12: Transporting Interactive Media

Mark Handley

Interactive Multimedia Sessions

- Telephony, conferencing, etc. don't tolerate delay:
  - 200ms RTT for interactive conversation.
- Streaming media also has limited delay budget, although much larger than telephony.

Result: not all lost packets can be retransmitted before their playout time.

- We need to minimise the *perception* of loss.

Application Data Units

An *Application Data Unit* (ADU) is the application's natural unit of data.

- Depends on the application and on the codec.

- Audio:
  - PCM: one sample.
  - GSM, LPC, CELP: one audio frame.
- Video:
  - One video frame? One DCT block?

Packetization: *not too big, not too small.*

IP packets have a 20 byte IP header, plus 8 bytes UDP, plus application headers. Perhaps 40 bytes overhead per packet.

- Need *enough bytes of payload* in a packet to be worth the overhead.

IP packets typically cannot exceed 1500 bytes (and sometimes less) without using IP fragmentation:

- If one IP fragment is lost, the remaining fragments of the packet need to be discarded at the receiver.
- **Don't rely on IP fragmentation for multimedia data!**

Application Data Units vs Packets

Audio: ADU usually determined by codec framing or by end-to-end delay requirements.

- No more than 80ms of audio per packet.

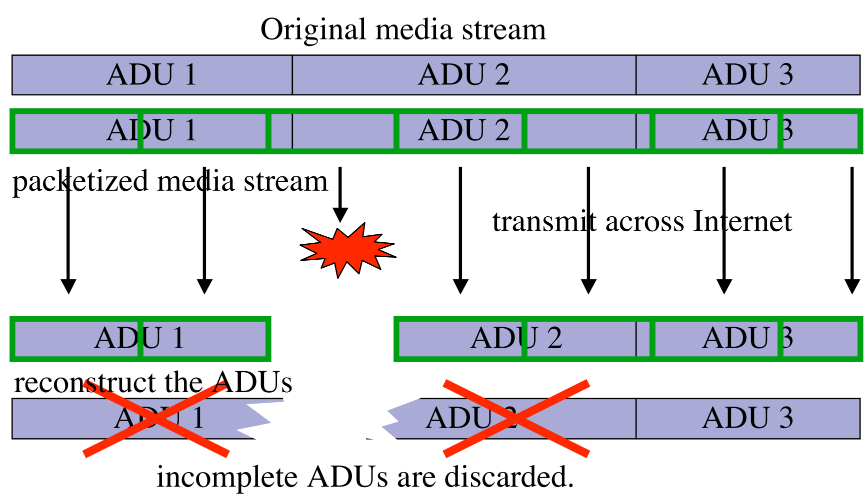
Video: ADU determined by codec resynchronization units.

Goal: *application data units should be* ***idempotent***.

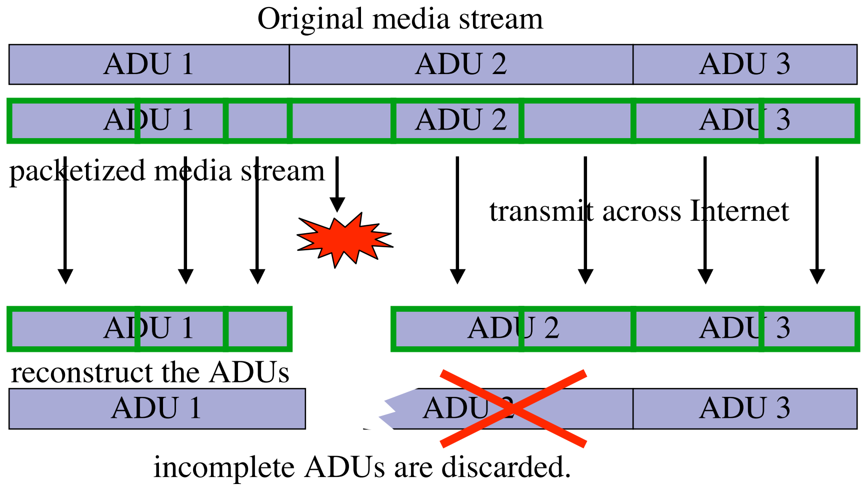
- Loss of an ADU should only affect the data carried by that ADU.

- **Problem 1**: what if an ADU is larger than a packet?
- **Problem 2**: cumulative codec predictor error.

Packetization



Packetize on ADU boundaries



Problem 1: Large ADUs

Large ADUs will need to span multiple packets, but loss of any of those packets will cause all the data in the ADU to be discarded.

- We can be smart about packetization.

- Even though we can't resync within an ADU, we can break ADUs at semantic boundaries that permit the ADU data before the lost packet to be used.
- We may be able to add a *small* amount of additional data to the packet header to add codec-specific resync points so some of the data after the lost packet is recoverable.

H.261 ADUs (RFC 2032)

- An H.261 video stream is composed of a sequence of images, which are themselves organized as a set of Groups of Blocks (GOB).
  - Each GOB holds a set of 3 lines of 11 macro blocks (MB).
  - Each MB carries information on a group of 16x16 pixels
- The H.261 Huffman encoding includes a special "GOB start" pattern, composed of 15 zeroes followed by a single 1, that cannot be imitated by any other code words.

A GOB is thus H.261's natural ADU - there is no smaller resynchronization unit in the H.261 bitstream.

- But a GOB can be up to 3KBytes in size.

Splitting H.261 on *Macroblock* boundaries

An additional H.261 RTP header is added to each packet:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|SBIT |EBIT |I|V| GOBN  |   MBAP  |  QUANT  |  HMVD   |  VMVD   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Some examples:

- **Start bit position** (`SBIT`): 3 bits

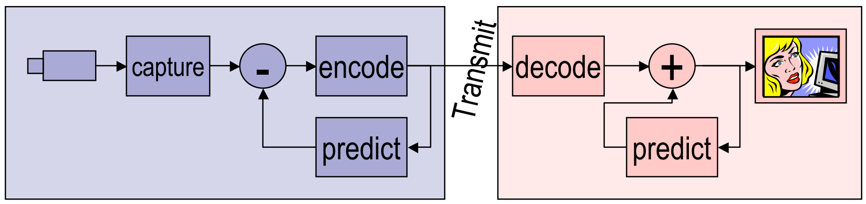
  Number of most significant bits that should be ignored in the first data octet.
- **GOB number** (`GOBN`): 4 bits

  Encodes the GOB number in effect at the start of the packet. Set to 0 if the packet begins with a GOB header.
- **Quantizer** (`QUANT`): 5 bits

  Quantizer value in effect prior to the start of this packet.

Problem 2: Predictor Error

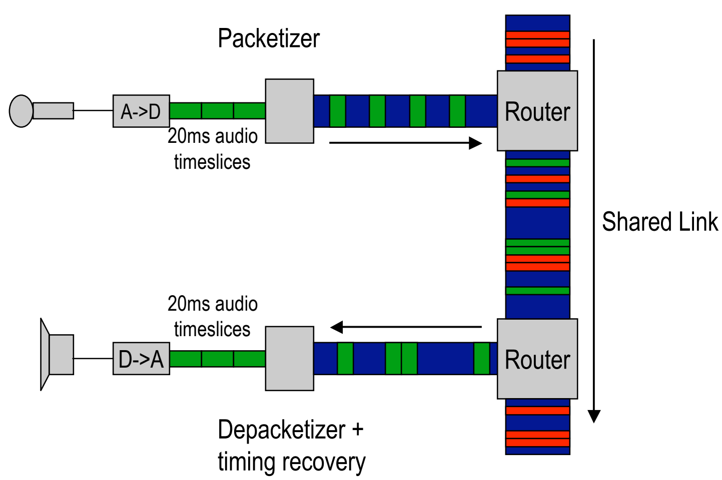
Codecs achieve high compression by removing redundancy.
A key technique is *prediction*:



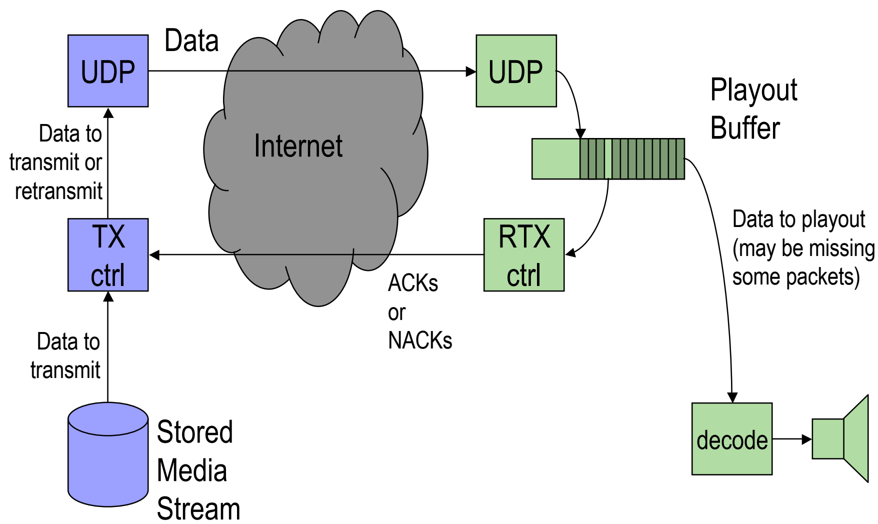
Packet loss causes the two predictors to get out of step.

- **Must design codec so errors aren't permanent!**

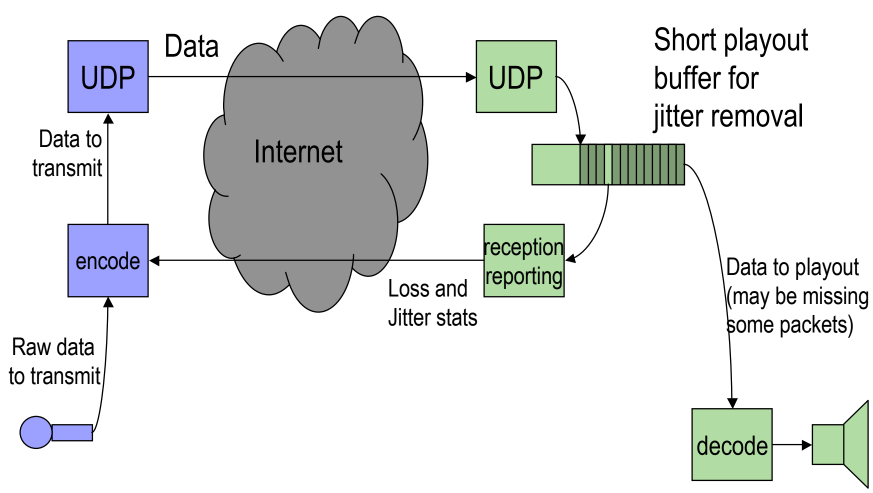
Jitter and Timing Recovery



Recap: Streaming Media with UDP

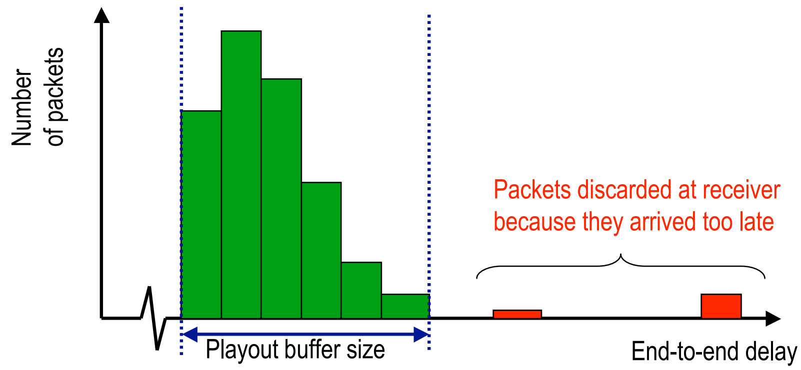


Interactive Media



Adaptive Playout Buffering

The goal is to delay playout at the receiver for *just long enough* to remove the jitter. Delaying any more than this adversely impacts interactivity.

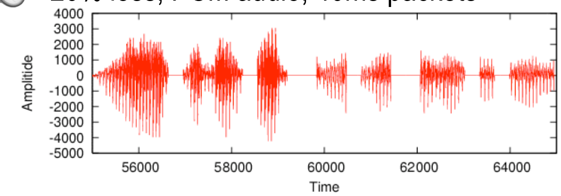


Loss Concealment

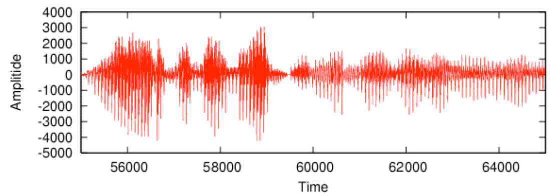
- Audio is moderately robust to loss.
  - Human brain interpolates across missing data.
  - Silence is treated as a "feature", so filling with silence is not optimal.
- Typical phoneme lengths range from 10-80ms.
  - Shorter packets help intelligibility in presence of loss, at the expense of header overhead.
  - Can also attempt to fill the gap with sounds from around the loss.

Loss Concealment

- 20% loss, PCM audio, 40ms packets



- 20% loss, PCM audio, 40ms packets, interpolating audio from the surrounding packets that did arrive.



References

D.D. Clark, D. L. Tennenhouse, "*Architectural Considerations for a New Generation of Protocols*", in Proceedings of SIGCOMM '90 (Philadelphia, PA, Sept. 1990), ACM.

RFC2032, *RTP Payload Format for H.261 Video Streams*, T. Turletti, C. Huitema.