envivio®

WHITEPAPER – IP Streaming of MPEG-4: Native RTP vs MPEG-2 Transport Stream

Authors: Alex MacAulay, Boris Felts, Yuval Fisher

October 2005

Contents

1. Summary

MPEG-4 AVC/AAC has been accepted as an audio-visual encoding format by a number of standards bodies and is poised for wide adoption in IP-based video services over consumer broadband connections (IPTV). In spite of the agreement on the encoding format, there are several competing specifications for the transport of MPEG-4 over IP networks. Because service providers have more experience with MPEG-2 as compared with pure IP-based delivery of services, some IPTV deployments have utilized MPEG-2 Transport Stream (TS) for the carriage of MPEG-4 data - even though this approach was not designed to make use of MPEG-4-specific encoding structures. TS-based transport of MPEG-4 comes in two flavors: TS over UDP/IP and TS over RTP/UDP/IP. In this paper we compare these transport mechanisms with the so-called “native RTP” transport of MPEG-4 – the direct carriage of MPEG-4 encoded data in RTP packets. We highlight the benefits of the native RTP approach, focusing on IPTV applications such as broadcast and video on demand (VOD).

2. Background

We begin with a brief overview of AVC/AAC and follow with a brief introduction to Internet Protocol, MPEG-2 TS and RTP, listing the features of each relevant to our comparison. In subsequent sections, we compare the carriage of MPEG-4 AVC/AAC data over these protocols.

2.1 AVC

The Advanced Video Codec (AVC) , also known as ITU H.264 and MPEG-4 part 10, is specified in [7]. Because AVC yields good video quality at bit rates currently achievable by ADSL and wireless connections, it has received attention from standards bodies and industry focused on broadband and wireless video services.

AVC is currently adopted by the following standards bodies:

- ITU and MPEG put together the H.264/MPEG-4 part 10 specification.
- DVD Forum and the Blue-ray Disk Association selected AVC as mandatory for their respective next-generation high definition DVD formats.
- DVB adopted AVC for use in both standard and high definition digital television, as well as in wireless transmission to handheld devices.
- 3GPP selected AVC as the primary codec for mobile video in its Release 6 specification.
- ISMA defines profiles and specifications for streaming AVC over IP networks.

AVC is also under consideration for adoption by 3GPP2 and ATSC.

The transport of AVC content is not uniform in these specifications. To aid in providing efficient and error resilient transport, the AVC specification defines a Network Abstraction Layer (NAL) that encapsulates the output of the encoder. NAL Units consist of video slices - independently-decodable groups of macro blocks with positioning, quantization and other data. NAL Units form the basic fragments of video that are transmitted to clients.

2.2 AAC

The Advanced Audio Codec was standardized by MPEG and is described in [8]. It is a high quality audio codec that significantly out-performs the well-known MP3 format, see for example [9]. It is currently used in Apple's iTunes software, as well as XM satellite radio, and is adopted by 3GPP, 3GPP2, the ISMA and DVB.

2.3 Internet Protocol

The internet protocol (IP) [5] is a packet-based-network transport protocol upon which the internet is built. IP packets are encapsulated in lower, hardware-level protocols for delivery over various networks (Ethernet, etc), and they encapsulate higher transport- and application-level protocols for streaming and other applications.

IP packets consist of a header and a payload. The header contains addressing and control information that allows a packet to be routed through packet-switching networks. The payload contains the data that is to be transmitted. In the case of streaming over IP networks, multiple protocols, such as RTP and UDP (described below), may be carried in the IP payload, each with its own header and payload that recursively carries another protocol packet. For example, Figure 1 shows video data carried in an RTP packet carried in a UDP packet carried in an IP packet. In this example, the total header information consists of 40 bytes and the final payload consists of 1125 bytes.

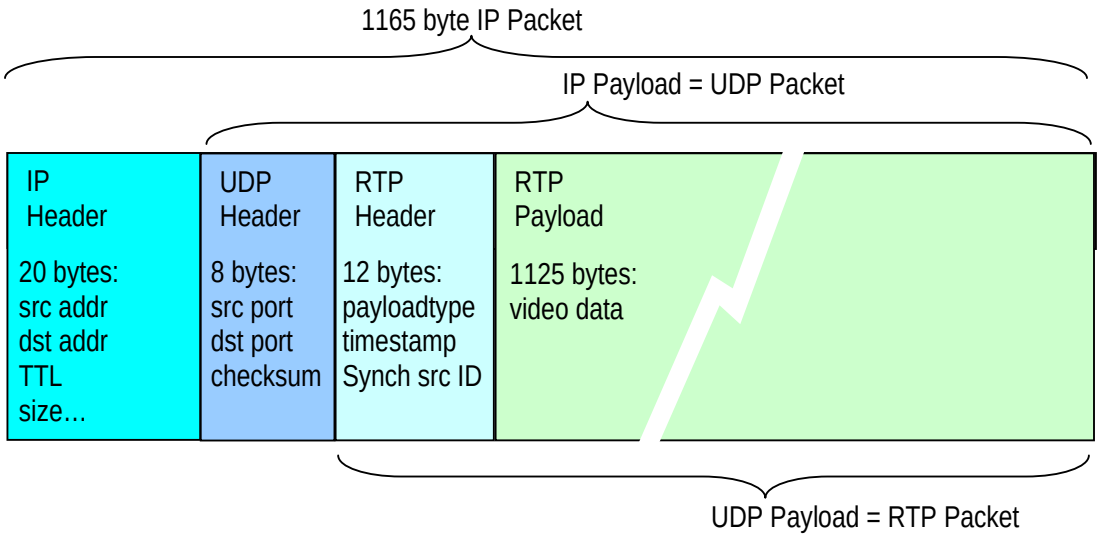

Figure 1. An IP packet encapsulating a UDP packet encapsulating an RTP packet carrying video payload.

2.4 MPEG-2 Transport Streams

MPEG-2 Transport Streams (described in [1]) are composed of 188 byte TS Packets, each with a 4 byte header. Some TS packets contain an optional Adaptation Field whose size depends on flags set in the packet header and which may contain timing information, pad bytes, and other data. TS packet payloads may contain program information as well as Packetized Elementary Streams (PES), typically video and audio streams. PES packets are broken into 184 byte chunks to fit into the TS packet payload. They have a variable-length byte header which must coincide with the start of a TS packet payload. It is thus necessary to pad a TS packet that carries the last chunk of a PES packet when there is insufficient PES data to fill it.

A Transport Stream contains multiplexed data, carrying TS packets with payloads from multiple PES packets – again, typically audio and video – as well as associated program information. See Figure 2. Because PES packet headers, as well as Adaptation Fields, contain timing information, no other signaling is necessary to synchronize multiple streams for playback.

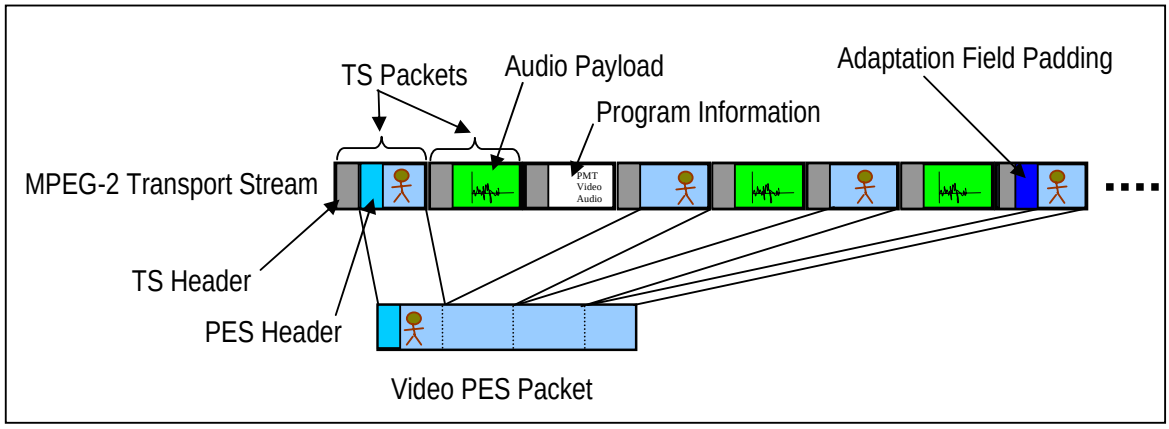

Figure 2. An MPEG-2 Transport Stream, multiplexing video, audio and program information.

2.5 Real Time Protocol (RTP)

The Real Time Protocol (RTP) was developed for carriage of real time data over IP networks. It is described in [3] and updated in [2]. RTP is a native internet protocol, designed for and fitting well in the general suite of IP protocols. Specifications exist for RTP embedded in RTSP and RTP over UDP; RTP is the implicitly recommended format when using the RTSP control plane [10]. Other non-MPEG media formats (such as AMR, H263, G.723, etc.) are supported by RTP and can be used in parallel with and synchronized to MPEG-4 audio and video streams over RTP.

RTP does not provide any multiplexing capability. Rather, each media stream is carried in a separate RTP stream and relies on underlying encapsulation, typically UDP (described in [4]), to provide multiplexing over an IP network. Because of this, there is no need for an explicit de-multiplexer on the client either. Each RTP stream must carry timing information that is used at the client side to synchronize streams when necessary.

RTP makes use of an associated protocol, the RTP Control Protocol (RTCP), which provides quality of service monitoring and information used to synchronize multiple RTP streams. RTCP is also described in [3].

2.6 MPEG-2 TS over IP

There are two methods currently utilized for the carriage of MPEG-2 TS over IP: The first simply selects a number of TS packets and carries them as the payload of the UDP datagram. Ethernet based networks have a Maximum Transmission Unit (MTU) of 1500 bytes, so this corresponds to 1500/188≈7 packets.

The second method specified by the IETF in [11] and by the DVB-IPI group in [12], uses RTP to carry MPEG-2 TS packets. In this case, the RTP payload again carries an integral number of TS packets, also 7 for Ethernet based networks. Interestingly, DVB specifically warns users not to make use of the direct UDP transport method, even though the RTP method has an extra level of packetization with some redundant information.

In both of these methods, sequential TS packets are carried in the payload without any specific knowledge about the content of the packets. This has implications for the consequences of packet loss, as discussed in Section 4.

2.7 Native RTP Carriage of AVC/AAC

Carriage of AVC video over RTP is defined in [13] and [14]. To increase error resiliency, NAL units are carefully mapped into the RTP payload.

Carriage of AAC audio over RTP is defined in [15] and [14]. For stereo encoded at 64kbps, AAC frames contain on average about 200 bytes [15], which means that 7 frames can be carried in one RTP payload (assuming an Ethernet based network).

As with AVC carriage, the ability to adapt the packetization specifically to the encoding format, for example by interleaving audio frames, leads to improved error resiliency.

3. Overhead

In this section we compare the packetization overhead of the various transport schemes. Table 1 summarizes the header size for each packetization scheme.

Packetization	Bytes in Header
IP	20
UDP	8
RTP	12
TS	4
PES	8, 13[1]

Table 1. Header size for various packetization methods.

[1] The PES header is variable sized. For audio or for video without B-frames, it is 8 bytes long; for PES packets holding video with B-frames, it is 13 bytes.

For 7 TS packets carried over UDP, there are 7*4 (TS) + 8 (UDP) + 20 (IP) = 56 bytes of header. For 7 TS packets carried over RTP/UDP there are 7*4 (TS) + 12 (RTP) + 8 (UDP) + 20 (IP) = 68 bytes. For TS, the overhead consists of not just the transport packet headers, but also the PES headers and the bytes used to pad TS packets that carry the end of the PES packet. However, these are hard to enumerate, since they depend on the size of the PES packet.

For native RTP, the header size is 12 (RTP) + 8 (UDP) + 20 (IP) = 40 bytes.

Thus, video using TS/UDP/IP requires 56 bytes, which is 40% more header, while video using TS/RTP/UDP/IP requires 68 bytes, which is 70% more header than native RTP, at 40 bytes. This calculation ignores the PES and adaptation field sizes, because these are not regularly distributed amongst the TS packets.

While the savings in header size for video are significant, they are still relatively small compared to the overall packet size. Seven TS packets form a payload of 7 x 184 = 1288 bytes, so the header represents just 56/1288 = 4.3% for TS/UDP/IP or 68/1288 = 5.2% of the payload for TS/RTP/UDP/IP. Overall, the 28 byte difference in header size between TS/RTP/UDP/IP and native RTP is just 2% of the payload. Nevertheless, the extra header bytes are a waste of bandwidth.

For audio carried over native RTP, where multiple frames are carried in the payload, there is an additional (approximately) 2 byte header per frame, as well as a fixed 2 byte header-length per packet. In this case the header length is (approximately) 2 (audio frame header-length) + 7*2 (audio frame header) + 12 (RTP) + 8 (UDP) + 20 (IP) = 56 bytes. For audio carried over TS, a 7 byte “ADTS” header needs to be prepended to each audio frame, giving approximately 7*7 = 49 extra bytes of header for a total of 105 or 117 bytes (for TS/UDP/IP and TS/RTP/UDP/IP respectively).

Thus, header size in the case of transmission of audio over native RTP is half the size of the header in the case of TS/UDP/IP or TS/RTP/UDP/IP. It represents 4.3% of the payload in RTP/UDP/IP, 9% in TS/RTP/UDP/IP and 8.1% in TS/UDP/IP.

Overhead	RTP/UDP/IP	TS/RTP/UDP/IP	TS/UDP/IP
Video	2%	5.2%	4.3%
Audio	4.3%	9%	8.1%

Table 2. Header size as a percentage of payload for video and audio for each of the transport methods.

For sake of simplicity, we ignored here some of the extra information necessary for broadcast:

- Clock information: transported in PCR in MPEG TS or over RTCP in the case of native RTP;
- Program information: transported in PAT or PMT tables in MPEG TS, or in SDP in the case of native RTP.

This information represents roughly the same size in both cases and contributes to the overhead in the same way.

4. Error Resilience in Broadcast

Probably the most important reason for avoiding TS/UDP/IP or TS/RTP/UDP/IP is the susceptibility of these transport mechanisms to errors resulting from packet loss. MPEG-4 AVC video provides numerous error resilience features, notably NAL units that allow independent decoding of packets in a video frame. The RTP packetization of AVC video allows intelligent mapping of video packets to RTP packets [13]. MPEG-2 Transport Stream makes minimal allowance for intelligent mapping of video packets to transport packets [18], but the video packets must either be smaller (implying some extra overhead for the resync marker and less efficient encoding due to smaller independent regions in the video), or risk cutting a video packet over two UDP packets if a video packet traverses several transport packets (and therefore increasing the damage caused by the loss of a single UDP packet).

For video transmission, the use of native RTP means that one lost packet will typically result in one lost NAL unit. However, in the TS case, the TS packets in the payload may belong to more than one PES packet, and thus to more than one NAL unit. This means that when a packet is lost, two NAL units may be lost.

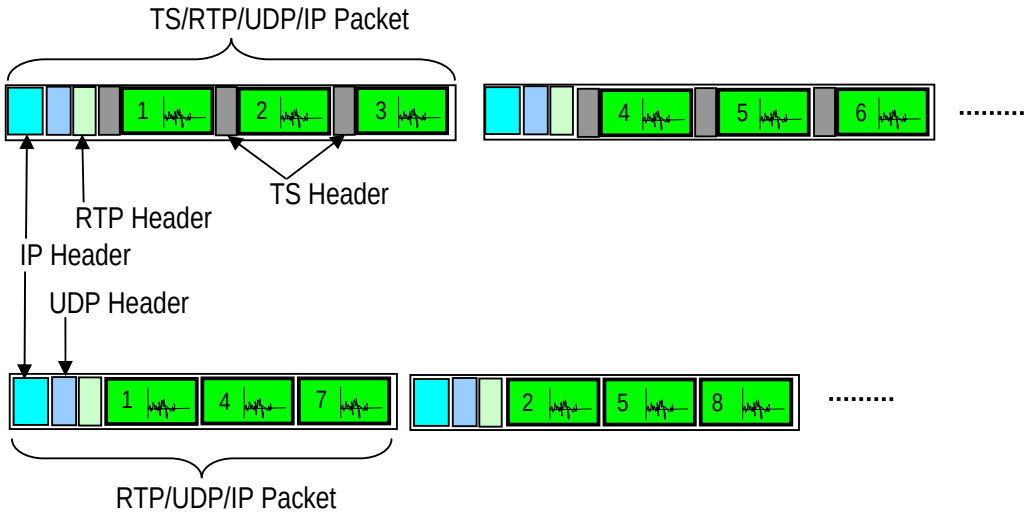

Figure 3. An example of interleaved audio in RTP packets, leading to improved error resiliency. If the first TS/RTP/UDP/IP packet is lost, audio samples 1,2 and 3 will be lost, leading to a long pause in the audio. If the first RTP/UDP/IP packet is lost, pakets 1, 4, and 7 are lost, and there will be short bits of silence spread over time.

In the audio case, the inability to interleave audio frames in TS packets means that packet loss is perceptually more disturbing. For example, in
, sequential RTP packets may carry every third audio frame, so that the loss of an RTP packet will result in smaller (and less perceptible) gaps in the audio, spread over a longer time, rather than a long gap corresponding to the loss of sequential audio frames.

5. VOD Services

In many applications, clients may require different sets of streams for the same content. Some example of this are:

- selection of an audio language for a video;
- closed captioning;
- subtitles in a particular language;
- alternative camera views

In native RTP it is possible to serve independent streams as required. Thus, for example, it is possible for a client to request a video stream and an associated audio stream in a specific language. In MPEG-2 TS, only two options are possible: send all the streams associated with the content, and let the client select which streams to render; or insist that the server select the appropriate PES packets, partition them, and multiplex them. The first option wastes bandwidth and may not be possible in many cases, for example when many languages are available. The second option imposes a computational burden at the server, increasing deployment costs.

Another service that is more complex using TS is trick-play, the ability to pause, and change play direction and speed of audio-visual streams. In native RTP streaming, hinting information at the server allows easy selection of video I-frames which are packetized and send to the client. In the MPEG-2 TS case, either the server must depacketize and repacketize the PES packets down to the level of the video frames on the fly, or repacketize/re-index the stream offline with several predefined speeds (e.g. x2, x4, x64 forward and backward). In any case the complexity of the operation is much higher.

6. Network

MPEG-2 TS is a unidirectional stream. There is no feedback to the server about packet loss, network jitter, and round trip latency. These statistics allow the server to detect client status, as well as adjust streaming parameters. For example, the server can reduce the video bit rate or drop video B-frames if too many packets are lost. This mechanism is even supported in a scalable multicast mode – all receivers broadcast quality feedback to the same multicast address. When coupled with RTP, the RTCP reports make this information available for RTP/UDP/IP. This is, of course, not available for TS/UDP/IP.

Another advantage of native RTP is the ability of the network to provide stream-based prioritized delivery. For example, when the network is congested, it is possible to deliver different RTP streams with different priorities and thus ensure an overall better client experience.

7. Misconceptions

In this section, we discuss several common questions about the use of MPEG-2 TS.

Do MPEG-2 TS-transported audio and video give better synchronization? Both RTP and PES headers have time stamp fields that are used by the decoder to determine when to display the decoded payload. In both cases, a stream can be synchronized with other streams or run in its own time base. Different Elementary Streams are synchronized by having them refer to the same clock reference. In MPEG-TS, the PCR (Program Clock

Reference) is carried within the transport stream and is used to synchronize streams with each other and synchronize the decoder clock with the encoder clock. In the native RTP case, the functionality of the PCR is fulfilled by an RTCP sender report that maps the time stamps in the RTP headers of different streams to wall-clock-time. Once the appropriate RTCP reports arrive at the client, any number of RTP streams may be synchronized. So in fact the mechanisms for MPEG-2 TS and native RTP synchronization are quite similar. TS multiplexes the synchronization information in the same stream, and RTP sends it in a different stream. In the RTP case, the multiplexing happens at the network level, rather than the stream level. In terms of reliability and effectiveness, both methods are identical. Packet loss or errors are not more or less likely in either scenario.

Native RTP requires the use of two ports per stream: one for the stream and another port for the RTCP reports. Does this represent a burden on the client? What about the server? The resource load on a client to maintain extra ports is inconsequential. The advantage of using just one port in the MPEG-2 TS case is irrelevant. On the server side, the same ports can be used or all clients, so there is no significant extra burden.

8. Conclusion

Native RTP brings several advantages over TS:

- Improved efficiency:
  - Native RTP has smaller packet header sizes for video and significantly smaller header sizes for audio carriage, leading to better bandwidth utilization.
  - Native RTP has fewer encapsulation packets to parse.
- Better error resilience:
  - Native RTP has support of numerous error resilience mechanisms that are well suited for transport over IP networks.
  - For both video and audio, native RTP causes less discernable artifacts due to packet loss.
- Improved networking:
  - Good integration with other internet protocols.
  - Reception quality feedback is standardized by RTCP; no such mechanism exists for MPEG-2 TS.
  - Flexibility to send the client just the streams the client needs.
  - Ability to give different streams different priorities and improve the client experience.
  - Native RTP gives highly optimized, network-based multiplexing and de-multiplexing. With MPEG-2 TS, there is a need for an explicit de-multiplexing step.
- Improved Services:
  - Native RTP allows additional streams containing user-specific or content-specific data to be streamed and synchronized.
  - Native RTP implementations have simpler trick-play implementation.

Given the multiple advantages of using native RTP, there is no technical reason to use TS packetization for the transport of MPEG-4 AVC/AAC data over IP.

9. References

An introduction to networking concepts can be found in [17]. The DVB specification [12] is a surprisingly good read, and the introduction to the MPEG-2 specification [1] is also digestible (though the going gets thick fast in later sections). An analysis of various ways of transporting MPEG-2 audio and video, concluding that simply using MPEG-2 Transport is not the most effective way of transporting MPEG-2 audio and video over IP can be found in [16]. The various RFCs referenced below can be found at http://www.faqs.org/rfcs/rfc**X**.html, where **X** is the RFC number.

[1] ISO/IEC 13818-1:2000 (ITU-T Recommendation H.222.0), "Generic coding of moving pictures and associated audio information: Systems", October 2000.

[2] Schulzrinne, H., Casner, S., Frederick, R., and V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications", STD 64, RFC 3550, July 2003.

[3] H. Schulzrinne, S. Casner, R. Frederick, V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications", RFC 1889, January 1996.

[4] J. Postel, "User Datagram Protocol", RFC 768, August 1980.

[5] J. Postel, "Internet Protocol", RFC 791, September 1981.

[6] Schulzrinne, H., Rao, A. and R. Lanphier, "Real Time Streaming Protocol (RTSP)", RFC 2326, April 1998.

[7] ISO/IEC International Standard 14496-10:2003.

[8] ISO/IEC International Standard 14496-3:1999.

[9] G. Stoll, F. Kozamernik "EBU listening tests on Internet audio codecs", http://www.ebu.ch/departments/technical/trev/trev_283-kozamernik.pdf

[10] Schulzrinne, H., Rao, A., and R. Lanphier, "Real Time Streaming Protocol (RTSP)", RFC 2326, April 1998.

[11] D. Hoffman, G. Fernando, V. Goyal, M. Civanlar, "RTP Payload Format for MPEG1/MPEG2 Video", RFC 2250, January 1998.

[12] DVB IP Phase 1 handbook , ETSI TS 102 034, "Digital Video Broadcasting (DVB); Transport of MPEG-2 Based DVB Services over IP Based Networks", March 2005.

[13] S. Wenger, M.M. Hannuksela, T. Stockhammer,M. Westerlund, D. Singer, "RTP Payload Format for H.264 Video", RFC 3984, February 2005.

[14] ISMA 2.0 Specification, see www.isma.tv.

[15] J. Van Der Meer, D. Mackie, V. Swaminathan, D. Singer, P. Gentric, "RTP Payload Payload Format for MPEG-4 Streams", IETF RFC 3640 , Nov 2003.

[16] Basso, A., Cash, G. L. and Civanlar, M. R., "Transmission of MPEG-2 Streams over non-guaranteed quality of service networks", Picture Coding Symposium (PCS-97), 10-12 Sept. 1997. [http://www.cs.columbia.edu/~hgs/papers/others/Bass9709_Transmission.ps.gz]

[17] BROADBAND IP NETWORKS AS BROADCAST CONTRIBUTION NETWORKS
by Pierre Clément,
http://www.broadcastpapers.com/broadband/ThalesBroadbandIPNetworks02.htm

[18] Amendment 3: Transport of AVC video data over ITU-T Rec H.222.0 |ISO/IEC 13818-1 streams, ISO/IEC 13818-1:2000/FDAM 3, July 2003.