

ATM Adaptation Layer Packet Reassembly during Cell Loss

G. J. Armitage

and

K.M. Adams

POSTAL ADDRESS for both Authors:

Dept. of Electrical & Electronic Engineering

University of Melbourne

Parkville 3052,

Victoria,

AUSTRALIA

FAX

+61 3 344 6678

Electronic Mail addresses for the Authors:

(INTERNET)

gja@mullian.ee.mu.OZ.AU

kma@mullian.ee.mu.OZ.AU

Telephone Numbers for the Authors:

G. J. Armitage +61 3 344 7976

K.M. Adams +61 3 344 7974

SUMMARY

Transportation of datagrams over Asynchronous Transfer Mode (ATM) networks by the ATM Adaptation Layer (AAL) involves their Segmentation into fixed sized cells, transmission across a virtual connection, and subsequent Reassembly. Cells may be lost through corruption, misrouting, or congestion control, resulting in incorrect or aborted datagram reassembly. This paper looks at the various cell loss modes, and their affect on Type 3/4 and Type 5 AAL services. AAL3/4 is shown to be capable of partial AAL Service Data Unit (AAL_SDU) reassembly, while AAL5 is not. AAL5 is seen to lose 2 AAL_SDUs if the end of one is lost, suggesting that selective discarding mechanisms should preserve End-Of-Message cells. AAL3/4 is shown to be completely unsuited to discarding AAL_SDUs at switch nodes as a method of congestion control. The value of passing up fragments of reassembled datagrams to AAL service users such as TCP/IP and LAN gateways is discussed.

1. INTRODUCTION

Broadband integrated services digital network (B-ISDN) technology has been evolving rapidly over the last 4 years, primarily under the guidance of CCITT's Study Group 18 (as of the beginning of 1993, CCITT has become the ITU Telecommunication Standardisation Sector, ITU-TSS). The promise of sophisticated customer services, such as integrated high speed data, voice, and video communication, has generated much interest in bringing B-ISDN to the marketplace. The decision to adopt the Asynchronous Transfer Mode (ATM) as support for B-ISDN has been made by the ITU-TSS [1, 2]. As a result of this decision the communication link between the customer and the network is limited to a single physical medium such as an optical fibre which will carry all types of signals in the form of ATM cells.

ATM is a packet switched data transport system based on short, fixed length cells. Each cell consists of 48 bytes of user data and a 5 byte header. Each cell carries a Virtual Channel Indicator (VCI) and Virtual Path Indicator (VPI) in its header. Cells are routed through switching nodes, using the combination of VPI and VCI as a label to associate each cell with established virtual connections.

Essential to the services offered by the new ATM networks is the ATM Adaptation Layer (AAL). The AAL is an ITU-TSS defined layer which "adapts" the cell based ATM physical layer to

packet, datagram, or bitstream oriented higher layers [3,4]. It exists at the endpoints of virtual connections, where higher layer peers wish to establish communication. The AAL has been evolving rapidly as the ITU-TSS attempts to bring B-ISDN to a standardised reality [5], since it is the AAL which is fundamental to the integration of different service types across ATM links.

Although very low error rates are being aimed for, the ATM layer makes no guarantee about the transport of individual cells. A variety of mechanisms exist which result in loss of cells across a virtual connection. This is an obvious problem for user services, as cell loss results in loss of user data. It is also an issue for the AAL, as mechanisms need to be provided for passing error indications to the higher layers.

This paper looks at the failure modes along a virtual connection that can cause cell loss, and analyses ways in which AALs cope. In section 2 the relation between VPIs, VCIs, and virtual connections is discussed. Section 3 looks at the AAL itself, with specific emphasis on the AAL3/4 and AAL5 service types. The sources of cell loss are described in Section 4. Section 5 discusses the effect that lost cells have on AAL3/4 or AAL5 type of service. The usefulness of the ability of AAL3/4 to pass up fragments of corrupted data to higher layer protocols is discussed in section 6. In section 7 consideration is given to implementing selective cell discarding within switching nodes, and the limitations imposed by each AAL. The paper is concluded in section 8.

2. VPIs, VCIs, AND ATM CONNECTIONS.

In section 2 of ITU-TSS Recommendation I.150 [1] ATM is described as “*a connection-oriented technique. Connection identifiers are assigned to each link of a connection when required, and released when no longer needed.*”. The VPI and VCI fields within each cell header (shown in Figure 1) are the connection identifiers. In section 3.1.1 of [1] an ATM connection is described as consisting “*..of the concatenation of ATM layer links in order to provide an end-to-end transfer capability to endpoints.*”. An ATM link consists of any cell path between points where the VPI or VCI are switched or terminated.

The VCI field is 16 bits wide. The VPI field has two ITU-TSS defined sizes. At the Network Node Interface (NNI) it is 12 bits, and at the User Network Interface (UNI) it is 8 (the remaining 4 bits carrying Generic Flow Control (GFC) information) [6]. ITU-TSS have established an hierarchical relationship between VPIs, VCIs, and the physical media [7]. Virtual channels are considered to exist within virtual paths, which are themselves unique only within a given physical path. Figure 2 illustrates the basic relationship.

Virtual Channel Connections (VCCs) provide end-to-end cell paths between users of the ATM layer (typically the AALs at either end). Both the VPI and VCI values may change along a VCC on a per-link basis. Virtual Path Connections (VPCs) carry bundles of VCCs across sequences of links where only the VPI is changed. Within a VPC, and hence a VCC, cell sequence is assured but cell integrity is not. For the rest of this paper the term 'virtual connection' should be taken to mean a VCC.

3. THE ATM ADAPTATION LAYER (AAL).

A virtual connection end point consists of a physical medium interface, an ATM layer, and an AAL (Figure 3). The ATM layer demultiplexes received cells based on their VPI and VCI before passing them upwards to the AAL. Cells which do not belong to any virtual connection currently open to the AAL are filtered out and dropped by the ATM layer. An overview of the AAL's functions will be given here. For a more detailed description ITU-TSS recommendations I.362 and I.363 [3,4] should be referred to.

Figure 3 shows multiple user services being simultaneously supported by the AAL. Each service may require different data transfer mechanisms - ITU-TSS is defining (at this stage) four different types of AAL to provide a range of basic functions. An example is having simultaneous video and file transfers across a single physical medium. Each service would have its own virtual connection, with different AALs encoding and decoding the data streams for the user services.

The AAL is considered to consist of two internal sublayers: the Convergence Sublayer (CS) and the Segmentation and Reassembly sublayer (SAR) . The CS function provides any necessary encapsulation of user data before the SAR function inserts the result into 48 byte cell payload fields. In general the SAR function provides transmission and error detection facilities on a cell by cell basis. The CS function provides transmission and error detection facilities over the 'natural' unit of data utilised by the user service (bytes, bitstreams, or variable length packets). In some applications the SAR or CS functions may be empty.

For constant bitrate services, section 2 of I.363 [4] specifies AAL type 1 (AAL1). For continuously variable bitrate services, AAL type 2 is being defined (AAL2, section 3 of [4]). AAL1 is reasonably well defined at this stage, and is expected to be used by digitised video and voice services. These services require clock signals to be provided by the AAL, and some measure of buffering within the AAL, to ensure that information crosses the AAL-higher layer boundary at a fixed rate. These two AALs will not be covered further in this paper as they are not intended to support packet or datagram services.

3.1 PACKET TRANSPORT SERVICES - TYPE 3/4 AND TYPE 5 AAL.

Many distributed computing communication protocols are based around the transmission of packets or frames of information, with arbitrary and unpredictable periods of zero activity between transmissions (e.g. DARPA Internetworking Protocol - IP [14,15]). Most LAN technologies aim to support such 'packet' transmissions (e.g. Ethernet [16], Token ring [17], LocalTalk [18], etc), and CCITT has worked to ensure at least one AAL existed to provide such support over ATM.

Initial support was provided by the development of AAL3 and AAL4, which have been subsequently merged into a single standard - AAL3/4 (section 4 of [4]). Parallel analysis of ATM and its fibre based medium led to the LAN industry proposing another packet oriented protocol, now being standardised as AAL5 [8,9]. AAL5 maximises the efficiency of cell utilisation by removing all 'per-cell' encapsulation, and depends on the underlying ATM virtual connection having very low error rates. AAL5 evolved from earlier work on the 'Simple and Efficient Adaptation Layer' (SEAL) [32], and the same algorithm is used in 'Network Compatible ATM for Local Network Applications' [10]. A simplified segmentation and reassembly layer was also defined at Bolt Beranek and Newman Inc. (BBN) in 1990 [33], to function with an undefined, and possibly null, convergence sublayer.

For both AAL3/4 and AAL5 the I.363 recommendations further subdivide the Convergence Sublayer into an upper half (the Service Specific Convergence Sublayer, or SSCS) and a lower half (the Common Part Convergence Sublayer, or CPCS). The CPCS contains only the basic processing necessary to transport a variable sized block of data across a virtual connection. Additional functions, such as flow control and the retransmission of lost or corrupted data, are implemented within the SSCS. In many early AAL implementations the SSCS will be a null layer, with the CPCS functions mapped directly to the AAL-User interface. In this paper we will assume a null SSCS, as the CPCS functions defined in I.363 provide sufficient processing for typical network/transport layers such as TCP/IP.

Discussion over which of these AALs will be recommended for connection oriented and connectionless services appears to be converging on AAL5 as the preferred choice. Examples of such services would be carrying MAC frames (e.g. Ethernet packets) across a virtual connection for LAN-LAN interconnection [19,20], or carrying Network layer Protocol Data Units (PDUs, e.g. IP packets carried either directly between ATM-capable IP hosts or between ATM-capable IP routers [21]). AAL3/4 is still of interest in its own right, both for comparison to AAL5 and because it is closely related to the protocol used by the IEEE 802.6 MAN standard [29], on which the interface protocol for Switched Multimegabit Data Service (SMDS) is based [30].

3.1.1 THE AAL3/4 SEGMENTATION PROCESS.

The basic AAL3/4 CPCS function provides a 'Non-assured' service, where each user frame (or packet) is contained in one AAL_SDU and transferred in one CPCS_PDU (section 4.3.2.2 of [4]). In this mode the AAL does not guarantee the successful transfer of any given AAL_SDU from peer to peer.

Figure 4 shows a simplification of the transmission process on a single virtual connection. An AAL_SDU (user PDU) is encapsulated to generate a CPCS_PDU. This is then segmented into a series of 44 byte SAR_PDU payloads, wrapped with 2 byte headers and trailers to create 48 byte SAR_PDUs, and inserted into the payload fields of cells for transmission.

The CPCS_PDU encapsulation contains information which enables the CPCS function to verify the correct reception of the entire CPCS_PDU, shown in Figure 5. The PAD field ensures the CPCS_PDU trailer is 32 bit aligned, and the AL field pads the trailer to 32 bits. The Common Part Indicator (CPI) specifies how the CPCS should interpret the remaining fields of the header and trailer. Presently a CPI of zero indicates that the BAsize field contains an estimate of the incoming CPCS_PDU's size in bytes, and the Length field contains the exact size of the CPCS_PDU payload, in bytes.

The BAsize field may be used by the reassembly machine to preallocate buffer space, a function considered unnecessary during the development of AAL5. The Length field aids in the detection of reassembly errors such as loss or gain of cells. Additional error detection is possible using the Btag and Etag fields. Both fields are set to the same value when the CPCS_PDU is transmitted, the actual value used being unimportant so long as it changes for successive CPCS_PDUs.

Of greatest interest to us is the process of segmenting and reassembling a CPCS_PDU. Figure 5 also shows a breakdown of the SAR_PDU, header, and trailer. The ST field indicates what type of information is carried within the SAR_PDU. Four types exist - Beginning Of Message (BOM), Continuation Of Message (COM), End Of Message (EOM), and Single Segment Message (SSM). If a CPCS_PDU is less than 45 bytes long it will fit within the payload of a single SAR_PDU - and will be sent as an SSM. Longer CPCS_PDUs are sent as a sequence of SAR_PDUs, beginning with a BOM, followed by zero or more COMs, and ending with an EOM.

While segmenting and sending a given CPCS_PDU, the SN field is incremented by one, modulo 16, for each SAR_PDU sent. The LI field indicates how many bytes of the payload are actually valid. It is always 44 for BOMs and COMs, but may be less for EOMs and SSMs corresponding

to CS_PDUs that do not finish on a 44 byte boundary. The CRC field provides bit-error detection across the entire SAR_PDU. This level of encapsulation probably reflects an early belief that bit errors on the fibre medium would be more common than technology now achieves.

A distinctive feature of AAL3/4 is that it can multiplex different streams of AAL_SDUs across a single virtual connection, unlike AAL5 where such multiplexing must be performed at a higher layer. Multiplexing is achieved through the 10 bit MID field. By associating each stream of AAL_SDUs, and hence CPCS_PDUs, with a distinct MID value, SAR_PDUs may be interleaved on the virtual connection and successfully extracted at the receiving end. When a user process is not multiplexing AAL_SDUs from distinct sources the MID field is set to zero.

3.1.2 THE AAL3/4 REASSEMBLY PROCESS.

The aim of reassembly is to recreate the CPCS_PDU. Once this is obtained, the original AAL_SDU may be extracted and passed up to the receiving user process. When a cell arrives it is considered to either represent a new CPCS_PDU (if it is a BOM or SSM) or add to a CPCS_PDU already being reassembled.

Under normal operation the receiving AAL will, on a given virtual connection, see a repeating sequence of BOM-COMs-EOM...BOM-COMs-EOM.... cells (interspersed with SSMs if the traffic is a varied collection of short and long AAL_SDUs). The first BOM causes the AAL to note the MID and SN fields, and then look for following COMs which contain the same MID and have correctly incrementing SN fields. The payload is extracted from each SAR_PDU to form the CPCS_PDU. Finally when an EOM arrives, in sequence and with a matching MID value, the CPCS function is passed the final payload with an indication that the CPCS_PDU should now be complete.

Final error checking involves matching the Etag with the Btag, and ensuring the Length field matches the received data. Note that AAL3/4 relies on the CRC in each SAR_PDU to ensure the CPCS_PDU's component parts are error free. Handling multiplexed AAL_SDU traffic requires several instances of the SAR and CPCS entities to exist simultaneously. Each instance of the SAR/CPCS is associated with a valid MID, and a particular AAL service user.

COM and EOM SAR_PDUs that arrive with a MID value not corresponding to a current CPCS_PDU are ignored. Those that arrive with an out of sequence SN field are considered to indicate an error in the transmission process, and the CPCS must abort reassembly of the current CPCS_PDU. The arrival of a BOM with the MID of a CPCS_PDU that is still being reassembled also causes the current CPCS_PDU to be aborted.

3.1.3 THE AAL5 SEGMENTATION AND REASSEMBLY PROCESS.

A major complaint against AAL3/4 is that it over-provides facilities for the AAL user. The CPCS and SAR overheads of 8 bytes per AAL_SDU, and 4 bytes per cell, are considered unjustifiably large. In contrast AAL5's philosophy shifts responsibility for multiplexing to a higher layer, simplifies error detection facilities, discards the 'BAsize' facility to preallocate reassembly buffers, and allows only one type of CPCS_PDU format. All SAR level encapsulation is removed, enabling 48 bytes of CPCS_PDU to be carried within each cell. The SAR_PDU 'type' is now indicated by the ATM-layer-user-to-ATM-layer-user (AUU) parameter in the Payload Type Indication (PTI) field of the ATM cell header (Section 2.2.4 of [6]). Figure 6 shows a simplified AAL5 transmission of an AAL_SDU.

Only two SAR_PDU types are used under AAL5. The AUU is set to one to mark the last (or only, in the case of a small CPCS_PDU) SAR_PDU of a CPCS_PDU. If the CPCS_PDU spans more than one cell, cells with the AUU set to zero carry the beginning and continuation SAR_PDUs.

The CPCS_PDU is shown in Figure 7. The Padding field, which may be between 0 and 47 bytes long, ensures that the Trailer always occupies the last 8 bytes of the last SAR_PDU, and that the overall CPCS_PDU is aligned on a 48 byte boundary. Primary error detection is provided by the 4 byte CRC field, which covers the entire CPCS_PDU (including the Length field, but excluding the CRC itself). The Length field specifies the payload size in bytes. The Length field also provides a backup protection against cell loss or gain, in the unlikely event it is not picked up by the CRC. (In [24], Wang and Crowcroft show that the 32 bit CRC provides extremely effective detection of cell misordering, a fault condition they considered possible under network failure conditions).

The Common Part Indicator (CPI) field currently has no defined purpose except to pad the trailer out to 64 bits. It must be set to zero; non-zero values are reserved for carrying management control messages (when defined in later editions of I.363). The CPCS User-to-User (CPCS-UU) indication allows a single byte to be carried, uninterpreted, between CPCS user entities.

Because AAL5 does not support the simultaneous multiplexing of CPCS_PDUs on a single virtual connection, all SAR_PDUs carry either the current CPCS_PDU or the next one. The first SAR_PDU to arrive after a SAR_PDU with AUU set to 1 is assumed to be the beginning of the next CPCS_PDU, irrespective of whether it is a SAR_PDU with AUU set to 1. A CPCS_PDU shorter than 41 bytes may be represented by a sole SAR_PDU with AUU set to 1 (40 bytes of CPCS_PDU and 8 bytes of Trailer fit within a single SAR_PDU).

4. SOURCES OF CELL LOSS

Between the segmentation layer of the source AAL and the reassembly layer of the destination AAL lie a variety of sources for cell loss. Cell loss occurs whenever a cell is discarded (either due to bit errors or congestion control), or a cell is erroneously routed onto another virtual connection (or a non-existent virtual connection). The probability of cell loss on an ATM virtual connection, through mechanisms unrelated to congestion, is expected to be better than 10^{-8} . Congestion induced cell loss is still hard to predict meaningfully, as models for bursty AAL3/4 and AAL5 traffic are often simplistic. We shall look at each major area in turn qualitatively, without attributing much in the way of loss probabilities to them.

4.1 ERRORS ON THE TRANSMISSION MEDIA.

The transmission medium (e.g. fibre) is the lowest level source of errors in cell contents. Bit errors occur evenly across both the cell header and payload, and may occur on each and every segment of fibre a cell traverses. Errors in the cell payload do not contribute directly to cell loss prior to the destination AAL. However at every point where an ATM layer receives cells (destination or intermediate switching node), header errors can cause problems.

The cell header is protected by an 8 bit Header Error Check (HEC) field (Figure 1). The HEC is capable of providing single-bit error correction and multi-bit error detection. A cell receiver operates in two modes - "correction" and "detection". In detection mode all cells with detectable header errors are discarded. In correction mode single-bit errors are corrected, while cells with multi-bit errors are discarded. Detecting an error while in correction mode causes the receiver to switch to detection mode after processing the current cell. Reception of a valid cell while in detection mode causes the receiver to process the cell and switch back to correction mode. The HEC function is described more fully in Section 4 of Recommendation I.432 [11], where it is also used to provide cell delineation.

Cell loss occurs when a corrupt cell is detected and properly discarded. Cell loss may also occur when header corruption is not detected, but has resulted in a new VPI or VCI. The cell then 'disappears' from the virtual connection it should have been on. A further mechanism for cell 'disappearance' is when a single-bit error is incorrectly reconstructed, again possibly leading to a new VPI or VCI. (These disappearances can affect another virtual connection, by injecting loose cells into it. This Cell Insertion has been shown to be much less significant than Cell Loss in most cases [12].)

4.2 DISCARDING CELLS FOR CONGESTION CONTROL

The broad effect of congestion control is due to the combination of Policing Units acting close to the cell source, and overload response mechanisms acting at the inputs to cell switching nodes. In general each virtual connection will have an associated agreed traffic parameter. Policing Units act to ensure the compliance of the network user with their agreed traffic parameters. A Policing Unit may simply discard cells sent in violation of the agreed parameters.

Most switching nodes will, by their design, discard cells when the switch fabrics and associated buffers are overflowing. The Cell Loss Priority (CLP) bit in the header is used to indicate to switch nodes which cells they should discard first, if possible. A variation on the discarding Policing Unit is a 'violation tagging' policing unit [13], where cells in violation of the traffic parameters are allowed through but tagged so that switch nodes know they may be discarded first.

A further consideration is where blocks of cells are discarded. Congestion is intimately linked to the load offered to the network by an AAL user. The transport layer protocols of most packet oriented services will retransmit a lost packet, so a single lost cell results in a retransmitted AAL_SDU. An obvious extension to congestion control of AAL3/4 or AAL5 virtual connections is, having discarded one cell, to discard any following cells associated with the same CPCS_PDU. The reasoning is that the rest of the CPCS_PDU will eventually be retransmitted anyway, so removing the extra cells reduces the load on the network links beyond the congestion point. However this mechanism cannot be applied to AAL3/4 connections, as it breaks layering rules and adds implementation complexity to policing units. In section 7 a method of applying this to AAL5 connections is described in more detail. A similar idea is described as "Selective Discarding of Cells" in [13].

4.3 PROCESSING ERRORS IN SWITCHING NODES AND ENDPOINTS.

A final area of cell loss is when header processing occurs incorrectly at a switching node or virtual connection endpoint, leading to possible complete loss or misrouting of cells. Actual symptoms in this area will depend on the physical and logical architecture of switching nodes along the path taken by any given virtual connection. A receiving node may incorrectly interpret a cell's VPI or VCI and fail to match it to an open virtual connection, thus losing the cell.

5. AFFECT OF CELL LOSS ON REASSEMBLY

Cell loss, whether through bit errors or congestion-related discarding, has different consequences for the AAL3/4 and AAL5 CPCS machines. Depending on where the error occurs, AAL3/4 may be able to pass partially reassembled CPCS_PDUs to the user along with an error indication. AAL5 can only pass up an error indication, losing any part of the current CPCS_PDU correctly received prior to the lost cell.

5.1 AAL3/4

When a receiver SAR/CPCS machine is in the idle state, it will reject all COM and EOM cells passed to it. The reception of a valid BOM is required to enter reassembly mode, so the loss of a CPCS_PDU's BOM results in the complete loss of the CPCS_PDU. This form of error does not allow the AAL to pass any information at all to the user, although statistics on each connection could be updated to show the reception of unexpected SAR_PDUs.

When the SAR/CPCS machine is in the reassembly state, incorrect sequence number (SN) progression between SAR_PDUs will reveal the loss of a COM. In the case where a multiple of 16 consecutive cells are lost, and the SN wraps around, the loss of data will be detected by the CPCS_PDU being undersized. Either of these error conditions leaves the CPCS with at least the first 40 bytes of the AAL_SDU correctly received (the BOM that originally began the reassembly, and any further COMs up to the point where cell loss occurred). If the error condition is detected through incorrect SN progression, data between the BOM and the last SAR_PDU received with a correct SN may be passed up to the AAL user as a partial CPCS_PDU. If the error is not detected until the Length of the CPCS_PDU is checked, then only the first 44 bytes of the reassembled CPCS_PDU may be legitimately retrieved.

Two mechanisms exist to detect EOM loss. If the BOM of the next CPCS_PDU on the same MID arrives before the EOM for the current CPCS_PDU, the partially reassembled CPCS_PDU must be released by the SAR/CPCS machine. The entire partially reassembled CPCS_PDU received to that point may be considered valid, and so passed to the AAL user along with an error indication. This mechanism works for cases where just the EOM is lost, or where a cell burst knocks out some COMs followed by the EOM.

A cell loss burst that knocks out the EOM and the following BOM, and slips past the SN checks, will be detected when the Btag and Etag fields fail to match (the Length indication may fail to pick up this error if the cell burst loses as many cells as are added by concatenating the two CPCS_PDU fragments). In this case only the first 44 bytes of the first CPCS_PDU may be legitimately retrieved. Nothing may be established about the second CPCS_PDU.

Another method is to attach a timer to each CPCS_PDU under reconstruction, and signal an error when it is not reassembled within a certain time frame. The advantage of this mechanism is that the time taken to detect a transmission error, and free CPCS resources, is bounded by the local timer rather than the unknown arrival time of the next CPCS_PDU.

5.2 AAL5

The lack of encapsulation and sequence checking of the SAR_PDU means that reassembly errors are only detected when the CPCS_PDU trailer arrives. The error detection mechanisms, and the minimal information carried by each SAR_PDU about its position within the CPCS_PDU makes it impossible to indicate how much of the CPCS_PDU has been received correctly.

Single bit errors in SAR_PDUs will not be picked up until the CPCS_PDU CRC is calculated, and result in the discarding of the entire CPCS_PDU. Loss of cells with AUU=0 will, in the first instance, be detected by an incorrect CRC when the trailer finally arrives. If the CRC fails to flag the error, the Length field mismatch will ensure the CPCS_PDU is discarded.

Loss of cells with AUU=1 is detectable in three ways. The SAR_PDUs of the following CPCS_PDU may be appended to the first, resulting in a CRC error (or a Length mismatch) being flagged when the second CPCS_PDU's trailer arrives. Alternatively the AAL may enforce a length limit. If the length limit is exceeded while appending the second CPCS_PDU, an error is flagged and the assembled data is discarded. Thirdly a timer may be attached to the CPCS_PDU reassembly. If it expires before the CPCS_PDU is completely received, the assembled data is discarded.

An important side effect of the second method is that both CPCS_PDUs will be lost, as an unknown amount of the second CPCS_PDU will be flushed along with the first. When the reassembly process restarts afresh on the remaining cells of the second CPCS_PDU, it will ultimately generate a CRC or Length error and be discarded. The first method will also lose both CPCS_PDUs, although it is theoretically possible to recover the second CPCS_PDU. The received Length field may be used to estimate where the second CPCS_PDU should have started, and a CRC recalculated from this point. If the CRC passes, then the second CPCS_PDU may be considered to have been recovered, losing only the first one. However implementing this recovery mechanism may be hard to justify in hardware AALs.

The vulnerability of AAL5 to 'concatenation errors' needs to be placed into perspective. The number of user bytes lost through concatenation errors is significantly less than the overhead

needed by AAL3/4 to avoid such errors (the 2 bit ST field). Assuming a cell loss probability of 10^{-8} , and 20 cell CPCS_PDUs, we can see that:

- $20 * 2/8 = 5$ bytes of an AAL3/4 CPCS_PDU is consumed providing the mechanism needed to avoid concatenation errors.
- Probability of the last cell of AAL5 CPCS_PDU being lost is 10^{-8} (concatenation loss does not depend on the history of previous cell loss within the CPCS_PDU).
- The average number of CPCS_PDU bytes lost through concatenation error is approximately $(48 * 20) * 2 * 10^{-8} = 1.92 * 10^{-5}$.

A cell loss rate of roughly $2.6 * 10^{-3}$ is needed before the AAL3/4 ST field becomes justifiable.

There are repercussions for the design of congestion control mechanisms such as selective cell discarding schemes in Policing Units. For AAL5 all cells up to the AUU=1 may be dropped, but the cell with AUU=1 must be passed to ensure the receiving AAL remains synchronised and loses only the intended CPCS_PDU.

6. THE BENEFITS OF AAL3/4 ERROR INDICATIONS TO AAL SERVICE USERS.

Network and transport layer protocols, like TCP/IP, do not expect the underlying data link layer to provide much indication of reception errors. This is one of the reasons AAL5 could be 'designed down' to provide its minimal service. The Internet Engineering Task Force (IETF) has already moved to recommend AAL5 for multiprotocol encapsulation [28]. However it is worth considering what we are losing by abandoning AAL3/4.

Network and transport layer protocols that try to provide guaranteed data delivery use sequence numbers, retransmission timers, and acknowledgments to detect lost packets (PDUs). Guaranteed delivery may be provided on a per-link basis by the network layer, or it may only be provided end-to-end by the transport layer.

Positive acknowledgment protocols retransmit PDUs not acknowledged by the receiving entity. A corrupted AAL_SDU will trigger such a retransmission, as the receiving network/transport entity will either not see the packet or discard it itself, and consequently not acknowledge it.

Retransmission is, therefore, only an indirect consequence of error detection in received PDUs. Providing a fragment of a corrupted AAL_SDU serves little purpose with such protocols.

More useful are protocols that negatively acknowledge the receipt of a corrupted PDU. The errored packet is assumed to contain enough information to identify it to the sender. This information is held in the header of most network or transport layer PDUs. A useful AAL_SDU fragment needs to contain at least the header of the layer providing retransmission. A similar scheme is 'Selective retransmission', where lost sequence numbers are used to trigger a request for missing PDUs.

As discussed in section 5.1, AAL3/4 allows for AAL_SDU fragments to be provided for analysis by the AAL service user. An obvious use for these fragments is to enable suitably 'aware' higher layer protocols to expedite retransmission of lost AAL_SDUs. In this section we will focus specifically on what AAL3/4 could offer some typical services, under the assumption that cell loss has already corrupted an incoming CPCS_PDU.

6.1 ASSURED MODE AAL SERVICE.

AAL users are expected to be offered both 'Non-assured' and 'Assured' delivery of AAL_SDUs when using AAL3/4. Non-assured mode is provided by the basic CPCS functions described in section 3. The provision of assured mode, where the AAL_SDU is guaranteed to reach the remote AAL user, comes about through the addition of an SSCS between the CPCS and the AAL user. The SSCS required to provide assured service will involve further encapsulation of the AAL_SDU into an SSCS_PDU (Service Specific CS_PDU) before becoming CPCS_PDU for segmentation.

As noted in section 5.1, AAL3/4 may be able to provide at least 44 bytes (from the BOM) of a CPCS_PDU that has suffered cell loss, which means at least the first 40 bytes of the SSCS_PDU may be recovered. Any SSCS providing assured delivery will incorporate PDU retransmission. Given that the SSCS_PDU header is fully contained within the first 40 bytes, and contains its own internal CRC checks, an intelligent SSCS design will incorporate a negative acknowledgment scheme using the headers of corrupted SSCS_PDUs. Given the tight link between the SSCS and CPCS this would seem a reasonable approach (whether such development is worthwhile will ultimately depend on what services chose to use AAL3/4 in the future).

6.2 TCP/IP.

Although IETF is proposing AAL5 to carry IP traffic, there is no technical reason to avoid AAL3/4. The DARPA IP layer does not attempt to provide guaranteed delivery, so retransmission algorithms will exist at the transport layer. As a consequence AAL_SDU fragments must be long enough to contain the IP header as well as the transport layer header before they are useful. The IP header size ranges from 20 bytes up to 60 [22]. Assuming no further encapsulation of the IP

datagram, we need to have received at least the first two cells of an errored CPCS_PDU (the BOM and COM, giving us 84 bytes of AAL_SDU) to be sure of capturing the IP header. IETF encapsulation adds an extra 9 bytes which would need to be captured. The IP entity would also need to be modified to pass along AAL_SDU fragments to the destination transport layer.

The Transport Control Protocol (TCP) [23] is often considered as *the* transport layer for IP. It does not provide for selective retransmission of lost PDUs, so it can make no use of AAL_SDU fragments. However some work is being done on selective retransmission for TCP (RFC-1072, October 1988, although recently deferred in RFC-1323, May 1992). A TCP header is 20 bytes long, with a possible addition of 44 bytes of TCP options. At least 2 cells, and possibly 3 if header options were being carried, would be needed from an errored CPCS_PDU before we could offer a useful fragment of TCP header to the receiving TCP layer.

6.3 XTP

The Xpress Transfer Protocol (XTP, [25]) uses selective retransmission. Loss detection involves the use of PDU sequence numbers. When running over network layers like IP, which do not guarantee PDU ordering, XTP is not able to make immediate assumptions about a missing sequence number (the FASTNAK facility must be disabled). However, the protocol might easily be modified to accept an AAL_SDU fragment as proof of a PDU loss, even across an IP link. An XTP header is fixed at 40 bytes, and contains its own checksum. To accommodate that, at least 3 cells of an errored CPCS_PDU must be received before passing the AAL_SDU fragment up.

XTP may also be run directly across an ATM virtual connection, with XTP packets becoming AAL_SDU's. In this case the XTP header is contained completely in the first 44 bytes of an errored CPCS_PDU, so provided the BOM is not lost selective retransmission is possible.

6.4 VMTP

For historical interest, we recall that the Versatile Message Transaction Protocol (VMTP) also used selective retransmission [26,27]. In its first draft it was positioned above the IP layer, but could be used directly over a data link layer. A VMTP packet header is fixed at 64 bytes (including a 40 byte Message Control Block capable of carrying the parameters for short request/response transactions in their entirety). At least 3 cells of an errored CPCS_PDU need to be received to provide a useful AAL_SDU fragment when running over IP.

6.5 LAN-LAN INTERCONNECTS

LAN-LAN interconnection is based on the exchange of MAC level packets between physical LANs, to emulate an extended LAN. A common feature of LAN MAC layers is that no guarantee of packet delivery is given. The network layers using the LAN are already designed on the assumption that packets will occasionally vanish.

Typically interconnection is provided through a MAC level router. The router promiscuously picks up packets it believes to be destined for the remote LAN and passes them across an ATM virtual connection. Providing the router with fragments of AAL_SDUs is pointless as there is no mechanism for passing this information to the originating network or transport layer. The AAL underlying the router can do little more than keep statistics on error types and occurrence rates for the virtual connection.

6.6 CLNP - CONNECTIONLESS NETWORK PROTOCOL.

The ITU-TSS's support for connectionless services over the B-ISDN is evolving within draft recommendation I.364 [31]. The 1992 version requires the Connectionless Network Access Protocol (CLNAP) to be layered over AAL3/4, and provides only non-assured service. CLNAP does not, therefore, need CPCS_PDU fragments itself. However, if it were supporting a transport protocol using negative acknowledgments, as described above, then some form of 'fragment pass through' facility may be useful. If AAL5 replaces AAL3/4 in support of CLNAP, this will not be necessary.

7. THE EFFECT OF SELECTIVE DISCARDING

As mentioned in section 4.2, one approach to link congestion is to drop all cells belonging to a given CPCS_PDU (or 'violation tag' them by setting the CLP bit) as soon as one cell fulfills the conditions for dropping or tagging. Congestion occurs at the input buffers to the switch when the offered load on all incoming ports exceeds the speed at which the switch can provide service. Dropping the remainder of a CPCS_PDU immediately reduces the transient load on the switch at the time when it is under the most pressure. This increases the likelihood of cells arriving on other ports being serviced without loss, as the apparent load offered by the port that has dropped its CPCS_PDU temporarily drops to zero.

However section 3.1 suggests some implementation problems with this idea. To implement selective discarding, a policing unit through which the cells travel must be "aware" of the SAR service being used on each virtual connection. To police entire CPCS_PDUs it must also track the

position of each SAR_PDU within each CPCS_PDU. For an AAL5 virtual connection this entails monitoring the transitions of the AUU indicator - a cell header function. For an AAL3/4 virtual connection the policing unit would have to be vastly more complex. To track SAR_PDU position, the ST field (BOM, COM, EOM, and SSM indications) would need to be monitored. Since a given connection may be simultaneously carrying multiple CS_PDUs, the MID field must also be monitored. These fields are protected by a CRC in the SAR_PDU trailer, so the entire SAR_PDU payload needs to be scanned before the ST and MID fields may be used.

The complex processing needed by AAL3/4 traffic would destroy the speed and delay figures for a cell switch implementing selective discarding. In addition it requires layering rules to be broken in order to function. As switch fabrics should not be interpreting the contents of cell payloads, this process should not be used.

Selective discard is legitimate and practical to implement for AAL5 traffic. The AUU indication is contained within the header, covered by the HEC field and already verified as part of the switching process. The tracking algorithm is very simple. If a cell with AUU=0 triggers a dropping or tagging action, then cells can continue to be dropped or tagged until one with AUU=1 is found. As noted in section 5.2, AAL5 uses cells with AUU=1 to synchronise the reassembly process. To avoid losing two CPCS_PDUs at the receiving AAL, selective discard should not drop cells with AUU=1. Tagging cells with AUU=1 will not cause immediate problems. However, it increases the probability of a later cell switch discarding it when congestion is imminent.

8. CONCLUSION.

Cell loss is going to exist between AALs, whether through bit errors at the physical level or congestion management mechanisms acting in switches and policing units. Segmentation means that cell loss may only affect parts of an AAL_SDU. However, neither AAL3/4 nor AAL5 are able to reconstruct errored AAL_SDU. The AAL3/4 algorithm is able to identify the cells received correctly before data was lost. If the first 2 or 3 cells are received correctly the headers of the Network or Transport layer PDUs may have been received intact. Network and/or Transport layers that utilise selective acknowledgment and retransmission may use these headers to expedite the retransmission of the lost PDU. It appears reasonable for a type 3/4 AAL to offer higher layers the option of receiving AAL_SDU fragments with error indication when cell loss is detected.

AAL5 offers no similar possibilities. Losing one cell causes the whole AAL_SDU to be discarded. Loss of the SAR_PDU with AAU=1 (end of message) actually results in the loss of the subsequent AAL_SDU too. However the simplified SAR_PDU identification of AAL5 opens the possibility of using selective discarding as a means of congestion control on AAL5 links. All but

the EOM cell (AUU=1) may be selectively discarded by a cell switch when the incoming cells violate traffic parameters or the switch is suffering congestion problems. Selective discarding offers benefits to the local switch fabric, and downstream links, as it sheds redundant cells after one cell of a CPCS_PDU is lost.

The issues surrounding cell loss raise many questions about the broader area of network management policy and implementation. The impact of cell loss through the proper functioning of network congestion control mechanisms depends on the reassembly service being used. Customer equipment needs to be able to control the cell injection rate in response to load indications from the network. In this way it can minimise the loss of AAL_SDUs, and hence minimise the overall impact on network and transport level protocols. Analysis of destructive load management (cell dropping) and constructive load management (rate control at user equipment) is vital to the creation of a stable and powerful BISDN service.

REFERENCES

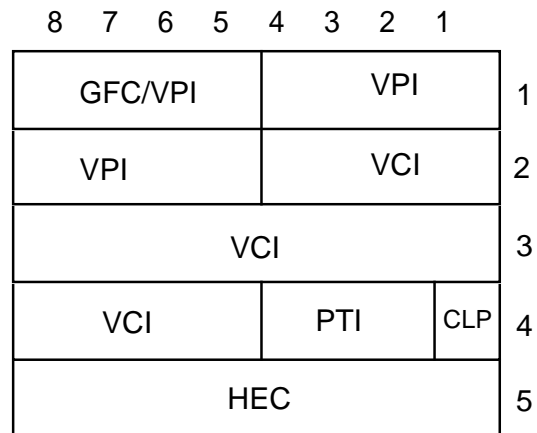
- [1] CCITT SGXVIII - Report R109, Draft recommendation I.150, "B-ISDN Asynchronous Transfer Mode Functional Characteristics", July 1992.
- [2] Martin de Prycker, "Asynchronous Transfer Mode - Solution for Broadband ISDN", Ellis Horwood, 1991.
- [3] CCITT SGXVIII Draft recommendation I.362, "B-ISDN ATM Adaptation Layer (AAL) functional description", June 1992.
- [4] CCITT SGXVIII Draft recommendation I.363, "B-ISDN ATM Adaptation Layer (AAL) specification", June 1992.
- [5] Scott Sutherland, "Worldwide Standards for Broadband ISDN - The Challenge to be faced", Australian Broadband Switching and Services Symposium '91, p.118, Sydney, 3-5 July 1991.
- [6] CCITT SGXVIII Revised recommendation I.361, "B-ISDN ATM Layer specification", June 1992.
- [7] CCITT SGXVIII - Report R111, Draft recommendation I.311, "B-ISDN General Network Aspects", July 1992.
- [8] CCITT SGXVIII Draft recommendation I.363, "B-ISDN ATM Adaptation Layer (AAL) specification", January 1993, Section 6.
- [9] ANSI T1S1.5/91-449, "AAL5 - A New High Speed Data Transfer AAL.", ANSI T1S1, November 1991.
- [10] Apple Computer, Bellcore, Sun Microsystems, Xerox, "Network Compatible ATM for Local Network Applications, Phase 1", Version 1.0, April 1992.

- [11] CCITT SGXVIII - Report R119, Draft recommendation I.432, "B-ISDN User-Network Interface - Physical Layer Specification", June 1992.
- [12] M.E. Anagnostou, M.E. Theologou, E.N. Prontonotarios, "Cell insertion ratio analysis in asynchronous transfer mode networks", *Computer Networks and ISDN Systems*, Vol. 24, p.335.
- [13] C.J. O'Neill, "Comparison of Broadband ISDN Congestion Control Methods", *Proceedings of Australian Broadband Switching and Services Symposium'91*, July 1991, Sydney, Australia, Vol. 1.
- [14] Jon Postel, "Internet Protocol - DARPA Internet Program Protocol Specification", RFC-791, September 1981, Section 1.4.
- [15] D. Comer, "Internetworking with TCP/IP - Principals, Protocols, and Architecture", Prentice Hall, 1988.
- [16] "The Ethernet - A Local Area Network", Version 1.0, Digital Equipment Corporation, Intel Corporation, Xerox Corporation, September 1980.
- [17] Thomas W. Madron, "LANs: applications of IEEE/ANSI 802 standards", John Wiley & Sons, Inc, 1989, ISBN 0-471-62049-1.
- [18] G.S. Sidhu, R.F. Andrews, A. B. Oppenheimer, "Inside AppleTalk", Addison-Wesley, March 1989.
- [19] S.L. Sutherland, J. Burgin, "Broadband ISDN Interworking", *Proceedings of Australian Broadband Switching and Services Symposium'92*, July 1992, Melbourne, Australia, Volume 2.
- [20] J. Lamont, M.H. Hui, "Some Experience with LAN Interconnection via Frame Relaying", *IEEE Network Magazine*, September 1989, p.21.
- [21] G.J. Armitage, K.M. Adams, "Some Observations on the ATM Adaptation Layer Services Needed to Support TCP/IP", *Proceedings of Australian Broadband Switching and Services Symposium'92*, July 1992, Melbourne, Australia, Volume 2.
- [22] Section 3.1 of [14].
- [23] Jon Postel, "Transmission Control Protocol - DARPA Internet Program Protocol Specification", RFC-793, September 1981.
- [24] Zheng Wang, Jon Crowcroft, "SEAL Detects Cell Misordering", *IEEE Network*, July 1992, p.8.
- [25] Protocol Engines Incorporated, "XTP Protocol Definition", Revision 3.6, 11 January 1992.
- [26] David Cheriton, "VMTP: Versatile Message Transaction Protocol - Protocol Specification", RFC-1045, February 1988.
- [27] David R. Cheriton, Carey L. Williamson, "VMTP as the Transport Layer for High-Performance Distributed Systems", *IEEE Communications Magazine*, June 1989.

- [28] Juha Heinanen, "Multiprotocol Encapsulation over ATM Adaptation Layer 5", IETF Network working group DRAFT RFC, February 1993.
- [29] "IEEE Std 802.6 - 1990 Distributed Queue Dual Bus (DQDB) Subnetwork of a Metropolitan Area Network (MAN)", Institute of Electrical and Electronic Engineers, New York USA, 1991.
- [30] "Generic System Requirements in support of Switched Multimegabit Data Service", Technical Advisory TA-TSY-000772, Bellcore Technical Advisory, Issue 2, March 1989.
- [31] CCITT SGXVIII - Temporary document 58, Draft recommendation I.364, "Support of Broadband Connectionless Data Service on B-ISDN", July 1992.
- [32] ANSI T1S1.5/91-292, "Simple and Efficient Adaptation Layer.", ANSI T1S1, August 1991.
- [33] Julio Escobar, Craig Partridge, "A Proposed Segmentation and Reassembly (SAR) Protocol for use with Asynchronous Transfer Mode (ATM)", Proceedings of the 2nd IFIP WG6.1/WG6.4 International Workshop on Protocols for High Speed Networks, Stanford, CA, November 1990.

ACKNOWLEDGEMENT

The authors gratefully acknowledge the support of Telecom Research Laboratories, Australia, under contract no. 7064.



Transmission of bits is left to right, and down the page.

Figure 1

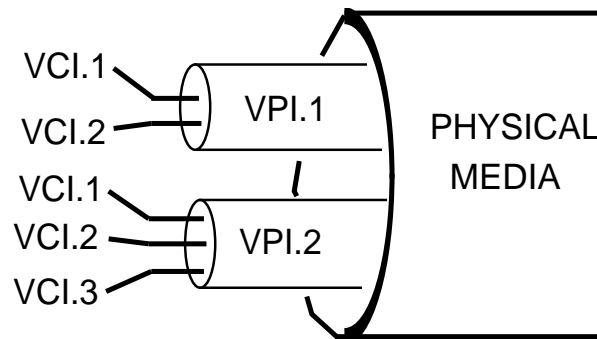


Figure 2

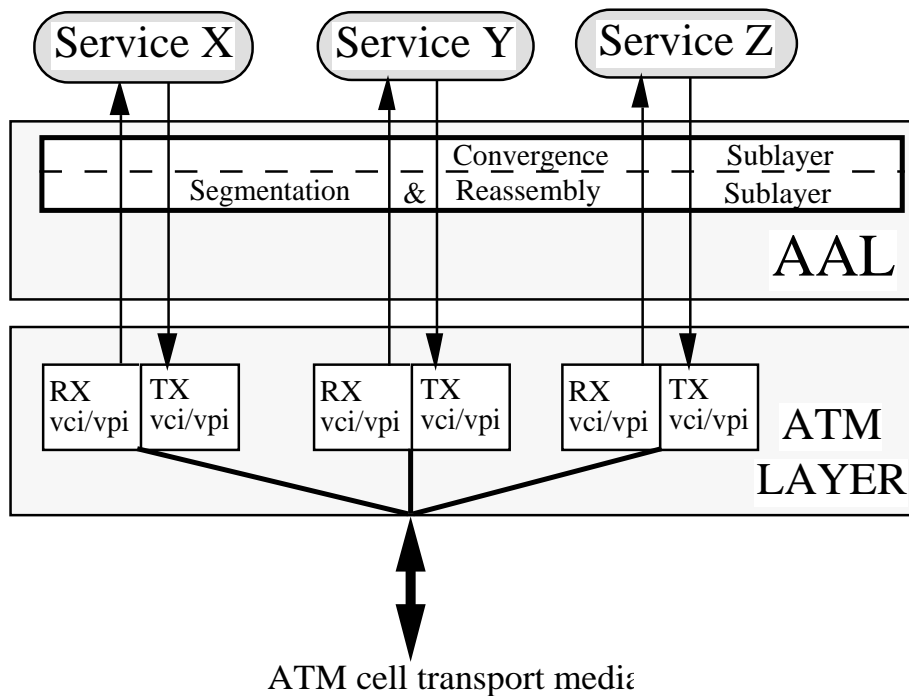


Figure 3

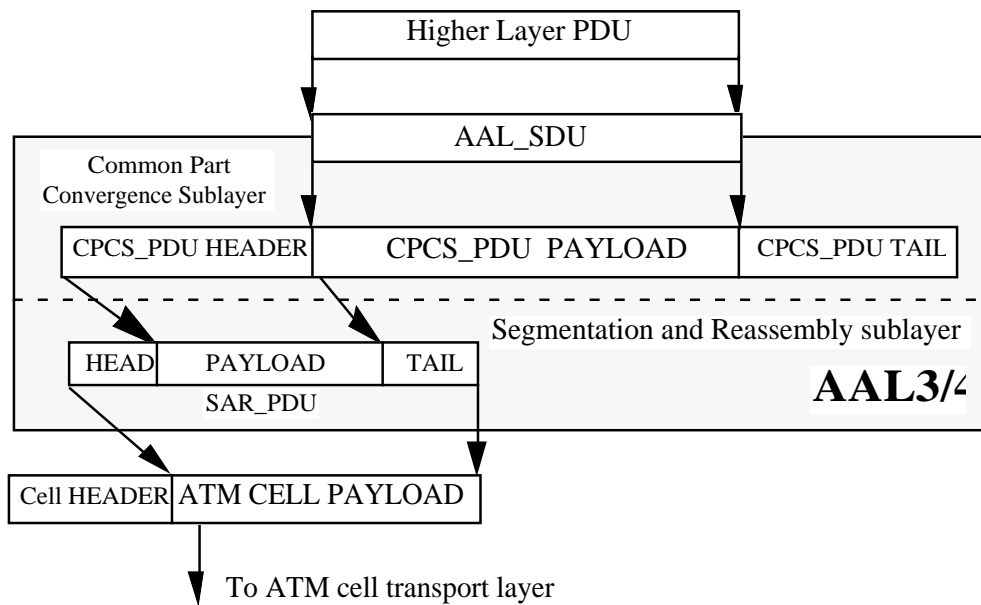


Figure 4

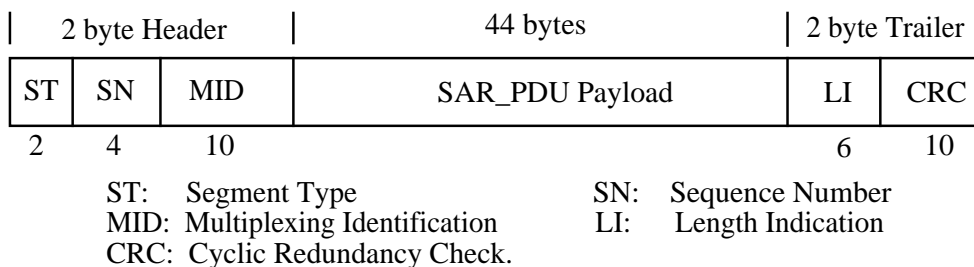
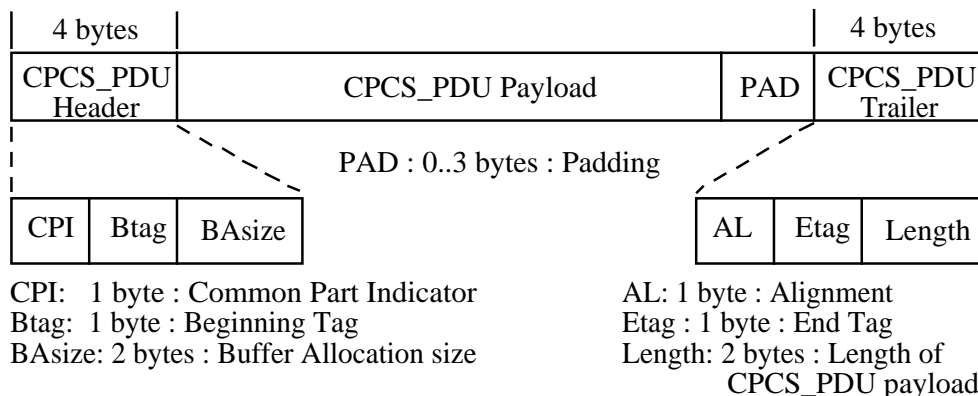


Figure 5

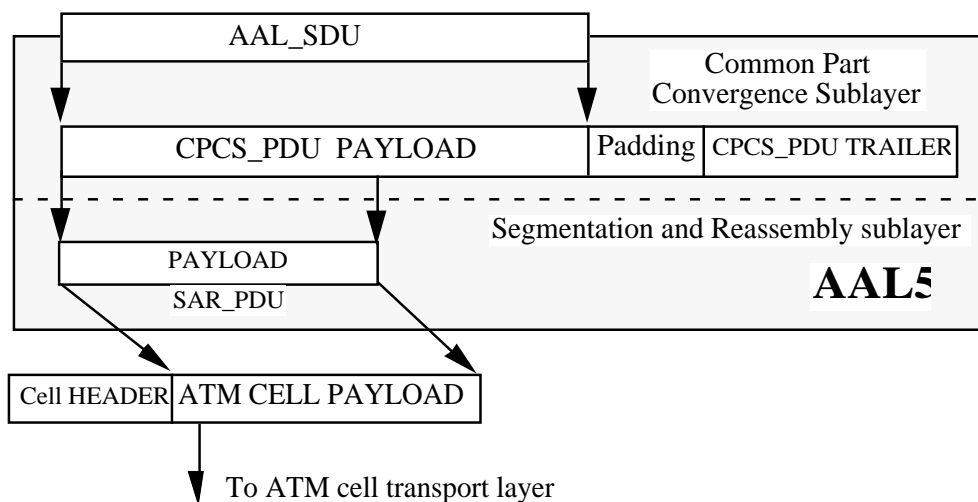
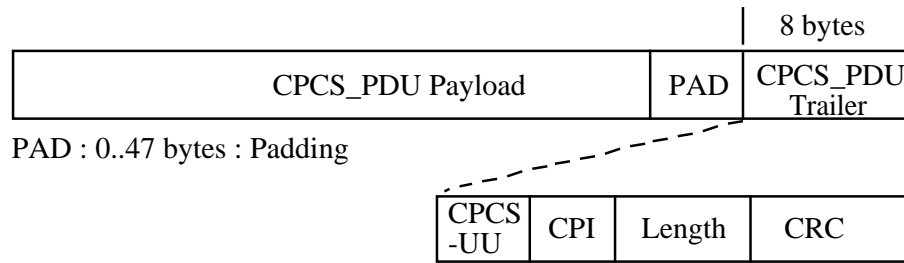


Figure 6



CPCS-UU: 1 byte : CPCS User to User Indication
 CPI: 1 byte : Common Part Indicator
 Length: 2 bytes : Length of CPCS_PDU payload.
 CRC: 4 bytes : Cyclic Redundancy Check

Figure 7