digital

Networks

Frequently Asked Questions on ATM and Digital's ATM Program

This paper focuses on how ATM is actually used to solve customer problems, and on current technical topics of importance to decision makers evaluating ATM products for use in their private_networks. It also explains key aspects of Digital Equipment Corporation's ATM program, strategy and current products. Sections of this paper can be read in any order if the reader has a basic familiarity with ATM concepts. For readers not familiar with ATM, it is recommended that the document be read in the order in which it has been presented and be supplemented with introductory ATM materials.

For more complete information on Digital's ATM products including GIGAswitch/ATM system, please refer to Digital's home page at:

http://www.networks. digital.com
http://www.networks.europe.digital.com

Digital Equipment Corporation
Network Product Business

June, 1996

EC-Y6194-42

This Page Intentionally Left Blank

Table of Contents

Figures and Tables

1. Public vs. Private ATM Networks

1.1. *What are the differences between ATM switches used by carriers vs. those used in private LANs and WANs?*

Switches for public carriers are much higher priced ($1 million and up) and provide much higher capacity (20 to 100 Gbps), than private ATM switches. Private ATM switches range in price typically from $20,000 to $300,000 with capacities in the 1 Gbps to 10 Gbps range because they must be competitive with other options for private networks such as FDDI and Fast Ethernet switches, backbone routers, and time division multiplexers.

There are also many feature differences. How they handle bursty traffic is one area of particular importance.

Carrier switches are not usually designed for efficiently handling bursty data traffic, such as that which arises from direct connections to individual workstations, servers or hosts. Carrier ATM switches work best when traffic from many sources is aggregated and shaped into an overall load that varies modestly within a predictable range. Many were designed specifically for voice or related services with constant data flows.

Likewise carriers' ATM services are tailored to this type of environment where customers specify their required capacity and obtain a **CBR (Constant Bit Rate)** service charged on a monthly basis similar to a private line. Sometimes it is more desirable (depending on whether the traffic is *loss sensitive*) to use a **VBR** (**Variable Bit Rate**) service. With VBR the customer specifies an expected **SCR** (**Sustained Cell Rate**) and desired **PCR** (**Peak Cell Rate**) and is charged according to a formula similar to Frame Relay services, with the risk that traffic in excess of the SCR may be discarded.

ATM switches for private LANs and WANs need to be able to handle bursty, unpredictable traffic loads (as is common on private computer networks), while also supporting CBR services for voice, video and other real-time traffic. Bursty, unpredictable computer traffic is best handled by **ABR** (**Available Bit Rate**) services. ATM switches for Private LANs should have well-developed ABR support.

VBR services are generally not needed in a private ATM network if it has good ABR and CBR support. (See Section 4: *Categories of Service*).

Other important differences are in the support and use of Virtual Paths (VP) vs. Virtual Circuits (VC), and in the use of Permanent Virtual Circuits (PVCs) and Switched Virtual Circuits (SVCs).

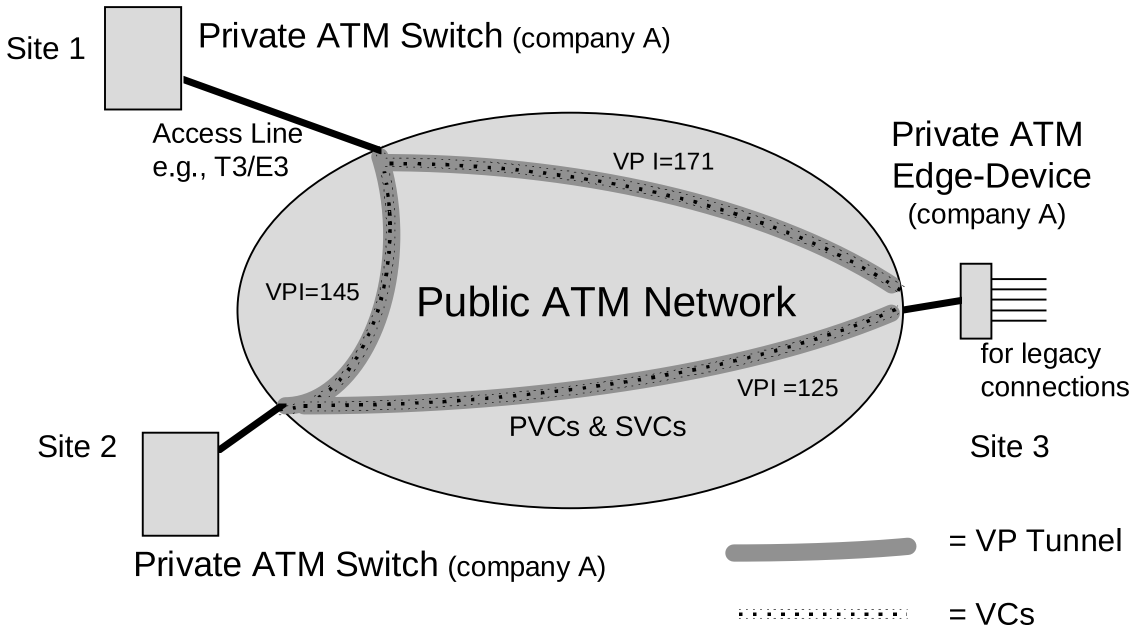
1.2. *How are VPs and VCs used in an ATM environment?*

A Virtual Path (VP) is like a pipe or a tunnel that can carry many Virtual Circuits (VCs) **--** up to 65,000. It may carry these only from switch to switch or all the way across the ATM network end-to-end. In all but the largest private LAN or WAN, 65,000 VCs per physical link are plenty today and support for multiple VPs is not really needed. Many ATM LAN vendors only support one VP, namely VPI =0. When only one VP is supported, it is not used end-to-end, so there is no constraint on whether the VCs stay within a given VP. This allows VCs to connect any set of stations regardless of VP. Data is always carried in a VC.

Carrier switches, on the other hand, must typically have support for hundreds or thousands of different VPs, and potentially millions of different VCs. Often carriers want to be able to offer their customers a dedicated VP end-to-end across their network between any two sites on a customer's network. When VPs are used this way it is called a **Virtual Path Connection (VPC)** or more naturally a "**VP Tunnel"**. It may carry both **Permanent Virtual Circuits (PVCs)** and **Switched Virtual Circuits (SVCs). See Figure 1**.

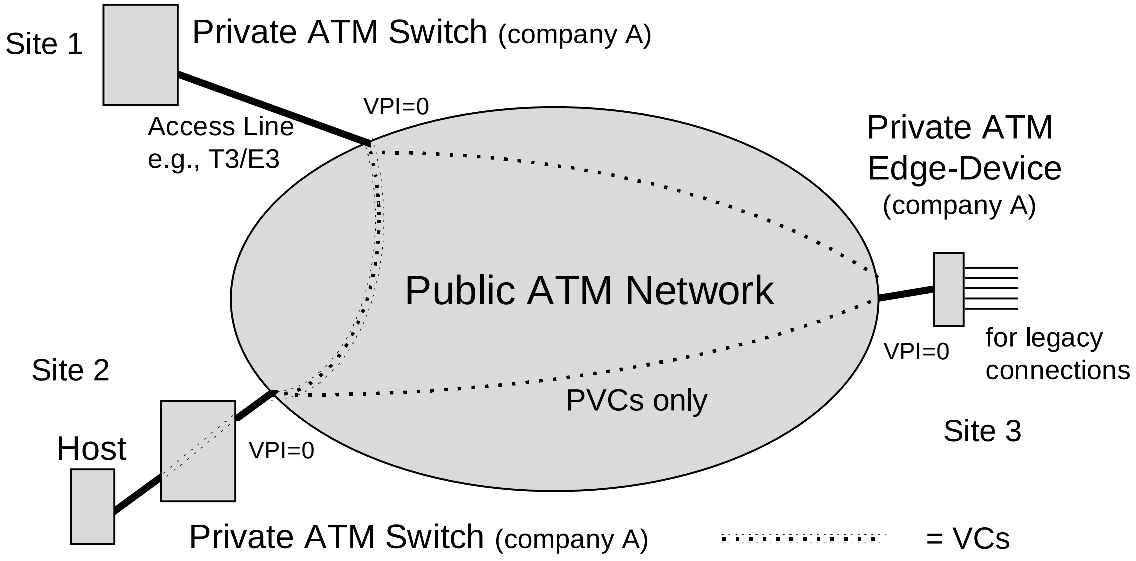
Inside a VP Tunnel, the carrier's customer can establish PVCs and SVCs without the carrier having to get involved in any way. Furthermore, the carrier's switch need not support the routing of SVCs since the path is alread determined. The VP Tunnel also provides a way to isolate different companies from each other on the public network. When using public ATM services this way, multiple VP Tunnels are required to interconnect the sites on a customer's network, to whatever degree is desired. Figure 1 shows two VP Tunnels per site in a three-site network, all for the same company. Other companies sharing this same public network would have separate VP tunnels.

Figure 1
Public ATM Network with VP Tunnels



In a public ATM network environment, if the carrier does not offer VP Tunneling capabilities (and some may not), then the carrier can offer only PVCs. This is because most carrier switches cannot support the routing of SVCs directly (and some may never) and because many carriers do not wish to support SVCs (it complicates billing and may raise issues of data security between companies). Without VP Tunnels, carriers typically use VPI=0 at the end-points of their network for originating and terminating PVCs. **See Figure 2**.

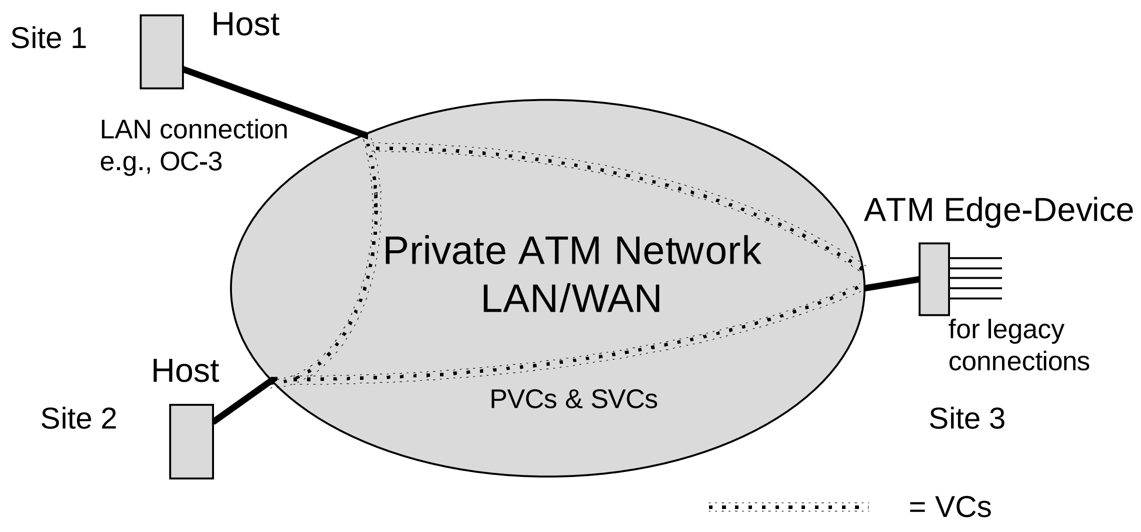
Figure 2
Public ATM Network without VP Tunnels



The PVCs in a public network are set-up by the carrier, requested in advance by the customer. Such PVCs are particularly useful for connections out to remote **"ATM edge-devices"** (e.g., routers, Ethernet or FDDI switches with ATM ports, ATM concentrators), that dynamically multiplex many non-ATM traffic sources over a single PVC back to, say, a headquarters location. Using a PVC between ATM hosts also limits communication to predetermined end-points. This is may be desirable on a public network.

Within a private network (LAN or WAN), SVCs are the preferred way to communicate between stations, since SVCs can be set-up on demand by the end-stations themselves. This is how most private non-ATM LANs and WANs work today. Therefore, private network ATM switches must support SVCs directly. PVCs are also sometimes used in a private network when an end-station or edge-device does not support SVCs or should not be allowed to request connections on demand. PVCs must be set-up by the network administrator in advance and are less robust than SVCs in the event of network element failure, since their path is predetermined. Within private ATM networks VP Tunnels are much less important or even unnecessary. **See Figure 3.**

Figure 3
Private ATM Net (no VP Tunnels)



1.3. *Do I have to use a public ATM carrier to connect ATM LANs?*

Absolutely not. Private ATM LAN switches and edge-devices can also be directly interconnected by "plain old" dedicated digital lines (e.g., T1/E1, T3/E3, or OC-3c if available) obtained from a carrier or privately constructed. This avoids all complications regarding what features are supported by the public ATM carrier. This offers a very straight forward way to build a fully-integrated private ATM LAN/WAN with consistent SVC support (and ABR support—see Section 4) throughout your private network. This option is portrayed (implicitly) in Figure 3, although the private ATM switches and dedicated WAN lines are not shown.

The main reason to use public ATM services is that the bandwidth costs for WAN connectivity can be lower (depending on your particular configuration and traffic patterns), but not always. Generally, public ATM services are cost-effective only for connecting many remote small sites into a central site or private WAN backbone network (as with Frame Relay). If there is a high concentration of traffic between two sites in a private network, then those sites may be more cost-effectively served via direct dedicated lines.

1.4. *Can I connect to a public ATM service using VPI=0?*

Yes. When VPI (Virtual Path Identifier) is equal to 0, there may be some restrictions on its use, depending on the public ATM service provider. Most carriers do not allow a VP Tunnel (VP Connection) to be established using VPI=0 at the end points since it is used for connections that don't involve VP Tunnels. This limits connectivity options to using individual PVCs between sites, as in Figure 2. Some carriers are working on ways to allow VP Tunnel support using VPI=0 due to the large number of private networking ATM switches and devices on the market that only support VPI=0.

1.5. *When is "VP Switching" used?*

VP switching is used by carriers to set-up entire VP Tunnels or to change the termination end-points of an existing VP Tunnel. This is something that would not be needed normally in a private LAN or WAN since VP tunnels themselves are not needed usually. Public ATM service providers, however, need to have this ability, at least to set-up new VP Tunnels for new customers.

1.6. What support do Digital's ATM products have for VCs and VPs?

Digital's ATM products today support both PVCs and SVCs. Each linecard in the GIGAswitch/ATM, for instance, supports over 16,000 VCs (4095 VCs per port) which can connect to any location on the network. When fully configured with 13 line cards per switch (52 ports at 155 Mbps each), GIGAswitch/ATM hardware supports over 200,000 VCs. (Current firmware support only about 4000 active VCs, but future firmware releases will allow nearly this many VCs per switch).

This allows for the formation of very large networks with many attached devices. However, only one VP (VPI=0) is currently supported in our ATM products. This may limit their use with public ATM services to PVC connections, unless the service provider allows VP Tunnels on VPI=0. Future versions of our ATM products will have support for multiple VPs (in addition to VPI=0) so that our switches and edge devices can connect to several concurrent VP Tunnels over a single interface line to the public ATM service.

In the future, GIGAswitch/ATM will support both VP Tunnels and VP switching, enabling other ATM equipment to tunnel SVCs and PVCs through a network of GIGAswitch/ATM systems.

2. ATM Switch Design for Private Networks

2.1. *What architectures are used in ATM switches for private networks?*

The major architectural approaches used today for the switching fabric in commercial ATM switches for private networks are: **cross-bar matrix**; **shared bus**; and **shared memory**; or some a hybrid of these. In cross-bar and bus-oriented switches, further important differences exist with regard to where the bulk of cell buffers are located. They can be either **input buffered** or **output buffered**, or sometimes a little of both. In a shared memory switch the cell buffers are, by definition, in the central shared memory between the input and output ports.

Any of these three different switch architectures can produce a well-performing switch for ATM private networks supporting both traditional bursty data traffic and real-time multimedia traffic.

There is one other type of switch design sometimes used for commercial ATM switches called **"Batcher-Banyan".** This design was developed originally for high-end voice circuit switches. It has proven to be very poor for ATM networks that require low cell loss, such as in data networking environments. Batcher-Banyan switches suffer from noticeable cell loss even under light loads. This cell loss is due to the very small buffers used at each junction of the switch fabric. The buffers can easily overflow. This wreaks havoc in loss-sensitive data networks causing many re-transmissions. (Despite these problems there are still vendors marketing switches based on Batcher-Banyan technology.)

2.2. *What design considerations determine overall ATM network performance?*

Within the commonly-used switch architectures, the buffer size and buffer management techniques plus flow control and traffic management techniques, are usually much more important in determining overall switch and network quality than the type of switch fabric used (with the notable exception of Batcher-Banyan switching fabric—see Question 2.1).

Key aspects of quality and performance which these design choices can affect are:

- ability to handle bursty data traffic without cell loss
- guaranteeing fairness across competing VCs
- providing maximum utilization of network bandwidth (particularly expensive WAN links)
- providing low cell-delay variation and traffic shaping for real-time CBR traffic
- ensuring network stability under heavy loads and minimizing the impact of misbehaving traffic sources.

Buffer size, buffer management, and flow control are particularly important for providing robust ABR support. However, ABR support requires cooperation by both the switch and by the end-station adapters and edge-devices used in the network in order to avoid cell loss. (See Section 6: *Flow Control and Traffic Management).*

Other key factors for the network, often overlooked, are the overall performance of the ATM adapters (or NICs) and the computer systems themselves. Many ATM adapters are not really designed to handle a sustained data transfer at full line rate, especially 155 Mbps. Many deliver only a small fraction of this rate. In some cases this does not matter since the computer systems in which they are used cannot transfer data at these line speeds for a sustained period. A typical PC today can only handle sustained rates of 30 Mbps or less (regardless of LAN technology used). Many high-end PCs can handle only 40 to 80 Mbps of sustained network I/O (in or out). These limits are typically related to operating system constraints.

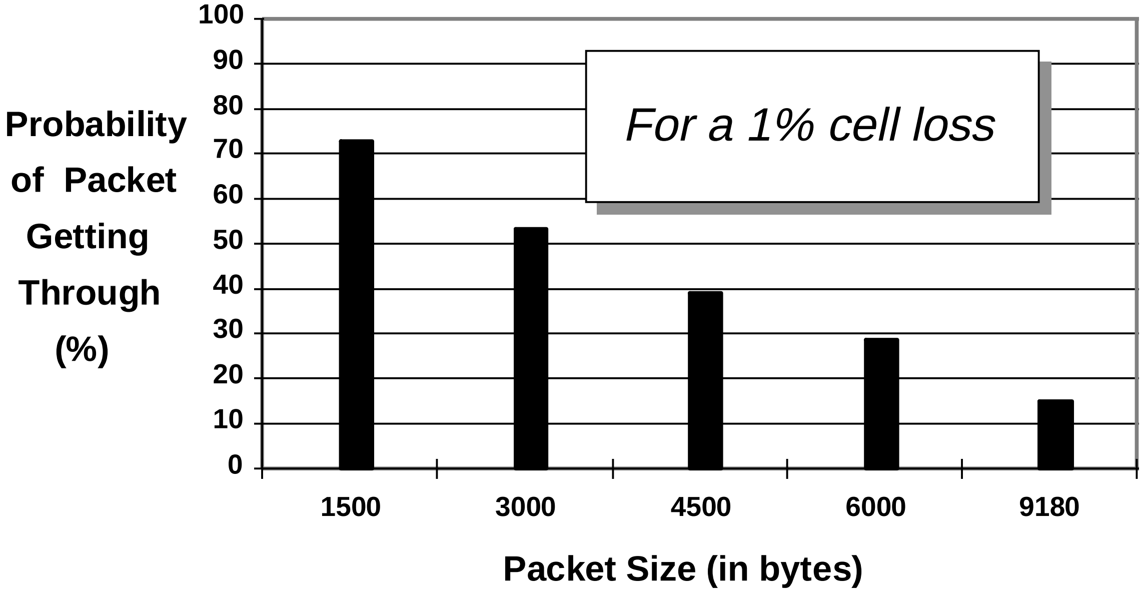
However, RISC UNIX workstations and servers can typically support much higher sustained throughputs on a 155 Mbps link, at or approaching 135 Mbps of packetized data, which is the full line rate after factoring out SONET (Synchronous Optical NETwork) overhead and ATM cell overhead of 5 bytes per 53 byte cell. Equipping these systems with high-performance adapters is essential to realizing the benefits of ATM. Digital provides two ATM adapters for PCI and TURBOchannel bus workstations and servers: the

ATMworks 350 and ATMworks 750 adapters, respectively. These are the only ATM adapters with throughput >130 Mbps, that can achieve such throughput on a sustained basis. As more higher performance PCs and PC operating systems enter the market, they too will be demanding higher-performance ATM adapters to achieve >100 Mbps throughput.

2.3. *Why is eliminating cell loss such a concern in ATM networks?*

Due to ATM's small cell size (53 bytes with at most 48 bytes of user data), a single cell is usually carrying only a small part of a much larger "chunk" of data. Ethernet frames, IP packets and other commonly used "chunks" in data networks must be carried by many, many cells logically associated together via the ATM Adaptation Layer (e.g., AAL-5). Even with a Cell Loss Rate (CLR) of just 1%, the packet loss rate can be much higher. **See Figure 4**.

Figure 4
Impact of Cell Loss in ATM (by packet size)

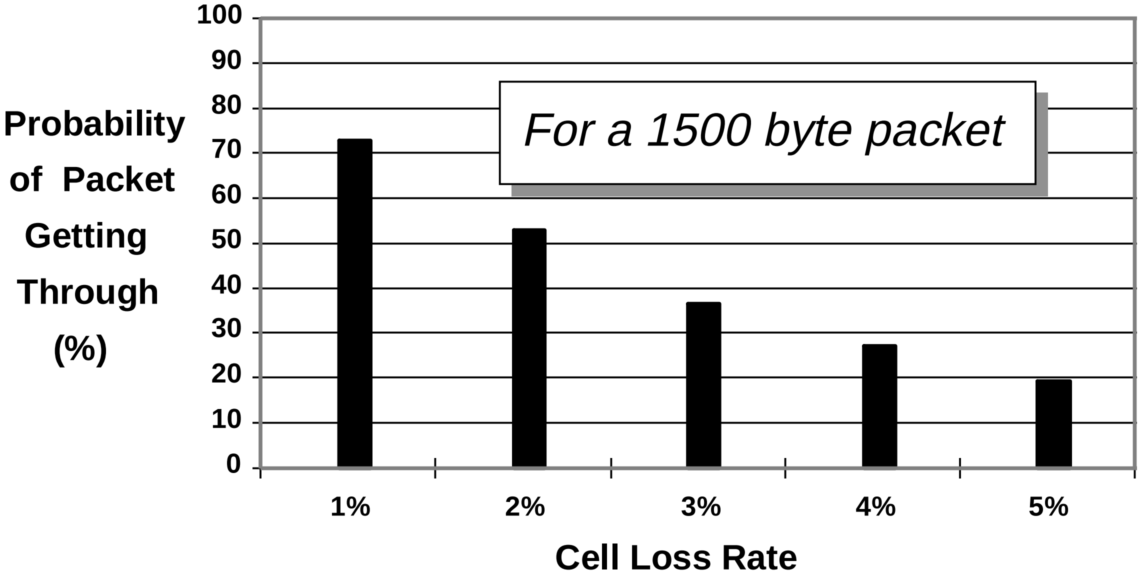


When using Classical IP (the Internet Protocol) over ATM, packet sizes may be as large as 9180 bytes requiring 192 cells to transfer it. A random cell loss rate of 1% results in just a 15% chance of getting the packet through (or 85% of packets lost). Smaller packets fair better, but not well at a 1% CLR.

As the cell loss rate goes up things get worse fast. With even much smaller 1500 byte packets (a common size on Ethernet LANs and over ATM LAN Emulation), a cell loss rate of 5% can result in more than 80% of the packets being lost. **See Figure 5.**

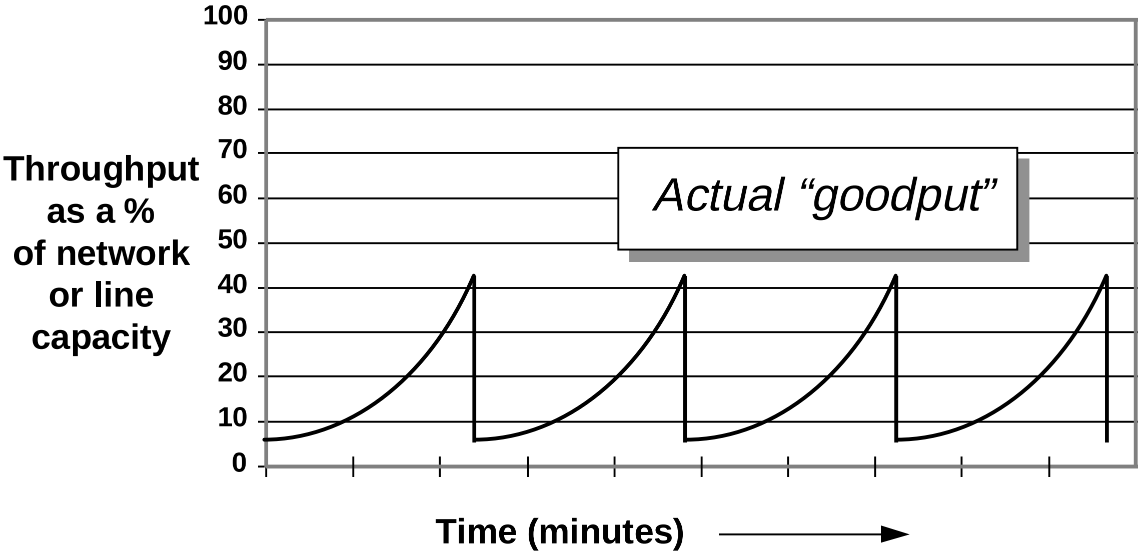
When even one cell is discarded due to congestion, all the other cells associated with that packet are now worthless, but may still be present in the network, causing congestion elsewhere. Higher layer protocols (e.g., TCP) will now re-transmit new packets for those damaged packets, resulting in 192 more cells per packet on the network for the Classical IP example above or 32 more cells in the case of a 1500 byte packet. This only makes a bad situation worse. Typically protocols like TCP will re-transmit their entire "window size" of outstanding packets (often 8 or more packets) if any one packet is damaged or lost, making matters even worse.

Figure 5
Impact of Cell Loss in ATM (by CLR)



When re-transmissions begin, congestion is very likely to get worse fast, unless by chance most of the other sources of data have suddenly gone quiet. When congestion gets worse, more cell loss will occur, causing more re-transmissions. This cycle can quickly snowball. While there are a lot of cells being sent, nothing usable is getting through; **"goodput"** has gone down to almost zero in what is called a **throughput collapse**. **See Figure 6**.

Figure 6
Throughput Collapse



The network conditions improve only after connections have timed-out and traffic stops (and users get angry). As traffic builds again, another collapse is likely unless some users have given up and gone home. This cycle of instability has been observed many times in ATM networks operated without effective flow control.

Clearly cell loss is to be avoided in ATM networks. Fortunately the high quality of digital transmission lines, especially fiber, prevents the lines from being a significant source of cell loss. By far the greatest cause of cell loss is congestion when combined with lack of good flow control. But with good flow control, plus good buffer and traffic management, cell loss can be essentially eliminated. (See Section 6: *Flow Control and Traffic Management*).

2.4. What are the key factors affecting switch capacity, scalability and cost?

The cost of ATM switches is determined by a number of factors. Obviously the total throughput of the switch and whether the switch is non-blocking are key factors. Other important factors are:

- Actual number and types of ports configured and supported on the switch (i.e., scalability)
- Flexibility in configuring those ports (e.g., Singlemode Fiber, Multimode Fiber, Unshielded Twisted Pair)
- Amount of memory provided for cell buffers
- Amount of CPU processing power and processor memory available for SVC set-ups and other management and value-added tasks (e.g., LAN Emulation)
- Amount of redundancy built into the switch
- Degree of integration in silicon (e.g., custom ASICs) used.

The switch architecture is often the determining factor in overall throughput capacity, scalability and blocking characteristics of the switch.

Cross bar technology lends itself to high throughput (~3 to ~20 Gbps), very scaleable, non-blocking switches. This is because multiple simultaneous paths are created between switch fabric input ports and output ports as more linecards (and thus user ports) are added.

Shared bus switches are good for mid-range throughput (~1 to ~3 Gbps), but may suffer from some scalability problems and blocking depending on the particular implementation. This is due to the fact that there is a single bus serving all ports. As ports are added, the bus capacity does not grow. Bus-oriented switches will suffer from blocking if the available capacity of the bus (which equals bus speed times bus width less overhead from bus arbitration and other factors) is not greater than the sum of the speeds of all the ports.

Memory-based switches can be very cost-effective for small switches with low-end throughput (~1 to ~2 Gbps). They too are bounded in their performance and typically come in fixed-port configurations. Memory-based switches can suffer from blocking too, if the speed of the memory I/O to the ports is not twice the sum of the speeds of all the ports.

Since there is one central buffer pool, memory-based switches tend to allow for very simple designs that also require less overall memory for buffers. This allows for a very low-cost switch (since memory is a large part of switch costs in all design approaches), yet with sophisticated buffer management techniques.

In terms of cost, high throughput cross-bar switches tend to be the most expensive switches followed by lower capacity shared bus switches. Memory-based switches tend to have the lowest cost *Cost per unit of performance* though can be a different story. Depending on the other factors detailed above, cross-bar switches can offer very low cost per unit of throughput, comparable or less than the other types of switches, when configured near capacity. They also allow customers to "pay as you grow", since most of the cost is in the linecards, rather than the switch fabric. Cross-bar switches clearly lend themselves to backbone applications or very high throughput workgroups (e.g., technical computing environments). Bus and memory-based switches are fine for small workgroup applications or as concentration points into the ATM backbone.

2.5. Why did Digital implement the GIGAswitch/ATM using cross-bar technology?

We chose cross-bar technology because we wanted an ATM switch that would scale with the growth in the ATM market and our customers' use of ATM over the next five years. Also, cross-bar switches can be designed with much better performance for "bursty" data traffic (e.g., LAN traffic), producing zero cell loss, than multi-stage switch fabrics or shared bus designs. This consideration was a primary factor in our decision.

We believed we could produce a very cost-effective cross-bar switch, and in fact, we did. The GIGAswitch/ATM has the best price/performance ratio on the market today, and we have plans in place that will enable us to maintain that lead over the next several years using the same core switching fabric sold in our GIGAswitch/ATM platform today, even with the significant decline in price of ATM technology expected over the next few years.

2.6. Some consultants claim cross-bar switches do not scale well. What do they mean?

They mean simply that the complexity of a cross-bar switch design increases by the square of the number of entry points into the cross bar matrix. Thus, it is difficult to design and build very large cross-bar switches (e.g., one with thousands of entry points).

What is missing from this simple observation, though, is the fact that today ATM cross-bar switches provide higher capacity (e.g., 10.4 Gbps for the GIGAswitch/ATM) than either shared bus or memory-based switches today. Cross-bar switches (also sometimes called "space division" switches) continue to be the preferred means of building high-throughput ATM switches that meet both the low-cell loss requirements of bursty data environments and the need for real-time CBR traffic.

Also missed is the fact that most cross bar switches multiplex many user ports into a single cross-bar "entry point". Thus, cross bar switches with hundreds of user ports are easily implemented today, where as shared bus and memory-based switches typically top out at 16 to 32 ports maximum (due to complexities of bus and memory access arbitration).

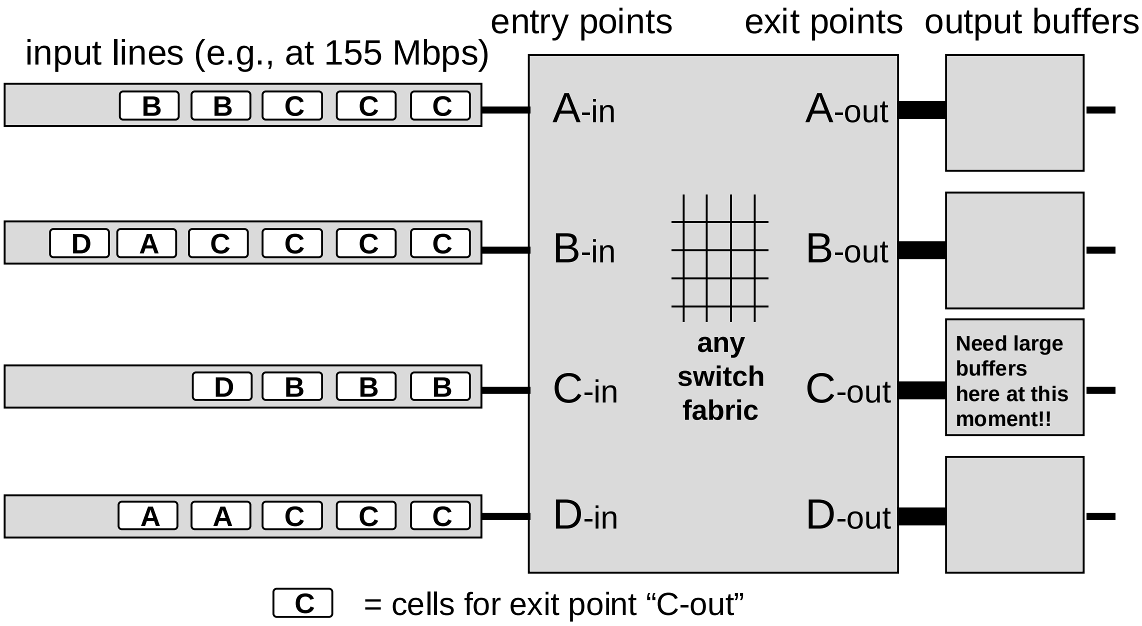
The GIGAswitch/ATM today multiplexes 4 user ports at 155 Mbps into each entry point of a 13x13 cross-bar matrix, providing 52 ports. Future linecards for the GIGAswitch/ATM platform can be designed to support 8 or more ports each, more than doubling the total user port density, and still use the same switch fabric that we ship today.

2.7. What are the important differences between input vs. output buffered switches?

Input buffered switches capture a cell as it enters the switch and only send the cell through the switch fabric if the output port to which the cell is going is free (i.e., no other input is sending to that output at that moment). Output buffered switches send cells across the switch fabric soon after they enter the switch regardless of the status of the output port.

While these two approaches do not seem to be that different, they have a big impact on buffer sizing and can have an impact on overall switch performance and stability, if not properly handled. This means that the buffers on output buffered switches have to be larger and faster than those on input buffered switches since it is highly likely that multiple inputs to the switch fabric can be sending cells to the same output port at the same time. **See Figure 7.**

Figure 7
Congestion in Output Buffered Switches



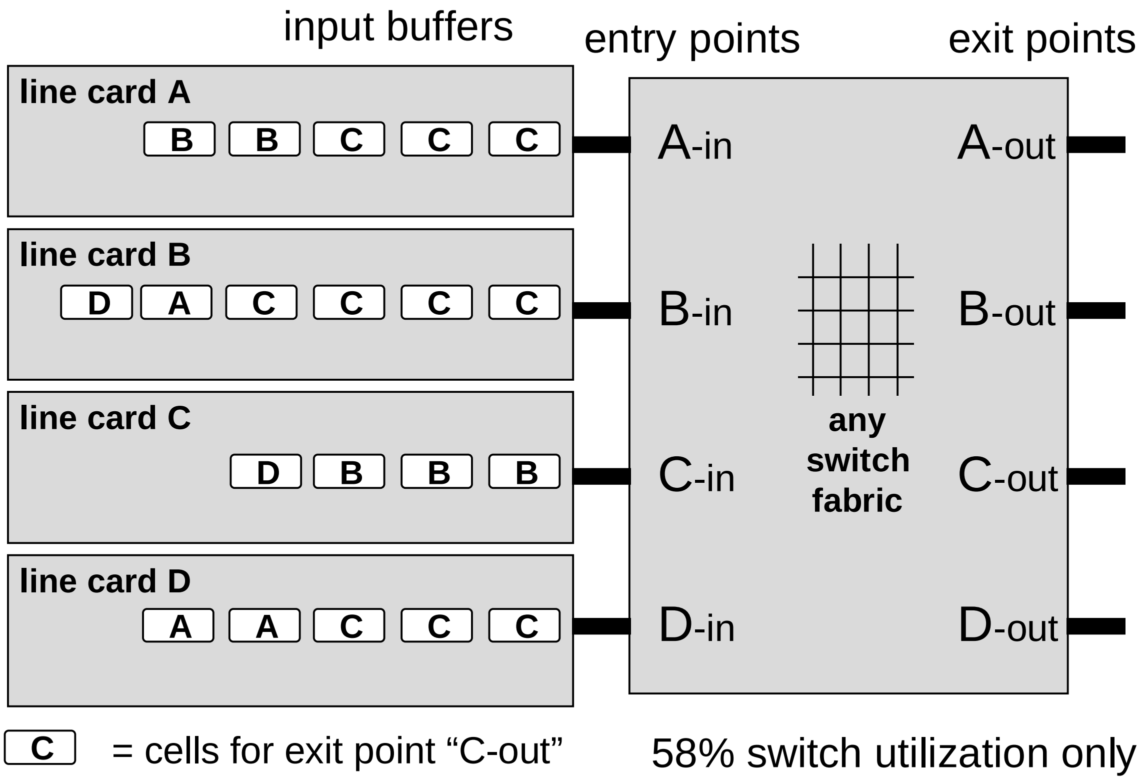
For example, on a 16 port switch it is quite possible that 6 input ports will be sending to the same output port during the same 10 millisecond interval. OC-3c 155 Mbps lines transmit at a rate of 353,000 cells per second (factoring out the SONET overhead). In a matter of 10 milliseconds, it is possible to have 21,180 cells (= 6 x 3530) attempting to go out through the same output port. Since only 3530 can actually be sent out the port in 10 milliseconds, a total of 17650 (= 5 x 3530) cells will have to be buffered at the output port in this example. If the output port buffers are not large enough, cell loss will occur.

In this same scenario an input buffered switch need only have the ability to support 17650/6 = 2941 cells per port in its buffers for that instant. Of course, good flow control mechanisms can help solve this problem without requiring such large buffers, but assume for a moment that no flow control is used, or that the lag time for the flow control to kick-in is more than 10 milliseconds.

Cell loss is a more likely occurrence on output buffered switches under conditions of even moderate load. Also, when cell loss occurs, it can the affect the VCs on many input ports on that switch all at once, (namely all input ports with VCs passing through the congested output port). This could necessitate re-transmissions (by higher layer protocols) for all the VCs affected. This exacerbates the likelihood of **"throughput collapse"**, since many more VCs are affected all at once. *In fact, this problem has been shown to exist even in one of the most widely used output buffered ATM switches available on the market today. Despite its relatively large cell buffers for the output ports (13,000 cells per four port linecard), a single unruly VC can ruin things for every other VC on the switch.*

On the other hand, input buffered switches have their challenges too. Since input buffered switches only send a cell across the switch fabric when the output port for that cell is free, the cells which are deeper in the input queues could be forced to wait unnecessarily even if their destination output port is free. This condition is known as **head of line blocking. See Figure 8.**

Figure 8
Head of Line Blocking



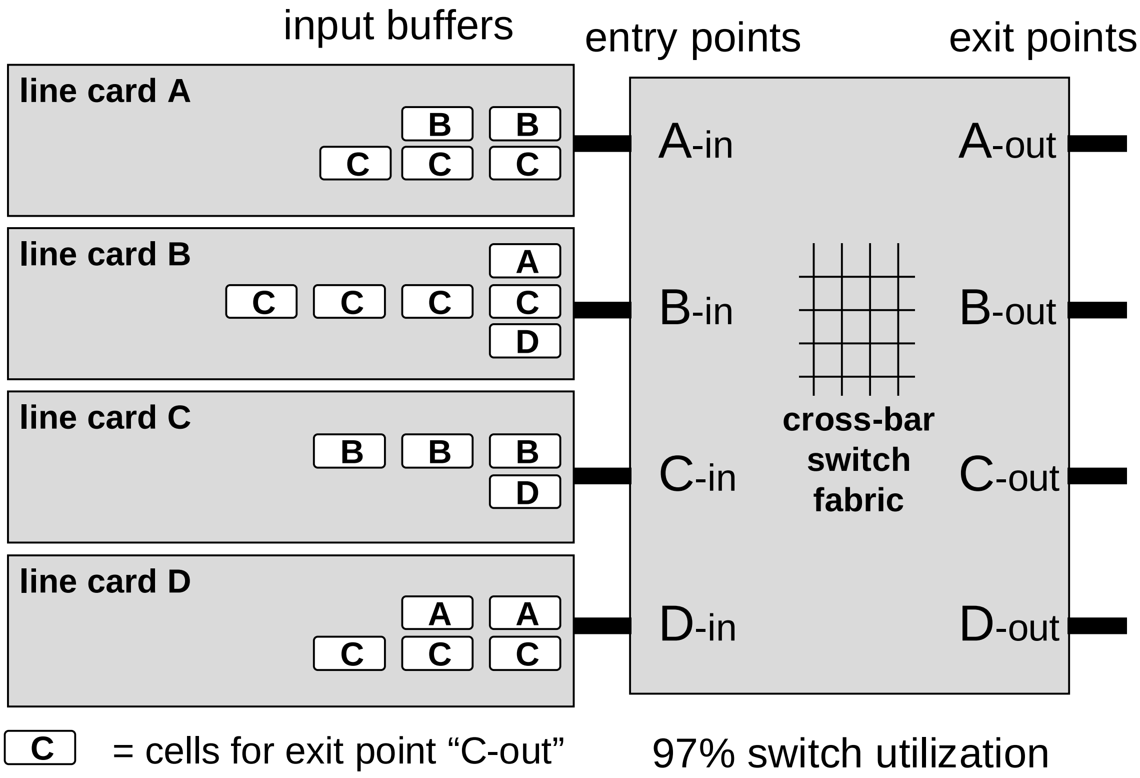
Head of line blocking can reduce a large switch's effective throughput to 58% of its theoretical maximum for a uniformly random distributions of cell destinations. It can also increase latency and cause cell loss if the input buffers are overrun. Any cell loss caused by overrunning input buffers, however, would be limited to VCs on that port alone, containing the impact to a much smaller part of the network. That is one major advantage of input buffering.

The second major advantage of input buffering is that the size of the buffers do not have to be as large as those on an output buffered switch, as discussed above, which allows the switch to be less expensive. The third major advantage of input buffered switches is that it is much easier to implement effective flow control mechanisms to assure zero cell loss, since knowledge of buffer status is local to the port that needs to communicate back to the traffic source—namely the input port.

2.8. *What is Digital's buffer management strategy for the GIGAswitch/ATM?*

The GIGAswitch/ATM product is an input buffered switch. The head of line blocking problem sometimes associated with input buffered switches has been solved by using a sophisticated algorithm called "parallel iterative matching". The cells in the input buffers to the cross-bar matrix are organized according to the output port to which they are destined. Therefore no cell waits behind a cell destined to another crossbar port. (These "per output port queues" may contain cells for many different VCs). Digital has patented this algorithm and markets it as **SWITCHmaster** advanced queue management. The SWITCHmaster algorithm is included as an integral part of GIGAswitch/ATM system. **See Figure 9.**

Figure 9
SWITCHmaster ™

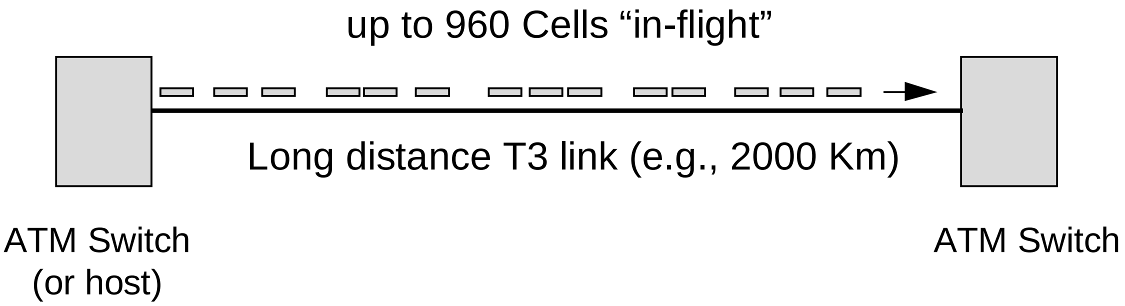


Lab and customer tests of the effectiveness of SWITCHmaster have shown that under random load conditions a utilization of 97% of the switch fabric is achieved. Therefore a full 10 Gbps of throughput can be obtained with GIGAswitch/ATM. Zero cell loss is achieved in the GIGAswitch/ATM by providing deep input buffers and implementing flow control and traffic management mechanisms. (See Section 6: *Flow Control and Traffic Management*).

The buffers in the GIGAswitch/ATM provide approximately 8,000 cells per linecard, which can be shared by ports on the linecard (up to 4000 cells per port). For applications with long distance links (e.g., T3/E3 lines running cross-country), a buffer upgrade will be offered which more than triples the buffer capacity to over 30,000 cells per linecard, or a maximum of 15,000 cells per port.

Larger buffers are needed on long distance links due to "flight time" latencies caused by the finite speed of light (about 5 microseconds per kilometer). A T3 line (45 Mbps) can send 96,000 cells per second (after accounting for T3 framing overhead). On a 2,000 km link, 960 cells could be in "flight" in just one direction on the link. Only with sufficiently large buffers and good flow control can both zero cell loss and full line-rate throughput on high-speed long distance links be maintained. **See Figure 10.**

Figure 10
ATM over Long Distances

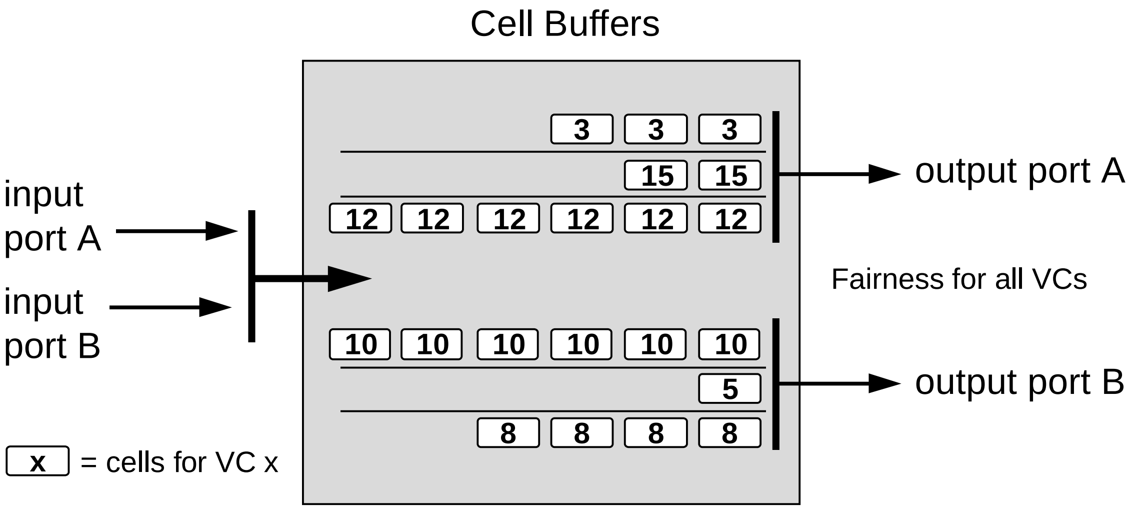

up to 960 Cells "in-flight"
Long distance T3 link (e.g., 2000 Km)
ATM Switch
(or host)
ATM Switch

In addition to the "per output port queues" of SWITCHmaster, separate queues or buffers are established for each VC in the GIGAswitch/ATM. This is called "***Per VC Buffering***" and is another essential feature of providing good traffic management. Also provided is ***Per VC Buffer Accounting.***

2.9. What is "Per VC Buffering", and why is it important?

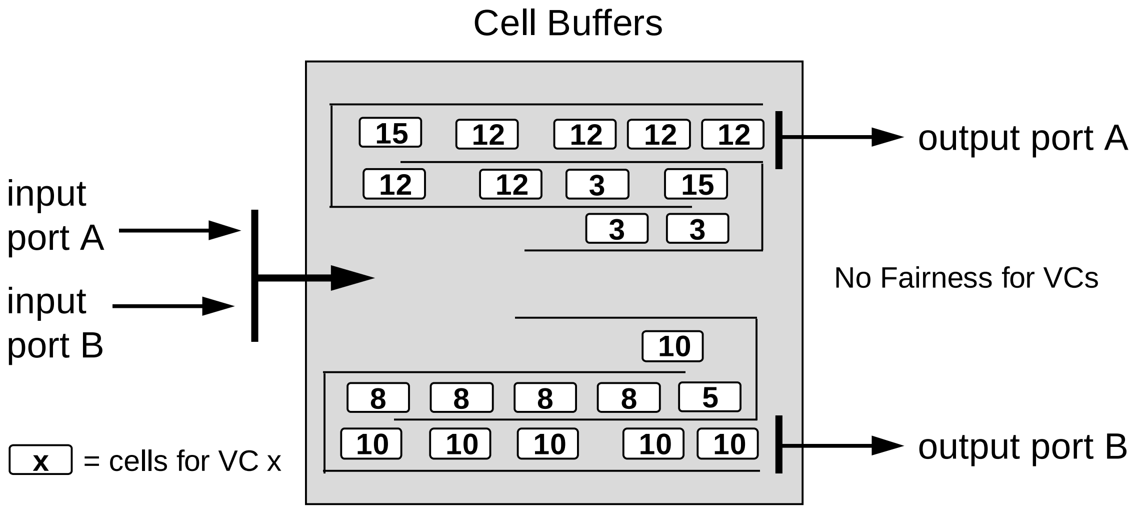
Per VC Buffering is a buffer management technique to assure **fairness** across all VCs in a switch in terms of latency, throughput and cell loss—if there is to be any cell loss at all (as on UBR). It can be used regardless of the type of switch fabric employed, or whether buffers are on the input ports, output ports or in a central memory. It is simplest to visualize for a memory-based switch. **See Figure 11.**

Figure 11
Per VC Buffering in a Simple Two Port Memory-Based Switch



Without separate buffers or queues per VC, all VCs contend for buffer space from a **simple FIFO** (first-in, first-out) buffer pool for each output port, and there can be no accounting of how much any one VC is using and no individual attention from the switch for each VC. This can result in widely varying latencies, throughput and cell loss by VC. **See Figure 12.** Notice in Figure 12 the "greedy" VCs #12 and #10 are able to monopolize the FIFO buffer, and impair through-put and latency of the "innocent" VCs #3 #5, #10 and #15. This does not happen with Per VC Buffering in Figure 11.

Figure 12
FIFO Buffering in a Simple Two Port Memory-Based Switch
(no Per VC Buffers)



Lab and customer tests of the GIGAswitch/ATM with a mix of traffic loads, congested ports and unruly UBR sources have shown that fairness in throughput and consistently low latency is assured with Per VC Buffering as implemented in the GIGAswitch/ATM. Furthermore, no VCs experienced cell loss (with flow control applied).

The GIGAswitch/ATM assigns dedicated buffer space only to “active VCs” and provides configuration options for how buffer space is managed across VCs, allowing more for long distance links. By using dynamic allocation of buffer space to “active VCs” only, total buffer size need not grow linearly with the total number of VCs allowed on the link.

2.10. *What is “Per VC Buffer Acccounting” and why is that important?*

While Per VC Buffering alone is sufficient to equalize throughput and latency of VCs, it is not sufficient to guarantee that no innocent VCs will unfairly suffer cell loss as a result of excessive loading by other VCs, when there is no per VC flow control deployed, as with UBR, VBR or CBR service. Without flow control one very greedy VC (e.g., #10 or #12) could nearly fill the entire available buffer space. Therefore it is necessary to also do **Per VC Buffer Accounting .** Per VC Buffer Accounting allows the switch to put a limit on how much buffer capacity any one VC can use. Therefore, if a VBR, UBR or CBR VC should “run wild” (or an ABR device should decide not to obey the flow control protocols) the network is protected. Properly behaving VCs will not be adversely affected by misbehaving VC. This protection applies to all types of VCs, whether they are SVC or PVC, and whether they are CBR, VBR, ABR or UBR.

With Per VC Buffer Accounting configuration parameters are used to enable the network manager to decide how much buffer space to allow per VC (for all VCs) on any given link. For long distance WAN links this number would want to be higher than for LAN links of the same speed since long distance VCs may have many cells “in flight”.

It might also possible to have different amounts of buffer space allowed for different VC on a link depending on the maximum throughput that VC will requested (e.g., the PCR) as determined at SVC set-up time.

Since Per VC Buffer Accounting is needed to identify which VCs are consuming more than their 'fair share' of the buffer pool, (or more than they said they would use) it is also an essential foundation for applying per VC flow control and other advanced traffic management techniques on a per VC basis.

Without Per VC Buffering and Per VC Buffer Accounting, techniques used for flow control and traffic management would have to be applied equally to all VCs on the congested link regardless of relative load per VC. This would result in an ATM network with very erratic behavior.

Per VC Buffer Accounting assures that misbehaving VCs can be identified and throttled back, if a "per VC" flow control mechanism is employed. If no flow control is available (or if no flow control is applicable: as on CBR, VBR or UBR), cells of the greedy or misbehaving VCs can simply be dropped. Dropping the cells is the best option in this latter case if the VC has already used more than its fair share (or more than its requested Peak Cell Rate), before a misbehaving VC destroys performance of many innocent VCs.

Per VC Buffer Accounting for all types of VCs is supported in the latest release of line cards (V2.0) for GIGAswitch/ATM shipping in the summer of 1996. Previously Per VC Buffer Accounting was applied only to ABR VCs using *FLOWmaster*.

2.11. What are typical ATM switch latencies? How does the GIGAswitch/ATM compare?

Typical ATM switching latencies are very low—usually less than 30 microseconds in well-designed ATM switches for ABR traffic. The GIGAswitch/ATM latency is around 10 microseconds for ABR traffic under most load conditions. Older packet or frame switching technologies usually have latencies measured in milliseconds—hundreds of times slower.

ATM switches achieve these low latencies by using "cut-through" switching techniques. Only the 5 byte header on the ATM cell needs to be examined before the cell is switched and on its way.

In a well-designed ATM switch using Per VC Buffering (see Question 2.9), congestion does not appreciably impact latency for other (uncongested VCs on the same switch) under randomly heavy loads. This is how GIGAswitch/ATM operates.

In a switch with simple FIFO buffering, it is possible to have very long latencies. With buffers sizes of, say, 3,500 cells per link, latency could be as bad as 10 milliseconds under congested conditions (~353,000 cells per second are transmitted on a 155 Mbps port). However, cell loss, not latency, is the real problem here. After buffers on that port fill-up, cells will be lost for that VC and all other VCs on that port unless flow control mechanisms "kick-in" quickly enough.

CBR latencies, on the other hand, should remain constant and unaffected by other traffic, as should the **Cell Delay Variation (CDV)** for CBR traffic. In the GIGAswitch/ATM, latencies for high-bandwidth CBR VCs remain under 20 microseconds, with CDV as low as 3 microseconds regardless of other loads. Lower bandwidth CBR services would have higher latencies and higher CDV. This is due to less frequent servicing of low-rate CBR queues by the switch (because they need less bandwidth). Nonetheless, both CBR latency and CDV are within limits needed for high quality real-time multimedia applications, regardless of the amount of bursty ABR data traffic.

2.12. What are the advantages of distributed vs. central control in ATM networks?

Distributed control provides for higher availability, more power and more flexibility than centralized control.

The control processes of particular importance in ATM networks relate to SVC set-up, enhanced services (e.g., LAN Emulation), and access to management functions of the switch via SNMP. Switches that function under centralized control are subject to a single point of failure for the entire network, are more limited in call processing power, are unable to support enhanced services, and may not fully support SNMP management (since proprietary mechanisms are used to control them remotely).

Some vendors' switches today still rely on an attached workstation running specialized software to support an entire ATM network and perform all control functions including SVC set-up and LAN emulation functions. This approach is particularly vulnerable to failures and performance bottlenecks. It also adds cost and complexity to the network. (See Section 5: *Setting up Virtual Circuits in ATM Networks* and Section 7: *Using ATM with Traditional LANs*).

Well-designed distributed control equips each switch internally with the power it needs to operate independently, to be managed by any standard SNMP managers, to set-up SVCs, and to support enhanced functions.

Most ATM switches today for private networks operate under distributed control. However, many have only a single processor for control in each switch. This exposes the switch to a single point of failure and performance bottlenecks (particularly for SVC set-ups). The best modern ATM switches support multiple control processors in each switch, as GIGAswitch/ATM does today.

2.13. What are the most important redundancy features to have in an ATM switch?

Where ATM switches provide the foundation of a company's backbone, having some level of redundancy in each switch is very important. The most likely components of a switch to fail are the power supplies and the fans. For backbone operations it is essential that these be fully redundant with automatic failover. (Obviously there should also be independent power sources, separate circuit breakers, or even an external UPS for such backbone equipment.)

The next most likely subsystems to fail are related to the control processor board which is where the software runs and handles SVC set-up, management tasks, etc. (see Question 2.11). In addition to the control processor board, the control software itself and the components that support I/O on the linecards are the next most likely failure points. It is highly desirable to have redundancy and/or hot swap capability in these components as well. Other components in an ATM switch (e.g., the switch fabric itself) are often designed with few active components and typically have MTBF ratings many times longer than these other components.

To seek full redundancy (with auto-failover) of all components in an ATM switch including the switch fabric would make the switch exceedingly expensive and a non-economical alternative to other technologies (Fast Ethernet and FDDI). In fact it is often not even technically feasible to attain full-redundancy since ATM connections are not inherently redundant—terminating at only one point on the switch (unlike DAS connections in FDDI).

For environments with extremely high-availability requirements, the prudent approach is to install two or more ATM switches in a topology that assures a high degree of availability for key users and services. This can done by equipping critical end-stations (e.g., servers, hosts), with two or more adapters and access lines into the network, each to a different switch. Other systems (e.g., users' PCs) can use access devices (e.g., ATM edge-devices or small ATM switches) which are themselves connected to two or more different backbone switches.

This approach has the additional benefit of providing twice as much total backbone capacity, and at much lower total cost, than seeking full redundancy in the ATM switches themselves.

2.14. How does Digital provide redundancy in the GIGAswitch/ATM?

Digital provides full redundancy in power supplies and fans in the GIGAswitch/ATM platform. Full redundancy is also provided in the switch control processing system since each linecard is equipped with the capability of being the master linecard (which handles all control and management processing). At switch initialization time a master linecard is elected. In the event that one linecard fails, another linecard will be elected master.

This approach has the added benefit that all linecards can participate in SVC set-up, LAN Emulation and other processing-intensive tasks with a firmware upgrade that is planned for the GIGAswitch/ATM. This provides valuable additional capacity for these critical functions.

The GIGAswitch/ATM platform will also support hot swapping of I/O components on each linecard (without changing the linecard), and hot swapping of the linecards themselves (in future versions of the linecards).

2.15. *What is the most effective way to support I/O interface flexibility in ATM?*

With ATM there are various physical interface options (OC-12c over Singlemode and Multmode Fiber, OC-3c over Singlemode and Multimode Fiber, STS-3 over UTP-5 copper, 25 Mbps over UTP-3, T3 and E3 over coaxial cable, and T1 and E1 over UTP, and other.) More options will also become available. If different linecards have to be built by vendors and bought by customers for each of these different physical interfaces, the costs and complexity of implementing ATM would be high.

The most effective way to implement flexibility for the I/O interface into an ATM device is for vendors to support the ATM Forum standard called UTOPIA (Universal Test & Operation Physical Interface for ATM). This defines a general interface at the cell level that is independent of the framing techniques used on the fiber or the wire and the speed of the media. UTOPIA allows for small, user-swappable I/O cards to provide the physical interface between the linecard and the physical cable or link. Thus one linecard design (and one linecard as purchased by a customer) can support a wide variety of physical I/O interfaces.

Digital is implementing support for I/O interface flexibility on its GIGAswitch/ATM and other ATM family products using the UTOPIA standard for a collection of daughter cards called “Mod Phys” (Modular Physical interfaces). This includes support for all of the 155 Mbps options described above plus the T3, E3, T1 and E1 options. Additional options will be provided in the future.

3. Differences between ATM in the Private LAN vs. the Private WAN

3.1. What differences exist between ATM LAN and private ATM WAN switches?

An ATM switch designed for the LAN backbone use must provide higher throughput than one designed just for private WAN use, since traffic loads in the LAN tend to be much higher. ATM LAN switches also have to be more price competitive than private WAN switches in order to compete effectively against 100 Mbps LAN technologies such as Fast Ethernet and FDDI.

A LAN backbone switch must provide support for SVCs since the LAN environment is one where "connectivity on demand" is the norm. Many private WAN switches, though, only provide support for PVCs since WAN connectivity typically is pre-arranged by the network administrator.

An ATM switch designed for a private WAN backbone places more emphasis on integrating multiple WAN technologies (e.g., Frame Relay and circuit emulation at relatively low speeds such as, T1, 56 Kbps) so that a single ATM backbone can carry all WAN traffic (voice and data). These features are not usually supported directly on an ATM LAN switch, instead requiring separate edge-devices.

A LAN switch should also provide support for ATM Forum standard "LAN Emulation" (or LANE) integral to the switch, so that existing LANs (e.g., Ethernets) can connect to the ATM network without modification and without the introduction of extra "LANE server" devices. ATM WAN switches may provide rudimentary support for point-to-point bridging of LANs (Ethernet, Token Ring) using bridge or router tunnels over the WAN, but this is very different from true LAN Emulation. (See Section 7: *Using ATM with Traditional LANs.* Also see Question 3.2)

3.2. What are the advantages and disadvantages of "hybrid" switches?

The advantage of the hybrid switches (i.e., switches with both ATM and non-ATM interfaces) is that a single vendor appears to provide a more complete solution in a single package, (e.g., support for Frame Relay, Ethernet, Voice, Video, etc.). The disadvantages can be significant, however. Depending on the specific configuration required, it may result in a much higher cost solution with less flexibility, and the solution may not fully meet the requirements. Some particular traps to watch for are:

- Generally, hybrid switches are much higher-priced per ATM port and per unit of throughput than a pure ATM switch -- sometimes three-to-six times higher. If lots of ATM ports (e.g., more than 6) and lots of throughput are required, the integrated switch is <u>not</u> an attractive way to go.
- Slots in the ATM switch chassis may become "crowded-out" by non-ATM technologies reducing the usable bandwidth of the switch to a few hundred megabits per second. This turns a very expensive multi-Gbps switch into a low throughput "ATM concentrator". The cost of using up a slot in an ATM switch chassis has to be factored into the cost of using that slot for other LAN or WAN technologies.
- Ethernet ports may be as much as 10 times higher in price per port than competitive Ethernet switches with an integral ATM uplink (e.g., $7,000 to $10,000 per port in the ATM switch vs. about $1,000 per port in an Ethernet switch with an ATM uplink). If only one or two LAN ports in total are required though, adding a LAN card to an ATM switch (e.g., to support direct Ethernet connections) could be less than buying a separate Ethernet switch with an ATM uplink.
- There is often inadequate support for LAN technologies in many of these hybrid switches. Many support just Bridge Tunnels which are point-to-point and must be manually configured as a PVC (see RFC 1483) vs. full LAN Emulation which uses SVCs and is dynamically configured.
- Voice and low-speed circuit emulation ports are several times higher in cost than those provided by external ATM concentrators, now available from several vendors. (See Question 3.3).
- Often, there is inadequate support for voice signaling or compression, or lack of support for individual voice calls. Point-to-point voice trunking is typically all that is provided. This lack of

support means that an expensive ATM switch is being used as a virtual private line, or TDM mux, rather than taking advantage of ATM's SVC capabilities to handle individual voice calls.

3.3. How do I connect other LAN and WAN technologies to a pure ATM switch?

Many vendors, including Digital, now offer backbone routers and Ethernet switches with 155 Mbps ATM uplinks which support ATM Forum LAN Emulation (LANE) and/or Classical IP. Token ring and FDDI switches with ATM "uplinks" and/or LANE support are also becoming available.

Coming from another vantage point, many WAN equipment vendors (e.g., ADC/Kentrox, Digital Link, GDC, Litton Fibercom, Onstream Networks, Premisys, and others) are now offering **"ATM concentrators"** that support Circuit Emulation Services at T1/E1 and lower rates over various standard interfaces. Explicit support for silence suppression, voice compression, HDLC bit streams, and Frame Relay are also sometimes provided. These ports send data only when there is something to send, intelligently conserving WAN bandwidth. Network level and service level interworking for Frame Relay devices is also being standardized across these "edge-device" vendors.

Interoperability between these various types of edge-devices and ATM switches (which is done via the UNI specification) is already here for basic connectivity (i.e., PVCs). SVC interoperability over UNI for these devices is also fairly extensive with Ethernet-to-ATM edge-device products and extending rapidly to include most vendors.

The ATM "edge-device" market will become a very competitive open market and will provide the most cost-effective, flexible way to integrate all manner of networking technologies onto an ATM backbone using "pure" ATM switches in the backbone to get the best overall price/performance and feature set.

3.4. Can ATM switches designed for a LAN backbone be used for a private WAN?

Yes, ATM LAN switches can be connected over the WAN if they support common ATM WAN interfaces such as T3/E3 and/or T1/E1, and have sufficiently large buffers for these interfaces. This approach actually represents the most cost-effective and viable approach to building an integrated ATM LAN/WAN backbone.

The bandwidth for WAN backbone links can be obtained as either dedicated private lines site-to-site, or as connections into a public ATM service. There are advantages and disadvantages to each alternative. (See Section 1: *ATM in Public vs. Private Networks*).

Other WAN technologies (voice trunks, video trunks, HDLC/SDLC circuits) can be integrated into this single LAN/WAN backbone by using ATM concentrators and other edge-devices available from a variety of vendors. (See Question 3.3).

3.5. Can ATM switches designed for a private WAN backbone be used in a LAN?

Yes. ATM switches designed for building a private WAN backbone can be used for a LAN backbone with limitations. If LAN backbone requirements grow appreciably, you will outgrow the ATM WAN switch in terms of throughput, ports or both. They are typically not designed for the high throughput and port concentration environment of the LAN, and tend to be much more expensive. You may also find that the ATM WAN switch does not provide support for SVCs, for ATM Forum LAN Emulation or the flow control needed for high quality ABR services.

3.6. What ATM line speeds are most viable in the LAN? In the WAN?

For ATM LAN backbones, 155 Mbps SONET connections (e.g., OC-3c when on fiber, STS-3 when on Category 5 UTP wire) are the most viable standards. OC-3c on multimode fiber (MMF) can easily extend to 2 kilometers. On singlemode fiber (SMF) distances to 25 kilometers can be readily obtained. These are adequate for even the largest LAN backbones. STS-3 (on Category 5 UTP) can extend to only 100 meters;

however, this may be quite adequate for desktops and servers connected to the backbone or for a workgroup environment.

Very soon 622 Mbps (OC-12c) lines will also be available for inter-switch connections in the LAN over both MMF and SMF. These capabilities provide very attractive backbone options for large LANs or high-performance environments. Digital will provide 622 Mbps inter-switch links on the GIGAswitch/ATM in early 1996.

Other speeds for ATM LAN connections are not as widely used and offer questionable benefits. For example, the 100 Mbps "TAXI" interface, common in the very early days of ATM is now falling out of favor, being replaced by the 155 Mbps SONET standard at the same or lower prices. And the 51 Mbps standard never really did catch on.

The 25 Mbps standard approved by the ATM Forum in early 1995 (based on IBM Token Ring framing technology and *incompatible* with 25 Mbps SONET connections) is interesting only if connections using it can be made at <u>significantly</u> lower cost than a 155 Mbps SONET connection. The jury is still out on that. While 25 Mbps options are lower in cost today, SONET framing technology is coming down rapidly in price. As computer systems grow in I/O capabilities, the 25 Mbps option may begin to appear too limiting (and non-competitive with 100 Mbps Ethernet), when trying to move large files quickly between systems.

One clear distinction of the 25 Mbps ATM standard, however, is its ability to use Category 3 UTP wire, which is more widely installed than Category 5. However, it is also possible to run SONET connections over Category 3 wiring (at both 155 Mbps and at lower speeds). At 155 Mbps on UTP-3, distances may be severely limited and certain common connectors may not used.

For ATM WAN backbones, OC-3c services are generally not available from most public carriers, and where they are they tend to be rather expensive. More readily available today are T3 or E3 services (at 45 Mbps and 30 Mbps respectively). These too, however, can be hard to obtain in many local markets and are still quite expensive. However, in the US and other countries with competitive public telecom markets alternative carriers are now providing T3/E3 services for rates as low as $4,500 per month or less across a metro area, forcing former monopoly carriers to be more competitive.

The other WAN option for ATM is to use T1 or E1 services (at 1.5 Mbps and 2.0 Mbps, respectively). While these are very low relative to ATM LAN speeds of 155 Mbps, they may be all that is needed if most traffic stays on the LAN, which is typical. Additionally there are now standards for how to **inverse multiplex** several T1 or E1 lines into a single logical higher speed line for use by ATM. This is likely to become one of the most popular means of connecting small remote ATM devices or LANs to each other, since T1 and E1 services are widely available and are much lower in cost. (For example, the AT&T T1 ATM access charge is well under $1000 per month.)

4. Categories of Service in ATM: CBR, VBR, ABR and UBR

4.1. What are the important differences between CBR, VBR, ABR, and UBR service?

CBR (Constant Bit Rate) service provides a guaranteed bandwidth through the ATM network. Just one traffic parameter, the **Peak Cell Rate (PCR)**, is established at set-up time for the VC. (A pair of VCs in each direction yields a full-duplex CBR service—however the two rates do not have to be equal in each direction). The **Cell Delay Variation Tolerance (CDVT)** may also be specified (though not with current UNI 3.1 signaling). If a CBR service is set-up as an SVC, the calling station must request the PCR for each direction. If the network cannot accommodate the requested PCR in each direction the call is rejected. (This is called **Connection Admission Control—or CAC**). For a PVC the network administrator sets the PCR in each direction.

Once accepted, the network guarantees delivery of the cells at that rate. However, the **Quality of Service (QoS)** provided with a CBR may vary between networks and over time on the same network. (See Question 4.2.)

If the PCR is not fully used by the requesting device, the capacity assigned to that VC may or may not go unused or wasted *(this depends on the switches involved!)*. Cells in excess of the PCR will typically be discarded by the network. This means that a device requesting a CBR service must be prepared to "police" itself to stay at or under the PCR or risk losing cells that it sends into the network. Few computer systems have software in them to do this self-policing or to make explicit bandwidth requests in the first place. Instead they simply drive the line as fast as they can. This is what they are accustomed to doing on any LAN today.

Other devices, such as ATM concentrators, that provide circuit emulation services for channelized T1/E1 or other legacy circuit technologies, usually police CBR VCs appropriately. In some cases this is simply because they are unable to use more than the PCR requested since the VC is mapped to a physical port on the concentrator that operates at a rate corresponding to what was requested (e.g., 1.5 Mbps).

CBR services are specified in Cells Per Second. To accommodate a 64 Kbps channel, a PCR of at least 167 cells per second is required (given only 48 bytes of user data per cell). However, depending on the ATM Adaptation Layer (AAL) used, a somewhat higher cell rate may be required (e.g., 171 for AAL1 where 1 byte of the user data is reserved for synchronization). The application or edge device must know what AAL it is using and request the appropriate PCR given the AAL overhead involved.

Additionally it is worth pointing out that different ATM switches may support various **granularities** for CBR services. Few if any support a granularity as low as one cell per second. While the user can request any integer amount of cells per second, the switch may round up to the nearest 1,000 or 10,000 cells per second, or some other increment particular to the design of that switch, sometimes making ATM unsuitable if there are many low-bit rate CBR VCs required.

VBR (Variable Bit Rate) provides a guaranteed bandwidth with the ability to exceed that bandwidth for occasional bursts of traffic. VBR service requires a user to specify three traffic parameters: a **Sustained Cell Rate (SCR),** a **Maximum Burst Size (MBS)** and **Peak Cell Rate (PCR**). The SCR is the guaranteed bandwidth. If the SCR and MBS parameters are exceeded (as judged per standard VBR traffic conformance algorithms), then cells are marked as ***discardable***. They may not be dropped immediately, but are nearly certain to be dropped later if they encounter congestion elsewhere in the network. If the PCR is exceeded, the extra cells are usually dropped immediately.

With the upcoming version of the UNI 4.0 specification, two types of VBR services are proposed: **Real-Time VBR (rt-VBR)** and **Non-Real-Time VBR (nrt-VBR).** Non-Real-Time VBR differs from Real-Time VBR in that cell delay variation and maximum cell delay time do not matter.

In either case, the net effect is that if VBR is used for loss-sensitive computer communications (e.g., file transfers, transaction-oriented data, etc.), it is essential to have ***both VBR traffic shaping and policing*** in the computer system or edge-device. Traffic shaping can help be sure that VBR traffic conformance algorithms are satisfied (but only as long as the offered load and burstiness from the computers is not too great!).

Since VBR traffic shaping and parameter request support is typically not available on computer systems, VBR service is not recommended. One might consider using VBR with a very large burst size as a workaround. Alternatively the SCR can be set to the PCR, defaulting to a CBR service. In either case this can be very expensive—requiring resources to be allocated to a single VC, that might not be used efficiently.

ABR (Available Bit Rate) services, unlike CBR and VBR, are intended to meet the needs of computer networks by working in a manner similar to current LAN technologies. Computers (and computer system users) on LANs want to send their data as soon as they have something to send, and they want it to go as fast as possible (i.e., at line speeds) but without congestion causing any cell loss. This is because computer system data is "loss sensitive", to the extent that if re-transmissions are required then good throughput (or "goodput") in the network can decline dramatically, potentially leading to **a throughput collapse.** When a given LAN user doesn't have something to send (which in fact is quite often), then others on the LAN must to be able to use the available capacity.

It is unacceptable to reserve capacity between systems in a data network. There are simply too many different possible combinations that might want to talk. Furthermore, the need to talk arises suddenly and ends suddenly, often lasting less than a millisecond. For example, in one millisecond a user can send a 16 Kbyte file on a 155 Mbps ATM connection (including overhead). That may be all the user has to send for the time being. It is rare that an ordinary data user would transmit for more than ten seconds at ATM speeds—not too many files are larger than 160 Mbytes! (Multimedia applications are different. See Section 8: *Using ATM for Multimedia Applications*).

ABR services are intended to meet the requirements of "bursty LAN traffic" and use the **available** capacity after commitments are fulfilled for CBR VCs and the Sustained Cell Rate (SCR) portion of VBR VCs. With ABR, since there need be no minimum capacity assigned, you can leave the connection up without wasting resources. With CBR or VBR connections, this cannot be done without wasting significant resources.

One might think that setting up and then quickly tearing down CBR or VBR SVCs might provide an alternative to ABR. This idea is very problematic, however, given that SVC set-up can be 'slow' for some ATM equipment e.g., 100 milliseconds, and this would put intolerable burdens on the SVC set-up capacity of the network. It cannot be supported in a large network. (See Section 5: *Setting up Virtual Circuits in ATM*).

The standards for ABR service have just recently been finalized by the ATM Forum. They will be supported as part of UNI 4.0. It took a couple years to develop the standard. The main stumbling block was the "**traffic management**" portion of the standard, in particular the issue of what **flow control mechanisms** should be used and how they should work in detail. The recently completed ABR specification actually includes several flow control mechanisms which can be used independently or in conjunction with each other. (See Section 6: *Flow Control and Traffic Management).*

The specification for ABR includes two traffic parameters for an connection: **Minimum Cell Rate (MCR)** and **Peak Cell Rate (PCR).** The MCR is designed to provide some base level of guaranteed bandwidth (for unique data applications that may require it), even though there is no parallel in LANs or WANs today. Typically the MCR will be set to zero. Any non-zero MCR would tie-up bandwidth and be likely to waste it. The PCR must be set to the maximum rate that VC will ever use. Typically the PCR will be set to be consistent with the line rate of the connection (or the maximum throughput capacity of the station), unless the VC is being set-up by an application with a known limited maximum bit rate. Selecting any lower PCR would risk cell loss.

UBR (Unspecified Bit Rate) service can be viewed as ABR service without flow control or any specified traffic parameters. (PCR is optional). It is strictly a "best effort" type of service. Some vendors today offer

UBR services but call them ABR services causing confusion in the market, even though they provide no flow control mechanism.

Public carriers generally offer just VBR and CBR services, since these are easier to bill and manage from their perspective, and are closest to the services they offer today: Private Line and Frame Relay. Whereas, on private networks just CBR, ABR and UBR services are desired.

4.2. *What “Quality of Service” parameters are associated with these services?*

In general the Quality of Service (QoS) parameters for ATM services are:

- Cell Delay Variation (CDV)
- Maximum Cell Transfer Delay (Max CTD)
- Mean Cell Transfer Delay (Mean CTD)
- Cell Loss Ratio (CLR)

However, not all QoS parameters apply to all categories of service.

Under UNI 3.0 and 3.1, QoS is requested indirectly according to a “Class of Service”. The various “Classes of Service” correspond roughly to the “Categories of Service” described above, with no ability to individually specify any QoS parameter. Each Class of Service has a “QoS Class” associated with it as follows:

- Service Class A (QoS Class 1): Suitable for voice Circuit Emulation and Constant Bit Rate video
- Service Class B (QoS Class 2): Suitable for Variable Bit Rate audio and video
- Service Class C (QoS Class 3): Suitable for connection oriented data transfer (e.g., Frame Relay)
- Service Class D (QoS Class 4): Suitable for connectionless data transfer (e.g., IP or SMDS)

Service Class A is CBR effectively, B is VBR, and Classes C & D are closest to ABR. There is also a “QoS Class 0” (which maps to UBR Service) that has no quality specified.

In none of these QoS Classes, however, are specific values of the above QoS parameters mandated as part of the ATM Forum standards. The Forum only offers methods of measuring them. Strongly implied, however, is that Service Classes A and B provide low CDV and low Max CTD, whereas Classes C and D provide no specific CDV or Max CDT. This is about all there is to QoS under UNI 3.0 and 3.1. A lot of terminology, and not a lot of substance. The ATM Forum has realized this and is not moving away from it.

Under the proposed UNI 4.0 specification, it is intended that QoS parameters be individually “negotiable” on a per VC basis at call set-up time. “Negotiable”, however, means simply that the network will inform the caller if it cannot meet a requested parameter, and reject the call. Whereupon the user can try a different (easier to meet) parameter. Table 1 shows the proposed applicability under UNI 4.0 of each QoS parameter to each service category.

Table 1 Applicability of QoS Parameters to Service Categories for UNI 4.0

	CBR	rt-VBR	nrt-VBR	ABR	UBR
CDV	yes	yes	n/a	n/a	n/a
Max CTD	yes	yes	n/a	n/a	n/a
Mean CTD	n/a	n/a	yes	n/a	n/a
CLR	yes	yes	yes	yes	n/a

4.3. What does the ATM "category of service" apply to: VC, VP or the entire interface?

Normally the "category of service" (and associated traffic and QoS parameters) applies to individual VCs. The same physical interface or line can have CBR, VBR, ABR and UBR VCs all running across it at the same time. Hence a single physical interface or line does **not** normally have a category of service assigned to it.

It is also possible, though, for a category of service (and associated traffic and QoS parameters) to apply to an entire VP Tunnel (VPC) and hence to the aggregate of all the VCs in the tunnel. In this case VCs within this VP Tunnel can contend for the bandwidth allotted to the VP Tunnel, and must in aggregate comply with the traffic contract of the VP Tunnel (or risk cell loss). All VCs in the VP Tunnel must typically be of the same category of service (e.g., all ABR or all CBR) for this to work sensibly. If there is only one VP Tunnel over the entire physical interface, then the traffic and QoS parameters do, indirectly, apply to the physical interface.

4.4. When will UNI 4.0 be complete and when will Digital support it?

UNI 4.0 will be completed by mid-1996. Digital will support it shortly thereafter. Typically 6 to 9 months is required to implement a new standard of this nature.

4.5. Where should someone use CBR vs. VBR vs. ABR?

CBR services are primarily useful for real-time traffic (e.g., voice, video) that require low Cell Delay Variation (CDV) and guaranteed access to bandwidth of a known rate. If less than the requested CBR rate is used, this is not necessarily a problem. It depends on how the unused capacity can be reused and who has access to it. In well-designed switches ABR and UBR traffic can use the unused CBR capacity.

VBR services are useful for traffic that is not loss sensitive and which varies slowly and modestly around some well-known level. There is some debate over what traffic types meet that requirement. Data people tend to see VBR as being good for voice; voice people tend to see VBR as being good for data. The reality is that VBR is not particularly good for either. Variable rate encoded video might be a good example, as long as viewers don't mind noise or lost pixels during the "action scenes", when bit rates are highest. Effective use of VBR really requires an aggregation (i.e., statistical multiplexing) of many sources of traffic shaped into a single VC without much variation.

ABR should be used for any data networking requirement where "goodput" (i.e., error free throughput) is important and it is impractical or wasteful to reserve capacity between all potential end-points. This is essentially all ordinary computer-to-computer communications and applies to both LAN and WAN links. In fact, on expensive WAN links it is even more important not to reserve and pay for capacity that will go largely unused, or to waste it sending re-transmissions.

Consequently, a public ATM service without ABR support is of questionable usefulness for computer networking. None offer true ABR service today. The best alternative in the meantime is to use direct dedicated digital lines (e.g., T1/E1, T3/E3 or OC-3c if available) between ATM LAN switches or edge-devices, by-passing public ATM services altogether, allowing the ATM LAN switches to implement ABR end-to-end.

Another alternative is to consider using the CBR services of the carrier with the PCR applied to an entire VP Tunnel so that all devices can contend for the reserved capacity. However, if more than one VP Tunnel is established over the physical interface, some mechanism must be used to assure that the PCR of each tunnel is not exceeded. (This may not be possible.)

A simpler approach entails using a single VP Tunnel (to a single destination) with the PCR parameter for the tunnel set according to the speed of the access line used (e.g., 45 Mbps). However, this is logically identical to obtaining a dedicated private line between the same two endpoints, and may actually be higher in cost than a dedicated end-to-end private line.

4.6. *Is ABR going to replace VBR for data traffic?*

In most private ATM networks, ABR/UBR is already the preferred option for non-real-time data traffic (i.e., almost all data traffic).

However, public ATM carriers will probably still offer VBR services for two very good reasons: a.) they are much easier to bill for than ABR services where cells would have to be counted, and b.) they are easier to make money with than CBR services. By offering VBR services, carriers can "play the odds" (similar to how airlines overbook seat reservations, and how banks reinvest the money in your checking account). VBR enables the carriers to offer customers greater "access to capacity" in total than the carrier actually has installed. If they offered only CBR, they could not do this.

What makes the most sense is to apply VBR service parameters to an entire VP. This allows all VCs in the VP to contend for the available VP capacity.

However, to play the odds successfully (i.e., to reduce 'excessive' cell loss to 'modest' cell loss at loads in excess of the SCR), the carrier must have backbone bandwidth many times higher than the access line speeds to customers. Today this is not generally true. ATM access line speeds are in the 45 to 155 Mbps range, and backbone lines are still mostly at 155 Mbps. When backbone links go up (to 622 Mbps), this will improve.

None the less, cell loss will always be an inherent "feature" of VBR services when used as intended, otherwise, the customer would have no incentive to commit to, and pay for, a substantial SCR. With most ATM applications being very sensitive to cell loss, it is not clear what uses customers will find for carriers' VBR services.

4.7. *What categories of service does the GIGAswitch/ATM support?*

The GIGAswitch/ATM supports CBR, rt-VBR, nrt-VBR, ABR and UBR. The ABR service provided supports both EFCI and ***FLOWmaster***. *FLOWmaster* is a highly effective traffic management mechanism to guarantee zero cell loss, instant access to available bandwidth and fairness across all ABR VCs. The EFCI support is available with the latest releast of line cards for the GIGAswitch/ATM, shipping summer 1996. (See Section 6: *Flow Control and Traffic Management*.)

CBR services are supported with a granularity of about 250 cells per second, making GIGAswitch/ATM suitable for VCs with low-speed WAN bandwidths (64 Kbps, 128 Kbps, 384 Kbps etc.) commonly used by voice and video conferencing applications.

A couple of technical details on the current ABR and VBR implementations in the GIGAswitch/ATM should also be noted. The ABR VCs currently default to an MCR of zero and PCR is determined by the line rate. This is the configuration most commonly desired for private LAN/WAN use. The VBR VCs also default to an SCR equal to the PCR—also highly desirable wherever minimizing cell loss is the top priority as in a private LAN/WAN. These minor limitations will be removed in future versions of the product.

4.8. *How does the GIGAswitch/ATM handle unused CBR and VBR capacity?*

The GIGAswitch/ATM system dynamically allows any unused CBR or VBR capacity to be used by ABR and UBR traffic. In this way CBR and VBR bandwidth is never wasted, as long as there are ABR or UBR services present on the network. Therefore, allocating more cells per second to a CBR or VBR VC than will be used is not a problem (up to a point), depending on how many CBR/VBR VCs are required.

4.9. What happens to CBR services in the GIGAswitch/ATM when there is a big burst of ABR or UBR traffic?

The GIGAswitch/ATM fully protects both CBR and VBR services from ABR and UBR services. CBR and VBR services are scheduled on a simple time division basis, with steady CDV (regardless of ABR and UBR loads) and excellent shaping. They are fully reshaped at each switch.

4.10. What benefits are there to supporting multiple traffic priorities in ATM?

Multiple traffic priorities are useful only if everyone is <u>not</u> able to send their traffic at the highest priority. Therefore they must be managed by a network administrator.

Among contending CBR traffic loads, having different priorities would make little sense, since once the connection is accepted the network is obliged to carry the traffic. Otherwise it would require “bumping” a current CBR VC off the network, in place of a high priority CBR VC that shows up later. Such schemes are very hard to administer and often have unexpected adverse technical and organizational consequences.

Among contending VBR traffic loads the same logic as above applies to the SCR (Sustained Cell Rate) part of VBR. The excess over the SCR is the only place where a traffic priority scheme might be used: to decide whose data to trash first! But this would also come with considerable administrative effort, and could have adverse organizational consequences. It also further undermines the viability of VBR for loss sensitive applications.

Among contending ABR traffic, having different priorities nullifies ABR’s ability to behave like successful LAN environments do today. In fact, the opposite, namely **fairness** for all users, is usually preferred, and part of the ATM Forum standards. If one wanted to apply different priorities to different ABR cells in a network, it is unclear as to what should be done. Nobody should suffer cell loss, regardless of their “priority” because of the adverse consequences to everyone else and the wasted bandwidth caused by re-transmission. And, any attempt to slow down low priority ABR cells as they transit the network would simply consume valuable buffer space needed by higher-priority users.

The bottom line is that ATM already supports the two essential and distinct traffic priorities needed for user’s traffic, namely CBR and ABR.

4.11. What is Digital’s approach to traffic or user priorities in ATM?

Digital believes the only priorities that make sense for users are differences that affect the total <u>throughput</u> available to different classes of users for their CBR and separately for their ABR traffic, <u>without affecting cell loss or latency.</u> This can be accomplished via a number of mechanisms, some of which Digital supports today and others which will be added in future versions of its products. One method supported today is to control the amount of credits issued to a VC when using credit-based flow control. However, this is a very different notion from that of traffic priority proposed or implemented by some vendors. (See Question 4.10)

A higher priority, of course, is provided for OA&M cells required for internal network Operations, Administration and Management, as per Forum standards.

5. Setting up Virtual Circuits in ATM

5.1. *How are SVCs set up in an ATM network?*

Unlike PVCs which are set-up through manual procedures by a network manager, SVC set-up is initiated directly by higher layer protocols (e.g., IP) in a host (workstation, server, etc.) or by an edge-device (e.g., LAN switch, router, ATM concentrator, etc.). Using the signaling protocols that are part of the ATM Forum's **UNI (User to Network Interface)** standard, devices simply request a connection to a certain ATM address, and a certain category of service (CBR, VBR, ABR or UBR) with certain traffic parameters and QoS, where applicable. The connection may be accepted or rejected (e.g., for lack of bandwidth).

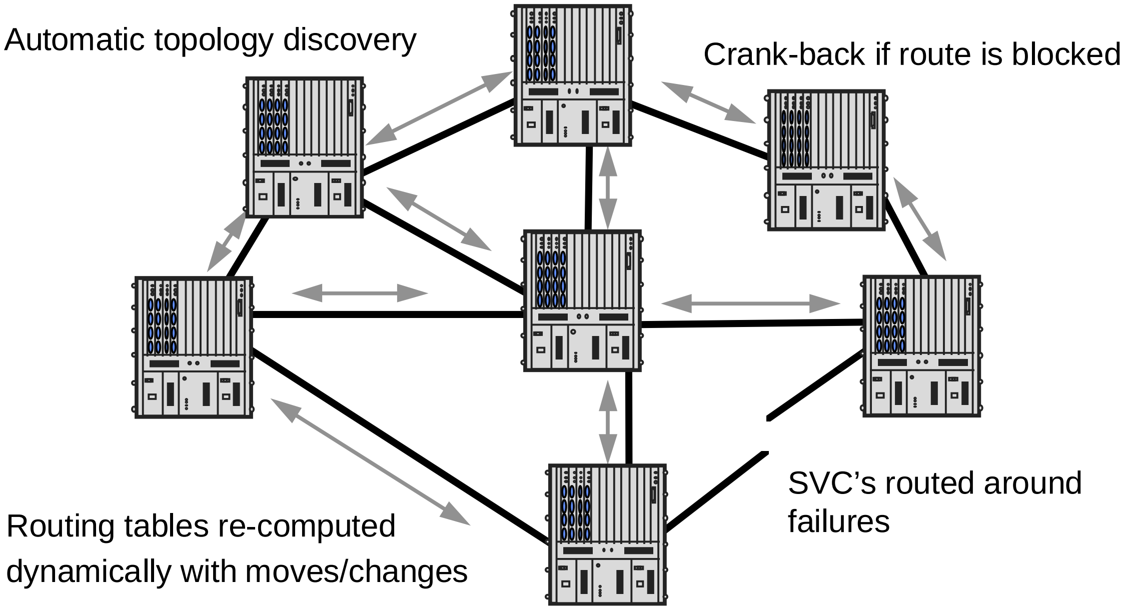
Some vendors (e.g., FORE Systems) still rely heavily upon and promote proprietary mechanisms for setting-up and routing SVCs (e.g., FORE's SPANS protocol), despite the widespread adoption of the UNI standard. Such approaches limit implementing ATM in a multivendor environment.

5.2. *What is the difference between static and dynamic routing in an ATM network?*

Static routing requires the network manager to manually enter or load a pre-determined routing table into each ATM switch in the network, in order for routing of SVCs to work. (This is the same as static routing within a traditional router network). Static routing is inflexible and labor intensive.

With dynamic routing the switches themselves discover the topology of the network and the needed routing information using mechanisms similar to traditional routers e.g., link state protocols with optimization for using the open shortest path first. Dynamic routing also allows for some other very beneficial features, namely: re-routing around failures in the network, load balancing across multiple links, and ease of reconfigurations (for both switches and end-stations). **See Figure 13.**

Figure 13
Dynamic Routing with GIGAswitch/ATM


Automatic topology discovery
Crank-back if route is blocked
SVC's routed around failures
Routing tables re-computed dynamically with moves/changes

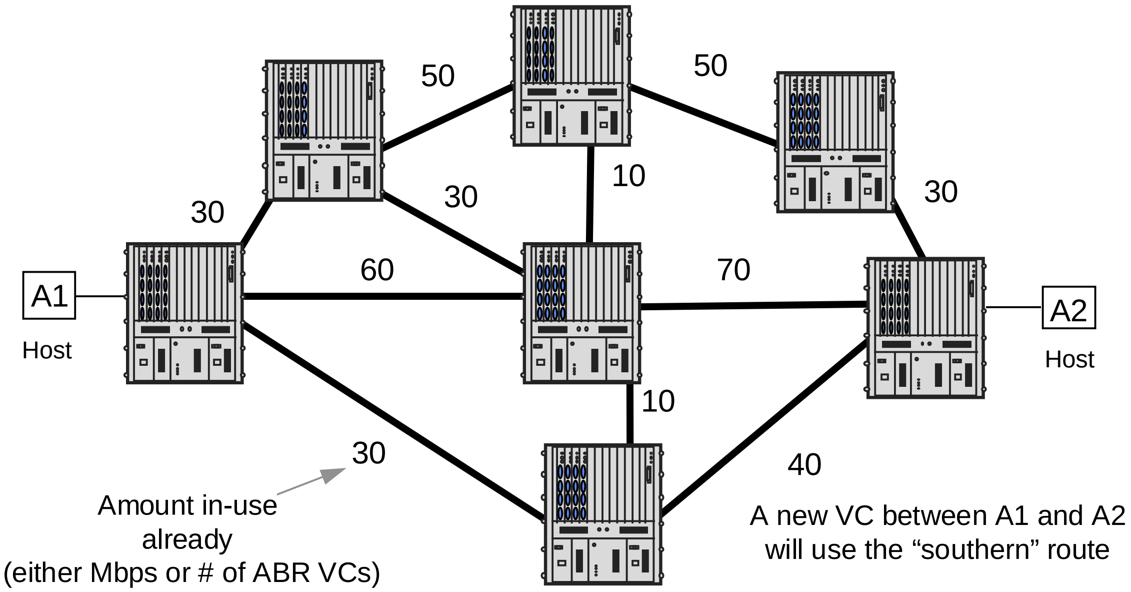
5.3. Does Digital support dynamic routing of SVCs?

Yes. Digital provides dynamic routing of SVCs plus a number of value-added features to support a dynamic networking environment with the GIGAswitch/ATM. **See Figure 13.**

The topology is automatically discovered by the GIGAswitch/ATM, and SVCs are routed along the shortest path (i.e., one with the fewest hops) where possible. Reconfiguration of the network topology or relocation of end-stations is handled automatically, as routing tables in each switch are re-computed dynamically. Re-routing of SVCs around failures is also supported as end-stations re-establish their SVCs automatically (see Question 5.7). In addition, Digital supports "crank-back", an algorithm to minimize the burden on the end-station when a path along the route is found to be blocked due to inadequate capacity remaining on that path. Instead of the SVC failing and reporting error conditions to the end-station, the route is "cranked back" and an alternate route, if available, is determined.

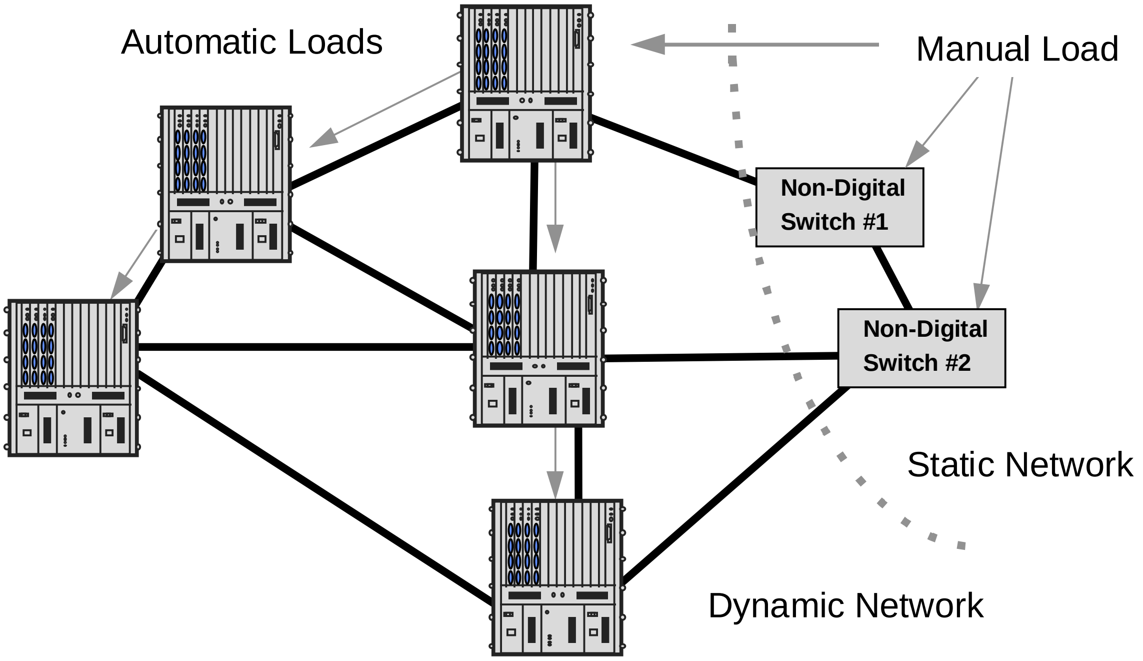
SVCs are automatically load balanced across alternative paths. Both ABR/UBR and CBR/VBR traffic are balanced (separately) among alternative paths. CBR/VBR is balanced by keeping running totals of the committed bandwidth on each link and equalizing appropriately. ABR/UBR is balanced by keeping running totals of the number of ABR/UBR VCs on each link. **See Figure 14**.

Figure 14
Load Balancing with GIGAswitch/ATM


50
50
10
30
30
30
60
70
A1
Host
A2
Host
10
30
40
Amount in-use already (either Mbps or # of ABR VCs)
A new VC between A1 and A2 will use the "southern" route

Finally, static routing tables from other types of switches that do not support dynamic routing are automatically distributed into each GIGAswitch/ATM. In this way stations located on any part of the static network can also be reached from the dynamic part of the network. **See Figure 15.**

Figure 15
Automatic Loads of Static Routing Tables with GIGAswitch/ATM



5.4. How long does it take to set up an SVC?

Call set-up typically takes between a few tens of milliseconds and a second. The set-up time depends on a number of key factors: type of switches, the number of switches and end-stations in the network, the distances involved, and the overall network complexity. The power and the number of the call processor(s) serving the network also has a major impact on call set-up time. In particular, if a single centralized call processor is used for the entire network, set-up times may grow to be very long and total set-up capacity will be very limited. Distributed call processing (in each switch) can significantly reduce set-up times and greatly improve SVC set-up capacity.

If the load of SVC set-up requests is in excess of the SVC set-up capacity of the call processor(s) for a sustained period, the wait for call set-up can be interminable. This situation is possible following a power outage, with all stations requesting their SVCs to be re-established at the same time.

5.5. How much SVC set-up capacity does a Private ATM network need to have?

This depends on how the network is used and how large it is. If it is used by edge-devices that have mainly PVCs or 'long-lived' SVCs between them, a very modest SVC set-up capacity is adequate (e.g., 1 per second), since legacy LAN traffic will be multiplexed inside a single VC. This is typically how many early ATM networks functioned.

If, on the other hand, the network is large (1000s of stations) and used by many directly attached workstations and servers, SVC set-up loads could be as high as ten or twenty SVC set-up requests per second, depending on the application environment. Many private ATM switches are not capable of handling this level of SVC set-up load, intended instead for static or slow changing environments. The impact of power and link outages on call set-up loads should also be factored into these requirements. When power and links come back up, call set-up loads can be very high.

5.6. *What SVC set-up load can GIGAswitch/ATM handle, and with what latency?*

GIGAswitch/ATM can handle SVC set-up rates of 30 per second, with a latency of less than 40 milliseconds, regardless of the size of the network. This is possible because each switch takes a lead role in SVC set-up for stations attached to it, multiplying the SVC set-up capacity by the number of switches in the network. This makes it suitable for very large networks with many directly attached stations today. With future versions of the GIGAswitch/ATM firmware, call processing in each switch will be distributed across all linecards in the switch, further multiplying the SVC set-up capacity as much as tenfold. This keeps SVC set-up latency low and grows the power for SVC set-up as the network grows.

5.7. *If a link or a switch goes down what happens to my SVCs? My PVCs?*

If a link goes down briefly, SVCs are automatically reconnected when the link is brought back-up, (as might happen if someone bumped a loose cable or other 'glitch' affected the link). Under UNI 3.0 switches and end-stations are supposed to hold and prepare to reconnect "crippled" SVCs for up to 90 seconds in the event of link failure. Under UNI 3.1 this timer was reduced to 10 seconds. This provides for automatic reconnect when the link comes back.

For failures on inter-switch links in a private network under the P-NNI standard, the same rule applies: SVCs on the link are to be held in waiting for the link to come back. However, if the link does not come back quickly there is nothing preventing end-stations from re-establishing SVCs as soon as they detect that the old one has failed—this may take 2 to 3 seconds depending on the high-level protocol running in the end-systems.

The same is true if a switch goes down. If an alternate path is available, dynamic routing of the SVC can re-establish it without the application being aware (other than a pause in traffic flow).

A PVC behaves differently. PVCs are static and will not be automatically re-routed (even if an alternate path is available). However, when the link or the switch eventually comes back—no matter now long—the PVC will be restored as it was before, and on the route that it was on before. PVC configuration data is stored in non-volatile RAM in each switch.

This is not true of SVCs. After 10 seconds of link outage (under UNI 3.1) SVCs are forgotten by the switches involved. If a switch undergoes a temporary power failure, all SVCs are lost (immediately) and must be re-established by the end-stations.

5.8. *What benefit is there to dynamic re-routing of existing SVCs?*

Re-routing existing SVCs is less beneficial than might be thought. Furthermore, doing so violates the ATM Forum specification unless ten seconds is passed. Normally, the SVC is re-established by the end-station automatically in a matter of 3-4 seconds when higher-layer protocols time out.

If desired, the ATM driver software in the end-station could arrange to have a new SVC re-established (to replace the old one) before the higher layer protocol times out (i.e., within 1 or 2 seconds). This approach, however, requires modifying the UNI rule for holding a "crippled" SVC open for at least 10 seconds (discussed above). This can optionally be done in certain situations, but can have other adverse consequences (e.g., like a sudden burst of SVC set-up requests whenever an outage occurs somewhere in the internals of an ATM network). The interests of stability are better served by waiting for stations' higher level protocols to individually time-out (at various different times).

5.9. *What are "Smart PVCs"?*

"Smart PVCs" are a feature that allows PVCs to be set-up by the network manager simply by specifying the end-points of the connection. The switches then determine the route through the network. Once established though, a Smart PVC behaves fully like a PVC (not an SVC). Without Smart PVC capability, the network manager must specify the route and enter the PVC configuration data into each and every switch in the path. While not bad for a small network, this can be a laborious task in a large mesh network of ATM switches. Digital will support Smart PVCs in its ATM switch products.

5.10. What degree of interoperability is there among vendors for PVCs and SVCs?

For the past two years there has been a very high degree of interoperability among PVC implementations. What is now happening is a high degree of interoperable SVC implementations via the ATM Forum UNI standards. At a recent interoperability event over 25 vendors, Digital included, demonstrated interoperable SVC implementations.

5.11. What is the SVC standard most commonly used?

There are two major SVC signaling standards in use for ATM: UNI 3.0 (which uses Q.93B) and UNI 3.1 (which uses Q.2931). These two standards themselves are not interoperable with each other, which means vendors must support both standards. There is a higher degree of interoperability among vendors today when using UNI 3.0, and in general more vendors support UNI 3.0 today than UNI 3.1. However interoperability among vendors using UNI 3.1 is also coming along.

FORE Systems' proprietary SPANS protocol is used by some vendors, but has fallen out of favor with the widespread acceptance of UNI standard from the ATM Forum.

5.12. What is the status of the P-NNI standard, and Digital's support for it?

P-NNI (Private Network to Network Interface) specifies how one private ATM switch must interface to another (e.g., for signaling and routing of SVCs). P-NNI Version 0 is the only standard implemented today. It is usually called **IISP—"Interim Inter-switch Signaling Protocol".** IISP is based on UNI 3.0. Digital's GIGAswitch/ATM supports it today, configurable on a per link basis. IISP, however, only supports static routing of SVCs. Therefore, it is usually desirable to use dynamic SVC routing within a network of GIGAswitch/ATM switches. When connecting a group of switches operating under IISP to a group of GIGAswitch/ATM, static routing information must be "handed-off" to switches in the dynamic portion of the network. Digital facilitates this with automatic distribution of static routing tables throughout a network of GIGAswitch/ATM systems (as discussed under Question 5.3).

P-NNI V1.0 will encompass dynamic routing, using link state protocols and shortest path algorithms, very similar to the dynamic routing already employed in the GIGAswitch/ATM platform today. P-NNI V1.0 should be finalized and approved by mid-1996, and Digital will support it soon thereafter.

Since P-NNI V1.0 is very similar to the current dynamic routing support provided already in GIGAswtich/ATM, it will not be a major transition for our customers to implement P-NNI.. P-NNI will be included in a standard release of the GIGAswitch/ATM firmware at no charge.

6. Flow Control and Traffic Management

6.1. For which ATM categories of service is flow control used?

Flow control is used mainly for ABR services. A certain type of flow control (EFCI) may also (optionally) be used with VBR services to control flows in excess of the SCR. It is unnecessary and it would be a violation of a VBR or CBR traffic contract for the network to impose flow control on the guaranteed portion of these services.

6.2. Other LAN technologies don't have flow control. Why is flow control important for ABR services in ATM?

Shared media LAN technologies do have access control (e.g., CSMA-CD in Ethernet, token passing in FDDI), which performs a "flow control-like" function. With the rapid emergence of LAN switching, flow control in switched configurations is becoming an important issue for traditional LANs as well. Primitive forms of flow control are being introduced into some LAN products today, where needed.

Without flow control, the same problems can occur in LAN switches as in ATM switches: when more than one input port is attempting to send frames to the same output port, buffers fill and may be overrun causing lost frames, which then causes re-transmissions and potentially more congestion.

The major difference is that frame loss in a LAN switch is "cleaner" than cell loss in an ATM network. When a frame is lost in a LAN switch, the entire packet is usually lost. Whereas, in an ATM network, when one cell is lost, dozens or hundreds of other cells composing the packet, e.g., through AAL-5, are probably still in the network, and still causing congestion for other traffic. This has led to the use of a technique in ATM called Early Packet Discard (EPD*).* (See Question 6.14)

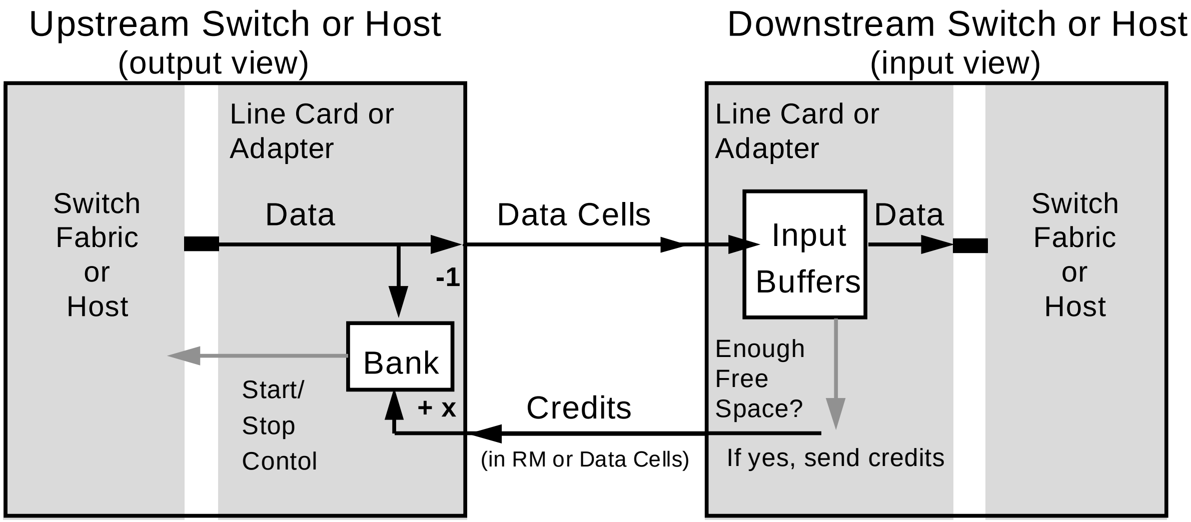
6.3. What are the major alternative approaches used for flow control?

At the highest level of abstraction there are just two approaches to flow control: **rate** and **credit**. These apply to both traditional data networks as well as ATM networks. A rate-based approach sends messages to the sender to either stop or slow down when congestion is encountered. A credit-based approach lets a sender continue sending until it has used up its "credits", where each credit corresponds to some quantity of data. In ATM, credits are conveniently measured in cells. As each cell is sent, one credit is deducted from the "bank" of credits (a simple integer) maintained by the sender. **See Figure 16.**

For each approach there must be some mechanism to allow stations to begin sending again, or to speed up when congestion dissipates. In rate-based approaches this may be a time-out, or it may be a message indicating that a new higher rate is allowed or simply saying "all clear". In credit-based approaches this message is always one that grants some amount of additional credits (say X), which can then be used, as needed, to send more data. In a good implementation of credit-based flow control, more credits are issued to the sender before the sender's credits are used up, if there is no congestion at the receiver. This allows the sender to maintain full line-rate throughput when there is no downstream congestion. In credit-based approaches for ATM the credits can be sent back in **Resource Management (RM)** cells or "piggy-backed" on data cells.

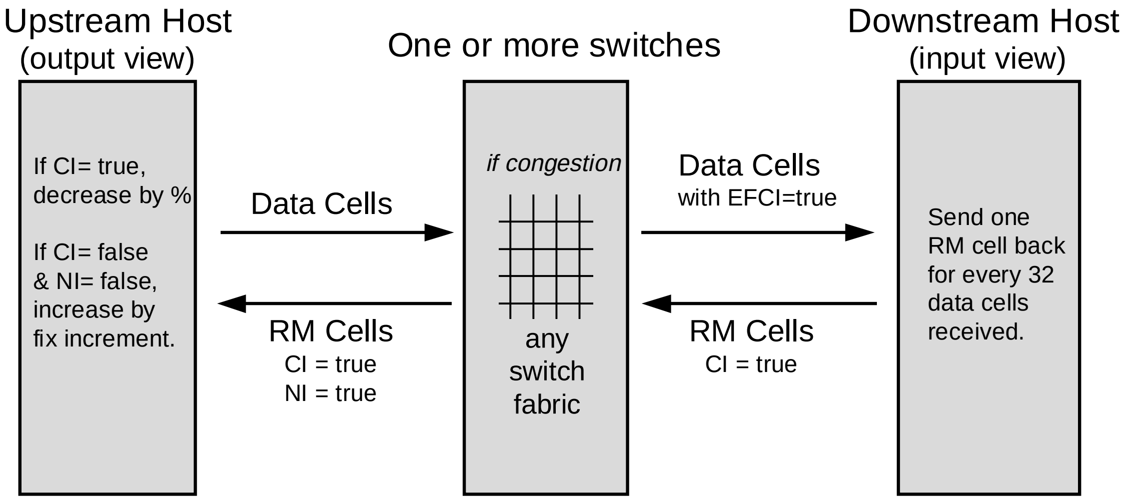
Within each approach there is the option to apply it either hop-by-hop (i.e., for each link in the network individually), or end-to-end (just between source and destination end-stations). Credit-based approaches in ATM are all done on a hop-by-hop basis as shown in Figure 16. Rate-based approaches (with the exception of GFC -- see Question 6.7) were conceived to and intended to operate on an end-to-end basis. **See Figure 17.**

Figure 16
Credit-Based Flow Control in ATM (hop-by-hop)



No data cells sent unless there are "credits in the bank". No cell loss.
RM = Resource Management

Figure 17
Rate-Based Flow Control in ATM (with EFCI end-to-end)



EFCI = Explict Forward Congestion Indication
CI = Congestion Indication, NI = No Increase
RM = Resource Management

Within a general mesh network of high-speed ATM switches, where 622 Mbps flows may be merging with each other and feeding into (i.e., congesting) a 155 Mbps pipe or even slower WAN links which run at 45 Mbps, 30 Mbps, or even 1.5 Mbps, it is almost essential to have hop-by-hop flow control on every link in the network. Consequently, rate-based approaches are now being adapted to be able to work on a hop-by-hop basis. However, rate-based approaches do not lend themselves to deployment or every inter-switch or host-to-switch link in an ATM network , (See Question 6.12).

One other key difference is that rate-based approaches sometimes are applied only to the aggregate of all ABR VCs passing through a congestion point in the network, and not individually per VC (because Per VC Buffering is often not supported in rate-based switches). All credit-based approaches in ATM are applied per VC using Per VC Buffering in each switch. This guarantees fairness and stability across VCs. (See Question 2.9).

Most data networks today use some form of credit-based flow control. For example the "windowing" mechanism in TCP (used heavily on the Internet and inside large corporate networks for file transfers and the World Wide Web) is a credit scheme. The window size in TCP determines the number of outstanding packets allowed to be sent without acknowledgment (ACK) from the other end. The ACK is the means of granting more "credits" in TCP. Similar mechanisms are used in SNA, DECnet and other transport protocols, but typically only between end-systems (not router-hop-by-router-hop).

The most common (and simplest) rate-based mechanism is XON/XOFF (Ctrl-Q/Ctrl-S) used by people at dumb terminals to stop the flow of data onto their screen where the rate is either "all or none". Modern rate-based mechanisms are significantly more complex it that they allow any intermediate rate to be "negotiated" between end-stations and sometimes also between an end-stations and the network.

6.4. Doesn't flow control add cost and overhead to ATM, reducing its benefits?

It is true that flow control adds some overhead in terms of flow control messages being passed over the line. However, the net benefit of good flow control far exceeds this overhead which is typically less than 6% of ABR traffic and zero if Digital's ***FLOWmaster*** is being used. Only switches with good flow control can achieve the full utilization of the transmission lines for ABR without cell loss, re-transmissions or network instability. This is particularly important over expensive WAN bandwidth, but no less important for LANs.

If ATM were not being used to carry loss-sensitive data (where re-transmissions were mandatory), flow control would be less important. Since ATM is being used to carry loss-sensitive data, this question has more recently come down to*: What mechanisms for flow control are most cost-effective?* The answer to this varies depending on who you are: a private network customer who has yet to buy an ATM switch, or a switch manufacturer with an existing architecture that may not be able to support certain flow control and buffering mechanisms (e.g., Per VC Buffering) without an expensive redesign.

Flow control algorithms add essentially nothing to the manufacturing cost of a switch. In fact good flow control algorithms actually allow reductions in the amount of buffer space required to achieve a given cell loss probability (even zero cell loss), and therefore can lower the cost of a switch.

6.5. What approach is used by Digital with FLOWmaster and how well does it work?

FLOWmaster flow control is a credit-based mechanism that works on each link in the ATM network (hop-by-hop) and for each ABR VC. It guarantees zero cell loss, instant access to available bandwidth, and full line rate throughput for an ABR VC, in an unlimited mesh topology of ATM switches. It can handle any mismatch in line speeds anywhere in the network and works over long distance as well as LAN links. Sources do not have to use up all their credits before they get more credit. Additional credits are sent as the destination's buffer space is freed-up, with credit information "piggy-backed" on data cells in the reverse direction, which allows it to be very efficient in its use of bandwidth. Review Figure 16.

6.6. What rate-based flow control mechanisms are used for ABR and how do they work?

Three different rate-based mechanisms are specified. One mechanism is called **EFCI (Explicit Forward Congestion Indicator).** This is a simple rate-based mechanism, similar to one defined for Frame Relay called FECN (but rarely used). Review Figure 17. The other two rate-based mechanisms are called **ER (Explicit Rate)** and **GFC (Generic Flow Control).**

With EFCI, if there is congestion in a switch along the way, a switch can set the EFCI bit in a data cell header to “true”. (EFCI is a one bit message, either true or false). The information is not useful, though, until it is returned to all the source stations causing the congestion. (This will be all of the sources that have VCs passing through that point of congestion unless Per VC Buffering is used.) This is done by having all destination end-stations send back congestion information using cells dedicated to this purpose called **Resource Management (RM)** cells. This is done every so often (e.g., for every 32 data cells it receives). The RM cells actually carry two bits of information back to the source, a **“Congestion Indication” (CI)** message and a **“No Increase” (NI)** message, each of which is either true or false. Switches on the return path may also elect to set the NI message in the RM cell to true (to prevent increases in load).

When a source receives a “CI = true” message it is supposed to slow down by a percentage of its current sending rate. When a source receives a “CI=false” message it may increase its rate by some fixed additive increment, as long as the NI message is also false. The percentage decrease and additive increase amounts are configurable, but there is no guarantee all stations will be set the same, resulting in a lack of fairness.

This mechanism suffers from several major weaknesses:

a.) It may take tens of milliseconds, or more, depending on distances, congestion already present in both directions, and other factors, for the RM cells to get back to the various traffic sources. By this time, the congestion may have already caused cell loss, or, it is possible the congestion may have totally dissipated, because individual bursts from end-stations are typically only a few milliseconds long or less (at 155 Mbps line speeds).

b.) The behavior rules for traffic destinations receiving EFCI messages from switches are not very sophisticated regarding whether to set “CI = true” in the RM cell. A destination will typically receive 32 EFCI messages (either “true” or “false”) for each RM cell sent in the return direction, yet the RM cell contents are determined solely by the last (most recent) EFCI message received. This could propagate spurious information back to the source.

c.) EFCI implemented without Per VC Buffering causes flow control to be applied unfairly.

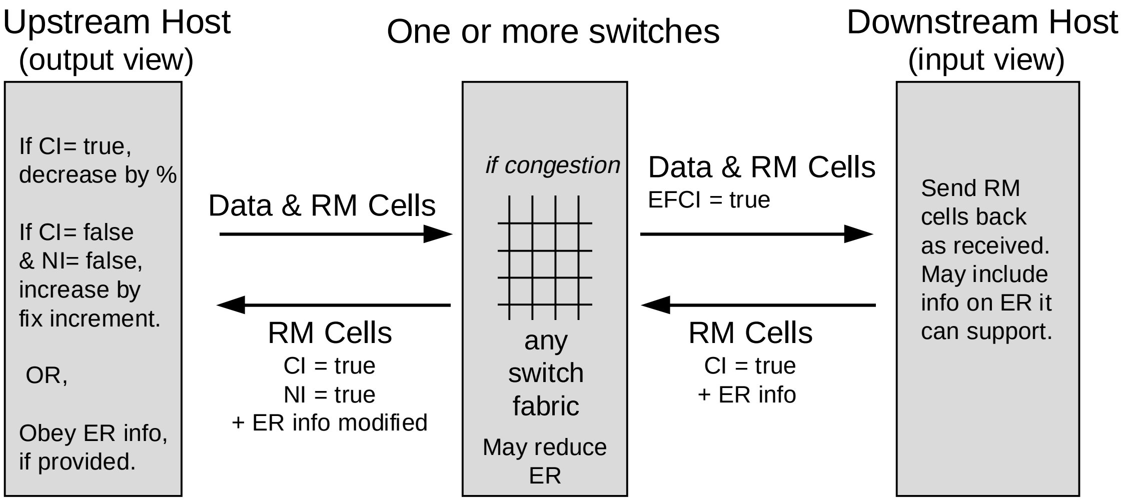
d.) By the time a source receives an RM cell it may have already stopped sending on that VC. What about the next time it wants to send a burst of data on that VC? How much can it send?

e.) What about new VCs set-up through the same congestion point, or VCs that were previously inactive. They are not receiving any RM cells. How much can they send?

To handle these latter two cases an **Initial Cell Rate (ICR)** notion is also defined as part of the standard for previously inactive VCs and new VCs. This too is a configurable parameter. If set high (i.e., close to the line rate) the chance of cell loss in the network increases. If set low, the ability to gain instant access to available ATM bandwidth in the network is severely limited.

Due to weaknesses of EFCI, the ATM Forum has also worked hard to define a better rate-based flow control mechanism called **Explicit Rate (ER).** As the name implies, this mechanism attempts to provide more explicit information on the rate at which stations should be allowed to send data into the network. That rate is based on feedback from all the switches in the path. Each switch can independently specify on the return path to the source how fast a given VC should be allowed to send data into the network. The switch that sets the lowest tolerance for load sets the rate for the VC. ER also requires all sources to send RM cells in the forward direction. **See Figure 18.**

Figure 18
Rate-Based Flow Control ER plus EFCI (end-to-end)



ER = Explicit Rate, CI = Congestion Indication,
NI = No Increase, RM = Resource Management

When end-stations support ER, they must still also support EFCI. This is due to the fact that many switches will not support ER. (It is difficult to do so, and it is not required by the ATM Forum.)

Explicit Rate suffers from many of the same problems as EFCI. The key difference is that the source end-station behavior will be able to be more tightly controlled by destination end-stations. It may also be more tightly controlled by switches experiencing congestion, but only if the switches support ER! Without switches also supporting ER, the feedback loop is no shorter than with EFCI. And even with ER support in switches, the feedback loop may still be too slow. This depends on where the congestion is relative to the traffic sources -- it may be in a remote switch and the distances of the links may be long.

The Explicit Rate specification is exceedingly complex. There are over 19 different parameters associated with ER—many of which are implementation specific (within a wide range) or configurable by the network administrator. Closure on the standard (initiated in 1994) bogged down for many months on several issues. In particular, there was difficulty defining the start-up behavior for a source that had not been sending for a while and for a new VC.

For the start-up behavior in ER, it was an objective to have something more flexible than the simple ICR parameter of EFCI as discussed above, but that did not happen. Given the brevity of most data transmissions at these high line rates (e.g. 16 Kbytes sent in one millisecond), it is possible that most transmissions could be considered “initial transmissions”. Therefore, a simple fixed ICR would unnecessarily throttle most stations most of the time. With poor settings of the parameters ABR turns back into something that looks more like VBR or CBR (but at a fixed low rate relative to line speed), undermining its benefit for high-performance computer networks. Fortunately, the ICR parameter is configurable by the network administrator and can be set high or at the line rate. However this poses risks of cell loss.

6.7. What is GFC (Generic Flow Control) and why can't it solve the flow control problem?

GFC (Generic Flow Control) is a very different approach and can only loosely be considered a rate-based mechanism. The use of the GFC field (the first field in an ATM cell header) has been defined by the ITU (International Telecommunications Union) -- the original body establishing ATM standards. GFC is defined and used only across the User-to-Network Interface (UNI), hence it is not available on switch-to-switch links nor does it work end-to-end. Its use is limited to the direct control of ATM end-stations or edge-devices by the network (not vice versa). It is a simple mechanism but not widely supported by vendors at this time. It is gaining more support lately.

GFC operates per link (not per VC). It allows the network to control the flow of traffic entering the ATM network on ABR and UBR VCs across the user access link to the switch. While it can prevent cell loss over the UNI connections, and it is very useful for supporting low-cost desktop access lines into an ATM network, it does not provide a full solution to the flow control problem. Congestion can occur in any part of a large ATM mesh network, not just over the UNI link.

6.8. What flow control support is required for compliance with the ATM Forum?

To offer ATM Forum compliant ABR services, it is only necessary to support EFCI on switches. End-stations must also support EFCI. Support for Explicit Rate (ER), is optional for switches but is intended to be required for end-stations (where all the complexity resides). However, vendors implementation parameters can be set such that the incremental benefit ER is effectively nullified. This was a political compromise due to the fact that many switch and adapter manufacturers could not support a flow control mechanism with all of the complexity of ER.

ER must be implemented in the chips on ATM adapters (or NICs) for computer systems and in the loadable software drivers for those adapters as provided by the vendor. *Solid interoperability of different adapter vendor's ER implementations, when they eventually appear, is not likely to be wide spread for many years to come.*

Support of GFC is also optional.

6.9. What alternatives to rate-based flow control are available to the ATM Forum?

Digital and others have presented credit-based flow control alternatives to the ATM Forum in the past. While credit-based approaches were recognized by many members of the Forum as being both simpler and more robust than rate-based mechanisms, they were not adopted. This is due largely to the difficulty existing telecom equipment vendors would face supporting credit-based approaches without major redesign of their switches. It was due also to a lack of understanding on the part of some voice/telecom-oriented Forum members about the fundamental differences between computer networks and voice switching networks. Many continued to view data traffic like it was voice.

In the last year, a consortium of ATM vendors formed to focus on private ATM networking for data environments. They developed a complete technical specification for a credit-based mechanism called **Quantum Flow Control (QFC)**. The QFC Consortium, which now has over 15 members including Digital, has made this specification available to the public. While the QFC Consortium would like the specification to be formally adopted by the ATM Forum as one alternative approach "blessed" by the Forum, it is unlikely at this point that the Forum will want to entertain additional mechanisms so quickly after completing the ABR standard.

Meanwhile numerous vendors have begun implementations of the QFC specification, including major chip vendors that plan to build the algorithm into high-performance chips (e.g., SAR chips, Link Control chips, etc.) for use in many vendor's ATM switches and adapters. QFC is similar in many ways to *FLOWmaster*, working on a hop-by-hop basis. It differs mainly in terms of how credits are communicated back to traffic sources and provides other features.

6.10. What are the key differences and similarities between QFC and FLOWmaster?

FLOWmaster uses a field in the ATM cell header which is normally used for VP information to "piggy-back" credit information in data cells back to sources. This allows *FLOWmaster* to very efficient since it adds no overhead to the line, but it does interfere with use of the VP field in the ATM cell head -- preventing concurrent use of VPs (Virtual Paths) with *FLOWmaster*.

With QFC, credits for multiple VCs are "batched" together in RM cells so that a single RM cell carries credits for many VCs at once. This makes it very efficient and allows full use of the VP field in the ATM cell header providing some useful features not available with FLOWmaster, namely:

- Support for ABR flow control on both individual VCs as well as on VP Tunnels (VPCs) for connections to public ATM networks.
- Support for creation of ABR-based VP Tunnels for other VCs to pass through.
- Support for longer distance, high-speed connections (with full line rate utilization and zero cell loss).

The similarities between *FLOWmaster* and QFC are many since they both use the same fundamental credit-based approach to ATM flow control. **Review Figure 16**.

- Both guarantee zero cell loss in spite of congestion and variations in available bandwidth due to higher priority traffic such as CBR and VBR.
- Both allow instantaneous access to available bandwidth, which may be the full line rate.
- Both operate hop-by-hop.
- Both can support multipoint ABR VCs (something not practical with rate-based approaches).
- Both scale to all ranges of ATM line speeds and distances.
- Both use Per VC Buffering to assure fairness across VCs.

FLOWmaster, in many ways, is the proof of concept for QFC since *FLOWmaster* has already been used successfully in many large production ATM networks.

6.11. What is Digital's position with respect to flow control in ABR services?

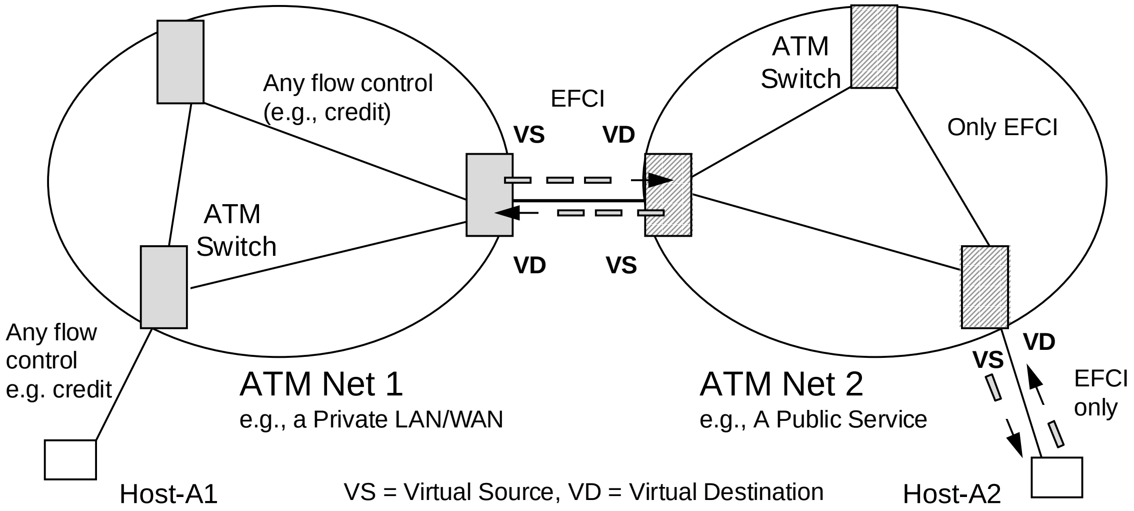
Digital will meet all ATM Forum requirements for flow control support in ABR by implementing both EFCI and ER. By also implementing open, value-added mechanisms such as QFC, we believe we can improve the suitability of ATM for computer networking, particularly for private LANs and WANs. Digital also plans to add support for GFC (Generic Flow Control) to all of its products to accelerate the use of ATM for low-cost desktop connections. Digital will continue to support *FLOWmaster* for those customers that desire to have the most efficient means of handling flow control and have no need for VPs.

These will be configurable options on a link-by-link basis. Digital plans to support Explicit Rate (ER) with particular priority on end-system adapters/interfaces, since that is where most of the work is required. EFCI , QFC and ER will be supported on a "per VC" and "per VP" basis. EFCI support will be enhanced later with the additional value of **"Virtual Source and Virtual Destination"** . (See Question 6.12).

6.12. What is "Virtual Source/Virtual Destination" (VS/VD)? What are its benefits?

One of the weaknesses of rate-based flow control for ATM is caused by the delay in getting feedback to the traffic sources. This is because it was conceived as an end-to-end flow control mechanism, rather than as a hop-by-hop mechanism. To remedy this, the ATM Forum, in its final documents on ABR, has included a "suggestion" for how EFCI (and ER if desired) might be operated on a VS/VD basis. **See Figure 19.**

Figure 19
Virtual Source/Virtual Destination with EFCI



VS/VD is in effect when ports on an ATM switch are made to function as virtual sources and as virtual destinations of traffic, for the purposes of rate-based flow control. This has the potential of providing additional robustness inherent in a hop-by-hop approach (namely: shorter feedback loops). However, it still cannot guarantee zero cell loss due to the same uncertainties about initiation of new flows and new VCs as described earlier, as well as the remaining delays in the feedback loop.

Furthermore, implementing VS/VD logic for rate-based flow control in an ATM switch adds overhead to a switch (potentially degrading performance), and supporting it also complicates the life of the network administrator. Due to these reasons it will most likely vendors will offer it for EFCI long before they offer it for ER. Furthermore, it is most likely to be applied by users only at the interface between two dissimilar networks (e.g., a between private and a public ATM network) or some other boundary (e.g., to a host that does not support credit based flow-control), rather than on every hop in a large network.

6.13. What benefit is there to supporting both rate-based and credit-based mechanisms?

EFCI, implemented end-to-end, provides useful feedback from and to EFCI-only hosts on how fast the other end can receive, but relatively little timely information about congestion conditions in the network. Supplementing EFCI with credit-based covers this latter need**..**

Despite the limitations of rate-based mechanisms, EFCI is likely to be the only mechanism available on some end-stations (hosts) and on public ATM services. Therefore, while we would recommend *FLOWmaster* (and QFC when available in products) for computer connections and inter-switch links, it is not always available. By combining credit-based and rate-based mechanisms in a network EFCI-only hosts can be better protected from cell loss, and when cell loss does occur it can be contained mainly to links at the edge of the network (e.g., where traffic is entering) and avoid wasting backbone bandwidth. There is no point in carrying traffic all the way across a network, only to drop it at the far end. **See Figure 20**

Figure 20
Credit and Rate on the Same VC in Same Private Network

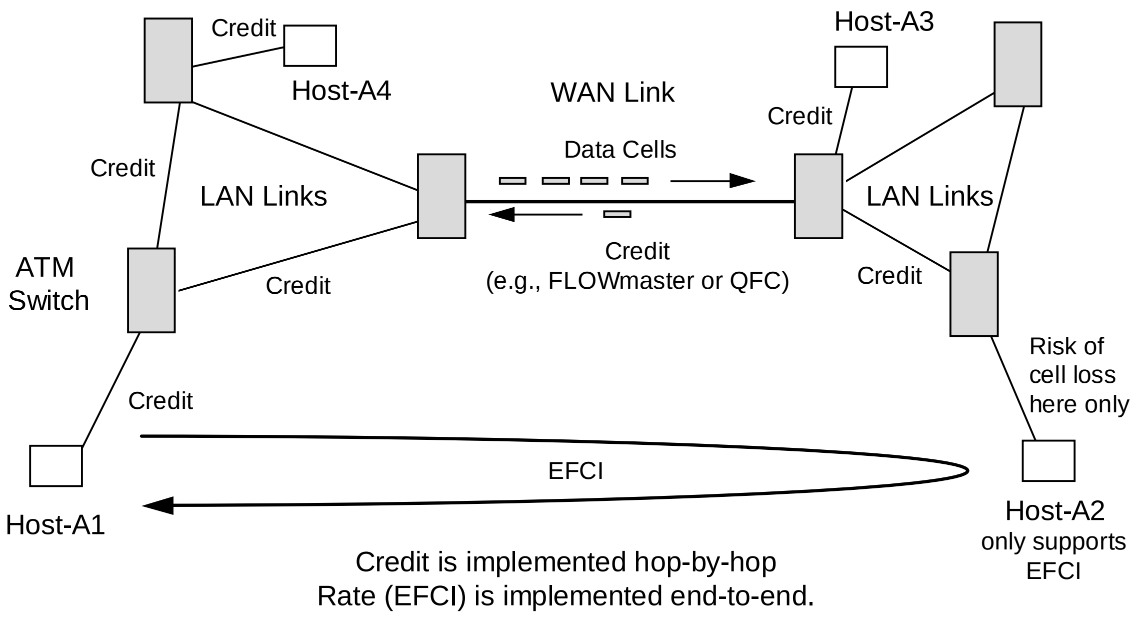


Figure 20 shows Host-A1 communicating with Host-A2, Host-A3 and Host-A4. However, Host-A2 only supports EFCI, so Host-A1 communicates to it using EFCI (end-to-end), but uses a credit mechanism (FLOWmaster or QFC) for all its VC connections to the network generally. Also the network internally (hop-by-hop) uses a credit mechanism for all switch-to-switch links. This assures no cell loss on the links to Hosts-A1, A3 and A4 and on all the intervening links. The only exposure to cell loss is on the link from the network to Host-A2, and mainly when Host A2 sends exceeisve amounts of traffic into the network. (On receive Host A-2 should not have a problem loosing cells since EFCI per VC is controlling the rate at which this host allows others hosts to send to him. The same logic applies if Host-A2 were to implement ER with EFCI (NOTE: In this case ER is not really an improvement over EFCI.)

For connections to public ATM services, EFCI again will be the only mechanism supported for some time to come, maybe indefinitely. Therefore, support of both Rate and Credit is useful here too. A likely scenario is to have EFCI rate-based flow control applied to an entire Virtual Path (VP) Connection (VP Tunnel) at the interface to the carrier, with that interface functioning as a virtual source and virtual destination of ABR traffic in both directions. The same approach would be recommended when connecting private ATM networking equipment that may not support QFC or *FLOWmaster* to public networks. Review Figure 19.

GFC flow control will be useful in these mixed environments for connecting to end-stations or edge-devices that do not obey other flow control mechanisms, yet very low cost desktop ATM connections are desired.

6.14. What is "Early Packet Discard" (EPD) and how well does it work?

Early Packet Discard (EPD) -- and its cousin Partial Packet Discard (PPD) -- are techniques used in switches to reduce congestion quickly once it is determined that some cell loss is likely to occur or has already occurred. They are not flow control mechanisms, but rather ways to mitigate congestion in an environment that lacks flow control, or lacks good flow control.

Basically these techniques seek to discard all cells that compose a single packet (e.g., under AAL-5) if it appears that any one cell in the packet will have to be discarded (EPD) or was already lost (PPD). This makes ATM switches behave similarly to legacy LAN switches (e.g., Ethernet switches) with respect to the loss of entire packets.

While these techniques improve performance under load, they are not a substitute for good quality flow control. In situations where not all devices are supporting flow control or the same type of flow control, they can be used to improve stability of the network. However, they cannot guarantee stability (i.e., the avoidance of throughput collapse).

Digital is supporting EPD and PPD in the new versions of its linecards for GIGAswitch/ATM and other new ATM switch products.

6.15. Why can't larger buffers in ATM switches solve the flow control problem?

Making buffers larger, after a certain point, only adds expense and latency to a switch, without solving the congestion problem. If multiple input ports at 155 Mbps (353,000 cells per second each) are attempting to send to the same 155 Mbps output port for any significant time, eventually the buffers will overrun if there is no flow control, regardless of buffer size. The problem can get much worse, where for example, a server is connected at 622 Mbps and sends a large file to a user connected at 25 Mbps.

6.16. How large should buffers be?

Buffers in switches should be sized with two key objectives in mind: minimal or zero cell loss and the ability of a VC to attain full line-rate throughput—less any overhead. The sizing of buffers to meet these requirements depends heavily on what flow control mechanism is used. It also depends on the switch's buffer management strategy and whether it is primarily an input or output buffered switch. Output buffered switches require more buffer space. If Per VC Buffering is used, buffer sizing will also depend on how many simultaneously active VCs are to be supported over the link. (See Section 2: *ATM Switch Design for Private Networks).*

If the above objectives are to be met, one general rule is that longer distance links require more buffering than shorter distance links. This is true regardless of the type of flow control mechanism used. It is a result of the delay caused by the finite speed of light, about 5 microseconds per Km in cable, or 5 milliseconds for a 1,000 Km link.

For example, with a rate-base mechanism the "slow down" or "no increase" message will be delayed longer getting back to the source on a long distance link. Therefore, more traffic will be "in flight", hence more buffers are needed so the cells in flight can "land safely" if the traffic is bursty and unpredictable (which ABR traffic is).

With a credit-based scheme the size of the credit (or window size) has to be made larger on a longer distance link so that a sender does not run out of credits while waiting for more credits (or an ACK) from the network or recipient. Otherwise throughput would be reduced over the link. With larger credits issued per message, buffers have to be made correspondingly larger.

6.17. How does the choice of flow control mechanism affect required buffer sizing?

With credit-based approaches, there is a simple relationship between the size of the buffers needed and the size of credits used. In this way it is simple to guarantee zero cell loss under conditions of heavy congestion with credit-based approaches. There is always enough room to catch whatever data is coming your way. Buffers under credit-based flow control are established separately for each VC from each end-station. "Per VC Buffering" is an inherent part of credit-based approaches used for ATM, and hence all of the other benefits of Per VC Buffering are also obtained, such as guaranteed fairness and low latency for each VC regardless of congestion. Using dynamic allocation of buffer space to "active VCs", buffer size need not grow linearly with the total number of VC's allowed on the link. (See Section 2: *ATM Switch Design for Private Networks).*

With rate-based schemes there is no simple rule to determine the size of the buffers needed, and there is no necessity to use Per VC Buffering. In fact, many rate-based implementations do not use Per VC Buffering, but instead opt to share one large FIFO buffer randomly among all VCs, consequently destroying fairness across VCs, and exposing the network to degradation from unruly or misbehaved VCs.

The proper sizing of buffers for a rate-based flow control mechanism depends on many factors:

- Whether the rate-based mechanism is end-to-end or hop-by-hop,
- The expected delays for rate reduction messages to reach each traffic source that is contributing to the congestion,
- How fast and by how much each traffic source will actually slow down,
- The number of sources that will begin sending new data in the next few moments
- How much data each new traffic source will likely send,
- Distances involved (diameter of network) and number of switches in network.

It is clear there is no way to predict with certainty the required buffer size for rate-based mechanisms. And, even with very large buffers, there is no way to guarantee zero cell loss under congestion using rate-based flow control.

In general, switches supporting only rate-based mechanisms will have to have larger buffers, to guarantee acceptable performance with bursty traffic sources. Interestingly, many WAN or Telco switches with rate-based flow control actually have smaller buffers than credit-based switches. They are counting on the use of traffic aggregation across many sources and traffic shaping at inputs to the network to eliminate burstiness, and they sometimes use only simple FIFO buffering schemes.

6.18. Why cannot higher level protocols (e.g., TCP) provide flow control for ATM?

While it is true that TCP and other higher layer protocols often do have their own flow control and that use of these protocols can help avoid cell loss in ATM, they are not a complete solution.

First of all, there are many widely-used computer applications that do not use a reliable higher layer protocol like TCP, hence they have no flow control. (e.g., Network File Services, NFS, which uses UDP). Secondly, the flow control mechanism in TCP, namely the "window size", dynamically varies depending on how well things are going and works only on an end-to-end basis between computers. If things are going well the window size increases, which means more data is sent without acknowledgment, perhaps many thousands or even tens of thousands of bytes. It often keeps increasing until problems are encountered (such as lost data), whereupon re-transmissions occur—not just of the lost packet, but of all data outstanding! TCP quickly ramps the window size back down to throttle the flow. This can cause an oscillating behavior that wastes bandwidth and prevents the user from attaining maximum sustained throughput end-to-end. TCP was designed for slower networks, and is not well optimized for the high speed of ATM networks.

If only one VC were running through a switch and the file being transferred was large enough (many Mbytes), TCP/IP would eventually find a "happy middle ground" and run smoothly. However, an ATM switch may have many TCP/IP sessions (as well as other non-TCP sessions) through it. This means the behavior of one session would affect others. For example, as a new session ramps-up an existing session might have cell loss imposed on it, causing it to ramp down, which allows the first to ramp up further, until the old one ramps back up and causes cell loss on the new session.

Meanwhile, certain TCP sessions may actually time-out if repeated cell loss is encountered, necessitating a re-initialization of the TCP connection, adversely affecting users with noticeable delays. Non-TCP sessions (e.g., UDP) would of course suffer cell loss and/or application level time-outs and disconnects of their own, as they are whipsawed by the impact of other VCs.

6.19. Doesn’t the Cell Loss Priority (CLP) bit help with congestion avoidance?

Yes, but the CLP bit (in the header of the ATM cell) typically applies mainly to VBR services, not ABR services. (It does not help with ABR flow control.) The CLP bit is set by the network when the Sustained Cell Rate (SCR) and Maximum Burst Size (MBS) in a VBR traffic contract is exceeded; it indicates that cell is “discardable” in the event of congestion. Most switches and adapters do not support CLP for ABR. (See Section 4: *Categories of Service in ATM*).

6.20. What is traffic shaping and where is that important?

Traffic shaping is used for VBR and CBR services, not ABR or UBR services. It should be done by the end-station itself before placing traffic into the network, or by an intermediate edge-device, to help insure that the traffic contract (e.g., maximum burst size and SCR) is not violated.

ATM switches themselves should also provide traffic shaping before handing-off traffic to the next switch or destination. This is important so that small bursts of VBR or CBR traffic do not build up in the network, and so that Cell Delay Variation (CDV) specifications are met end-to-end.

6.21. What is traffic policing and where is it important?

Traffic policing is the process of enforcing the traffic contract. Like traffic shaping, it is applicable only to CBR and VBR services, not ABR or UBR, and may be applied by both the end-stations themselves and the switches in the network.

Policing is accomplished by discarding cells in excess of the traffic contract, or by marking cells as “discardable” (using the CLP bit) as in the case of moderately excessive VBR traffic.

End-stations requesting VBR or CBR services should police themselves (or risk cell loss). Switches in the network should also police VBR and CBR services if they do not want users of those services to consume more bandwidth than they requested (and agreed to pay for), and also if they do not want buffers in their switches consumed by VCs that are violating their CBR and VBR contracts.

6.22. What are GCRA and the Leaky Bucket Algorithms?

GCRA (Generic Cell Rate Algorithm) is a means of determining if a VBR or CBR traffic stream is conforming to its traffic contract given its traffic parameters (e.g., SCR, PCR, MBS for VBR). The algorithm for VBR is often expressed in terms of cells entering a “leaky bucket” of size roughly equal to the maximum burst size (MBS), and being “leaked” onto the output line at the constant (SCR) rate. If the bucket overflows, cells are discarded. The “Dual Leaky Bucket Algorithm” adds additional sophistication to handle enforcement of all VBR service parameters and traffic parameters.

6.23. What is Digital’s support for traffic shaping and traffic policing?

Digital in GIGAswitch/ATM and other ATM switches provides traffic shaping of both VBR and CBR traffic, fully reshaping the cell flow at each switch. In this way GIGAswitch/ATM provides consistently low CDV for both CBR and VBR. Traffic policing of the CBR and VBR services is provided by simply placing limits on the maximum buffer space a CBR or VBR VC can use in proportion to its PCR. Full support of the “dual leaky bucket algorithm” is not planned. It is not needed on private networks where strict policing of user’s VBR traffic patterns (e.g., burst size in relation to PCR) is a moot issue and imposing unnecessary cell loss on users is generally undesirable. Also VBR itself is not generally used within a private network environment. (See Section 4: *Categories of Service in ATM*).

7. Using ATM with Traditional LANs

7.1. *What are the various ways to carry traditional LAN traffic on ATM networks?*

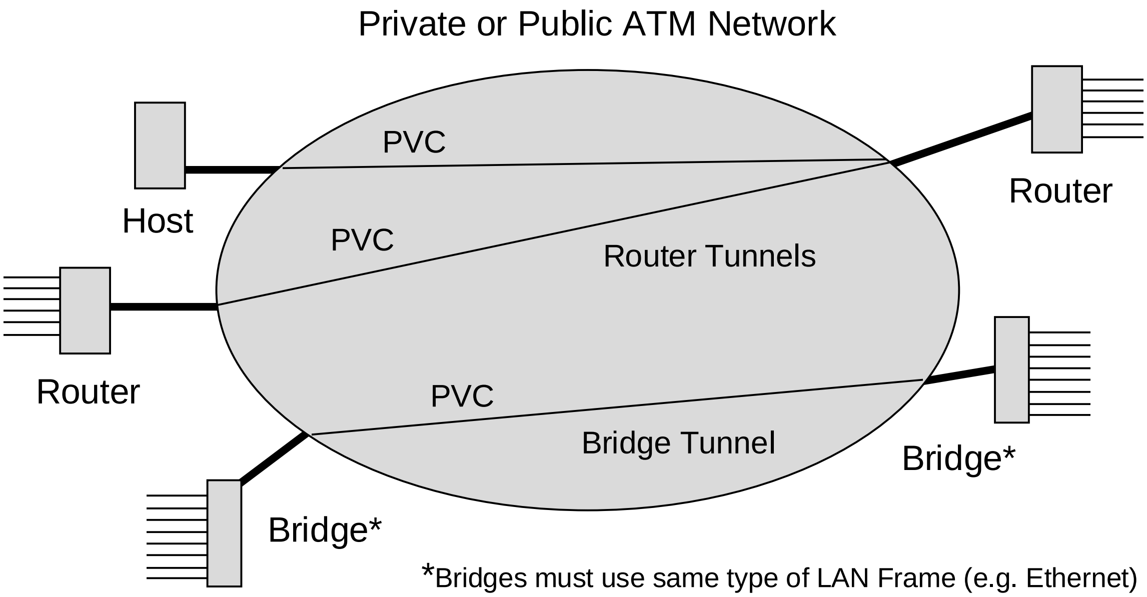
There are three different standard ways to carry traditional LAN traffic on ATM networks:

- Multiprotocol Encapsulation (RFC 1483) -- a.k.a. "Bridge & Router Tunnels"
- Classical IP (RFC 1577)
- LAN Emulation (LANE)

All three methods use AAL-5 (ATM Adaptation Layer 5) which is the most efficient way to carry traditional data packets with a series of ATM cells.

Bridge & Router Tunnels (RFC 1483) -- completed July 1993 -- actually describes several different approaches to multiprotocol encapsulation, only one of which is widely used, called "LLC encapsulation". LLC encapsulation uses a "Link Layer Control" header to allow multiple protocols and/or LAN frames to share a single PVC. However, it must be configured as either a "tunnel" between bridges of the same type (e.g., both Ethernet, both FDDI, etc.) or as a tunnel between two routers/hosts. (It has no relation to "VP Tunnels"). This is because different encodings are specified for each different LAN type (e.g., Ethernet, Token Ring, FDDI, SMDS, etc.) for use between bridges ("bridge tunnels"), and still different encodings are given for ISO vs. Non-ISO protocols for use between directly attached hosts and/or routers ("router tunnels") without the LAN frame present. In all cases, ATM is used like a "private line" with a PVC for the tunnel. This approach does not take advantage of ATM's powerful SVC and routing capabilities. However, this does make Bridge and Router Tunnels a viable option on Public ATM networks that may only support PVCs. **See Figure 21**.

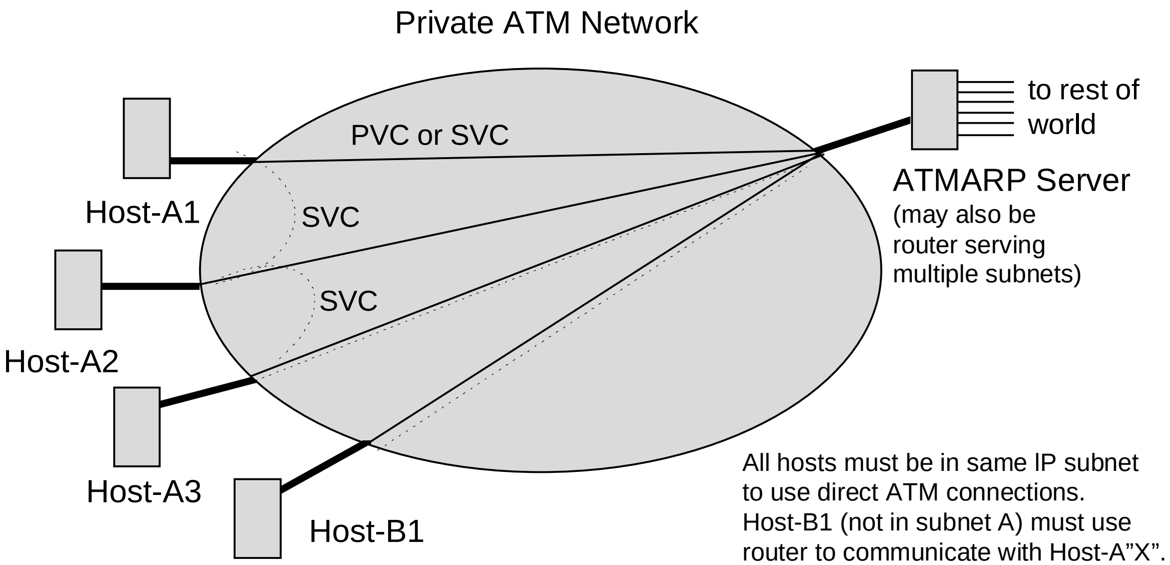
Figure 21
Bridge and Router Tunnels (RFC 1483)


Private or Public ATM Network
PVC
Host
PVC
Router Tunnels
Router
Router
PVC
Bridge Tunnel
Bridge*
Bridge*
*Bridges must use same type of LAN Frame (e.g. Ethernet)

Support for RFC 1483's Bridge & Router Tunnels is something that is put into bridges and routers that have ATM uplinks. The ATM network itself does not need to know or do anything to support them.

Classical IP (RFC 1577) **--** completed January 1994 **--** was developed largely to take advantage of ATM's dynamic SVC capabilities to support directly attached ATM stations. Classical IP works only for IP (and ARP). Classical IP uses the LLC encapsulation method of RFC 1483 for router tunnels, but adds to it the ability of ATM end-stations to dynamically establish connections with any station in the same logical IP subnet that is also directly connected to the ATM network. Each station maintains a local cache that maps IP address to ATM address. This local cache is populated with the assistance of an ATMARP Server, which registers the mapping between ATM address and IP address for all stations as they join the subnet. It then also responds to ATMARP requests from any station in the subnet. **See Figure 22.**

Figure 22
Classical IP (RFC 1577)


Private ATM Network
PVC or SVC
SVC
SVC
Host-A1
Host-A2
Host-A3
Host-B1
to rest of world
ATMARP Server
(may also be router serving multiple subnets)
All hosts must be in same IP subnet to use direct ATM connections. Host-B1 (not in subnet A) must use router to communicate with Host-A"X".

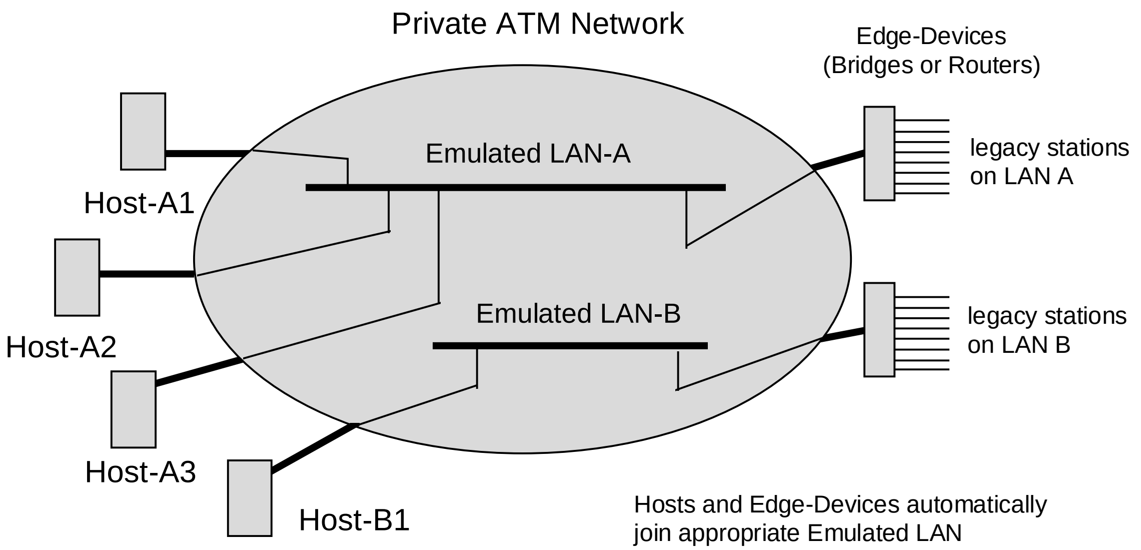
In this way, the broadcast nature of a traditional LAN-based IP subnet and its response to ARPs is emulated by using a point-to-point VC to the ATMARP Server. When a station wants to talk to another IP address in the same subnet, it can query the ATMARP server if it does not already know the ATM address from its local ATMARP cache. If it wants to talk to a station outside the subnet, it must go through an IP router—often installed in the same edge-device as the ATMARP Server.

Support for Classical IP is something that is associated typically with either routers or IP hosts (e.g., workstations, servers). The ATM network itself need not do anything to support it other than support SVCs to use Classical IP in its intended way. Classical IP can also be done in a network or with devices that only support PVCs. This is done by establishing a full-mesh of PVCs between all members of the subnet in advance. However that imposes management burdens and reduces flexibility significantly. It also does not scale well to large networks.

If multiple subnets are installed on the ATM network, there may be many ATMARP Servers. If each of these is set-up as a router as well, then communication between subnets on the ATM backbone can take place via the various router/ATMARP Servers. However, this is a less than ideal way to integrate traditional routing functionality with ATM. (See Question 7.7).

LAN Emulation (LANE) -- completed by the ATM Forum in May 1995 -- was developed to allow ATM networks to appear as one or more broadcast LANs for any protocol, integrating smoothly with existing LANs. LANE also takes full advantage of ATM's SVC capabilities to allow additional LANs or new directly-attached ATM stations to join the appropriate emulated LANs dynamically and automatically. **See Figure 23.**

Figure 23
ATM Forum LAN Emulation (LANE)
Logical Representation



Hosts and Edge-Devices automatically join appropriate Emulated LAN

LANE involves several components (none of which are shown in the above logical representation):

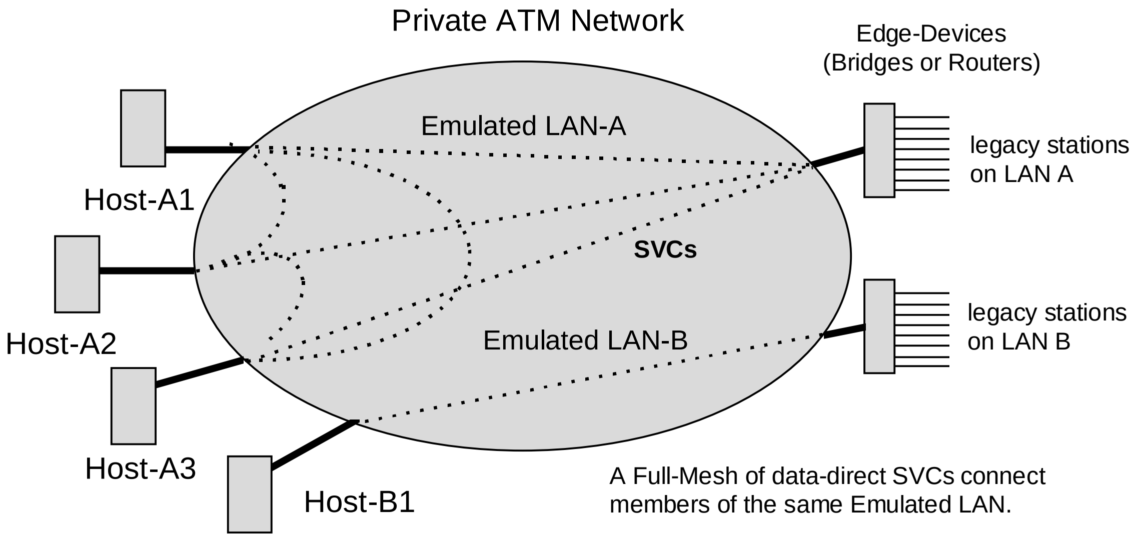
- LAN Emulation Configuration Server (LECS) -- controls the total environment
- LAN Emulation Servers (LES) -- one for each emulated LAN
- Broadcast and Unknown Servers (BUS) -- one for each emulated LAN
- LAN Emulation Client (LE Client or LEC) -- one or more for each station or edge-device

The LECS determines which stations or edge-devices join which emulated LANs (based on their address). It holds the master configuration database.

The LES registers each participant into its assigned emulated LAN. This registration may include legacy LAN destinations (e.g., Ethernet MAC addresses) accessed via an edge-device, as well as ATM addresses of directly attached stations. The LES also answers queries to resolve the mapping between a MAC address and an ATM address (so-called "LE-ARPs"). The BUS function typically resides with the LES and handles all broadcast, multicast or unknown unicast traffic out to other LE Clients. This is done using ATM's point-to-multipoint VC capability and is done very efficiently.

All unicast traffic with known destination addresses (MAC and ATM address) is sent by an LE Client directly to its destination LE Client via a "data direct" SVC. Using LANE a full-mesh of data direct VCs can be automatically established between all ATM end-stations and edge-devices. **See Figure 24.**

Figure 24
ATM Forum LAN Emulation (LANE)
Physical Representation



While a full-mesh of SVCs may be established, if the communications between any two ATM attached devices is not need no SVC is set up. Likewise if communication goes quiet for extended period the data direct SVCs between those two devices can be timed out. When communication needs to be re-established between them a new SVC is set up (typically within a few tens of milliseconds). In this way SVCs are re-used and very large emulated LANs can be built.

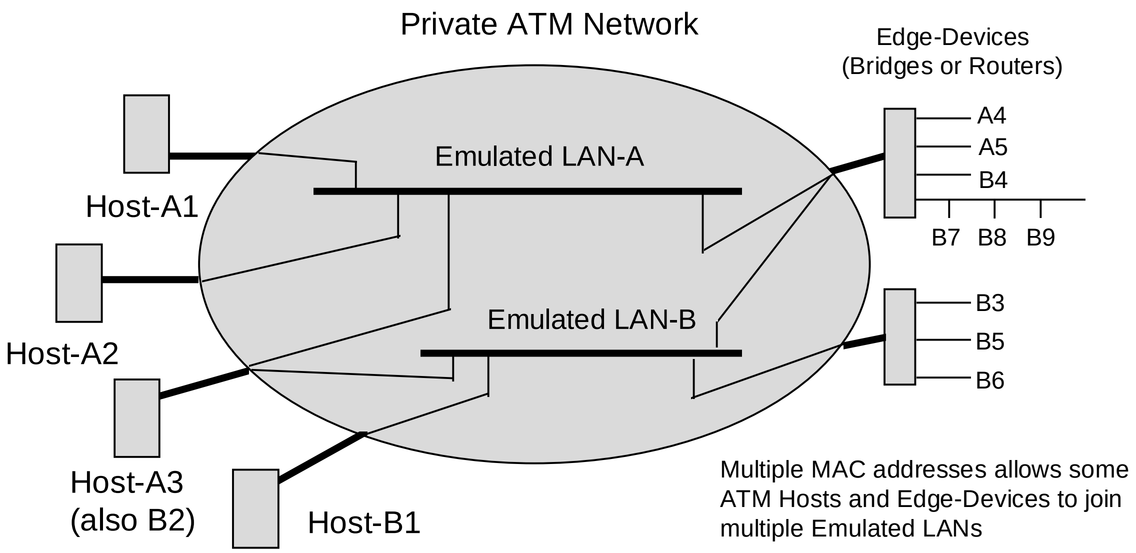
The actual location of the LECS, the LES, and the BUS is determined by vendor implementation and user configuration options. They may be located inside of the ATM switches themselves, in dedicated attached workstations or in edge-devices (switches or routers). (See Question 7.9).

The LE Clients always reside in each directly attached host or edge-device (typically in the driver software for the adapter or ATM interface). It is the entity that handles all tasks associated with participating in an emulated LAN. If an edge-device or end-station has more than one LE Client in it, then it can participate in more than one emulated LAN. This is particularly useful for edge-devices since the legacy LAN stations they represent can then be members of many different emulated LANs. A directly attached host can also be a member of more than one emulated LAN—and do so over a single physical interface if its ATM adapter supports multiple MAC addresses and its operating system supports multiple LAN connections. **See Figure 25.**

LANE defines two encodings for LAN frames: one for Ethernet and one for Token Ring. These are not interoperable. The encoding for Ethernet, though, can also be used to connect FDDI networks to ATM backbones with full-sized FDDI frames of 4500 bytes. These two encodings are different than that proposed by RFC 1483. While both include the MAC frame, LANE includes a 2-byte LAN Emulation header and replaces the 4-byte LAN Frame CRC with the AAL 5 CRC. This actually reduces overhead. Whereas, "Bridge Tunnels" include an 8- to 12-byte LLC header depending on LAN type, and may or may not replace the LAN Frame CRC—resulting in higher overhead.

LANE is, by far, the most powerful of the three alternatives.

Figure 25
LANE with Multiple LANs per Edge Device



7.2. *What are the advantages of using LANE vs. Classical IP?*

LANE is the most effective method of using ATM as a backbone for existing LANs while also allowing directly attached ATM stations to participate in those LANs dynamically and automatically.

Classical IP (RFC 1577) is limited to logical IP subnets composed of stations directly attached to the ATM backbone. It does not extend out to stations on legacy LANs, except via a router that is itself attached to the ATM backbone, in which case the stations must be members of a different IP subnets.

Classical IP of course is also limited to IP, whereas LANE supports all protocols (e.g., IPX, OSI, DECnet, etc.) including non-routable protocols (e.g., NetBIOS or LAT).

7.3. *What are the advantages of using LANE vs. Bridge or Router Tunnels?*

While Bridge and Router Tunnels (RFC 1483) do allow ATM backbones to be used by devices on existing LANs, they are not as dynamic or flexible, requiring much more administrative support. For instance, consider a network with say 10 different edge-devices (such as bridges/switches with ATM uplinks). To interconnect them fully over the ATM backbone requires 45 (10 x 9/2) different PVCs to be set-up and maintained. When a new edge-device is added many new tunnels (and PVCs) would have to be added to achieve full pair-wise inter-connectivity. If there is an outage in the ATM network, the PVCs will not re-route around the failure. The same weaknesses also apply to router tunnels.

With LANE all of this work is handled automatically. Data direct SVCs are set-up automatically between all pairs of edge-devices and stations in the same emulated LAN creating *a full-mesh topology*. SVCs are also automatically set-up between each LE Client and the LE Configuration Server, the LES and the BUS to be used as needed. The membership rules of the LE Configuration Server determine which LE Clients join which emulated LANs, i.e., to which LES and BUS they connect, and therefore to which other LE Clients they connect. If there is a failure in the network, all SVCs will re-route around the failure dynamically, if paths are available.

The other major advantage of LANE is that a directly attached ATM station can also be a member of one or more emulated LANs. Therefore, as an example, a big file server or host can be given a 155 Mbps access line or two directly off the ATM backbone. All legacy LAN stations can then access it even if they are in many different emulated LANs, as long as this big file server is a member of each emulated LAN. LANE can do all this via just one physical ATM interface, if desired, as long as it supports multiple LE Clients. With bridge or route tunnels this would not be practical or may not even be feasible. (Computer systems often do not support RFC 1483 Bridge Tunnels, preferring instead to support Classical IP and LANE.)

Finally, each emulated LAN appears to an edge-device as a single LAN connection or port (even though it may be using of hundreds of SVCs to connect directly to hundreds of other edge-devices or hosts). Whereas a bridge or router tunnel appears as one connection -- and, of course, connects to only one other device. This makes it nearly impossible to build large networks with Bridge or Router tunnels since most edge-devices do not support hundreds of bridging or routing ports over a single ATM port. In fact many support only one or a handful of such tunnels.

7.4. *What are the advantages of ATM Forum LANE vs. proprietary forms of LAN emulation?*

ATM Forum LAN Emulation (LANE) is particularly strong in terms of its flexibility to support multiple distinct emulated LANs in a single ATM backbone and allow both edge-devices and directly attached ATM stations to be members of one or more of these emulated LANs. LANE is also strong in its dynamic re-configurability. This is not true of most vendor-specific proprietary LAN emulation schemes that were developed prior to the ATM Forum standard.

The other major advantage is that interoperability between edge-devices, adapters and switches of multiple vendors is obtained. There is no single vendor lock-in. At recent multi-vendor interoperability tests, many different ATM vendors successfully interoperated using LANE. This number is growing monthly. (Digital's LANE implementation was particularly successful at interoperating with other vendors. It worked with over 30 different vendors.)

To further promote LANE interoperability, Digital has released its source code for the ATM Forum LANE Client into the public domain (access via Digital's Home Page on the Internet). This is being used as a reference implementation by other vendors or institutions interested in doing their own implementations.

7.5. *What ATM "category of service" is used for Tunnels, Classical IP and LANE?*

Any ATM category of service can be used (CBR, VBR, ABR or UBR), in principle. However, UBR or preferably ABR services with a high-quality flow control mechanism are recommended. (See Section 4: *Classes of Service* and Section 6: *Flow Control and Traffic Management*). Use of CBR service for traditional computer-based data applications and LANs is problematic as described in Section 4.

7.6. *What is MPOA? What problem is it solving? And what is its status?*

MPOA stands for "Multi-Protocol Over ATM". MPOA is an effort that has been underway within the ATM Forum for some time now to accomplish many of the same goals as LAN Emulation: namely, the ability to support any protocol conveniently over an ATM backbone, regardless of whether the station is directly attached or attached via some edge-device. Where MPOA goes beyond LANE is in trying to address how traditional routers for IP and other protocols should be integrated into an ATM backbone environment. (See Question 7.7).

The MPOA effort, however, has bogged down in many complexities and may not be completed until 1997 or may come out looking very different than its current "rough draft". Furthermore, it has not taken advantage of the work completed for LANE. Instead, much work was actually done in conflict with LANE in several important areas (e.g., encoding of packets).

At the October 1995 meeting of the ATM Forum, it was agreed that MPOA should take advantage of LANE. At the December 1995 meeting, proposals for how to build upon LANE were discussed, but no agreement was reached. MPOA is now moving to be move in line with LANE and will most likely be built upon features proposed for LANE V2.

7.7. How are traditional routers integrated into an ATM backbone environment?

With LANE and Classical IP, traditional routers are on the edges of the ATM network and ignorant of the ATM network topology. (The same is true of Bridge and Router Tunnels, of course). This means when data traffic needs to pass through a router (e.g., because source and destination are in different subnets), it might need to exit the ATM backbone, pass through the router, and then re-enter the ATM backbone to get to its destination. This is not a very effective way to use routers with an ATM backbone.

With Classical IP, this is the standard way in which routers are used with an ATM backbone, and has given rise to the concept of the **"one-armed router",** defined as a router with a single high-speed port, plugged into the ATM backbone through which traffic both enters and leaves. The one-armed router is also set-up usually to be the Classical IP ATMARP server for several logical IP subnets on the ATM backbone. (Review Figure 22 -- but imagine connections to "rest of world" removed.)

With LANE, it is possible to arrange the topology such that legacy LAN traffic between two subnets can use a router either before it enters or after it exits the ATM backbone. This is particularly simple if the edge-device is both a bridge/switch and a router. If the traffic is within the same emulated LAN (e.g., same IP subnet), it need not use a router at all -- it is simply switched directly to its destination using LANE. However, communications between two directly attached ATM stations in different emulated LANs (e.g., different IP subnets) would have to make use of a router.

It may also sometimes be useful with LANE to configure an emulated LAN dedicated solely to interconnecting multiple routers, with full pairwise direct connectivity. In this way a cluster of routers (up to a few hundred perhaps) could all be joined into one emlated LAN each with their own 155 Mbps connection using multi-Gigabit ATM switches to communicate among them. This is simplest way to build a very high-performance routing backbone using standards today and assure that no end-station is more than two router hops from any other.

7.8. How is the concept of a "route server" for ATM networks different?

A **"route server"** is similar to a one-armed router on the ATM backbone. In some implementations it may be no more than a Classical IP ATMARP server with routing capabilities (discussed above).

In other implementations it may be set-up to maintain a mapping between ATM address and legacy LAN addresses (both MAC and layer 3 network addresses) and architected to communicate (via some proprietary mechanism) with specialized **"packet forwarders"** distributed around the edges of the network as part of some edge-device. These "packet forwarders" appear to be traditional routers to the legacy LAN devices (e.g., Ethernet stations) attached to them. However, only the router server would fully support routing protocols like RIP and OSPF while the packet forwarders would not.

The basic idea is to make the packet forwarders "fast and dumb" while letting the route server do all the thinking about where to send the data as new destination 'requests' arise (i.e., as packets with new destination addresses arrive). While this approach has some appeal, it can suffer from several serious weaknesses:

- The route server (and its access line and I/O port on the ATM backbone!) is a critical single point of failure for the entire network.
- The route server or its access line can become a bottleneck, degrading performance on the entire network; (e.g., delays communicating with the route server can cause buffers in the packet forwarders to overflow.)
- If the route server is located off-site, vulnerability and performance issues are exacerbated. In a multi-site network, only one site can have a route server (in these architectures).

- The approach is proprietary; packet forwarders and route servers from different vendors cannot be intermixed.
- Packet forwarders are not equipped with enough intelligence to handle even the most basic of local routing tasks for unfamiliar destination addresses without communication to the central route server.
- This approach is often implemented in a fashion that is incompatible with LANE.

These problems can all be avoided by using full-function distributed routing capabilities located in ATM edge-devices (as discussed for use with LANE).

7.9. When using LANE where should the LES, BUS and LECS functions be located?

There are three choices for where to locate these functions: either in the ATM switches themselves, in some special end-station on the ATM backbone (e.g., a dedicated workstation), or in an edge-device (e.g., a router or switch with ATM uplink).

There are several advantages to locating these directly inside your ATM switches:

- It will have the best possible bandwidth and connectivity to all your end-stations and edge-devices, eliminating the bottleneck of a single access line to the edge
- It eliminates the need for the purchase and maintenance of additional edge equipment
- It will have the same high-availability environment as the ATM switch itself (redundant power, fans, UPS, protected machine room, etc)
- It improves the overall availability of the LAN emulation service on the ATM network, since fewer separate pieces of equipment are involved.

The LECS should especially be located in a high-availability location.

7.10. Can there be multiple LES/BUSes serving the same emulated LAN?

Under the current LANE V1.0, this is not possible. However, this will be included in LANE V2.0. The advantages of this are two fold: it gives the option for adding additional BUS performance for environments with a lot of unicast or broadcast traffic, and it will provide LES and BUS redundancy for emulated LANs.

7.11. Can there be multiple LE Configuration Servers in a single ATM network?

Yes, there may be multiple LE Configuration Servers (e.g., for back-up) in a single ATM network. They would usually be set-up to have identical configuration databases, although one can imagine a scenario where different configurations might be used for different times of the day or week or month (thus providing a simple means to change the mapping of which user stations are members of which emulated LANs).

7.12. How do ATM Emulated LANs relate to Virtual LANs?

Virtual LANs (VLANs) are a generic industry term that encompasses any approach (proprietary or otherwise) that provides software-based management control over the creation of LANs within a backbone network and control over which users are connected to those LANs. In this context a LAN is typically defined as a “broadcast domain”—meaning the set of stations to which a MAC address broadcast (e.g., an ARP message) is distributed—which is the standard definition.

Today, ATM Forum LAN Emulation (LANE) offers the only standard method of provisioning multiple VLANs over a single physical network of switches and lines—in this case, an ATM network. Proprietary VLAN techniques for FDDI backbones and other networking technologies (e.g., Fast Ethernet) have been invented and various approaches are being proposed to standards bodies (e.g., IEEE 802.1). These proposals, however, are only now starting to converge on a standard frame format, with a final standard sometime in 1997.

With ATM's tremendous scalability (both in bandwidth and geography) and its standards-based approach to VLANs through LAN Emulation, it is clearly the preferred technology in which to implement VLANs. When implemented in conjunction with edge-devices (e.g., Ethernet switches) that support software configurable "bridge groups", ATM LANE provides a very effective VLAN solution for legacy LAN stations. Review Figure 25.

7.13. How is Digital supporting LANE and these other LAN integration options?

Digital supports all three options: Bridge and router tunnels, Classical IP and LANE. Tunnels are supported on the DECNIS router family, the GIGAswitch/FDDI platform and the DECswitch 400. Classical IP is supported on the DECNIS router family, the DECswitch 400 and Digital Equipment Corporation UNIX systems and Digital's Windows NT and NetWare support. (OpenVMS support will also be provided). The LE Client is supported on Digital's ATM adapters today and will be supported on the DECswitch 400 by mid-1996. Digital is also offering LES/BUS and LE Configuration Server capability directly on GIGAswitch/ATM today, for increased availability and performance. Other options for the LES and BUS will be offered for customers that may not have a GIGAswitch/ATM system include new low-end ATM switches for the DEChub 900. As new edge-device products come out they will support many LE Clients per device, to allow membership in multiple emulated LANs.

Digital's LANE implementation has been tested in numerous interoperability Forums and has successfully interoperated with over 30 other vendors. To further promote LANE interoperability, Digital has released its source code for the ATM Forum LANE Client into the public domain (via Digital's Home Page on the Internet).

7.14. What is Digital's VLAN strategy with respect to ATM?

Digital is supporting VLANs using both ATM backbones and other backbone technologies (e.g., FDDI). In general, the more advanced VLAN capabilities will tend to be available for ATM backbones, given its scalability and configuration flexibility. LANE plays a central role in our VLAN strategy for ATM. Each VLAN on an edge device is mapped to a distinct emulated LAN using LANE in order to extend the VLAN across the ATM backbone. (Review Figure 25). This is the only standard way to implement VLANs today and Digital is one of the first vendors to promote this approach to VLANs. It is also the only approach that allows the host to be directly connected to the ATM backbone. For a more complete discussion of Digital's VLAN strategy, please refer to the white paper on ***enVISN***—enterprise Virtual Intelligent Switched Networks.

7.15. What is "IP Switching" and Digital's strategy with respect to this new technology?

"IP Switching" is a set of techniques to allow the use of high-speed switching fabrics to carry traffic directly between IP subnets, by-passing routers, when the traffic meets certain criteria. These techniques require implementation on routers such that the routers being by-passed cooperate in the process for specified classes of traffic. These techniques remove the typical throughput bottleneck associated with routers, since traffic is now forwarded at multi-Gigabit switching speeds.

When fully integrated with ATM it is possible to use newly proposed IP Switching protocols (combined with traditional IP routing protocols such as OSPF) as a replacement for P-NNI in the set-up and handling of SVCs across the ATM network, tightly integrating IP networking with ATM networking—two worlds heretofore largely isolated from each other. This can have several benefits for organizations wanting to base all their networking upon IP, and wanting to build and support large ATM networks with the same tools and techniques they have used successfully for their traditional (non-ATM) IP networks. ATM networks based on IP switching are particularly effective for large multimedia and multicast applications, demanding the highest performance.

The newly proposed IP Switching protocols are “open” and available for other vendors to implement.

Digital’s strategy with respect to IP switching is to support it both in the context of ATM Forum standards such as P-NNI (and LANE), as well as in the pure case where P-NNI is replaced with IP switching and traditional OSPF protocols.

8. Using ATM for Multimedia Applications

8.1. I can do multimedia applications on my LAN today. Why do I need ATM?

Traditional LANs (e.g., Ethernet, Token Ring, FDDI) all operate effectively like ABR services, which tend to work well only for applications that can tolerate either high amounts of latency and buffering, or high amounts of jitter—therefore poor quality.

Using CBR services, ATM provides the low-latency and guaranteed bandwidth required by high-quality real-time multimedia applications which is not attainable over traditional LANs. While there is plenty of bandwidth available with other backbone technologies such as FDDI (or even Fast Ethernet), the way in which this bandwidth is managed, on the wire, in the switches and in the end-system adapters and drivers, does not allow for multimedia applications such as video conferencing, PC telephony, or other highly interactive multimedia communications. Even one-way (non-interactive) multimedia applications (e.g., video on demand) can be problematic without benefit of the CBR services of ATM. (See Question 8.2).

8.2. What ATM category of service should be used for multimedia applications?

This depends on the nature of the application. If it is a real-time data stream that is loss-sensitive (such as MPEG video or voice calls where quality matters), CBR services should be used. If the encoding of the information is not sensitive to cell loss and varies smoothly over time, VBR could be considered, but only if VBR service actually offers a lower cost option than CBR and does not degrade quality.

In private ATM networks CBR costs no more to use than VBR services when unused CBR capacity is made available to ABR services, such as with Digital's ATM products. CBR services will guarantee zero cell loss (if the PCR is not exceeded), and in general will provide a lower **Cell Delay Variation (CDV)** resulting in less jitter and a higher quality transmission. (See Section 4: *Categories of Service*).

Since ABR does not provide any guarantee on CDV, Maximum Cell Delay, or throughput, it is not useful for real-time multimedia traffic, unless end-systems can provide large buffers (to smooth over grossly uneven and delayed cell arrivals). At the high data rates required for high-resolution, full screen, compressed video (e.g., 4 to 8 Mbps), these buffer requirements can become very large if using ABR. It is quite conceivable that ABR services could be throttled back for several seconds, requiring buffers in the end-systems of several Mbytes. This then imposes additional significant latency, making ABR only good for "one-way" multimedia. Even that is questionable, given the large buffer requirements and ever present chance of the buffers overflowing or underflowing (i.e., running dry at the receiver).

8.3. What ATM Adaptation Layer (AAL) should I use?

Generally speaking AAL-5 and AAL-1 are the best choices. The decision on "which one should be used where" though depends on a number of factors, including both the application type and the devices involved. AAL-5 is the "simple and efficient adaptation layer" designed for computer networking. AAL-1 is the synchronous adaptation layer designed for carrying voice trunks (e.g., T1/E1 or T3/E3) over ATM.

The only other ATM Adaptation Layer fully defined and implemented by vendors is AAL-3/4. It was intended to be used by Frame Rely and SMDS and other delay-insensitive variable rate or bursty data traffic. In most cases it is now being passed over in favor of AAL-5 which is simpler and more efficient. AAL-2, intended for variable bit rate video, never got off the ground and is not offered by vendors.

Various technical committees of the ATM Forum are now putting forth standards for how different types of media (MPEG video, high quality audio, 3 KHz voice etc.) should be adapted to the ATM environment. For instance, the Audiovisual Multimedia Services (AMS) technical committee has put forth a proposal for Video on Demand over ATM. The specification details how to pack MPEG-2 transport streams into AAL-5 and the QoS parameters required for those applications.

Another technical committee is proposing how legacy voice services can be originated or terminated at a native ATM end-station. Called VTOA—Voice Telephony over ATM—this proposal includes a specification for how signaling of narrowband ISDN and the Public Switched Telephone Network (PSTN) calls are passed into and out of an ATM network to achieve full interworking between ATM SVCs and traditional telephone calls. This committee has proposed that voice calls should be carried as AAL-1 out to ATM end-stations. *This approach has some drawbacks, though, for use with standard PC or computer systems. Essentially all computer system adapters and operating systems are designed to handle AAL-5, and not AAL-1!* While AAL-1 was originally designed for voice over ATM, to date it has been used only for point-to-point trunks using **ATM Circuit Emulation Services (CES)**, between specialized edge-devices (e.g., ATM Concentrators). These devices are designed to connect sychronous equipment to an ATM network.

8.4. What are Circuit Emulation Services (CES), and where are they useful?

Circuit Emulation Services (CES) provides a CBR VC through an ATM network that performs just like a synchronous, point-to-point private line (e.g., T1 or E1) including all channelization and telephony signaling, where required. This is useful when one wants to make a private ATM backbone network handle connections between devices like Telephone Company Central Office switches, PBX's, T1 multiplexers, video codecs or modems that only accept these types of synchronous connections. This is usually done via an **ATM concentrator,** which has appropriate hardware interfaces on the "user side" of the box and which multiplexes many such connections over a single ATM port on the "network side" of the box. The ATM port typically operates at T3/E3 (45/30 Mbps) or OC-3c (155 Mbps) speeds, and thus can handle many lower speed (e.g., T1/E1) circuit emulation connections.

Public carriers may also use CES over ATM to provision traditional private data lines at 9.6 Kbps, 56 Kbps, 64 Kbps, fractional T1/E1 and even fractional T3/E3, using an ATM Backbone built on top of OC-3c (155 Mbps), OC-12c (622 Mbps) or even higher speed fiber optic based ATM links.

CES requires AAL-1 to maintain synchronous operation end-to-end across the ATM network, and make ATM function very much like a TDM multiplexer. CES also requires low Cell Delay Variation (CDV), low Maximum Cell Transfer Delay (CTD) and guaranteed through-put. Therefore, CBR services are mandatory.

While CES is useful for interconnecting "legacy" synchronous devices, CES is not generally used for connections directly into computer systems via their ATM adapters, since there are more effective ways to use ATM for computer communications. Also, as mentioned, AAL-1 is not supported in computer system adapters or operating systems today, except possibly in some rare cases.

8.5. Do Digital's ATM products support Circuit Emulation Services (CES)?

Digital's GIGAswitch/ATM supports CES. The Cell Delay Variation (CDV) requirements for CES from Bellcore (in the US) are 750 microseconds for both T1 and T3. GIGAswitch/ATM provides a CDV of less than 400 microseconds for T1 and less than 12 microseconds for T3. Maximum Cell Transfer Delay (CTD) requirements can also be met given the GIGAswitch/ATM's latencies which are as low as 20 microseconds for a CBR VC for high bandwidth VCs (e.g., 45 Mbps). For low bandwidth CBR VCs' (e.g., 1.5 Mbps), latencies may be as high as a few hundred microseconds, however, configuration options can lower this considerably .

Interfaces to the GIGAswitch/ATM switch for CES, however, must be provided using an ATM concentrator (or other ATM switch or edge device) that provides appropriate physical interfaces, such as those available from ADC/Kentrox, Digital Link, GDC, Onstream Networks, Premisys, Litton Industries, and others.

8.6. How is multicast accomplished in ATM?

Multicast is accomplished by branching at each node in the network as needed. This conserves bandwidth since only those branches with stations addressed by the multicast have to carry the multicast data flows. Multicasts, in theory, can be either CBR, VBR, ABR, or UBR however particular switch implementations may not allow all possibilities.

ABR multicasts are problematic with rate-based flow control and are not likely to be offered by any vendor, using end-to-end, rate-based flow control mechanisms. This is because merging of the backwards RM cells as they return to the source poses many difficult implementation problems. The net effect will not provide useful congestion status information when a variety of different destinations are involved. Hop-by-hop, credit-based flow control (such as FLOWmaster and QFC) will work quite nicely for multicast ABR applications.

8.7. What support does Digital provide for ATM multicasts?

Digital provides support for multicast using either CBR, VBR, ABR or UBR services (and any AAL type) in GIGAswitch/ATM. For CBR and VBR multicasts, the CDV and Max CTD is maintained the same as for unicasts. For ABR multicasts the ability to send at full line rate with zero cell loss is maintained the same as for unicast, if all destinations have full line rate receive capacity. ABR multicasts will also work (but at lower throughputs) when destinations have various receive capacity or congestion in the access lines. This is because they can be implemented on a hop-by-hop basis using credit-based flow control mechanisms (FLOWmaster or QFC when available).

8.8. How should I use ATM to the PC desktop for video-on-demand applications?

We would recommend following the proposal of the Audiovisual Multimedia Services (AMS) technical committee of the ATM Forum, discussed above in question 8.3. That is, to use AAL-5 with CBR services, since video-on-demand is effectively a computer-to-computer real-time application. The video source is either stored on a computer system, or generated in real-time by a computer system because it must be significantly compressed.

8.9. How should I use ATM to the PC desktop for interactive multimedia conferencing?

This is a more difficult question. On the one hand, it is desirable to use AAL-1, as per the VTOA proposal discussed above, since this is how voice trunks were intended to be handled on ATM. On the other hand, since the application is one involving conferencing between PC desktops (and not synchronous devices like PBX's or modems), AAL-5 is really the more attractive option—especially given that AAL-1 is not supported in computer systems and adapters today (except possibly in special cases, and when it is it adds significant cost!). AAL-5 is also more attractive because there will be other information types besides voice involved in a multimedia conference (e.g., video, application windows with data, virtual white boards, etc.). In any case, though, CBR services should be used to assure a low CDV and guaranteed bandwidth.

8.10. When will I be able to run a single set of wires to each desktop for telephone service, data networking, interactive multimedia conferencing, and video-on-demand, all using ATM?

This is technically feasible today, and will become more widespread as standards and applications for doing this easily and inexpensively are developed in the next couple of years.

APPENDIX

Glossary of Common ATM Acronyms

ABR Available Bit Rate—a category of ATM service

CBR Constant Bit Rate—a category of ATM service

CDV Cell Delay Variation—a Quality of Service parameter for CBR and real-time VBR service.

EFCI Explicit Forward Error Indication—a standard part of ABR service providing a form of rate-based flow control

ELAN Emulated LAN—a “LAN-like” capability formed using LAN Emulation

GFC Generic Flow Control—a field in the ATM cell header and one standard for flow control.

LANE LAN Emulation—a standard for supporting legacy LAN traffic over ATM networks.

MBS Maximum Burst Size—a traffic parameter for VBR service.

MCR Minimum Cell Rate—a traffic parameter of ABR service.

PCR Peak Cell Rate—a traffic parameter for VBR and CBR service.

P-NNI Private Node to Node Interface—the standard interface between ATM switches in a private network

PVC Permanent Virtual Circuit—a virtual circuit (VC) established by network management.

QoS Quality of Service—a term usually used to refer to a set of specific performance parameters.

SCR Sustained Cell Rate—a traffic parameter of VBR service.

SVC Switched Virtual Circuit—a virtual circuit (VC) established on demand by an attached device.

UBR Unspecified Bit Rate—a category of ATM service

UNI User to Network Interface—the standard interface for connection to an ATM switch from a user device either directly attached host or some edge-device (bridge, router or concentrator).

VBR Variable Bit Rate—a category of ATM service

VC Virtual Circuit—the logical connection established across an ATM network used for the transport of cells.

VP Virtual Path—a logical path across a link within which there may be many VCs.

VPC Virtual Path Connection—an end-to-end VP across an ATM network between two attached devices

Digital believes the information in this publication is accurate as of its publication date; such information is subject to change without notice. Digital is not responsible for any inadvertent errors.

Digital conducts its business in a manner that conserves the environment and protects the safety and health of its employees, customers, and the community.