SOME OBSERVATIONS ON THE ATM ADAPTATION LAYER SERVICES NEEDED TO SUPPORT TCP/IP.

Grenville Armitage (gja@mullian.ee.mu.OZ.AU)
Keith Adams (kma@mullian.ee.mu.OZ.AU)
Ph. (03) 344-7976
Fax. (03) 344-6678
Department of Electrical and Electronic Engineering
University of Melbourne
Parkville, 3052
AUSTRALIA

ABSTRACT

This paper examines four issues:
1. The datagram transport needs of the DARPA Internetworking Protocol,
2. The basic data transport facilities provided by CCITT's evolving ATM Adaptation Layer,
3. The relationship between these facilities and the needs of IP,
4. Encapsulation of IP datagrams, and the provision of dynamic Address Resolution.

Comparisons are also made with LAN support of IP services.

1. INTRODUCTION.

The DARPA Internetworking Protocol (IP), and the overlying Transport Control Protocol (TCP) and User Datagram Protocol (UDP), precedes many of the workstation environments it is currently used in. Within the academic and research communities TCP/IP has become almost synonymous with internetworking between UNIX platforms. It also provides connectivity for a vast network of different machines - from PC's running packages such as KA9Q NOS to VAXen running VMS. Its importance to the ATM community now is due to the size of the installed IP user base.

It is arguable whether telecommunication service providers can simply ignore the IP community - there are certainly pushes to have more OSI compliant systems predominate by the late 1990's. However, it is highly likely that the IP community will wish to make use of evolving BISDN interfaces to provide at least backbone services regardless of our attitude towards them. At best, however, we might expect that a happy merger of IP and ATM technology will provide an increased incentive for ATM products to be developed and placed on the market - benefiting all people involved in the BISDN area. To this end this paper provides a brief overview of the issues associated with this merger, looking at the services being offered by CCITT's ATM Adaptation Layer [1,2,3] and the services needed by IP [4].

2. IP DATAGRAM TRANSPORT

The fundamental unit of data transport in the Internetworking Protocol is the IP datagram. It is an independent entity unrelated to any other IP datagram by logical connection or circuit (virtual or otherwise) [5]. The header of each IP datagram contains source and destination IP addresses. These addresses identify the IP entity which originated the datagram, and the IP entity to which the contents of the datagram are to be delivered. The function of the layer beneath the IP entity is to provide a means of physically transporting an IP datagram from one IP entity to another. Generally IP entities are associated with particular physical layer interfaces, however, each interface will have its own unique addressing scheme specific to the technology

being used. Irrespective of the presence or lack of physical layer address checking, IP entities always check the addresses in the IP datagram before accepting them.

The Internetworking Protocol transports its datagrams over multiple physical networks to reach arbitrary destinations. IP gateways copy datagrams from one physical interface to another, based on the destination address in the IP header, thereby overcoming the localised nature of many physical layer technologies. When a gateway knows it doesn't have a direct physical connection to the destination IP entity it sends the IP datagram to another gateway which it believes is "closer" to the desired destination. Additionally IP entities make no assumptions about whether the underlying physical network is capable of guaranteeing the transport of any given packet.

IP also supports the concept of multicast or broadcast addresses. A broadcast address implies that the datagram should be sent to 'all directly connected' IP entities. Multicast addresses require a datagram to be sent to a specific group of 1 or more IP entities. The ability to provide broadcast IP transmissions usually depends on the physical layer having its own broadcast capabilities.

2.1 LANS

One of the most popular small scale physical technologies is Ethernet [6] (and its variant, IEEE 802.3). It provides moderately high speed datagram transport, with best-effort non-guaranteed delivery, almost identical to what the IP layer expects. Each Ethernet packet contains an Ethernet address specifying the immediate destination of the packet, and another address specifying the immediate source of the packet. An Ethernet 'Type' field in each packet identifies the protocol entity to which the packet contents ought to be delivered. An Ethernet type code of 0x800 means the contents are delivered to the IP entity at the destination, this type code is standardised world-wide [7]. In an IP environment the immediate Ethernet destination is either the destination IP entity or an IP gateway which is believed to be "closer" to the destination IP entity.

Most LANs pre-filter received packets to ensure only those specifically directed to the local physical interface address are passed up to the local IP entity. The exceptions to this are broadcast and multicast packets. Broadcast facilities in LANs enable IP entities to blindly 'locate' the physical layer addresses of IP entities on the local LAN segment (e.g. Address Resolution Protocol, RARP, and Proxy ARP [9,10,11]).

Different LAN technologies impose limits on the size of individual IP datagrams. For example Ethernet imposes a length limit of 1500 bytes, IEEE802.3 variants limit the datagram to 1492 bytes [8]. Other LAN technologies, for example IEEE802.5 [8] and FDDI [23], have also found their places as IP datagram transport systems. In common with Ethernet they also provide directed packet delivery and broadcast/multicast facilities for dynamic IP address resolution schemes.

2.2 POINT TO POINT LINKS

This phrase can be used to cover a range of techniques. The main characteristic is that these links have only two 'ends' exchanging data. This ranges from 1200baud modems using public telephone lines to high speed (tens to thousands of Kbytes/second) dedicated lines supplied by telecommunication companies.

Many low-performance systems utilise non-standard encapsulation schemes like SLIP [12] to carry IP datagrams across byte orientated modem connections. SLIP is a recognition of the minimum service required to connect two IP entities - convey the IP datagram with its boundaries intact. Modem interfaces generally only delineate the bit-streams into bytes, a SLIP

layer restores datagram boundaries to the byte stream. Environments using SLIP generally have no physical layer addresses to worry about, and the IP entities at either end are pre-configured with each other's IP address(es).

More advanced interface protocols have been developed to better use higher speed modem connections, and even dedicated serial lines. PPP is one of these [13]. They tend to address broader issues like allowing more than one higher layer protocol to share the link, and automated link establishment and tear-down. However the bottom line is that they restore datagram boundaries to information transmitted across byte orientated lines.

Circuit switched ISDN connections fall under the category of point-to-point links too, whether or not the interface provides a bitstream or full datagram support. With ISDN connections, a new range of issues arrives for the layer underneath the IP entity - a significant one being the need to establish virtual connections "on demand" to reduce connection time and costs.

A major problem with point-to-point networks is that they have no support for broadcast or multicast transmissions. This makes dynamic address resolution protocols rather more difficult to establish than in LAN environments.

3. IP GATEWAYS, ADDRESS RESOLUTION, AND ROUTING ISSUES.

IP addresses inherently provide limited routing information. IP address structures assume that the underlying physical networks and gateways can be characterised in a broadly hierarchical system of networks, sub-networks, and hosts. Gateways rarely store specific host location information, relying on the network and sub-network fields of IP addresses to identify which physical network a datagram should be forwarded on [20].

A transmitting IP entity differentiates between destination IP entities directly accessible through the local physical layer, and ones accessible through a gateway on the local network. In either case a mapping needs to be created between the IP address of the destination or gateway and their actual physical layer address. For example Ethernet requires 48 bit destination addresses, for ISDN it would be some form of E.164 number.

LANs may use a broadcast ARP technique to locate IP entities on the same LAN segment [9]. A broadcast ARP request goes to all nodes on the physical network, and elicits a response from the node whose IP address is being queried. IP entities using point to point links are still developing techniques for dynamic address resolution; most implementations currently use static mapping tables instead.

The original IP entity needs only to locate its nearest, or most useful, IP gateway before it can send a datagram outside its immediate network. An often used solution is to have a default-gateway defined in a local look-up table, assuming that the gateway's address on the physical network is reasonably stable.

4. AN OVERVIEW OF ATM ADAPTATION LAYER (AAL) SERVICES

The following discussion of ATM and AAL layer services is potentially contentious as the recommendations covering these areas are constantly evolving and being re-clarified.

The ATM Adaptation Layer (AAL) [14] provides an interface between the ATM cell network and higher layer datagram or data stream services [15,16]. AALs provide the ultimate sources and destinations of ATM cells within a network. They exchange data in variable sized AAL_SDUs with higher layers, and fixed sized cells with the ATM layer [17]. Each cell header

contains a Virtual Channel Identifier (VCI) and a Virtual Path Identifier (VPI) which enables AALs to associate a given cell with a given virtual connection.

Of the four basic SAR services identified by CCITT, Type 4 is the one most oriented towards provision of datagram transport between endpoints. This mode is expected to function over point-to-point and multicast virtual connections [17, section 5.1.1].

4.1 POINT TO POINT VIRTUAL CONNECTIONS

The most basic connection provided is point to point. The link management plane is called to create a virtual connection between one AAL and another. Each AAL in the network will have a unique E.164 identifier to locate its ATM layer interface. The ultimate endpoints of a virtual connection is the interface between an AAL and higher protocol layers using its services [18]. For this paper we will call the E.164 address an ATM Terminal Interface Address, ATIA. The higher layer protocol will be specified by an AAL Endpoint Protocol Identifier, AEPI. An AAL_SDU can be seen as being transferred, with boundaries preserved, between two (ATIA, AEPI) pairs. Any given AAL may support multiple point to point virtual connections originating from, or terminating on, themselves. An AAL may provide each virtual connection with different service types, agreed on during the connection establishment phase. An alternative conceptualisation is that each virtual connection has an 'instance' of an AAL, providing only one service type per instance.

Figure 4.1

4.2 MULTICAST CONNECTIONS

A more complex situation is also envisaged where AAL_SDUs are transmitted by one AAL and reassembled independently at multiple remote AALs. Again the expectation is that the AAL_SDU is transferred with boundaries intact to all destinations. How the underlying network establishes these multicast virtual connections is not yet clear.

4.3 CONNECTIONLESS EXTENSIONS

Protocols such as Ethernet provide connectionless transfer of packets, whereas the underlying nature of ATM is virtual connection orientated. CCITT have moved to define a layer above the Type 4 AAL services to provide connectionless datagram transport services between any two AALs in a B-ISDN network [19]. The similarities between this and LAN transports like Ethernet is best explained by a quote from [19] : "*The connectionless service layer provides for the transparent transfer of variable size data units from a source to one or more destinations in a 'best effort' manner, i.e. lost or corrupted data units are not retransmitted. This transfer is performed using a connectionless technique, including embedding destination and source addresses into each data unit*".

The connectionless mode data unit has a format reminiscent of Ethernet packets, containing destination and source addresses, CRC check fields, and Higher-Layer-Protocol-Identifier fields (amongst others) [19, section 3.5]. The source and destination addresses have E.164 format.

5. WHAT MODES OF SUPPORT ARE AVAILABLE TO THE IP LAYER ?

The risk with standards is that we may misinterpret the intent of draft recommendations and working party pronouncements. Be that as it may, the following ideas for IP make use of our simple model of ATIA and AEPI identifiers.

We can see the entire B-ISDN network as a cloud of AALs and ATM interfaces, all uniquely identifiable by ATIAs. Every AAL is the support base for one or more higher layer protocols (e.g. Video, telephony, ATM Connectionless mode, IP, etc). Each service which lies immediately on top of an AAL can be located by an AEPI value. So the B-ISDN network can be seen as a cloud of protocol entities identifiable by unique (ATIA,AEPI) pairs, and the network is capable of creating virtual connections between any (ATIA,AEPI) pair. What traffic is carried by each connection is up to the protocol entities at each end of the link.

5.1 POINT TO POINT CONNECTIONS

The most obvious implementation is to use B-ISDN in a similar way to present work in the ISDN and Serial Line world - a high speed point to point link. This could serve a host wishing to connect to another host, or a host wishing to connect to a gateway. Two issues need solutions here - resolving IP addresses to ATM interface addresses, and encapsulation of IP datagrams for ATM transmission.

Taking a 'small is beautiful' line to encapsulation will lead us to realise that no more than basic Type 4 SAR/CS service is *required* between any two AALs. This service provides non-assured datagram transport with boundaries kept intact. If an AEPI value is reserved for IP entities then protocol multiplexing and demultiplexing will essentially be provided at call set-up time. So an (ATIA, AEPI) pair is the connection oriented analog of an (Ethernet address, Ethernet type) pair on an Ethernet LAN. However 'small' does not always interwork with all higher layer software in today's so-called 'open systems' environment, no matter how beautiful. Some form of encapsulation will probably be used, although it is likely to come from the IP camp rather than the B-ISDN camp in the form of modified PPP or IEEE802.2 conformant interface layers. (As an example both the SMDS and FDDI RFC's note that the encapsulation provided for 802.2 is not strictly needed for either service to support IP and ARP).

In B-ISDN, as with ISDN and dial-up lines, there will probably be cost savings if the virtual connection is not always present. By analogy to ISDN it will be feasible to have virtual connections established when an IP datagram is ready for transmission, and torn-down when the connection has been idle for some period of time. To this extent our 'encapsulation' layer, whatever other services it provides, needs to hide the existence of the virtual connection from the IP entities.

5.2 MULTICAST GROUPS

A multicast ATM connection offers us the ability to group AALs into pseudo-broadcast networks. In this case each AAL_SDU (IP datagram) sent by one AAL will be received by every other AAL in the multicast group. The IP entities on each AAL must then accept or drop datagrams based on their destination IP address fields. As with a unicast virtual connection the smallest encapsulation scheme could use a Type 4 SAR/CS service directly. However, the dictates of 'open systems' will probably suggest that IEEE802.2 style services will be layered over the raw Type 4 connection, and IP layered over this.

This scheme is far more processor intensive at the IP layer than most broadcast LANs are, it would be analogous to sending directed IP datagrams in physical layer broadcast packets. However it should provide good support for applications requiring IP level multicasting - the multicast ATM connection could be set up on-demand in response to an IP multicast datagram.

5.3 CONNECTIONLESS MODE TRANSPORT

The encapsulation issue again sees a conflict between 'simple' and 'open', with connectionless data units providing the datagram transport the IP needs. However, our encapsulation/interface layer between IP and Connectionless Transport is now freed from the dynamic management of virtual connections. This may be considered a major bonus of connectionless transport. It would not be wise, though, to assume connectionless mode will be always better. By its nature it internalises the routing and connection establishment phase within the B-ISDN network interface. This creates a hidden source of transport delay which might be avoided on specific unicast virtual connections.

5.4 RESOLVING IP TO E.164 ADDRESSES

Address resolution is difficult when broadcast facilities are unavailable or unwieldy. In small systems of interconnected gateways using B-ISDN as a high speed backbone, it may be sufficient to have semi-static look-up tables. Look-up tables might also be appropriate for a single IP host using B-ISDN to connect itself to another IP network, where it only needs to know the E.164 address of its default gateway. However if we wish to interconnect multiple IP hosts using B-ISDN directly then an address resolution technique needs to be established.

Once the E.164 address is known a direct virtual connection can be established. This raises questions as to how we map the IP concepts of networks and sub-networks onto the B-ISDN address space. One solution is to create closed groups of E.164 addresses, each group representing a logical IP subnet. In a sense we create an association of physical interfaces not unlike that forced on Ethernet interfaces which connect to the same cable, except we are freed from the geographically localising effects of cable. Within such a group a multicast transmission could carry ARP requests for that subnet. Establishing the E.164 address of IP entities outside the group is a little more complicated. In a 'traditional' IP subnet at least one node is a gateway or router which not only knows how to get to remote IP entities, but transfers the datagrams as well. We might provide nodes which belong to multiple B-ISDN groups, each acting as a gateway/router between logical IP subnets. This technique uses the ATM network inefficiently, however, since IP datagrams are unnecessarily shuttled between these 'logical routers' even though the underlying network is capable of creating a 'direct' link.

A different scheme would be to introduce a hierarchical ARPing scheme. An IP entity could ARP request its logical IP subnet members for an E.164 address. The logical router on the subnet may recognise the destination IP address and know of another router 'closer' to the desired destination (based on the network/subnetwork fields of the IP address). It passes back the E.164 address of this new router to the original IP entity, which is then able to query the new router with a direct request. This procedure of direct ARP requests could repeat until the originating IP entity locates a router which claims exact knowledge of the destinations E.164 address. At this point a new mapping can be inserted into the local ARP mapping table, and the IP datagram transferred (either using connection or connectionless mode).

6. A DYNAMICALLY CREATED IP-OVER-ATM LINK

As part of an on going research project we developed a simple point-to-point IP link running over an emulated ATM network between two SUN workstations [21]. We implemented a simple signalling protocol (described in [22]) and 3 node ATM network. The ATM network contains two AALs on separate 'fibres' connected through an emulated Cell Switch. A virtual connection between participating IP layers is only requested when packets need to be transferred. When the IP link is idle for a sufficient period of time the virtual connection is removed, freeing up resources in the underlying ATM network.

The IP layer in SunOS was provided with an interface module which looked after the dynamic virtual connection creation and deletion. The IP interface module was allocated a fixed AEPI value of 0x800. Each end of the link was provided with a table mapping IP addresses to ATIAs.

The IP interface is in two parts - a STREAMS module which links the top side of the AAL with the 'sockets' based underside of the SunOS IP layer, and a user level daemon which executes signalling function calls to create and remove virtual connections when requested by the STREAMS module. Figure 7.1 shows how it all fits together.
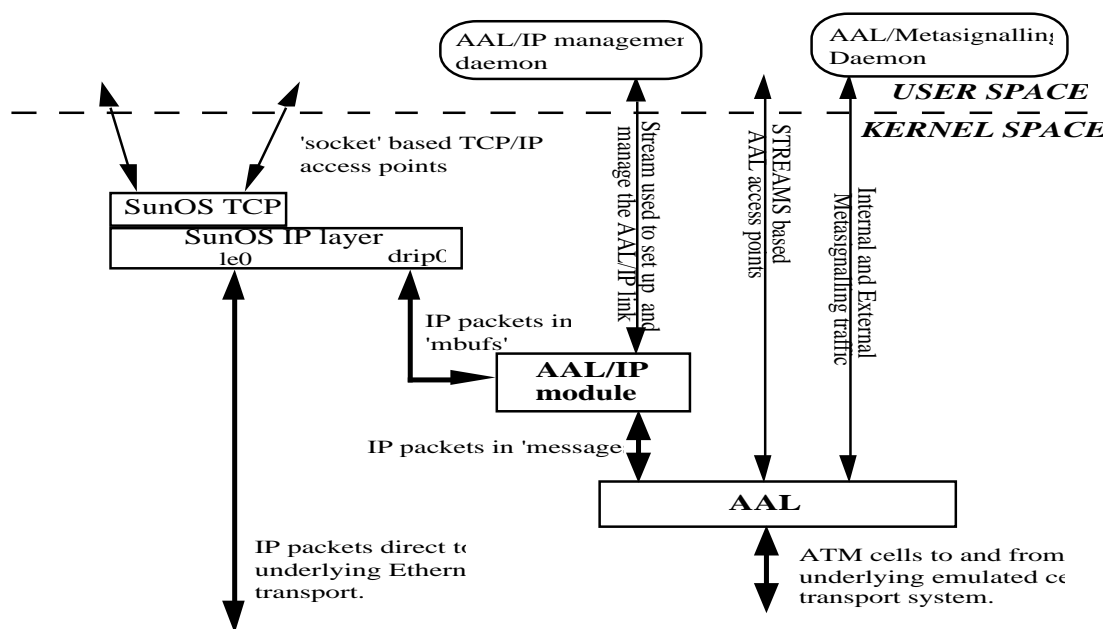

Figure 7.1

The interface provides no further encapsulation of the IP packets. The IP interface module transparently calls the daemon to re-create the virtual connection each time a packet needs to be transmitted after the connection has previously been torn down. Using the STREAMS modules' ability to defer processing of messages, the first packet is not lost but is transmitted when the daemon informs the interface module that the virtual connection has been re-established.

7. CONCLUSION

The ATM Adaptation Layer being defined by CCITT provides us with basic point to point services, and additional layers are aiming to provide connectionless services as well. Both of these are capable of exchanging IP datagrams in ways similar to current physical networks which IP has been run over. The biggest problem facing utilisers of the B-ISDN is not so much one of encapsulation of IP datagrams but the provision of Address Resolution services. Early use of B-ISDN will probably be as a backbone between gateways, as this has less need for dynamic address resolution service. Evolution to IP hosts actually existing on the B-ISDN with their own E.164 addresses will probably use logical IP subnets and some modified dynamic ARP procedure.

REFERENCES

[1]   CCITT SGXVIII Draft Recommendation I.362, "B-ISDN ATM Adaptation Layer (AAL) Functional Description", June 1990.

[2]   CCITT SGXVIII Draft Recommendation I.150, "B-ISDN ATM Functional Characteristics", June 1990.

[3]   Martin de Prycker, "Asynchronous Transfer Mode - Solution for Broadband ISDN", Chapter 3, Ellis Horwood, 1991.

[4]   D. Comer, "Internetworking with TCP/IP - Principals, Protocols, and Architecture", Prentice Hall, 1988.

[5]   Jon Postel, "Internet Protocol - DARPA Internet Program Protocol Specification", RFC-791, September 1981, Section 1.4.

[6]   "The Ethernet - A Local Area Network", Version 1.0, Digital Equipment Corporation, Intel Corporation, Xerox Corporation, September 1980.

[7]   C. Hornig, "A Standard for the Transmission of IP Datagrams over Ethernet Networks", RFC-894, April 1984.

[8]   J. Postel, J. Reynolds, "A Standard for the Transmission of IP Datagrams over IEEE 802 Networks", RFC-1042, February 1988.

[9]   D.C. Plummer, "An Ethernet Address Resolution Protocol", RFC-826, November 1982.

[10]   R. Finlayson, T. Mann, J. Mogul, M. Theimer, "A Reverse Address Resolution Protocol", RFC-903, June 1984.

[11]   D. Comer, "Internetworking with TCP/IP - Principals, Protocols, and Architecture", p.196, Prentice Hall, 1988.

[12]   J. Romkey, "A Non-standard for transmission of IP datagrams over serial lines: SLIP", RFC-1055, June 1988.

[13]   Drew D. Perkins, "The Point-to-Point Protocol for the Transmission of Multi-Protocol Datagrams over Point-to-Point Links", RFC-1171, July 1990.

[14]   CCITT SGXVIII Draft Recommendation I.362, "B-ISDN ATM Adaptation Layer (AAL) Functional Description", June 1990.

[15]   CCITT SGXVIII Draft Recommendation I.150, "B-ISDN ATM Functional Characteristics", June 1990.

[16]   Martin de Prycker, "Asynchronous Transfer Mode - Solution for Broadband ISDN", Chapter 3, Ellis Horwood, 1991.

[17]   CCITT SGXVIII Draft Recommendation I.363, "B-ISDN ATM Adaptation Layer (AAL) Specification", June 1990.

[18]   CCITT WPXVIII/8 Meeting Report - Annex 3 (Part B), "Text of WP XVIII/8 B-ISDN Recommendations", Appendix 9 - I.363, Sub-appendix 3, "Functional Model for the AAL Type 3/4", Melbourne, December 1991.

[19]   CCITT WPXVIII/8 Meeting report - Annex 3 (Part B), "Text of WP XVIII/8 B-ISDN Recommendations", Appendix 11 - I.cls, December 1991.

[20]   Reference [4], Chapter 4.

[21]   G. J. Armitage, K. M. Adams, "Implementing an ATM Adaptation Layer in a Multi-Media Terminal", Proc. ABBSS'91, Vol.1, July 1991.

[22]   G. J. Armitage, K. M. Adams, "Towards Shared ATM Fibres, A Pseudo Metasignalling Protocol, and ATM LANS", Proc. ABBSS'92, Melbourne, July 1992.

[23]   D. Katz, "A Proposed Standard for the Transmission of IP Datagrams over FDDI Networks", RFC-1103, June 1989.