Internet Engineering Task Force
INTERNET-DRAFT
draft-ietf-avt-qt-rtp-00

Audio-Video Transport WG
A. Jones, A. Periyannan, D. Singer
Apple Computer, Inc.

July 22, 1997
Expires: January 22, 1998

RTP Payload Format for QuickTime Media Streams

Status of This Memo

Abstract

This document specifies the payload format for encapsulating QuickTime media streams in the Realtime Transport Protocol (RTP). This specification is intended for QuickTime media/codec types that are not already handled by other RTP payload specifications. Each QuickTime media track within a movie is sent over a separate RTP session and synchronized using standard RTP techniques. A static QuickTime payload type (if assigned) or a dynamic payload type may be used. A QuickTime header within the RTP payload is defined to carry the media type and other media specific information. A packetization scheme is defined for the media data. This specification is intended for streaming stored QuickTime movies as well as live QuickTime content.

1 Introduction

This document specifies the payload format for encapsulating

QuickTime media streams in the Realtime Transport Protocol (RTP) [1]. RTP is a generic protocol designed to carry realtime media data along with synchronization information over a datagram protocol (mostly UDP over IP). The protocol itself does not address the encapsulation of specific media types, but instead leaves it to various profile specifications. An accompanying RTP profile document [2] contains various payload specifications to carry audio and video over RTP for conferencing applications and specifies the static payload types for various audio/video compression schemes. Other documents specify the encapsulation format used to carry specific compression schemes such as JPEG, MPEG and H.261 [3,4,5].

The QuickTime file format and architecture support an extensible set of media types and compression schemes. Many of these are not covered by the profile specifications available today. Hence, it is desirable to have an RTP encapsulation scheme that will handle all QuickTime media/codec types that are not covered by specific RTP payload types.

This specification proposes a scheme to carry QuickTime media/codec types over RTP. The scheme specified here handles all loss-tolerant media and a few loss-intolerant media such as text. Support for other loss-intolerant media such as MIDI and 3D will be added in future. This specification is intended for streaming stored QuickTime movies as well as live QuickTime content.

2 QuickTime Overview

QuickTime consists of a software architecture for multimedia authoring/playback and a movie file format to store multimedia presentations. These two aspects of QuickTime are independent of each other but are often combined when referring to QuickTime. It is possible to playback/author movies in other file formats such as AVI, AIFF, etc. using QuickTime software. Similarly it is possible to use QuickTime files independent of the software, for example, streaming movies over the Internet. The QuickTime movie file format is specified in [6]. More information on the QuickTime software architecture can be obtained from [7,8,9].

For the purpose of this document we will mostly be concerned with streaming QuickTime content using RTP. "QuickTime content" refers to content as specified in the QuickTime movie file format specification [6]. This does not preclude live QuickTime content. We merely use the file format specification as way to specify the format of the content.

QuickTime movie files contain the media data and synchronization information for the movie. A movie consists of multiple tracks, each of which contains a specific media type such as video, sound, MIDI,

text, etc. Not all media types are loss-tolerant. Table 2.1 lists the common QuickTime media types and whether they are loss-tolerant. The loss tolerant media can be carried over RTP/UDP in classic RTP-style. This will not however work for loss-intolerant data. RTP over TCP or using the Realtime Streaming Protocol (RTSP) [10] are some of the options for loss-intolerant media data. Another option is to achieve semi-reliability through redundant transmission. This specification uses this latter option to handle QuickTime "text" media over RTP.

QuickTime Media Type	Loss Tolerant?
Sound	Yes
Video	Yes
Timecode	No
Text	No
Music (MIDI)	No
MPEG	Yes
Sprite	No
Tween	No
3D	No

Table 2.1 QuickTime Media Types

QuickTime Timescales

QuickTime has a concept of timescales. A timescale defines the number of units of time that pass in every second of real time. Any time value has to be specified with respect to a timescale. A QuickTime movie has a timescale associated with it. Each of the tracks (medias) have a timescale associated with them. All of these timescales could be different. The RTP timestamp will be based on the timescale of the track associated with the RTP session. The timescale of a track never changes during its lifetime.

QuickTime Sample Descriptions

Every QuickTime media type has a sample description format associated with it. The sample description specifies how the sample is interpreted. For example, the video media sample description specifies the compression scheme, quality, bit depth and other such information. The sample description may change during the life of a track.

QuickTime Track Parameters

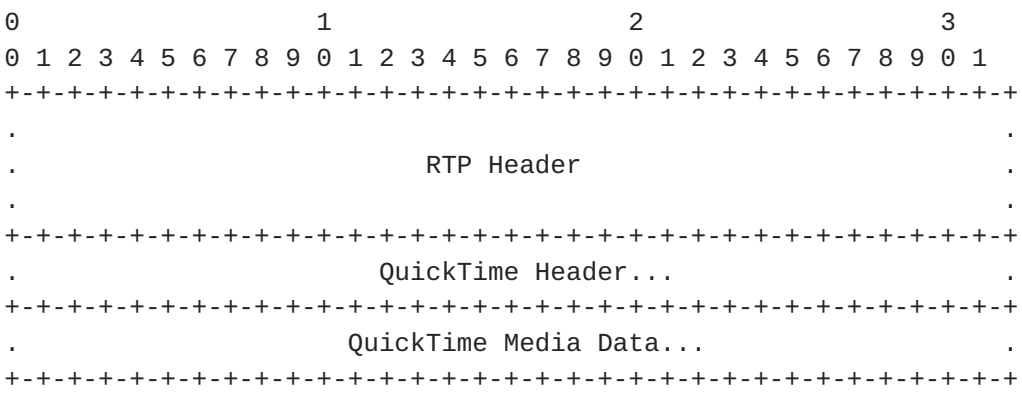
Every QuickTime track has a number of parameters associated with it such as height, width, transformation matrix, etc. These are as important to the presentation as the sample description.

3 RTP Encapsulation Format

The encapsulation scheme described here requires that each QuickTime media track within a single movie be sent over a separate RTP session and be synchronized using standard RTP techniques.

The QuickTime information is carried as payload data within the RTP protocol. There is a variable length QuickTime header immediately following the RTP header. The media data is packetized and placed in the RTP packet following the QuickTime header.

The RTP packet is formatted as follows:



3.1 RTP Header

The format and general usage of the RTP header fields are described in [1].

The following fields of the RTP header will be used as specified below:

- The payload type should specify the static QuickTime payload type (if assigned) or one of the dynamic payload types. (The need for a static payload type for QuickTime is up for discussion at the IETF AVT working group.) If a dynamic payload type is used, it should be agreed upon through some non-RTP means.

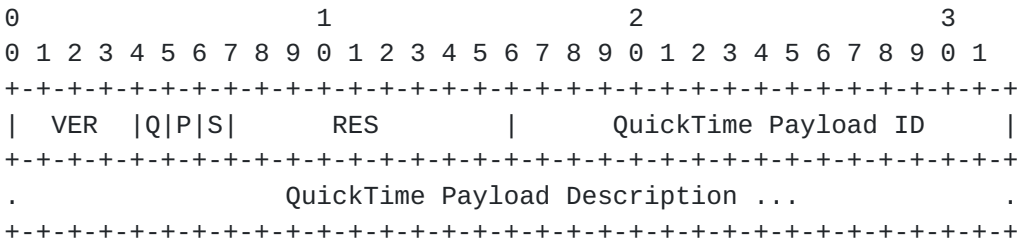
- The RTP timestamp is based on the timescale specified in the QuickTime header. The timestamp encodes the sampling instant of the first media sample contained in the RTP data packet. Multiple samples may be contained in one RTP packet or a single sample may require multiple RTP packets. The packetization rules are specified in a subsequent section. If a media sample occupies more than one packet, the timestamp will be the same on all of those packets. Packets

containing different samples must have different timestamps so that samples may be distinguished by the timestamp. The initial value of the timestamp is random (unpredictable) to make known-plaintext attacks on encryption more difficult, see RTP [1].

- The marker bit (M-bit) of the RTP header is set to one in the last packet of a sample and otherwise, must be zero. If one or more samples are fully contained within an RTP packet the M-bit must be set to one. Thus, it is possible to easily detect that a complete sample has been received and can be decoded and presented.

3.2 QuickTime Header

The QuickTime Header is defined as follows:



The fields in the QuickTime Header have the following meanings:

VER: 4 bits
A version field that must be set to zero by transmitters implementing this specification.

Q bit: 1 bit
The Q-bit is set to one if there is a payload description as part of this QuickTime header. Otherwise it is set to zero.

P bit: 1 bit
The P-bit is set to one if there are multiple samples which are of different sizes or durations carried in the payload. Otherwise it is set to zero. When the P-bit is set, two or more samples are placed in the QuickTime media data portion of the RTP packet along with individual timestamp and length information. The format for this packing is explained in a subsequent section. When the P-bit is not set, one or more samples or a partial sample is placed directly in the QuickTime media data portion of the RTP packet.

S bit: 1 bit
The S-bit is set to one if this packet contains data from a sync

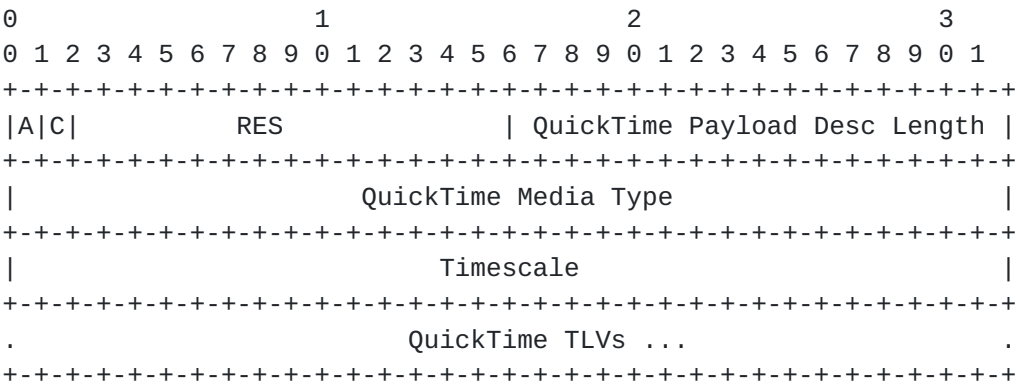
sample, i.e. key sample. Otherwise it is set to zero. When the packet contains more than one sample the S-bit is set to one if the first sample is a sync sample.

RES: 8 bits
Reserved for future use. Transmitter must set these bits to zero. Receivers must ignore these bits.

QuickTime Payload ID: 16 bits
A payload identifier that identifies the format of the QuickTime media data carried in this RTP session. The payload ID associates the QuickTime payload description (that is transmitted periodically) with the QuickTime media data. This identifier is changed every time the payload format changes. Payload IDs are explained further in a subsequent section.

QuickTime Payload Description: variable length
This field is present only if the Q-bit is set to one. It contains the QuickTime payload format description such as media type, timescale, sample size, compression information, etc. The header must be padded to a 32-bit boundary.

The QuickTime Payload Description is defined as follows:



The fields in the QuickTime Payload Description have the following meanings:

A bit: 1 bit
The A-bit is set to one if all samples are sync (key) samples for this payload description. Otherwise it is set to zero.

RES: 7 bits
Reserved for future use. Transmitter must set these bits to zero.

Receivers must ignore these bits.

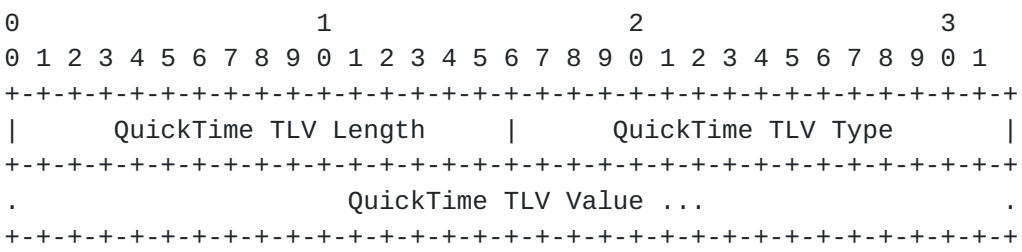
QuickTime Payload Description Length: 16 bits
Number of bytes in the QuickTime payload description (not including padding of 0 to 3 bytes). The QuickTime Media Data starts at the RTP data offset plus the QuickTime fixed header of 4 bytes plus the payload description length plus padding (of 0 to 3 bytes).

QuickTime Media Type: 32 bits
A 4 character media type that identifies the QuickTime media [6], example: 'vide' for video, 'soun' for sound, etc.

Timescale: 32 bits
The number of units of time that pass in 1 second of real time for this QuickTime payload ID. This is the timescale used by the RTP timestamp for this session.

QuickTime TLVs: variable length
One or more QuickTime parameters that describes this payload. The parameters are expressed as a Type-Length-Value triplet. The TLVs are not padded and can begin at any byte boundary.

A QuickTime TLV is formatted as follows:



The fields in a QuickTime TLV have the following meanings:

QuickTime TLV Length: 16 bits
Number of bytes in the QuickTime TLV. The next QuickTime TLV starts at the offset of the current TLV plus the current TLV length.

QuickTime TLV Type: 16 bits
A 2 character TLV type that identifies the QuickTime parameter.

QuickTime TLV Value: variable length
The value of the TLV as specified by the type. Values must be sent in network byte-order (i.e. big-endian format).

Note: Section 3.3 lists the set of currently defined TLVs. Some TLVs

are mandatory and must be present if the QuickTime Payload Description is being sent. Other TLVs will assume their default values if they are not sent. Any TLV not recognized by a receiver must be ignored and skipped over.

3.3 QuickTime TLVs

Sample Description (mandatory)
Type: 'sd'
Length: variable length
Media-specific QuickTime sample description. The format for this TLV for each of the currently defined media types can be found in [6] (starting pg. 59).
Default: none

QuickTime Atom
Type: 'qt'
Length: variable
Default: none
This TLV is used to transparently send a QuickTime Atom as defined in [6] (pg. 3). For example, this can be used to send User Data Atoms, Track Reference Atoms, Track Input Map Atoms, etc. The QuickTime atoms sent depends on the media type associated with the QuickTime payload description.

Track ID
Type: 'ti'
Length: 8
Default: 0
Track ID as defined in [6] (pg. 18).

Layer
Type: 'ly'
Length: 6
Default: 0
Layer as defined in [6] (pg. 18).

Volume
Type: 'vo'
Length: 6
Default: 255
Volume as defined in [6] (pg. 18).

Matrix
Type: 'mx'
Length: 40
Default: identity matrix

Matrix as defined in [6] (pg. 18 and 77).

Translation Matrix
Type: 'tr'
Length: 8
Default: identity matrix
h, v -- two 16-bit signed numbers indicating translation values (in pixels).This TLV is sent instead of the Matrix TLV when only translation is required.

Track Width
Type: 'tw'
Length: 8
Default: 0
Track Width as defined in [6] (pg. 19).

Track Height
Type: 'th'
Length: 8
Default: 0
Track Height as defined in [6] (pg. 19)

Language
Type: 'la'
Length: 6
Default: 0
Language as defined in [6] (pg. 32 and 75).

3.4 Media Data Packetization

The RTP packetization for QuickTime is designed to take into account the needs of a varied set of media types and compression schemes. Hence, 3 different packetization schemes are defined.

The following pieces of information are required at the transmission end to make packetization decisions:

- Maximum QuickTime Media Data size (MQD) that can be accommodated in a single RTP packet.
- Whether all samples for this media type are of constant size? (CQS)
- Whether all samples for this media type are of constant duration? (CQD)
- Sample size of all samples (when they are constant) (CSS).
- Sample size of a specific sample (SS).

Based on the above pieces of information, one of the following packetization schemes is adopted:

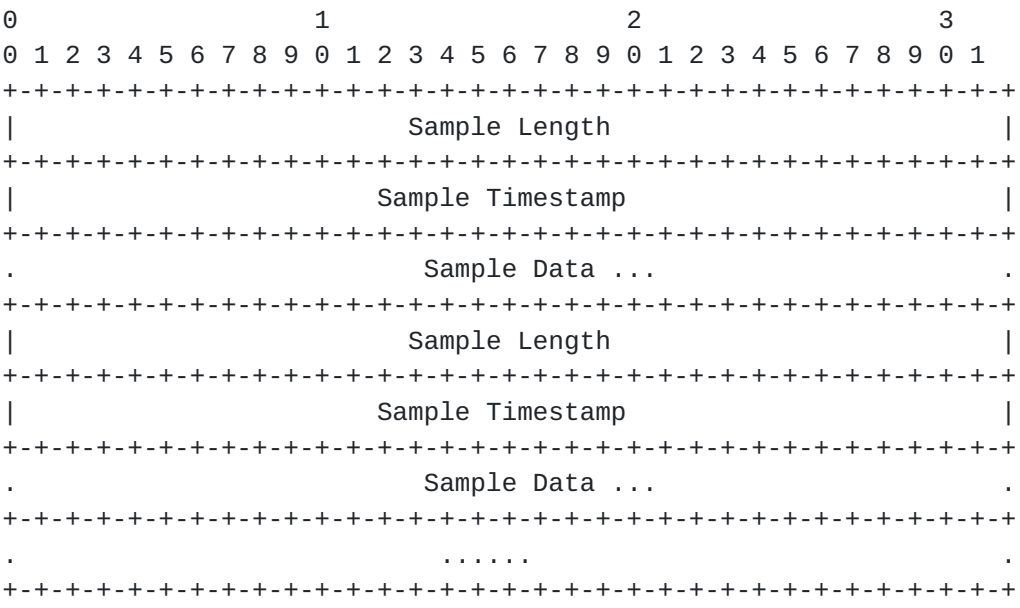
Scheme I : (CQS=true) AND (CQD=true) AND (CSS <= 0.5*MQD)

Multiple samples are packed into one RTP packet. The RTP header M-bit is set to one on all packets. The QuickTime header P-bit is set to zero on all packets.

Scheme II: ((CQS=false) OR (CQD=false)) AND (SS <= 0.5*MQD)

Multiple samples are packed into the QuickTime Media Data portion of an RTP packet. The RTP header M-bit is set to one in this packet. The QuickTime header P-bit is set to one.

The samples are packed using the format illustrated below:



The fields in the QuickTime Media Data have the following meanings:

Sample Length: 32 bits
Number of bytes in the sample data (not including the length field, timestamp field and padding).

Sample Timestamp: 32 bits
This field contains the timestamp for this sample in the timescale associated with this QuickTime payload ID. The first timestamp must be the same as the timestamp in the RTP header.

Sample Data: variable length
This field contains the sample data. The data must be padded to a

32-bit boundary.

All receivers are required to handle this scheme. A transmitter may choose to not implement this scheme in which case it will default to scheme III.

Note: This scheme leads to more efficient packing than scheme III for certain media/codec types. However, there is a trade-off between efficiency and losing multiple samples when a packet is lost.

Scheme III: Cases not covered by schemes I and II

A single sample is placed in one or more RTP packets. The RTP header M-bit is set to one in the last packet and is otherwise set to zero. The QuickTime header P-bit is set to zero in every packet.

The packetization boundaries may be chosen intelligently to respect the compression/decompression algorithm requirements. However, this is not a requirement. When intelligent boundaries are not chosen, a single packet loss will lead to the entire sample being lost in the case of multi-packet samples.

3.5 Payload Information

Payload ID

The QuickTime payload ID identifies the format of the QuickTime media data carried in an RTP session. It associates the QuickTime payload description (that is transmitted periodically) with the QuickTime media data. This identifier is an arbitrary 16-bit number that is changed every time the payload format changes. When streaming QuickTime movie tracks, the payload format changes usually when the sample description changes during the life of the track.

The following restrictions apply when picking payload IDs,

- The payload ID must be unique among all QuickTime RTP sessions originating from a given source canonical name. This is to ensure efficient mapping of payload IDs to payload descriptions using a single receiver-side table per canonical name.

- A payload ID must not be reused for a different payload description during the lifetime of the session. This allows receivers to cache the payload descriptions for the duration of the session.

Payload Description

The QuickTime payload descriptions are transmitted as part of the

QuickTime header. The payload descriptions specify the format of the QuickTime media data. The information for the specific fields in a payload description can be found in [6]. These fields do not include all of the information associated with a QuickTime track. For example, information on transformation matrices, layers, etc. is not included. This information needs to be communicated through non-RTP means.

Payload Description Transmission

The payload description must be transmitted in the first RTP packet which contains media samples that require the payload description. After the first packet, the payload description must be retransmitted at a periodic interval until the format of the media samples changes. The maximum retransmission interval should be 1 second, unless packets are being transmitted at less than 1 packet/second in which case the payload description must be transmitted with each packet.

The retransmission interval may be negotiated to an arbitrary value through non-RTP means. Note: This includes the case in which the payload descriptions are never sent over RTP, i.e. a retransmission interval of infinity. In this case the payload descriptions are communicated through some non-RTP means.

A transmitter may send an RTP packet that contains only a payload description and no QuickTime media data. This payload description must be cached by the receiver and used to interpret data that may arrive in the future.

3.6 Loss-intolerant Media Types

Loss-intolerant media types can not be easily handled within the standard RTP framework. Hence, we may need to use some non-RTP techniques to transmit these media types. However, some of the media types, notably Text and Tween media can be sent over RTP by the use of redundant transmissions. (Tween media is used to alter the characteristics of other media streams. For example, Tween samples may contain a series of values that change the volume of an audio stream.) The use of this technique is experimental.

Redundant Transmissions

The redundant transmission technique is one in which the RTP packet is retransmitted multiple times within the duration of the sample. The RTP packet is resent as a whole with the same RTP sequence number, timestamp and other information, i.e. it is an identical packet when seen on the wire. This technique is not bandwidth friendly when used with high bandwidth media types. Hence it will be

used only with the low bandwidth media types such as "text" and "tween" media.

The rationale for using the same RTP sequence numbers in the retransmitted packets is as follows: If the sequence numbers were incremented for each of the retransmitted packets we would require an additional field to identify the duplicate samples. In the proposed scheme, the receiver can discard duplicates by simply keeping track of the sequence numbers of the packets received.

The interval between retransmissions depends on the media type and the current congestion situation in the network. This interval can be a simple fixed interval, say 4 retransmissions equally spaced within the duration of the sample, or it could be more complex, say exponentially increasing intervals within the duration of the sample. This specification does not currently recommend a preferred scheme to use for determining the retransmission interval.

4 Open Issues

The following open issues need to be resolved:

- How to handle loss-intolerant media with "key" and "update" samples?
Loss-intolerant media samples can be retransmitted multiple times with fixed or variable intervals between transmission. The samples can be classified as key samples and update samples and handled appropriately. Update samples need not be periodically retransmitted. For example, in sprite media, key samples will contain the sprite image and update samples will contain the motion vectors. Whereas, in text media, all samples will be key samples.

- What is the appropriate interval between redundant transmissions for "text" and "tween" media samples?

- Should there be sample size TLV (that specifies bits per sample)?

Acknowledgments

The authors would like to thank Joe Pallas (of Apple ATG) and all the members of the QuickTime Streaming team, Jay Geagan, Andy Grignon, Sylvain Rouze and Kevin Gong for their valuable input in writing this proposal.

References

[1] H. Schulzrinne, et. al., "RTP : A Transport Protocol for Real-Time Applications", IETF RFC 1889, January 1996.

[2] H. Schulzrinne, et. al., "RTP Profile for Audio and Video Conference with Minimal Control", IETF RFC 1890, January 1996.

[3] L. Berc, et. al., "RTP Payload Format for JPEG-compressed Video", IETF RFC 2035, October 1996.

[4] D. Hoffman, et. al., "RTP Payload Format for MPEG1/MPEG2 Video", IETF RFC 2038, October 1996.

[5] T. Turletti, C. Huitema, "RTP Payload Format for H.261 Video Streams", IETF RFC 2032, October 1996.

[6] Apple Computer, Inc., "QuickTime File Format Specification", May 1996.

[7] Apple Computer, Inc., "Inside Macintosh: QuickTime", Addison Wesley Press.

[8] Apple Computer, Inc., "Inside Macintosh: QuickTime Components", Addison Wesley Press.

[9] Apple Computer, Inc., "QuickTime 2.5 Developer Guide", Developer Press.

[10] H. Schulzrinne, et. al., "Real Time Streaming Protocol", IETF Draft ietf-mmusic-rtsp-02.txt, March 24 1994, Expires: August 20 1997.

Authors' Contact Information

Alagu Periyannan
Email: alagu@apple.com
Tel: (408) 862 5387
Fax: (408) 974 0234

Anne Jones
Email: astoria@apple.com
Tel: (408) 862 1170

David Singer
Email: singer@apple.com
Tel: (408) 974 3162

Apple Computer, Inc.
One Infinite Loop, MS:302-3MT
Cupertino CA 95014
USA