Packet Switching -
Asynchronous Transfer Mode

Joseph Spring
School of Computer Science
BSc - Computer Network Protocols & Arch’s
Based on Chapter 3, Peterson & Davie, Computer Networks: A Systems Approach, 3rd Ed

Areas for Discussion

3.3 Cell Switching (ATM)
- Introduction
- Cells
- Segmentation and Reassembly
- Virtual Paths
- Physical Layers for ATM

Introduction

ATM - Introduction

- One particular example of a switching technology
- A telephone industry standard

ATM Characteristics

- Provides a connection-orientated, packet switching service
- Hence uses Virtual Circuits (See Section 3.1.2)

Connection Request Message

- In ATM the connection set up phase is called signalling – main signalling protocol Q.2931
- Q.2931 responsible for
  - Discovering suitable route
  - Allocating resources at switches along the circuit
- Objective here is to ensure a specific QoS - Quality of Service; one of main ATM strengths

Addresses

- With any virtual circuit the address of the destination must be put into the source message
- In ATM the address can be in one of several formats
  - Most common formats are E.164 and NSAP (Network Service Access Points)
  - Details unimportant here, note addresses different to the MAC addresses used in traditional LAN’s
ATM Introduction – Packets and Cells

• In ATM the packets are of fixed size
  • An unusual feature (See p. 202 / next slide)
  • 53 bytes = 5 bytes of header + 48 bytes of payload
  • Compromise between Europe (32 bytes) and US (64 bytes)
• To distinguish between these fixed length ATM packets and the more general variable length packets we refer to the ATM packets as **Cells**
• ATM – the canonical example of cell switching

ATM Introduction – Packets and Cells

Page 202

‘… As is so often the case with standards the end result was a compromise that pleased almost no one: 48 bytes was chosen as the ATM cell payload. Probably the greatest tragedy of this choice is that it is not a power of 2, which means that it is quite a mismatch to most things that computers handle like pages and cache lines. …’

Advantages of Fixed Length Cells

**Variable Length Packets**

• Some nice characteristics:
  – 1 Byte to send, (e.g. to acknowledge receipt of packet) then put it into a minimum sized packet
  • No padding required!

Advantages of Variable Length Packets

**Variable Length Packets**

– Large file to send, then break up into maximum sized packets as required
  • Ratio of header to data bytes reduces improving on bandwidth efficiency
  • Minimises total number of packets sent, hence minimising the total processing incurred in per - packet operations
  • Particularly important in increasing throughput
  – Many network devices limited by number of packets / sec rather than bits / sec that they can process

Fixed Length Cells

**Why Fixed Length Cells?**

– Mainly to facilitate implementation of hardware switches
  • ATM created mid – late 80’s
  • 10 Mbps Ethernet was cutting edge in speed
  • To increase speed people looked to H/W
  • In telephone world – think big for switches—telephone switches serve k x 10^6 customers
  • Fixed length packets very useful for fast, scalable switches.
Advantages of Fixed Length Cells

1. Easier to build H/W
   - to do simple jobs
   - job of processing packets simpler if you know how long each packet will be

2. If packets same length, can have lots of switching elements
   • doing the same thing in parallel – improves scalability!
   • Each taking the same time to do its job

3. Reduces latency of high-priority packets
   - Impractical to preempt a packet
   - a switch may have to wait until it has transmitted the largest allowed packet before it is able to transmit the next one
   - Fixed length cells mean that queue output is tied up for a fixed time taken to transmit a cell (less than for a general packet)
   - Potential for more control over behaviour of (not length of) queues – though do tend to be shorter

4. Fixed length cells may require less buffering in a switch
   - Consider a maximum length packet
   - Switch has to buffer the entire packet before it can forward it
   - Now split the packet into say 10 cells
   - Earlier cells can be forwarded by the switch long before the last switch has been received
   • It follows that queues of cells tend to be shorter than queues of packets implying less delay for all traffic

5. Suitable for Digital Voice Transmission
   - ATM is telephone industry standard
   - If it takes too long to fill each packet with digitised voice samples or too long to store and forward long packets then customers will complain regarding the perceived excessive latency

ATM Cell Format

UNI – User-Network Interface format: Host to switch
NNI – Network-Network Interface format: switch to switch

Virtually no difference between the two formats

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>GFC</td>
<td></td>
<td></td>
</tr>
<tr>
<td>VPI</td>
<td></td>
<td></td>
</tr>
<tr>
<td>VCI</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Type</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CLP</td>
<td></td>
<td></td>
</tr>
<tr>
<td>HEC (CRC-8)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Payload</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

384 (48 bytes)

Fig 3.16: ATM Cell Format at the UNI
**ATM Cell Format - Fields**

Type: Management functions versus user data
- 3-bit so eight possible values
- Four values, when the first bit is set
  - management functions
- Remaining four values, when the first bit is clear
  - user data
  - Second bit indicates EFII ‘explicit forward congestion indication: set by congested switch to inform an end node that it is congested
  - Third bit used in conjunction with ATM Adaptation Layer 5 to delineate frames

---

**ATM Cell Format - Fields**

CLC: Cell Loss Priority
- User or network may set this element
- A one bit flag indicating whether the cell should be dropped preferentially in the case of overload
  - E.g. Video coding application could set this bit for cells that would not significantly degrade the quality of the video
  - A network element might set this bit for cells that have been transmitted by a user exceeding their agreed limit

---

**ATM Cell Format - Fields**

HEC: Header Error Check
- An 8-bit CRC
- Uses the CRC-8 polynomial:
  \[ C(x) = x^8 + x^2 + x + 1 \]
- Provides error detection and single bit error correction on the header only
- Protecting the cell header is important since an error in the VCI will lead to misdeliveries

Payload: 48 bytes = 384 bits

---

**Segmentation and Reassembly**

Assumption
- A low level protocol will just accept packets handed down to it by a high level packet, attach its own header and pass the packet on down
- With ATM this is not possible
  - Packets handed down are often larger than 48 bytes and hence do not fit into an ATM payload

---

**Segmentation and Reassembly**

How does ATM handle packets over 48 bytes?
General Mechanism
- Fragment high-level message into several low-level fixed length packets at the source
- Transmit the individual low-level packets over the network
- Reassemble the fragments back together at the destination
Segmentation and Reassembly

• This method is usually called *Fragmentation and Reassembly*
• In ATM it is often called *Segmentation and Reassembly* (SAR)

ATM Adaptation Layer

• To handle segmentation and reassembly an additional protocol layer, called the ATM Adaptation Layer (AAL) is added between the ATM layer and the variable length packet protocols that might use ATM, such as IP

---

SAR – Multiple Adaptation Layers

• ATM was designed to support
  – Voice, Video and Data
• Initially Four Layers defined:
  – AAL1: Supporting applications that require guaranteed bit rates; e.g. voice and video
  – AAL2: As with AAL1
  – AAL3: Supports connection orientated data
    e.g. Virtual Circuits
  – AAL4: Supporting connectionless data
    e.g. datagrams

---

SAR – Multiple Adaptation Layers

• Later AAL3 and AAL4 were combined into the protocol AAL3/4
  – since reasons for keeping them separate felt to be insufficient
• However, many felt that AAL3/4 suffered from shortcomings – too complex. Hence a new protocol was developed:
  – AAL5: Introduced to simplify AAL3/4

---

The four current AAL protocols are:

– AAL1
– AAL2
– AAL3/4
– AAL5
• AAL 1 and AAL2 were designed to support applications such as voice, … that require guaranteed bit rates
• AAL3/4 and AAL5 were designed to provide support for packet data running over ATM

---

ATM Adaptation Layer3/4 and 5

We consider the two computer communication protocols AAL3/4 and AAL5

AAL3/4

• This protocol provides enough information to allow variable length packets to be transported across an ATM network
• ‘Since we are now working at a new layer of the network hierarchy, convention requires us to introduce a new name for a packet’
• A packet at this level is called a Protocol Data Unit (PDU)
**ATM Adaptation Layer 3/4**

SAR involves two different packet protocols:
1. The Convergence Sublayer Protocol data Unit (CS-PDU)
2. ATM cell format for AAL3/4

CS-PDU defines an encapsulated format (adds header and trailer) for variable length PDU’s passed down to the AAL layer before they are segmented into ATM cells. The ATM cell format partitions the CS-PDU into fixed length ATM cells and adds sufficient information within the ATM payload to allow the original CS-PDU packet to be reassembled after traversing the ATM network.

---

**Fields of AAL 3/4 Packet Format**

**CPI:** Common Part Indicator
- 8 bit
- Indicates version of CS-PDU format in use
- Only value 0 currently defined

**Btag:** Beginning tag
- Should match with Etag (End tag)
- Protects against loss of end of one PDU and beginning of next PDU leading to inadvert join of two PDU’s into one which is then passed up the protocol stack.

---

**ATM Cell Format for AAL 3/4**

- The CS-PDU packet must now be fragmented into fixed size ATM cells.
- To achieve this the packet is divided into 44 byte chunks to which an AAL3/4 header and trailer totalling 4 bytes is added.
- All 48 bytes are then encapsulated in a standard ATM cell.

---

**ATM Adaptation Layer 3/4**

- Convergence Sublayer Protocol Data Unit (CS-PDU) format for variable length PDU’s
- Packets are then split into fixed length ATM cells.

<table>
<thead>
<tr>
<th>Field</th>
<th>Name</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPI</td>
<td>Btag</td>
<td>Beginning of message</td>
</tr>
<tr>
<td>BAsize</td>
<td></td>
<td>Continuation of message</td>
</tr>
<tr>
<td>Pad</td>
<td>Etag</td>
<td>End of Message</td>
</tr>
<tr>
<td>Len</td>
<td></td>
<td>Single-segment message</td>
</tr>
</tbody>
</table>

---

**Fields of AAL 3/4 Packet Format**

**BAsize:** Buffer Allocation Size
- An approximate indication of the packet size
- Used at receiving end to allocate buffer size
- Information not definitive since size of packet may not be known when value assigned to field

**Pad:** Padding up to 3 bytes of zeros
- Padding with 0-filled byte ensures trailer aligned on a 32-bit boundary for more efficient processing

**CS-PDU Trailer:**
- Consists of Etag field and Len, the real length of the PDU.
Fields in ATM Cell Format for AAL 3/4

ATM Header
- Standard ATM 5 byte header described earlier

Type:
- Indicates whether a cell is in the start, middle or end of a packet
- Allows CS-PDU to be reformed at the destination

SEQ: A four bit sequence number
- Used to detect lost ATM cells
- Clearly too few bits to detect all cases of multiple lost cells

MID: Multiplexing Identifier
- Allows multiple PDU’s to be multiplexed onto a link

Length: Number of payload data bytes
- Six bit field
  - This will indicate 44 bytes for BOM and COM
  - Possibly less than 44 bytes for end of message

CRC: CRC-10
- A 10-bit CRC
  - Uses the CRC-10 polynomial:
  \[ C(x) = x^{10} + x^9 + x^5 + x^4 + x + 1 \]

ATM Adaptation Layer 3/4

ATM Adaptation Layer 5 - AAL 5

Motivation for AAL5
- Too much overhead generated in AAL3/4
- Yet segmentation and reassembly quite simple

Changes introduced in AAL5
1. Two bit type field in AAL3/4 replaced by single bit in ATM cell header
   - A set bit indicates the last cell in a packet
   - Just enough information to reassemble the packet

AAL3/4 Overhead in ATM Cells

Note:
- Combining AAL3/4 with ATM results in a poor overhead to payload ratio
  - Data: 44 bytes (max)
  - ATM Header: 5 bytes
  - AAL3/4 Header: 2 bytes
  - AAL3/4 Trailer: 2 bytes

Minimum Overhead: \( \frac{9}{44} = 17\% \) (approx)
i.e. overhead:payload = \( \frac{9}{44} = \ldots \)

Changes introduced in AAL5 (Continued)

2. Four bytes no longer ‘wasted’ in each ATM cell
3. All AAL3/4 fields in the ATM cells that provide protection against lost, corrupt or mis-ordered cells are now provided by fields in the AAL5 CS-PDU packet format
ATM Adaptation Layer 5 Packet Format

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Data</td>
<td>Pad</td>
<td>Reserved</td>
<td>Len</td>
</tr>
</tbody>
</table>

Fig 3.21: ATM Adaptation Layer 5 Packet Format

Data: Variable length
Pad: Data is padded out so that the CS-PDU will fit exactly into a group of ATM cells

ATM Adaptation Layer 5 Packet Format

Reserved: Currently zero
CRC-32: A very powerful 32-bit CRC check
  - Power derives from the number of bits provided
Functionality
1. CRC-32 checks for lost or misordered cells and for errors in the data
   - But mis-ordered cells can no longer be reordered
2. The single 32-bit CRC check is more powerful than the 10-bit CRC’s previously provided in each ATM
3. The multiplexing feature is lost

Summary

- Generally, AAL5 is more popular with the CS community than AAL3/4
Virtual Paths
- Two VCI fields in ATM provide a simple Virtual Circuit hierarchy
- See Section 3.3.3

Summary

Physical Layers of ATM

- ATM is very closely related to SONET
- Nonetheless, ATM and SONET are distinct protocol layers that have been defined separately.
- ATM therefore doesn’t have to run on SONET

Summary

3.3 Cell Switching (ATM)
- Introduction
- Cells
- Segmentation and Reassembly
- Virtual Paths
- Physical Layers for ATM