN version of LAPB (i.e., X.25 layer 2). As in any interface specification, FR does not attempt to define the technology to be used in the network. Any backbone network technology that is capable of provide the services required by the network interface specifications is adequate.

FR has been positioned as a LAN interconnect service. It provides interconnection between pairs of access points through Permanent Virtual Circuits (PVCs). Functionally, a FR network is a collection of (virtual) private lines. From a service point of view, FR merges the advantages of private line networks with those of packet networks. This is achieved by defining a service contract for each PVC, where a minimum throughput, referred to as Committed Information Rate (CIR), and a minimum buffer storage, referred as Committed Burst Size (CBS), are guaranteed. Usually, conformance to the contract for each PVC is verified at the network access by a leaky-bucket mechanism. Frames that cause a contract violation condition while entering the network are either dropped or at risk of being dropped. Some FR implementations drop the violating frame on access. Other implementations allow the violating frame to enter the network with a Discard Eligibility (DE) bit marked. Frames carrying the DE bit marked are dropped whenever they encounter congestion in the network.

FR can also provide congestion management information in the form of Forward Explicit Congestion Notification (FECN) and Backward Explicit Congestion Notification (BECN) signals. This congestion management information can improve access of end users to excess capacity while still maintaining some control over data loss. End users can take advantage of adaptive control algorithms that require network feedback to control theirs transmission rates.

FR can operate at very low speeds all the way up to T1 or E1 speeds. FR is a desirable network interface for this range of speeds particularly because of its low overhead. There is no technical reason for not implementing FR at higher speeds. However, since flexibility of offering a wide range of services rather than low overhead is more relevant at higher speeds, other interface standards (e.g., B-ISDN) become more appropriate.

Addressing in FR is simple. A PVC at a given interface is identified by a Data Logical Channel Identifier (DLCI). Frames associated with a given PVC traversing an interface carry a common DLCI, independent of the direction in which it travels. The DLCI identifying a PVC at the interface at one end does not need to be the same at the other end; the network is responsible for appropriately translating the DLCI field of every frame crossing a FR interface. The typical DLCI assignment is sometimes referred to as local addressing. Here, DLCIs only have significance locally at the FR interface. **Figure 10** depicts four locations fully interconnected over the FR network, where DLCIs are assigned according to local addressing.

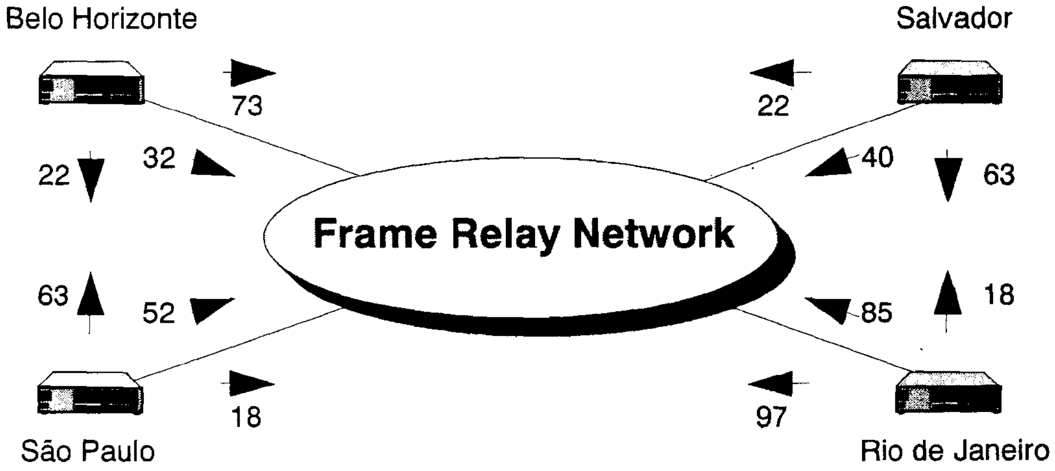


Figure 10 - Local addressing in a FR network

The only constraint is that DLCIs have to be unique on each interface. There is a more restrictive DLCI assignment, sometimes referred to as global addressing. Here, DLCIs have global meaning and cannot be reused within a logical network. **Figure 11** depicts the same fully interconnected locations, where the DLCIs are assigned according to global addressing. DLCIs identifying PVCs destined to a common destination are made identical.

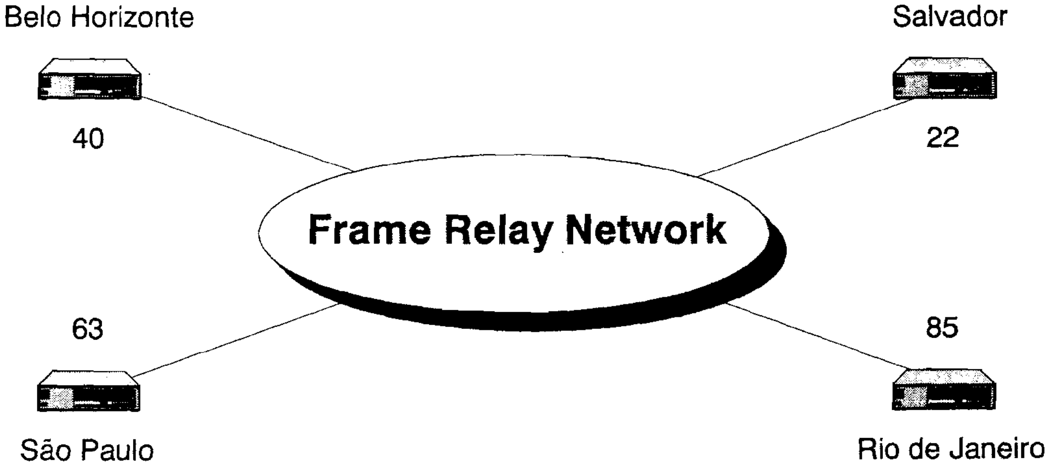


Figure 11 - Global addressing in a FR network

Global addressing, however, decreases the address space significantly and is only practical for small networks.

4.2.2 Switched Multi-megabit Data Service

The Switched Multi-megabit Data Service (SMDS) is the combination of a DTE-DCE interface specification, based on the IEEE 802.6 point-to-point Standards, and a set of service guarantees. It is based on ATM cells and the adaptation layers AAL types 3 and 4. Similarly to FR, SMDS defines the service at the network access interface and does not define the technology to be used in the network. Any backbone networking technology that is capable of providing the services required by the network service specification is adequate.

SMDS has been positioned as a LAN interconnect service. It provides interconnection between pairs of access points as well as multicast capabilities. Multicast makes SMDS functionally equivalent to a LAN. Consistently with the functional concept of a LAN, the service guarantees apply to an SMDS interface as opposed to connections between pairs of nodes (e.g., as in FR). The service contract for each SMDS interface is similar to a FR PVC contract and consists of a minimum throughput, referred to as Committed Information Rate (CIR), and a minimum buffer storage, referred to as Committed Burst Size (CBS). Usually, conformance to the contract for each SMDS interface is verified at the network access by a leaky-bucket mechanism. ATM cells that cause a contract violation condition while entering the network are at risk of being dropped. The violating cell is allowed to enter the network with the Cell-Loss Priority (CLP) bit set. Cells carrying the CLP bit set are dropped whenever they encounter congestion in the network.

SMDS does not provide explicit congestion notification. Adaptive algorithms have to rely on round-trip delay measurements to adjust the rate of transmission to congestion in the network.

One drawback of SMDS, as well as of other services based on ATM technology, is its high overhead. This is not a concern at high speeds since flexibility to offer a wide range of services more than compensates for the high overhead. However, the lack of bandwidth efficiency makes SMDS not so attractive for speeds of E1or less .

Addressing in SMDS is simple. **Figure 12** depicts in an SMDS network. There is an E.164 address associated with each SMDS interface. The interface E.164 addresses are equivalent to MAC addresses in LANs. A given E.164 interface can belong to one or more multicast group. There is a multicast E.164 address associated with each multicast group. All E.164 addresses are unique in the SMDS network.

4.2.3 Broadband ISDN

The Broadband Integrated Services Digital Network (B-ISDN) is a CCITT standard for a high-speed digital interface. Its purpose is to provide access to a wide range of services, including voice and non-voice services requiring rates greater than the Integrated Services Digital Network (ISDN) primary rate (i.e., T1 or E1). The motivation to incorporate broadband features into ISDN is documented in CCITT Recommendation I.121 [CC188]. After much discussion about possible ways of supporting higher rates, including various hierarchical solutions, a decision was reached in the form of a flexible non-hierarchical rate structure, referred to as Asynchronous Transfer Mode (ATM) [CC189] [CC189b].

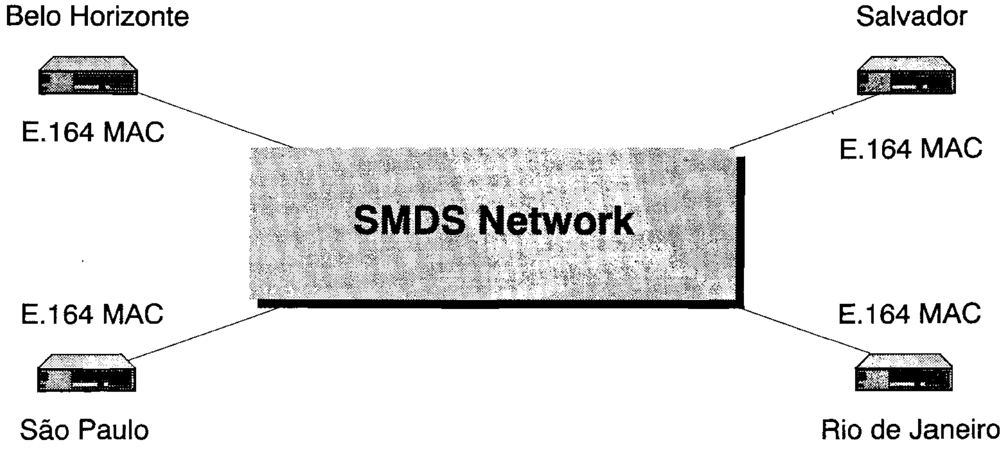


Figure 12 - Addressing in a SMDS network

For those that see in ATM the next infrastructure technology for providing the transmission services that are currently available (e.g., circuits), the B-ISDN work is complete. In fact, the SDH/SONET transmission format, depicted in **Figure 13**, supports the ATM cell structure. However, many believe that the cost of a major overhaul of the infrastructure can only be justified if new services can be provided. ATM provides the flexibility required by B-ISDN to achieve its purpose. However, ATM alone does not suffice. Other mechanisms must be defined and created before this flexibility reaches the end user. B-ISDN attempts to make this flexibility available through its service requirements, defined based on a set of potential services and applications. These service requirements allow B-ISDN to support services like FR and SMDS, as well as Constant Bit Rate (CBR) and compressed video applications.

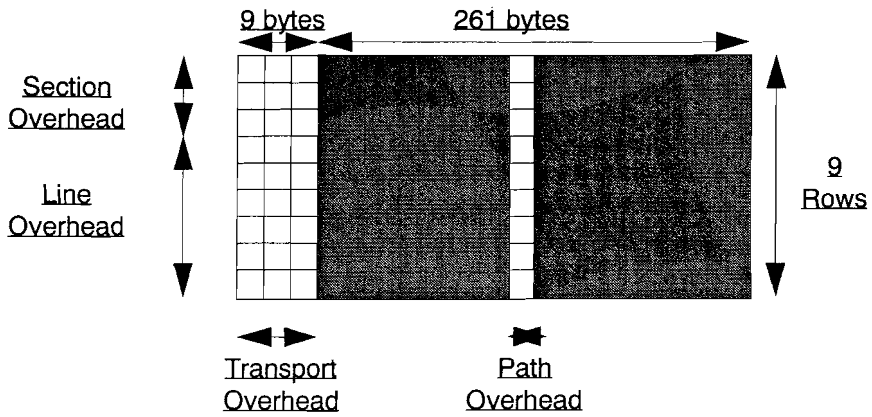


Figure 13 - SDH/SONET (STM-1)

5 High-Speed Data Networking Challenges

There is a vast body of work in the form of flow control, congestion control and resource allocation; see [Ger80], [Tur86], [Zha87], [Max90] and [Fen91]. As high speed data networking develops, we face new issues. Some of these issues are conceptually simple from a system point of view; those that make the system behave like a scaled version of a lower speed system. The issues that are most challenging are those that make a higher-speed system behave fundamentally differently from a similar lower speed system.

In wide-area high-speed data networking, propagation delays dominate other delays in the network. Unlike other parameters that scale with speed, propagation delays remain fixed since they are consequence of physical distances and the speed of light. Propagation delays combined with high transmission rates result in a large delay-bandwidth product, making the tradeoff between efficiency and congestion more complex and traditional bandwidth allocation and congestion control mechanisms ineffective. Resolving the issues of congestion and controllability in networks with large delay-bandwidth products is the key for making the high-speed networking resources that are currently being deployed usable.

5.1 Adaptive Control Schemes

Network performance requirements usually include high bandwidth utilization, low loss and low delays. However, these are conflicting goals; high bandwidth utilization requirements lead to instability; instability leads to either high loss or high delays, depending on the size of queues in the network. The traditional approach to resolve this tradeoff is to use windowing flow control algorithms in order to limit the amount of outstanding data. However, the windows have to be very large to achieve high utilization in a network with a large delay-bandwidth product, defeating its purpose.

The natural solution is to have the users adapt to the conditions of the network, making their window sizes large or small, depending on the number of users contending for the network resources. Feedback control to dynamically change the size of a window has been incorporated into some versions of TCP/IP [Jac88]. The feedback information is typically derived from round-trip delay measurements; an increase in round-trip delay, interpreted as an increase in network congestion, causes a decrease in the window size, slowing down each source, and vice-versa. However, early experiences with adaptive windows have shown that large oscillations in queue occupancy may occur, raising questions about the effectiveness of adaptive mechanisms in high-speed networks. Several adaptive feedback control schemes have been proposed; see [Jai86], [Ram88]. Later, Bolot and Shankar [Bol90] modeled an adaptive scheme, similar to the one proposed by Ramakrishnan and Jain, as a set of coupled differential equations, and obtained analytical closed form solutions for the transient and steady state behavior of this model. Mukherjee and Strikwerda [Muk91] analyzed a similar adaptive scheme and provided some fundamental insight about the stability behavior of linear-increase, exponential-decrease control. Later studies have shown that an optimal static window size exists [Mit90] and that a dynamic adaptive algorithm [Mit91] [Mit92] approaches the performance of the optimal static system.

A class of end-to-end adaptive algorithms for rate control has been proposed and analyzed [Fen92]. The system model is depicted in **Figure 14.** There is a single source which transmits at rate λ (t). The data reaches a bottleneck queue after experiencing a deterministic propagation delay of τ_1. The data is served from the bottleneck queue at a (deterministic) rate of μ and leaves the bottleneck queue carrying information about the queue length process x(t) upon the time of departure. This information returns to the source through acknowledgment packets at a rate of $\zeta(t,\tau_1,\tau_2)$ after experiencing a deterministic propagation delay of τ_2, where $\tau = \tau_1 + \tau_2$.

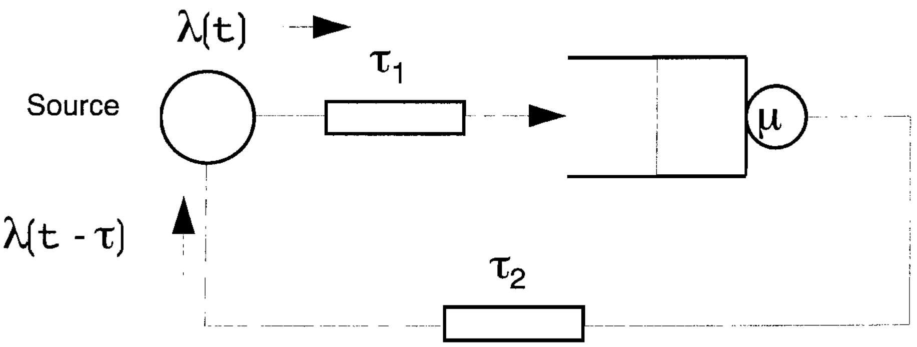


Figure 14 - The system model

The behavior of this model is captured by the set of deterministic delay-differential equations (1). The queue length x(t) changes in proportion to the difference between the mean arrival and departure rates. The source adapts its transmission rate as a function F_μ of the queue length (τ_2 later).

$$\frac{\partial}{\partial t} x(t) = \begin{cases} \lambda(t-\tau_1) - \mu \;;\; x(t) > 0 \text{ or } \lambda(t-\tau_1) - \mu < 0 \\ 0 \;;\; \text{otherwise} \end{cases}$$

$$\frac{\partial}{\partial t} \lambda(t) = \begin{cases} F_\mu(0)\,\lambda(t-\tau) \;;\; x(t-\tau_2) = 0 \\ F_\mu(x(t-\tau_2)) \;;\; \text{otherwise} \end{cases} \qquad (1)$$

Scaling, stability and instability results were obtained. The vector field depicted in **Figure 15** shows how the system evolves for any monotone non-increasing control function F. The horizontal axis denotes the occupancy of a bottleneck queue in the network. The vertical axis denotes the transmission speed at the source. The designed operating point ($\overline{x}$, λ) is unstable in a very strong sense; any perturbation will make the system oscillate. The cause of this instability is the propagation delay in the feedback loop which adds into the system a pole with positive real part. The system does not oscillate without bounds due the boundary condition at $x = 0$ (i.e., the queue cannot have negative occupancy). The system has a spiral flow, directed towards the rectangle C when outside C and out away from the center C when inside.

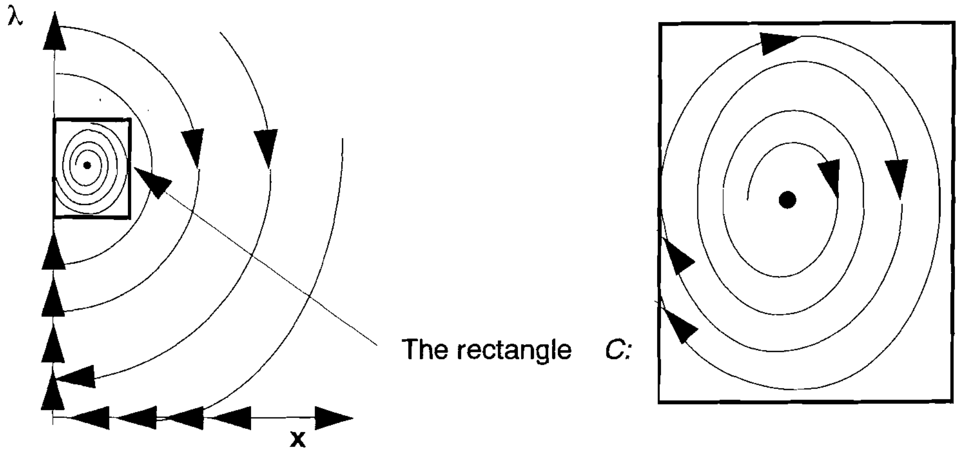


Figure 15 - Vector field of a typical rate control strategy

When the control function F is scaled as in Equation (2), where f is a monotone non-increasing function with $f(0) = 0$, the lengths of the sides of the rectangle C became O ($\sqrt{\mu}$).This implies that the rate λ (t) does not settle down to a single point, but oscillates with relative amplitude of $1/\sqrt{\mu}$. Thus, if one uses the scaling proposed here, the efficiency of the system actually increases with speed, as opposed to what is commonly believed. This scaling also requires buffers that increase with the square root of the network speed, which is remarkable.

$$F_{\mu}(x) = \frac{1}{\sqrt{\mu}} f\left(\frac{x}{\sqrt{\mu}} - b\right) \qquad (2)$$

Extensions to this model have been proposed and analyzed in [Fen93] and [Fen94]. In these models we consider several sources with diverse characteristics (e.g., non-adaptive and adaptive sources with different feedback delays, and different constraints on transmission rates) and finite-buffer ATM switches. These extensions take into account the randomness and burstiness of traffic from other sources and th interaction between different adaptive sources. The arrival process of non-adaptiv traffic is modeled as a general stochastic fluid with bounded positive rates. A detaile description of the queue-length and rate processes are obtained. Stability of th proposed scheme is proved, in that the rates of transmission of the adaptive sources an their queue-length processes rapidly converge to a neighborhood of the designe operating point. Not only does this extended model confirm the scaling of the model' parameters obtained in the deterministic analysis so that the control scales well wit increasing trunk rates, but it also determines the appropriate scaling with respect to th burstiness of the non-adaptive sources. Based on this new result, engineering rules fo designing globally effective access control mechanisms (e.g., leaky bucket) can b derived. Related applications can be found in [Ber92] and [Ber93].

Making further assumptions about the nature of the network can lead to more efficien controls. That is the case of [Kes91] and [Hah91], where the network nodes ar assumed to implement a processor-sharing service discipline.

Introducing memory to the system can transform an unstable equilibrium point into a stable one [Hal77]. This effect can be modeled with the introduction of a loss term in the differential equation describing the control. In [Elw 94], Elwalid proposes design procedures and interprets the choice of parameters to achieve local stability in such systems.

Most adaptive flow control systems have been devised as end-to-end mechanisms, often implemented by the network users. A hop-by-hop adaptive flow control mechanism, that has to be built into the network, has been proposed in [Mis92]. This work revives "back pressure" as an alternative architecture to congestion control. It provides an interesting tradeoff between processing/bandwidth overhead and system stability.

5.2 Resource Allocation

User application requirements translate, through service requirements, into performance guarantees that need to be provided by the underlying (high-speed) network platform. The application requirements usually include bandwidth, delay, response time and loss performance. Often, the performance expectation differs from user to user, even for the same application. The challenge of the designers of algorithms, systems and services is to join forces and define a system, made of a set of services and performance guarantees, that can meet the requirements of a broad set of applications and still leave room for each user to make their own tradeoffs between levels of efficiency and performance.

Adaptive control for high-speed networks, along with proper buffer sizing and scaling, is very effective in avoiding congestion and allowing for a high bandwidth utilization.

However, the performance guarantees that can be provided this way only apply over relatively large time scales. Such guarantees are often not sufficient, particularly in a high-speed network that is shared among applications with significantly different requirements, including real time applications (e.g., data, CBR, voice and video). For some discussion on time sensitive applications, see [Gol92], [Hym91], [Woo90]. Consequently, some explicit scheme for providing more stringent (e.g., real time) guarantees is necessary. These schemes take the form of resource allocation schemes, (see [Hui88]).

To be effective in supporting a wide array of services, a resource allocation scheme must provide enough coordination among network elements so that some level of performance guarantees can be provided. On the other hand, to be efficient in the use of network resources, it has to rely on as little coordination as possible among network elements due to the relatively small time scales of congestion at high speeds.

The traditional approach to resource allocation relies on strong coordination among network elements. Upon call setup, buffers and bandwidth are dedicated to that connection along its path. Techniques and concepts that could help engineer such systems can be found in [Pan89] and [Elw93]. However, such schemes are efficient only when most sources behave like CBR sources with long connections. In fact, this approach is limited in scope since it does not provide a straightforward means for supporting connectionless services. Furthermore, the inefficiencies of such schemes become exacerbated in networks with large delay-bandwidth product; at a given time, most data from a particular source will be in the pipeline, while most of the reserved resource along the path of the data will remain idle [Mit90].

Access control, on the other hand, is an approach that relies on very little coordination among network elements, (see [Lua88], [Eck89] and [Eck91]). The traffic that enters the network is tested (e.g., with a leaky bucket algorithm) against the user's service contract. The service contract includes a predetermined amount of bandwidth and buffers. The cells (or frames) that cause the contract to be violated are marked. The other nodes along the path of the data simply discard the marked cells (or frames) upon congestion. We refer to this scheme as the Delete Eligibility (DE) scheme. Although such an approach is very simple and can be made very efficient, it has some major drawbacks. One problem is insuring that a given circuit is allowed to actually make use of the contracted bandwidth. The problem arises when the algorithm interacts with windowing flow controls employed by the user's end-to-end protocol. Another problem is the inability to allocate excess bandwidth fairly among active circuits. A user will be able to grab most of the excess bandwidth along the path of its circuit simply by being more aggressive, regardless of their contracts. Lastly, upon congestion, the delay experienced by all circuits degrades significantly even for those users that are well within their contracts.

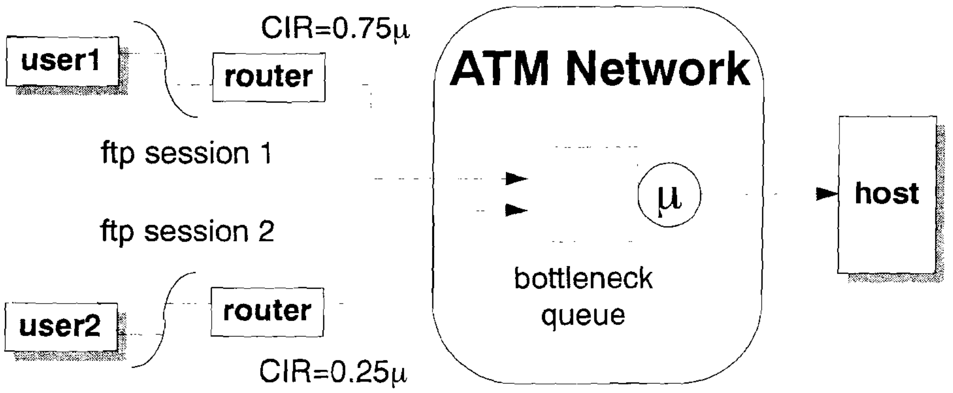


Figure 16 - Loss priority contract enforcement failure

Now we present an example that highlights the inability of loss priority based systems to enforce the bandwidth contract. Consider the two circuits sharing a link, as depicted in **Figure 16**. Each circuit has a contracted for bandwidth (CIR) of 0.75μ and 0.25μ, where μ is the link speed at the bottleneck queue. These circuits follow the same path through the ATM network and they interconnect each of the two LANs with a common host. Two users, one at each LAN, start a ftp file transfer at the same instant. The throughput of the two users will be identical and about 0.5μ each; the two circuits are symmetric since the loss threshold is never reached (it is greater than the sum of the two TCP windows). This is a situation in which user 1 is not able to make use of its contracted bandwidth while user 2 is able to exceed its contract. The contract enforcement mechanism never gets activated since congestion never really occurs. A more detailed description of these problems can be found in [Col93].

Schemes that overcome the above mentioned problems with a minimum increase of coordination among network elements have been proposed in [Tur 92], [Col93] and [Pan93]. The third proposal prescribes how to statistically multiplex unused slots among active sessions, and relies on end-to-end mechanisms for dealing with congestion. The first two proposals are similar in that a minimum amount of bandwidth and buffers is reserved along the path of a circuit. However, the first proposal does not specify how excess capacity is shared among contending users. In [Col93], this is achieved with ATM cells being served at each node (or line) according to a weighted round-robin discipline, where the service weight is proportional to the contracted bandwidth. The minimum amount of bandwidth reserved for each circuit is a consequence of the number of circuits sharing each trunk along with their service weight. Congestion at a resource (e.g., a link) is said to have occurred when the total work in the queue exceeds a predetermined global threshold. Cells in a circuit that has accumulated more work than a predetermined local threshold are discarded upon congestion. The ability of this scheme to detect and isolate the effects of congestion allows for very high efficient use of resources at the same time that it provides very strict performance guarantees. Upon congestion, users that violate their contract the most have their data dropped first. We refer to this scheme as the Global/Local Threshold (GLT) scheme.

Simulation results comparing the performance of an access control scheme and the GLT scheme under congestion are shown in Table 1. The configuration parameters are chosen so that the two schemes have identical performance in one special case. There are 24 circuits, each with 1/24 of the bandwidth guaranteed. Here, 50% of the circuits offer traffic at a rate of 0.9 times their guaranteed rate and 50% of the circuits offer traffic at a rate ten times their guaranteed rate. Although with different traffic intensities, the arrival process is identical for all circuits, with messages arriving according to a Poison process with sizes exponentially distributed with mean of ten cells. Each simulation run consisted of the transmission of 10^8 messages over the congested link.

Table 1
Loss performance comparison between the de and the glt schemes for two classes of users

Thresholds	Mean Cell Loss				Normalized Mean Cell Delay (ms)			
(G/L)	below		exceeds		below		exceeds	
	DE	GLT	DE	GLT	DE	GLT	DE	GLT
15000/60	$1.0x10^{-2}$	$3.6x10^{-4}$	$9.0x10^{-1}$	$9.0x10^{-1}$	1510	236	1508	2634
15000/40	$1.4x10^{-2}$	$8.8x10^{-1}$	$9.0x10^{-1}$	$9.0x10^{-1}$	1510	230	1508	2639
15000/40	$1.9x10^{-2}$	$2.2x10^{-3}$	$9.0x10^{-1}$	$9.0x10^{-1}$	1509	217	1507	2648
15000/30	$2.7x10^{-2}$	$5.4x10^{-3}$	$9.0x10^{-1}$	$9.0x10^{-1}$	1508	193	1506	2661

The GLT scheme provides better performance for both cell loss and cell delay for users who do not exceed their guaranteed rate. Note that, being work conserving, the overall mean cell delay and cell loss of the two systems are the same, thus the improved performance of the well-behaved users under the GLT scheme is being compensated by a degradation of performance of the ill-behaved users. Note that the extra loss incurred by the ill-behaved users on behalf of the improved loss performance for the well-behaved users is negligible. Note also that the increased delay experienced by the ill-behaved users (e.g., 2359/1509 = 1.56) is relatively smaller than the corresponding improvement in delay performance experienced by the well-behaved users (e.g., 1511/620 = 2.44). The GLT scheme provides a better package of performance tradeoffs than the DE scheme. The effectiveness of the GLT scheme is, to a great extent, due to its ability to shield well-behaved users from the effects of congestion. This allows each user to make their own tradeoffs between desired levels of efficiency and performance. Furthermore, this allows that high levels of performance be achieved even under congestion; the engineering issues are restricted to selecting threshold values enforcing a maximum number of ill-behaved users.

A bandwidth allocation algorithm at the network nodes is invaluable not only for shielding, to a reasonable extent, well-behaved users from the effects of congestion, but it also solves critical problems resulting from interactions with end-to-end windowing flow control algorithms. It is reasonable to expect that end-to-end adaptive algorithms, in general, will have a more graceful interaction with resource allocation algorithms (or congestion control schemes) that perform bandwidth allocation than with others that do not. However, significantly more work needs to be done before we have a complete picture of a flexible, efficient, general-purpose high-speed network platform.

References

[Ber92] A.Berger, W.Witt, "The Brownian approximation for rate-control throttles the G/G/1/C queue", Discrete Event Dynamic Systems: Theory and Applications, 2, pp.7-60, 1992.

[Ber93] A. Berger, W.Witt, " Asymptotics for open-loop window flow control, " preprint, 1993.

[Bol90] J.C.Bolot, A.U.Shankar, " Dynamic behavior of rate-based flow control mechanism", Computer Communication Review, 1990.

[CCI88] " Broadband aspects of ISDN, " CCITT COM XVIII Draft Recommendation 1.121, 1988.

[CCI89] " Meeting report of sub-working party 8/1 - ATM, " CCITT COM XVIII-TD. 14-E, Geneva, 1989.

[CCI89b] "Results of drafting activity on ATM adaptation layer, " CCITT COM XVIII- TD.32-E, Geneva, 1989.

[Col93] R.G. Cole, M.A.Rodrigues, "Flexible service guarantees for ATM networks, " AT&T Symposium on Performance Analysis, Holmdel, NJ, 1993.

[Eck89] A.E.Eckberg, D.T.Luan, D.M.Lucantoni, " Bandwidth management: a congestion control strategy for broadband packet networks - characterizing throughput-burstiness filter", Int. Teletraffc Congr. Specialists Seminar, Adelaide, Australia, 1989.

[Eck91] A.E.Eckberg, B.T.Doshi, R.Zoccolillo, "Controlling congestion in B-ISDN/ATM: issues and strategies", IEEE Communications Magazine, pp. 64-74, 1991.

[Elw93] A.Elwalid, D.Mitra, "Effective bandwidth of general Markovian traffic sources and admission control of high-speed networks", IEEE/ACM Transactions on Networking, to appear.

[Elw94] A.Elwalid, "Adaptive rate-based congestion control for high-speed wide area networks: stability and optimal design", preprint.

[Fen91] K.W. Fendick, D.Mitra, I.Mitrani, M.A. Rodrigues, J. B.Seery, A.Weiss, "An approach to high performance, high speed data networks, " IEEE Communications Magazine, pp. 74-82, 1991.

[Fen92] K.W.Fendick, M.A.Rodrigues, A.Weiss, "Analysis of a rate-based feedback control strategy for long-haul data transport", Performance Evaluation Journal, Special Issue on High-Speed Communications Systems, 1992.

[Fen93] K.W.Fendick, M.A.Rodrigues, "An adaptive framework for dynamic access to bandwidth at high speeds", Proc. ACM SIGCOMM, 1993.

[Fen94] K.W.Fendick, M.A.Rodrigues, "Asymptotic analysis of adaptive rate control for diverse sources with delayed feedback", to appear in the IEEE Transactions on Information Theory.

[Ger80] M.Gerla, L.Kleinrock, "Flow control: a comparative survey, " IEEE Transactions on Communications, vol. COM-28. no. 4, pp. 553-574, 1980.

[Gol90] S.J.Golestani, " A stop-and-go queueing framework for congestion management", Proc. ACM SIGCOMM, pp. 8-18, 1990.

[Gol92] S.J.Golestani, " Duration-limited statistical multiplexing of delay sensitive traffic in packet networks", Proc. IEEE INFOCOM, vol. 1, pp. 323-332, 1 990.

[Hah91] E.L.Hahne, C.R.Kalmanek, S.P.Morgan, " Fairness and congestion control on a large ATM data network with dynamically adjustable windows", 13th International Teletraffc Conference, 1991.

[Hal77] J. Hale, Theory of Functional Differential Equations, Springer Verlag, 1977.

[Hui88] J.Y. Hui, " Resource allocation for broadband networks, " IEEE Journal on Selected Areas in Communications, vol. 6, pp.1598-1608, 1988.

[Hym91] J.M.Hyman, A.A.Lazar, G.Pacifici, "Real time scheduling with quality of service constraints", IEEE Joumal on Selected Areas in Communications, pp. 1052-1063, 1991.

[ISO91] ISO 10589.

[Jai86] R.Jain, " A timeout-based congestion control scheme for window flow-controlled networks", IEEE Journal on Selected Areas in Communications, Vol. SAC_4, No. 7, 1996.

[Jac88] V.Jacobson, "Congestion avoidance and control, " Proc. ACM SIGCOMM, pp. 314-329, 1988.

[Kes91] S.Keshav, " Congestion control in computer networks, " Dissertation, University of California at Berkeley, 1991.

[Lua88] D.T.Luan, D.M.Lucantoni, "Throughput analysis of a window-based flow control subject to bandwidth management", Proc. IEEE INFOCOM, 1988.

[Max90] N.F.Maxemchuck, M.E.Zarki, "Routing and flow control in high speed wide area networks", Proc. IEEE, vol. 78, no. 1, pp. 204-221, 1990.

[Mis92] P.P.Mishra, H.Kanakia, " A hop by hop rate-based congestion control scheme", preprint.

[Mit90] D.Mitra, " Optimal design of windows for high-speed data networks, " Proc. IEEE INFOCOM, pp. 1156-1163, 1990.

[Mit91] D.Mitra, J.B.Seery, "Dynamic adaptive windows for high-speed data networks with multiple paths and propagation delays", Proc. IEEE INFOMCOM, pp. 39-48, 1991.

[Mit92] D.Mitra, "Asymptotically optimal design of congestion control for high speed data networks", IEEE Trasactions on Communications, 40, No 2, pp. 301-311, 1992.

[Moy91] John Moy, "The OSPF specification, version 2, " IETF Draft, 1991.

[Muk91] A.Mukherjee, J.Strikwerda, "Analysis of dynamic congestion control protocols - a Fokker-Plank approximation", Proc. ACM SIGCOMM, 1991.

[Pan89] S.S.Panwar, T.K.Philips, M.S.Chen, " Golden ratio scheduling for low delay flow control in computer networks", Proc. Globecom, 1989.

[Pan93] S.S.Panwar, T.K.Philips, M.S.Chen, " Golden ratio scheduling for flow control with low buffer requirements", IEEE Transactions on Communications, to appear, 1993.

[Ram88] K.K.Ramakrishnan, R.Jain, " A binary feedback control scheme for congestion avoidance in computer networks with a connectionless network layer", Proc. ACM SIGCOMM, 1988.

[Rod90] M.A.Rodrigues, " Erasure Nodes: performance improvements for the IEEE-802.6 MAN", Proc. IEEE INFOCOM, 1990.

[Tur86] J.S.Turner, " New directions in communications (or which way to the information age?)", IEEE Communications Magazine, 1986.

[Tur92] J.S.Turner, " Managing bandwidth in ATM networks with bursty traffic", IEEE Network Magazine, september 1992.

[Woo90] G.M.Woodruff, R. Kositpaiboon, " Multimedia traffic management principles for guaranteed ATM network performance", IEEE Journal on Selected Areas in Communications, vol. 8, 1990.

[Zha87] L.Zhang, " Designing a new architecture for packet switching communication networks", IEEE Communication Magazine, pp.5-12, 1987.

Manoel Alberto Rodrigues Neto: Nasceu no Rio de Janeiro, RJ, em 17 de janeiro de 1959. Concluiu o curso de Engenharia Elétrica na Universidade Federal do Rio de Janeiro em 1981, o Mestrado e o Doutorado, ambos em Engenharia Elátrica na universidade da Califórnia, San Diego, de 1984 a 1987. Foi membro do corpo técnico na AT&T Bell Labs de 1987 a 1993. Está atualmente com a OTL Odebrecht Automação e Telecomunicações Ltda, onde é responsável por Tecnologia de Telecomunicações. Suas áreas de interesse incluem: arquitetura de redes de dados, integração de serviços digitais, análise de desempenho de algorítmos e sistemas, e otimização de roteamento.