0.0 Preamble
--------
This tutorial is an attempt to qualitatively present the ATM concepts, and to introduce it gently to readers unfamiliar with the subject. The purpose of this writeup is to bring interested readers up to speed on what the heck is this "ATM". There are also some "opinions" presented in this writeup. Please feel to disagree. And if I may have mis-stated some fact or something is in error, I would be happy to learn about it.

If the reader is interested in knowing the exact bit locations in the ATM header, or how to design and implement an ATM interface card, or other facts and figures about interconnect speeds etc, he/she is directed towards the copious ATM standards committees documents where the latest and greatest information is available in the most excruciating detail.

1.0 What is this acronym ATM ?
-----------------------------
ATM stands for (no not automated teller machines) "Asynchronous Transfer Mode". It is primarily driven by telecommunications companies and is a proposed telecommunications standard for Broadband ISDN.

2.0 Motivation for ATM
-----------------------
In order to understand what ATM is all about, a brief introduction to STM is in order. ATM is the complement of STM which stands for "Synchronous Transfer Mode". STM is used by telecommunication backbone networks to transfer packetized voice and data across long distances. It is a circuit switched networking mechanism, where a connection is established between two end points before data transfer commences, and torn down when the two end points are done. Thus the end points allocate and reserve the connection bandwidth for the entire duration, even when they may not actually be transmitting the data. The way data is transported across an STM network is to divide the bandwidth of the STM links (familiar to most people as T1 and T3 links) into a fundamental unit of transmission called time-slots or buckets. These buckets are organized into a train containing a fixed number of buckets and are labeled from 1 to N. The train repeats periodically every T timeperiod, with the buckets in the train always in the same position with the same label. There can be up to M different trains labeled from 1 to M, all repeating with the time period T, and all arriving within the time period T. The parameters N, T, and M are determined by standards committees, and are different for Europe and America. For the trivia enthusiasts, the timeperiod T is a historic
legacy of the classic Nyquist sampling criteria for information recovery. It is derived from sampling the traditional 4Khz bandwidth of analog voice signals over phone lines at twice its frequency or 8Khz, which translates to a timeperiod of 125 usec. This is the most fundamental unit in almost all of telecommunications today, and is likely to remain with us for a long time.

On a given STM link, a connection between two end points is assigned a fixed bucket number between 1 and N, on a fixed train between 1 and M, and data from that connection is always carried in that bucket number on the assigned train. If there are intermediate nodes, it is possible that a different bucket number on a different train is assigned on each STM link in the route for that connection. However, there is always one known bucket reserved a priori on each link throughout the route. In other words, once a time-slot is assigned to a connection, it generally remains allocated for that connections sole use throughout the life time of that connection.

To better understand this, imagine the same train arriving at a station every T timeperiod. Then if a connection has any data to transmit, it drops its data into its assigned bucket (time-slot) and the train departs. And if the connection does not have any data to transmit, that bucket in that train goes empty. No passengers waiting in line can get on that empty bucket. If there are a large number of trains, and a large number of total buckets are going empty most of the time (although during rush hours the trains may get quite full), this is a significant wastage of bandwidth, and limits the number of connections that can be supported simultaneously. Furthermore, the number of connections can never exceed the total number of buckets on all the different trains (N*M). And this is the raison-d'etre for ATM.

3.0 Advent of ATM

-------------------

The telecommunications companies are investigating fiber optic cross country and cross oceanic links with Gigabit/sec speeds, and would like to carry in an integrated way, both real time traffic such as voice and hi-res video which can tolerate some loss but not delay, as well as non real time traffic such as computer data and file transfer which may tolerate some delay but not loss. The problem with carrying these different characteristics of traffic on the same medium in an integrated fashion is that the peak bandwidth requirement of these traffic sources may be quite high as in high-res full motion video, but the duration for which the data is actually transmitted may be quite small. In other words, the data comes in bursts and must be transmitted at the peak rate of the burst, but the average arrival time between bursts may be quite large and randomly distributed. For such bursty connections, it would be a considerable waste of bandwidth to reserve them a bucket at their peak bandwidth rate for all times, when on the average only 1 in 10 bucket may actually carry the data. It would be nice if that bucket could be reused for another pending connection. And thus using STM mode of transfer becomes inefficient as the peak bandwidth of the link, peak transfer rate of the traffic, and overall burstiness of the traffic expressed as a ratio of peak/average, all go up. In the judgement of the industry pundits, this is definitely the indicated trend for multimedia integrated telecommunications and data communications demands of global economies in the late 90's and early 21st century.

Hence ATM is conceived. It was independently proposed by Bellcore, the
research arm of AT&T in the US, and several giant telecommunications companies in Europe, which is why there may be two possible standards in the future. The main idea here was to say, instead of always identifying a connection by the bucket number, just carry the connection identifier along with the data in any bucket, and keep the size of the bucket small so that if any one bucket got dropped enroute due to congestion, not too much data would get lost, and in some cases could easily be recovered. And this sounded very much like packet switching, so they called it "Fast packet switching with short fixed length packets." And the fixed size of the packets arose out of hidden motivation from the telecommunications companies to sustain the same transmitted voice quality as in STM networks, but in the presence of some lost packets on ATM networks.

Thus two end points in an ATM network are associated with each other via an identifier called the "Virtual Circuit Identifier" (VCI label) instead of by a time-slot or bucket number as in a STM network. The VCI is carried in the header portion of the fast packet. The fast packet itself is carried in the same type of bucket as before, but there is no label or designation for the bucket anymore. The terms fast packet, cell, and bucket are used interchangeably in ATM literature and refer to the same thing.

4.0 Statistical Multiplexing
---------------------------

Fast packet switching is attempting to solve the unused bucket problem of STM by statistically multiplexing several connections on the same link based on their traffic characteristics. In other words, if a large number of connections are very bursty (i.e. their peak/average ratio is 10:1 or higher), then all of them may be assigned to the same link in the hope that statistically they will not all burst at the same time. And if some of them do burst simultaneously, that there is sufficient elasticity that the burst can be buffered up and put in subsequently available free buckets. This is called statistical multiplexing, and it allows the sum of the peak bandwidth requirement of all connections on a link to even exceed the aggregate available bandwidth of the link under certain conditions of discipline. This was impossible on an STM network, and it is the main distinction of an ATM network.

5.0 The ATM discipline and future challenges
--------------------------------------------

The discipline conditions under which statistical multiplexing can work efficiently in an ATM network are an active area of research and experimentation in both academia and industry. It has also been a prodigious source of technical publications and considerable speculations. Telecommunications companies in the US, Europe, and Japan as well as several research organizations and standards committees are actively investigating how BEST to do statistical multiplexing in such a way that the link bandwidth in an ATM network is utilized efficiently, and the quality of service requirements of delay and loss for different types of real time and non real time as well as bursty and continuous traffics are also satisfied during periods of congestion. The reason why this problem is so challenging is that if peak bandwidth requirement of every connection is allocated to it, then ATM just degenerates into STM and no statistical advantage is gained from the anticipated bursty nature of many of the future
broadband integrated traffic profiles.

Thus the past few years publications in "IEEE Journal of Selected Areas in Communications" and the "IEEE Network and Communications Magazines" are filled with topics of resource allocation in broadband networks, policing metering and shaping misbehaving traffic and congestion avoidance and control in ATM networks, and last but not least, multitudinous mathematical models and classifications speculating what the broadband integrated traffic of the future might actually look like, and how it might be managed effectively in a statistics based nondeterministic traffic transportation system such as an ATM network. The more adventurous readers desirous of learning more about ATM networks are encouraged to seek out these and the standards committees publications.

Fortunately however, these are problems that the service providers and ATM vendors like the telecommunications companies have to solve, and not the users. The users basically get access to the ATM network through well defined and well controlled interfaces called "User Network Interface" (UNI), and basically pump data into the network based on certain agreed upon requirements that they specify to the network at connection setup time. The network will then try to ensure that the connection stays within those requirements and that the quality of service parameters for that connection remain satisfied for the entire duration of the connection.

6.0 Who are the standards bodies investigating ATM ?
-----------------------------------------------

In the US, ATM is being supported and investigated by T1S1 subcommittee (ANSI sponsored). In Europe, it is being supported and investigated by ETSI. There are minor differences between the two proposed standards, but may converge into one common standard, unless telecommunications companies in Europe and America insist on having two standards so that they can have the pleasure of supporting both to inter-operate. The differences however are minor and do not impact the concepts discussed here. The international standards organization CCITT has also dedicated a study group XVIII to Broadband ISDN with the objective of merging differences and coming up with a single global worldwide standard for user interfaces to Broadband networks. No conclusions yet.

7.0 Types of User Network Interfaces (UNI) for ATM
-----------------------------------------------

It is envisioned that the ATM network service providers may offer several types of interfaces to their networks. One interface that is likely to be popular with companies that build routers and bridges for local area networks is a Frame based interface. One or more of the IEEE 802.X or FDDI frames may be supported at the UNI, with frame to ATM cell conversion and reassembly being done inside the UNI at the source and destination end points respectively. Thus a gateway host on a local area network might directly connect its ethernet, token ring, fddi, or other LAN/MAN interface to the UNI, and thus bridge two widely separated LANS with an ATM backbone network. This will preserve the existing investment in these standards and equipments, and enable a gradual transition of the ATM networks into the market place.
An alternate interface likely to be more popular in the longer run, and for which the concept of Broadband-ISDN really makes sense, is direct interface at the UNI with standard ATM cells. Such a streaming interface can hook subscriber telecom, datacom, and computer equipment directly to the network, and allow orders of magnitude greater performance and bandwidth utilization for integrated multimedia traffic of the future. Thus it is by no accident that the IEEE 802.6 packet for the MAC layer of the Metropolitan Area Network (MAN) DQDB protocol (Distributed Queue Dual Bus) looks very much like an ATM cell.

It is quite likely that companies may crop up (if they have not already done so) to design ATM multiplexers for interface to the UNI of a larger ATM backbone network. Especially if the CCITT succeeds in standardizing an interface definition for UNI, it will be an additional boon to this market. The multiplexers with multiple taps on the user side can connect to one fat ATM pipe at the network side. Such a multiplexer would hide the details of ATM network interface from the user, and provide simple, easy to use, low cost ATM cell taps to hook the user equipment into.

Companies with investment in existing STM networks such as T1 and T3 backbones, are likely to want a direct T3 interface to the UNI, thus allowing them to slowly integrate the newer ATM technology into their existing one. Thus it is possible to see a flurry of small startups in the future rushing to make large T3 multiplexers for connecting several T3 pipes into one large ATM pipe at the UNI.

Typically, an ATM network will require a network management agent or proxy to be running at every UNI which can communicate and exchange administrative messages with the user attachments at the UNI for connection setup, tear down, and flow control of the payload using some standard signalling protocol. A direct user attachment at the UNI is likely to cost more and be more complex, then a user attachment to something which in turns interfaces to the UNI.

8.0 What does an ATM packet look like
------------------------------------

An ATM cell or packet as specified by T1S1 sub-committee is 53 bytes. 5 bytes comprise the header, and 48 bytes are payload. The header and payload are specified as follows:

```
<--------------- 5 bytes --------------->|<-------------- 48 bytes --------->|
-------------------------------------------------------------
| VCI Label | control | header checksum | optional adaptation | payload |
| 24 bits   | 8 bits  |   8 bits        |   layer 4 bytes     |44 or 48 |
-------------------------------------------------------------
```

The 48 bytes of payload may optionally contain a 4 byte ATM adaptation layer and 44 bytes of actual data, or all 48 bytes may be data, based on a bit in the control field of the header. This enables fragmentation and reassembly of cells into larger packets at the source and destination respectively. (Since the header definition may still be in flux, it is possible that presence or absence of an adaptation layer information may not be explicitly indicated with a bit in the header, but rather implicitly derived from the VCI label). The control field may also contain a bit to specify whether this is a flow control cell or an ordinary cell, an advisory bit to indicate whether this cell is dropable in the face of congestion in the network.
or not, etc.

The ETSI definition of an ATM cell is similar, 53 bytes cell size, 5 byte header, 48 bytes data. However the difference is in number of bits for the VCI field, number of bits in the header checksum, and semantics and number of some of the control bits.

For a more detailed specification of the ATM header, see the appropriate standards committees documents.

9.0 Connections on an ATM network
---------------------------------

As in STM networks, where a datum may undergo a time-slot-interchange between two intermediate nodes in a route, the VCI label in an ATM cell may also undergo a VCI label interchange at intermediate nodes in the route. Otherwise, the connections in the ATM network look remarkably similar to STM networks.

An Example:

Assume an ATM network with nodes in NY, ATLANTA, DALLAS, and SF. Say that Chuck while vacationing in NY decides to play Aviator with his buddies in MtN view who are still grinding away on MPsniff. Also assume that we have ATM cell interfaces at UNI's in both NY and SF. This is what can happens: Chuck's portable $3K laptop makes a connection request to the UNI in NY. After an exchange of connection parameters between his laptop and the UNI (such as destination, traffic type, peak and average bandwidth requirement, delay and cell loss requirement, how much money he has got left to spend, etc), the UNI forwards the request to the network. The software running on the network computes a route based on the cost function specified by Chuck, and figures out which links on each leg of the route can best support the requested quality of service and bandwidth. Then it sends a connection setup request to all the nodes in the path enroute to the destination node in SF.

Lets say that the route selected was NY--AT--DA--SF. Each of the four nodes might pick an unused VCI label on their respective nodes and reserve it for the connection in the connection lookup tables inside their respective switches. Say, NY picks VC1. It will send it to AT. AT in turn picks VC2, associates it with VC1 in its connection table, and forwards VC2 to DA. DA picks VC3 and associates it with VC2 in its connection tables and forwards VC3 to SF. SF picks VC4 and associates it with VC3 in its connection tables, and pings the addressed UNI to see if it would accept this connection request. Fortunately, the UNI finds Chuck's buddies and returns affirmative. So SF hands the UNI and Chuck's friends VC4 as a connection identifier for this connection. SF then acks back to DA. DA acks back to AT and sends it VC3. AT puts VC3 in its connection tables to identify the path going in the reverse direction, and acks to NY sending it VC2. NY associates VC2 in its connection tables with VC1, and acks the originating UNI with VC1. The UNI hands chuck's laptop VC1 and connection is established.

Chuck identifies the connection with VCI label VC1, and his buddies identify the connection with VCI label VC4. The labels get translated at each node to the next outgoing label like so:

NY       AT       DA       SF
Chuck -> VC1 -> VC2 -> VC3 -> VC4 -> buddies
Chuck <- VC1 <- VC2 <- VC3 <- VC4 <- buddies

Other scenarios are also possible and would depend on a vendor's implementation of the ATM network.

When Chuck has had enough playing Aviator and wants to get back to some serious scuba diving off the Atlantic coast, the connection is torn down, and the VCI labels are reused for other connections.

10.0 What Assumptions can a user attachment make for a VCI label?
-----------------------------------------------

As is probably obvious from the above example, none. The VCI labels are owned by network nodes, and get randomized quite quickly as connections come and go. A VCI label is handed to a user attachment only as an opaque cookie, and not much can be assumed about its spatial distribution other than quite random.

It may be possible to have certain reserved VCI labels similar in concept to "well known port definitions of UDP and TCP", as identifiers for special well known services that may be provided by the network. However very little can be assumed about the dynamically assigned VCI labels for most user related connections.

A service provider is unlikely to accede to any special request by any one service requester to allocate it a chunk of VCI labels, unless the network itself is owned by the service requester. Furthermore, the address space of the VCI labels is limited to 24 bits and only designed to identify the connections between two points on a single link. The address space would disappear rather quickly if customers started to requisition portions of the VCI label for their own semantics.

If there is a specific need to assume semantics for the VCI label outside of the ATM network, i.e. require it to be within a certain range on the user attachments at the UNI, it is probably best to provide a lookup table in hardware inside the user attachments which can map the pretty much randomized VCI label assigned by the network to n bits of a new label to which the user attachment can assign its own semantics to its silicon's content.

11.0 What Protocol layer is ATM?
-------------------------------

As is probably evident by now, ATM is designed for switching short fixed length packets in hardware over Gigabit/sec links across very large distances. Thus its place in the protocol stack concept is somewhere around the data link layer. However it does not cleanly fit in to the abstract layered model, because within the ATM network itself, end-to-end connection, flow control, and routing are all done at the ATM cell level. So there are a few aspects of traditional higher layer functions present in it. In the OSI reference model, it would be considered layer 2 (where layer 1 is the physical layer and layer 2 is the datalink layer in the internet protocol stack). But it is not very important to assign a clean layer name to ATM, so long as it is recognized that it is a hardware implemented packet switched protocol using 53 byte fixed length packets.

What is perhaps more relevant is how will all this interact with current TCP/IP and IP networks in general, and with applications which
want to talk ATM directly in particular. A convenient model for an ATM interface is to consider it another communications port in the system. Thus from a system software point of view, it can be treated like any other data link layer port. Thus for instance, in IP networks connected via gateways to ATM backbones, the model would be no different then it presently is for a virtual circuit connection carried over an STM link except that an IP packet over an ATM network would get fragmented into cells at the transmitting UNI, and reassembled into the IP packet at the destination UNI. Thus a typical protocol stack might look like this:

```
-------------
Data
-------------
TCP
-------------
IP
-------------
ATM Adaptation Layer
-------------
ATM Datalink layer
-------------
Physical Layer (SONET STS-3c STS-12 STS-48)
-------------
```

Thus just like an ethernet port on a host is assigned an IP address, the ATM port may also be assigned an IP address. Thus the IP software in a router decides which port to send a packet to based on the IP address, and hands the packet to the port. The port then does the right thing with it. For an ethernet port, the ethernet header is tacked on and the frame transmitted in ethernet style. Similarly, for an ATM port, the IP datagram is fragmented into cells for which an ATM adaptation layer is specified in the standards. The fragmentation and reassembly is done in hardware on the sending and receiving sides. A VCI label acquired via an initial one time connection establishment phase, is placed in the header of each cell, and the cells are drained down the fat ATM datalink layer pipe. On the receiving side, the cells are reassembled in hardware using the ATM adaptation layer, and the original IP packet is reformulated and handed to the receiving host on the UNI. The adaptation layer is not a separate header, but is actually carried in the payload section of the ATM cell as discussed earlier.

For direct interface to an ATM cell stream from an application, new interfaces have to be designed in the software that can provide the application with nice and fast mechanisms for connection establishment, data transfer, keep alive, tear down, and even application level flow control. In this case the software processing steps may look like this:

```
---------------------------
Application Streaming Data
---------------------------
OS interface to application
---------------------------
ATM virtual circuit management/signalling
---------------------------
Driver interface to ATM
---------------------------
ATM
```
where the ATM virtual circuit management represents software which understands the ATM header specifics, sets up and tears down connections, does demultiplexing of the payload to appropriate connections, and responds to whatever standard signalling protocol is employed by the ATM interface at the UNI for connection management.

12.0 The Physical Layer
------------------------

The physical layer specification is not explicitly a part of the ATM definition, but is being considered by the same subcommittees. T1S1 has standardized on SONET as the preferred physical layer, and the STS classifications refer to the speeds of the SONET link. STS-3c is 155.5 Mbit/sec. STS-12 is 622 Mbit/sec, and STS-48 is 2.4 Gbit/sec. The SONET physical layer specifications chalk out a world wide digital telecommunications network hierarchy which is internationally known as the Synchronous Digital Hierarchy (SDH). It standardizes transmission around the bit rate of 51.84 Mbit/sec which is also called STS-1, and multiples of this bit rate comprise higher bit rate streams. Thus STS-3 is 3 times STS-1, STS-12 is 12 times STS-1, and so on. STS-3c is of particular interest as this is the lowest bit rate expected to carry the ATM traffic, and is also referred to as STM-1 (Synchronous Transport Module-Level 1). The term SONET stands for Synchronous Optical Network and is the US terminology for SDH (since they had to differ in something). So much for the acronym soup.

The SDH specifies how payload data is framed and transported synchronously across fiber optic transmission links without requiring all the links and nodes to have the same synchronized clock for data transmission and recovery (i.e. both the clock frequency and phase are allowed to have variations, or be plesiochronous). The intention being that products from multiple vendors across geographical and administrative boundaries should be able to plug and play in a standard way and the Broadband ISDN network be a true international network. And guess what the fundamental clock frequency is around which the SDH or SONET framing is done? You guessed it, 8Khz or 125 usec.

However all of this sits below the ATM layer and the ATM cells are transported across the physical layer as opaque payload, also called the SONET payload or the Synchronous Payload Envelope (SPE). The physical layer is independent of the payload type, and can just as easily carry STM cells as ATM cells. Refer to the standards documents for more details.

13.0 Flow control in ATM
------------------------

Unlike the reactive end to end flow control mechanisms of TCP in internetworking, the gigabits/sec capacity of ATM network generates a different set of requirements for flow control. If flow control was left on end to end feedback, then by the time the flow control message was received at the source, the source would have already transmitted over several Mbytes of data into the ATM pipe exacerbating the congestion. And by the time the source reacted to the flow control message, the congestion condition might have disappeared altogether unnecessarily quenching the source. The time constant of end to end
feedback in ATM networks (actually feedback\_delay * link\_bandwidth product) may be so large that solely relying on the user attachments to keep up with the dynamic network is impractical. The congestion conditions in ATM networks are expected to be extremely dynamic requiring fast hardware mechanisms for relaxing the network to steady state, and necessitating the network itself to be actively involved in quickly achieving this steady state. Thus a simplistic approach of end to end closed loop reactive control to congestion conditions is not considered sufficient for ATM networks.

The present consensus among the researchers in this field is to use a holistic approach to flow control. They recommend employing a collection of flow control schemes along with proper resource allocation and dimensioning of the networks to altogether try and avoid congestion, to try and detect congestion build up early by closely monitoring the internal queues inside the ATM switches and reacting gradually as the queues reach different high watermarks, and to try and control the injection of the connection data into the network at the UNI such that its rate of injection is modulated and metered there first before having to go to the user attachement for a more drastic source quenching. The concept is to exercise flow control in hardware very quickly, gradually, and in anticipation rather than in desperation. Rate based schemes which inject a controlled amount of data at a specified rate that is agreed upon at connection setup time, and automatically modulate the rate based on the past history of the connection as well as the present congestion state of the network have seen much press lately. The network state may be communicated to the UNI by the network very quickly by generating a flow control cell whenever a cell is to be dropped on some node due to congestion (i.e. the queues are getting full). The UNI may then police the connection by changing its injection rate, or notify the user attachment for source quenching depending on the severity level of the congestion condition.

The major challenge during flow control is to try and only affect those connection streams which are responsible for causing the congestion, and not affect other streams which are well behaved. And at the same time, allow a connection stream to utilize as much bandwidth as it needs if there is no congestion. This topic is still an area of active research, experimentation, and prolific publications including several PhD thesis.

14.0 Does an ATM network provide inorder delivery ?
--------------------------------------------------

Yes. An ATM cell may encounter congestion and suffer variable delay due to bufferring within the ATM switches, and may even be dropped either due to congestion control or due to header checksum error. However an ATM connection always obeys causality, the cells in a connection (i.e. cells with the same VCI label) arrive inorder at the destination. This is so because there is no store and forwarding in the network, cells travel over a single virtual circuit path, the ATM switches do not switch the cells in the same VCI out of order, and no retransmissions is done at any point in the ATM network.

Connectionless services are also supported on ATM networks, but these are implemented as a higher layer service layered over the ATM datalink layer. Thus cells in a connectionless service may arrive out-of-order because there might be multiple VCIs over multiple paths setup to deliver the connectionless datagrams and cells may arrive
over different paths in different order. Thus the fragmentation reassembly engine which implements the connectionless datagrams, and which is layered on top of the basic connection oriented service of the ATM layer, must carry sequence numbers in the adaptation layer in each cell and correct any reordering of the cells at reassembly time. This is what the IEEE 802.6 protocol for MAN does to support its connectionless service class.

15.0 Does an ATM network provide reliable delivery?
-----------------------------------------------

No. There is no end-to-end reliable delivery service at the ATM layer. The ATM layer does not do any retransmissions and there are no end-to-end acknowledgements for what has been received. Reliable delivery service can be implemented as a layer on top of the basic connection oriented ATM layer, where acknowledgement of received data and retransmission of missing data can be done for connections requiring reliable delivery. Thus a TCP type transport layer protocol (layer 4 in the OSI model) layered on top of the ATM layer is required for guaranteed delivery.

16.0 Performance of an ATM interface
------------------------------------

Unlike STM networks, ATM networks must rely on considerable user supplied information for the traffic profile in order to provide the connection with the desired service quality. There are some sources of traffic which are easier to describe than others, and herein lies the cost/performance challenge for best bandwidth utilization in an ATM interface.

An ATM network can support many types of services. Connection oriented as well as connection less. It can support services which may fall in any of the four categories (loss sensitive, delay sensitive), (loss insensitive, delay sensitive), (loss sensitive, delay insensitive), and (loss insensitive, delay insensitive). It can further reserve and allocate a fixed bandwidth for a connection carrying a continuous bit stream for isochronous traffic (repeating in time such as 8khz voice samples), allocate a bandwidth range for a variable bit stream for plesiachronous traffic (variable frequency such as interactive compressed video), as well as allocate no specific amount of bandwidth and rely on statistical sharing among bursty sources. It may also provide multiple priorities in any of the above categories. The services can span the entire gamut from interactive such as telephony and on-line data retrieval, to distributed such as video and stereo Hi-Fi broadcasts and multicasts for conferencing and database updates.

Thus the performance that one might get from ones ATM connection is very much dependent on the parameters that are specified at connection setup time. Just because the link bandwidth may be an STS-12, does not necessarily imply that the end to end payload bandwidth that the ATM interface can sustain will also be STS-12. It will in fact be considerably lower based on connection setup parameters and the quality of service request, and whether bandwidth was reserved or statistically multiplexed, and the load on the ATM network.

Typically, the ATM network may not permit 100% loading of any link bandwidth, and in fact user available bandwidth may not be allowed to exceed more than 80% of the peak bandwidth of the link. The UNI may
start policing and/or denying new connection requests on the link if utilization exceeds this amount. Add the approx 10% overhead of the 5 byte header in the 53 byte cell, and the max sustainable payload throughput on an ATM cell stream interface may peak at 72% of the peak link bandwidth. And this does not include any adaptation layer overhead if present, signalling overhead, or physical layer overheads of SDH SONET framing and inter-cell spacing gaps.

And of course, application to application bandwidth may be even less, unless the software datapath from the interface driver through the OS to the application (and vice versa) is very carefully optimized. It would hardly be received very well if the end-to-end throughput from application to application would turn out to be no better for an ATM port than for an ethernet or fddi port due to software overheads.

How many cells might be realistically received or transmitted at a sustained rate on an ATM cell interface in a processor? Hard to say for sure as there is no existence proof as yet.

However, what can be stated is that the transmitter and receiver performance is independent of each other. The transmitter side is constrained by the flow control of the simultaneous connection streams by pacing the injection rate according to the respective negotiated class of service and bandwidth requirements. The receiver side is constrained by asynchronous reception of cells at a variable rate, and with buffering capacity for a large number of simultaneous connections each of which can be receiving data simultaneously. And if an adaptation layer is used, then the reassembly of these cells into a higher layer protocol data unit (PDU) must also be done in hardware by the receiver side. Thus a lot of thought is required in designing an ATM interface to a host system, poor design of which can cripple the system performance.

17.0 When can I have my own connection to an ATM network?

The Broadband ISDN with ATM is an enabling technology. It will enable new kinds of applications and new types of usage which are only in peoples imagination today. It is a complete overhaul of the copper based low bandwidth telecommunications technology that has existed until now, and represents a massive investment both in research and development, as well as deployment and integration. The software investment required to make the ATM network work is tremendous, and many of the algorithms and theories about how to manage the ATM network are still in their infancy and mostly on paper. Considerable work is also required in developing new network management paradigms and protocols to effectively control and manage the vasts quantities of bandwidths and services that the revolution in communication technology is promising to offer.

At the present time, there are no commercially available ATM networks in the US (to my knowledge), though there are several ATM prototype switches and experiments in existence. The earliest anticipated roll out of commercial ATM switches is expected no sooner than 1995 time frame. And full fledged deployment of ATM networks with "COST EFFECTIVE" multi-media integrated services to end-users is still a lot farther away, probably closer to the end of this decade. But its coming...hang on.