H.264视频的RTP荷载格式

Status of This Memo

This document specifies an Internet standards track protocol for the Internet community, and requests discussion and suggestions for improvements. Please refer to the current edition of the "Internet Official Protocol Standards" (STD 1) for the standardization state and status of this protocol. Distribution of this memo is unlimited.

Abstract

This memo describes an RTP Payload format for the ITU-T Recommendation H.264 video codec and the technically identical ISO/IEC International Standard 14496-10 video codec. The RTP payload format allows for packetization of one or more Network Abstraction Layer Units (NALUs), produced by an H.264 video encoder, in each RTP payload. The payload format has wide applicability, as it supports applications from simple low bit-rate conversational usage, to Internet video streaming with interleaved transmission, to high bit-rate video-on-demand.

目录

1. 介绍

1.1. H.264 Codec

本文指定一个RTP荷载规范用于ITU-T H.264 视频编码标准（ISO/IEC 14496 Part 10 [2]）(两个都称为高级视频编码
AVC)。 H.264建议在2005年5月被ITU-T采纳，草案规范对于公共回顾可用[8]. 本文H.264 缩写用于codec和标准,但是
本文等价于采纳 ISO/IEC相似的编码标准.

H.264 视频
codec又非常广泛的应用覆盖所有格式的数字压缩视频格式,从低带宽的Internet流应用到HDTV广播和数字
影院应用。和当前的技术状态比较,
整个H.264的性能被报告节省50%的位率。例如，数字卫星TV质量被报告在1.5 Mbit/s,
就可以实现，而当前的MPEG 2的操作点在大约3.5 Mbit/s [9].

该codec规范自己概念上区分[1]视频编码层(VCL)和网络抽象层(NAL). VCL包含Codec的信令处理功能;以及如转换，量化，

运动补偿预测机制；以及循环过滤器。他遵从今天大多数视频codec的一般概念,基于宏快的编码器，使用基于运动补偿的
图像间预测和残余信号的转换编码。VCL编码器输出片断:
一个位串包含整数数目宏快的宏块数据，以及片断头信息(包含
片断内第一个宏快的空间地址，初始量化参数以及相似信息).
片断内的宏快按照扫描顺序安排，除非指定一个不同的宏块
分配,通过使用被称为灵活宏块顺序语法Flexible Macroblock Ordering syntax.图像内的预测只用于一个片断内部。更多
信息在[9]提供.

(NAL)编码器封装VCL编码器输出的片断到网络抽象层单元(NAL units),它适合于通过包网路传输或用于面向包的多路复用

环境。H.264的附录B定义封装过程传输这样的NAL单元通过面向字节流的网络。本文档范围，附录 B 不相关的。

NAL使用NAL单元. 一个NAL单元由一字节的头和荷载字节串组成。

头指示NAL单元的类型，是否有位错误或语法冲突在NAL

单元荷载中,以及对于解码过程该NAL单元相对重要性的信息。本RTP荷载规范被设计成不了解NAL单元荷载的位串。

H.264的一个主要特性是传输时间，解码时间，图像以及片断采样演示时间完全的解耦合。H.264中指定的解码过程是不知道
时间的,
并且H.264语法没有运送如跳过帧数目(在早期视频压缩标准，时间参考格式中是普遍的)信息.同时，有的NAL单元

影响许多图像，因此固有的是无时间性的。因为这样的原因，处理RTP时戳要求对于采样或演示时间没有定义或者在传输时间
不知道的NAL单元进行一些特殊的考虑。

1.2. 参数集概念

H.264一个非常基本的设计概念是产生自包含包,
使得如RFC2429的头重复或MPEG-4的头扩展编码（HEC）[11]机制变得不必要。

这是通过从媒体流解耦合不止一个片断的相对信息来实现的。高层meta信息应该可靠/异步的发送,事先不和包含片断包的RTP

包流发送。(对于没有通过带外传输信道发送本信息的应用，通过带内发送本信息也提供了手段)。高层参数的组合被称为参数集。

H.264规范包括两类参数集:顺序参数集和图像参数集。一个活动顺序参数集在一个编码视频序列中保持不变,一个活动图像参数集

在一个编码图像里保持不变。顺序和图像参数集结构包含如图像大小，采用的可选的编码模式，宏块到片断组映射等信息。

为了改变图像参数(如图像大小)而不用同步传送参数集修改给片断包流,编码器和解码器可以维护不止一个顺序和图像参数集的
列表。每个片断头包含一个码字指示使用的顺序和图像参数集。

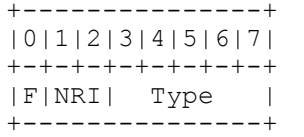
本机制允许从包流中解耦合参数集的传输,通过外部手段传输他们(即,作为能力交换的副作用),或通过一个(可靠或不可靠)控制协议
他们从没有被传送但是被应用设计规范修复甚至是可能的。

1.3. 网络抽象层单元类型

可以在[12], [13],[14]中找到关于NAL设计的学习信息.

所有NAL单元有一个单个NAL单元类型字节,他也作为本RTP荷载格式的荷载头.后面立即跟随NAL单元的荷载。

NAL单元类型字节的语法语义在[1]中指定,但是NAL单元类型的基本属性总结如下。NAL单元类型字节格式如下：



NAL单元类型字节部件的语义在H.264规范中制定，简要描述如下.

F: 1 bit
forbidden_zero_bit. H.264规范声明设置为1指示语法违例。

NRI: 2 bits
nal_ref_idc.
00值指示NAL单元的不用于帧间图像预测的重构参考图像。这样的NAL单元可以被丢弃而不用冒参考

图像完整性的风险。大于0的值指示NAL单元的解码要求维护参考图像的完整性。

Type: 5 bits
nal_unit_type. 本部件指定NAL单元荷载类型定义在[1]的表7-1中和本文后面。为了参考所有当前定义的NAL单元类型
和他们的语义，参考 [1]的7.4.1.

本文引入新的NAL单元类型，在5.2演示.
定义在本文的NAL单元类型在[1]中标记为未指定。但是,本规范扩展了F和 NRI的
语义，象5.3描述的那样.

2. Conventions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14, RFC 2119 [3].

This specification uses the notion of setting and clearing a bit when bit fields are handled. Setting a bit is the same as assigning that bit the value of 1 (On). Clearing a bit is the same as assigning that bit the value of 0 (Off).

3. Scope

This payload specification can only be used to carry the "naked" H.264 NAL unit stream over RTP, and not the bitstream format discussed in Annex B of H.264. Likely, the first applications of this specification will be in the conversational multimedia field, video telephony or video conferencing, but the payload format also covers other applications, such as Internet streaming and TV over IP.

4. 定义和缩写

4.1. 定义

本文档使用[1]中的定义. 为了方便以下定义在[1]中的词语总结出来：

access unit:
一组NAL单元总包括一个主要的编码图像。除了主要的编码图像,一个 access unit也可以包含

一个或多个冗余编码图像或其他的不包括片断或编码图像片断分区数据的NAL单元。access unit的解码总是
导致一个解码的图像。

coded video sequence: A sequence of access units that consists, in decoding order, of an instantaneous decoding refresh (IDR) access unit followed by zero or more non-IDR access units including all subsequent access units up to but not including any subsequent IDR access unit.

IDR access unit: An access unit in which the primary coded picture is an IDR picture.

IDR picture: A coded picture containing only slices with I or SI slice types that causes a "reset" in the decoding process. After the decoding of an IDR picture, all following coded pictures in decoding order can be decoded without inter prediction from any picture decoded prior to the IDR picture.

primary coded picture: The coded representation of a picture to be used by the decoding process for a bitstream conforming to H.264. The primary coded picture contains all macroblocks of the picture.

redundant coded picture: A coded representation of a picture or a part of a picture. The content of a redundant coded picture shall not be used by the decoding process for a bitstream conforming to H.264. The content of a redundant coded picture may be used by the decoding process for a bitstream that contains errors or losses.

VCL NAL unit: A collective term used to refer to coded slice and coded data partition NAL units.

In addition, the following definitions apply:

decoding order number (DON): A field in the payload structure, or a derived variable indicating NAL unit decoding order. Values of DON are in the range of 0 to 65535, inclusive. After reaching the maximum value, the value of DON wraps around to 0.

NAL unit decoding order: A NAL unit order that conforms to the constraints on NAL unit order given in section 7.4.1.2 in [1].

transmission order: The order of packets in ascending RTP sequence number order (in modulo arithmetic). Within an aggregation packet, the NAL unit transmission order is the same as the order of appearance of NAL units in the packet.

media aware network element (MANE): A network element, such as a middlebox or application layer gateway that is capable of parsing certain aspects of the RTP payload headers or the RTP payload and reacting to the contents.

Informative note: The concept of a MANE goes beyond normal routers or gateways in that a MANE has to be aware of the signaling (e.g., to learn about the payload type mappings of the media streams), and in that it has to be trusted when working with SRTP. The advantage of using MANEs is that they allow packets to be dropped according to the needs of the media coding. For example, if a MANE has to drop packets due to congestion on a certain link, it can identify those packets whose dropping has the smallest negative impact on the user experience and remove them in order to remove the congestion and/or keep the delay low.

缩写

DON: 解码顺序号
DONB: 解码顺序基
DOND: 解码顺序号差
FEC: 向前纠错

FU:	分片单元
IDR:	瞬间解码刷新
IEC:	国际电子委员会
ISO:	国际标准化组织
ITU-T:	国际电联-通信标准部门
MANE:	美提感知网络元素
MTAP:	多时刻聚合包
MTAP16:	16位时戳位移的MTAP
MTAP24:	24位时戳位移的MTAP
NAL:	网络抽象层
NALU:	NAL单元
SEI:	补充增强信息
STAP:	单时刻聚合包
STAP-A:	STAP类型A
STAP-B:	STAP类型B
TS:	时戳
VCL:	视频编码层

5. RTP 荷载格式

5.1. RTP头的使用

RTP 头的格式在RFC 3550
[4]中指定为了方便在图1又显示出来。本载荷格式使用头中域的方式和该规范一致。

当一个 NAL 单元封装在每个RTP包中,
推荐的RTP荷载格式在5.6节指定。对于聚合包/分片包的RTP荷载 (以及
一些rtp头域的设置) 在5.7和5.8节指定。

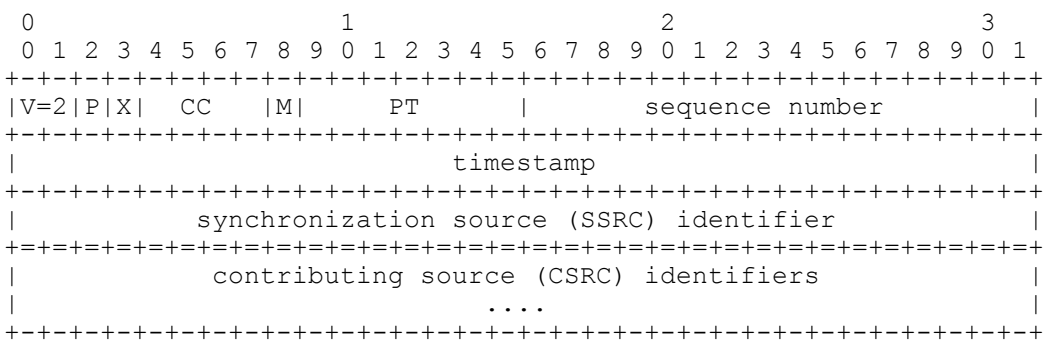


图 1. RTP 头。

根据RTP荷载格式设置的RTP头信息按如下设置:

Marker bit (M): 1 bit

对于RTP时戳指示的访问单元的最后一个包本位进行设置,符合视频格式M位的常规使用,以允许有效

缓冲处理布局。对于聚合包(STAP, MTAP),RTP头中的M位必须设置成最后一个NAL单元如果被传送在

单个RTP包中时M位对应的值。解码器可以使用本位作为早期最后一个包的指示,但是不可以依赖本
属性。

注: 运送多个NAL单元的聚合包只有一个M位相关联。因此,如果一个网关重新打包一个聚合包为几

个包，它可能不会可靠设置这些包的M位。

Payload type (PT): 7 bits

本新的包格式的荷载类型的值超过本文档的范围，在此不指明。荷载类型的赋值或者通过profile或者
通过动态方式。

Sequence number (SN): 16 bits

根据RFC 3550设置使用. 对于单个NALU与非交错打包方式,
序号用于对定NALU解码顺序。

Timestamp: 32 bits

RTP时戳设置为内容的采样时戳。必须使用90 kHz 时钟频率。

如果NAL单元没有他自己的时间属性(即,parameter set and SEI NAL units),RTP时戳设置成访问单元主编码图像
的RTP时戳,根据[1]的7.4.1.2节。

MTAPs时戳的设置在5.7.2定义.

接收者应该忽略包含在访问单元（只有一个显示时戳）的任何图像时间SEI消息，相反，接收者应该使用RTP时戳
同步显示过程。

RTP发送者你不应该传送图像时间 SEI消息对于不支持被显示成多个场的图像。

如果一个访问单元有多于一个显示时戳在图像时间SEI消息中,
SEI消息中的信息应该被对待成相对于RTP时戳的,
最早事件发生在RTP时戳给定的时间,
后续事件发生的时间由SEI消息中图像时间值差给定。假设tSEI1, tSEI2, ...,
tSEIn 为SEI消息中运送的显示时间戳, 其中tSEI1
是所有这样时间戳的最早值。tmadjst()是一个函数，他调整
SEI消息时间到90-kHz时间.TS是RTP时戳.则,和tSEI1关联的显示时间是TS.
和tSEIx[x=[2..n]]关联事件的显示时间为
TS + tmadjst (tSEIx - tSEI1).

注释：在一个3：2折叠的操作中需要显示编码的帧作为场,
在其中组成编码帧的电影内容使用隔行扫描显示。

图像定时SEI消息使得运送相同编码图像的多个时戳,因此3:2折叠过程正确控制。图像定时SEI消息机制是必须
的，因为在RTP时戳中只可以运送一个时戳。

注释:因为H.264允许解码顺序可以和显示顺序不同,
RTP时戳的值针对于RTP序号可以不是单调非减的。而且
RTCP报告中的抖动区间值可以不是网络性能问题的指示, as the calculation rules
for interarrival jitter (section 6.4.1 of RFC 3550) assume that the RTP timestamp of a packet is directly proportional to its transmission time.

5.2. RTP 荷载格式的公共结构

荷载格式定义三个不同的基本荷载结构。一个接收者可以识别荷载结构通过RTP荷载的第一个字节,

他也共享为RTP荷载头，某些情况下,作为荷载的第一个字节。本字节总是结构化为NAL单元头.
NAL单元类型指示目前使用那个结构. 可能的结构如下:

单个NAL单元包: 荷载中只包含一个NAL单元。NAL头类型域等于原始NAL单元类型,即在范围1到23之间. 5.6指定

聚合包:
本类型用于聚合多个NAL单元到单个RTP荷载中。本包有四种版本,单时间聚合包类型A (STAP-A), 单时间
聚合包类型B (STAP-B), 多时间聚合包类型(MTAP)16位位移(MTAP16), 多时间聚合包类型(MTAP)24位位移(MTAP24)。
赋予STAP-A, STAP-B, MTAP16, MTAP24的NAL单元类型号分别是 24, 25, 26, 27。见5.7.

分片单元:
用于分片单个NAL单元到多个RTP包。现存两个版本FU-A, FU-B,用NAL单元类型28, 29标识。见5.8.

Table 1. 单元类型以及荷载结构总结

Type	Packet	Type name	Section
0	undefined		-
1-23	NAL unit	Single NAL unit packet per H.264	5.6
24	STAP-A	Single-time aggregation packet	5.7.1
25	STAP-B	Single-time aggregation packet	5.7.1
26	MTAP16	Multi-time aggregation packet	5.7.2
27	MTAP24	Multi-time aggregation packet	5.7.2
28	FU-A	Fragmentation unit	5.8
29	FU-B	Fragmentation unit	5.8
30-31	undefined		-

注释:
他也共享为RTP荷载头，某些情况下,作为荷载的第一个字节。本字节总是NAL单元大小为65535字节。

5.3. NAL单元字节使用

NAL单元字节的结构语义在1.3节介绍。为了方便,NAL单元类型字节的格式在下面列出:

```
+---------------+
|0|1|2|3|4|5|6|7|
+-+-+-+-+-+-+-+-+
|F|NRI|  Type   |
+---------------+
```

本部分根据本规范指定F和NRI的语义。

F: 1 bit
forbidden_zero_bit. A value of 0 indicates that the NAL unit type octet and payload should not contain bit errors or other syntax violations. A value of 1 indicates that the NAL unit type octet and payload may contain bit errors or other syntax violations.

MANEs SHOULD set the F bit to indicate detected bit errors in the NAL unit. The H.264 specification requires that the F bit is equal to 0. When the F bit is set, the decoder is advised that

bit errors or any other syntax violations may be present in the payload or in the NAL unit type octet. The simplest decoder reaction to a NAL unit in which the F bit is equal to 1 is to discard such a NAL unit and to conceal the lost data in the discarded NAL unit.

NRI: 2 bits
nal_ref_idc.
0值和非零值的语义与H.264规范保持一致。换句话,00值指示NAL单元的内容不用于重建引用图像的

帧见图像预测。这样的NAL单元可以被丢弃而不用冒引用图像完整性的风险。大于00的值指示NAL单元的解码要求维护
引用图像的完整性。

除了上面指定的外，根据本RTP荷载规范，大于00的NRI值指示相对传输优先级，象编码器决定的一样。 MANE可以使用
本信息保护更重要的NAL单元。最高的传输优先级是11，依次是 10，01;00 最低。

注释：任何非零的NRI在H.264
解码器的处理是相同的。因此,接收者在传送NAL单元给解码器时不必操作NRI的值。

H.264编码器必须根据H.264规范设置NRI值(subclause 7.4.1)当nal_unit_type范围的是1到12. 特别是，H.264规范
要求对于nal_unit_type为6，9，10，11，12的NAL单元的NRI的值应该为0。

对于nal unit type等于7，8
(指示顺序参数集或图像参数集)的NAL单元,H.264编码器应该设置NRI为11
(二进制格式)

对于nal_unit_type等于5的主编码图像的编码片NAL单元(指示编码片属于一个IDR图像)，H.264编码器应设置NRI为11。

对于映射其他的nal unit types到NRI值,以下的例子可以使用并且在某些环境有效[13].其它的映射也可以，依赖于应用
以及使用的H.264/AVC Annex A profile.

注释：在某些profile中数据分区不可用，即 ，在Main或Baseline profiles. 因此，nal单元类型2，3,4 只出现在

视频流符合数据分区被允许的profile情况下，不会出现在符合MAIN/Baseline profile的流中。

Table 2. 编码片和主编码参考图像数据分区的编码片的NRI值的例子

NAL Unit Type	Content of NAL unit	NRI (binary)
1	non-IDR coded slice	10
2	Coded slice data partition A	10
3	Coded slice data partition B	01
4	Coded slice data partition C	01

注释：像以前提起的，非参考图像NRI值是00.

H.264编码器应该设置冗余编码参考图像的编码片和编码片分区NAL单元的NRI值为01(二进制格式).

对于NAL单元类型24~29的NRI的定义在本文5.7，5.8给出。

对于nal_unit_type范围在13到23的NAL单元的NRI值没有推荐的值,因为这些值保留给ITU-T, ISO/IEC.

对于nal_unit_type为0或30, 31的NAL单元的NRI值也没有推荐的值, 因为这些值的语义本文没有指定。

5.4. 打包方式

本文指定三种打包方式:

- o 单NAL单元方式
- o 非交错方式
- o 交错方式

单NAL单元方式目标是常规的系统, 该系统兼容ITU-T H.241 [15] (12.1).
非交错方式目标是常规系统, 可以不符合
ITU-T H.241建议.在非交错方式,
NAL单元按照NAL单元解码顺序传送。交错模式目标是不要求非常低端到端延迟的系统。
交错方式允许传送NAL单元不按照NAL单元解码顺序。

使用的打包方式可以通过OPTIONAL packetization-mode
MIME参数的值指定或外部手段。使用的打包方式控制那个NAL
单元类型在RTP荷载中允许。表3
总结对每个打包方式允许的NAL单元类型。有些NAL单元类型值(在表3中指示为没有定义)
保留为将来扩展.
那些类型的NAL单元不应该被发送者发送, 接受者必须忽略他们。例如:
1-23, 相关的包类型"NAL unit",允许出现在 "单NAL单元方式" 和"非交错方式",
不允许在"交错方式".
打包方式在第六节更详细解释。

表 3. 每个打包方式允许的NAL单元类型总结(yes = 允许, no = 不允许, ig = 忽略)

Type	Packet	Single NAL Unit Mode	Non-Interleaved Mode	Interleaved Mode
0	undefined	ig	ig	ig
1-23	NAL unit	yes	yes	no
24	STAP-A	no	yes	no
25	STAP-B	no	no	yes
26	MTAP16	no	no	yes
27	MTAP24	no	no	yes
28	FU-A	no	yes	yes
29	FU-B	no	no	yes
30-31	undefined	ig	ig	ig

5.5. 解码顺序号(DON)

在交错打包方式,
NAL单元的传输顺序允许和NAL单元的解码顺序不同。解码顺序号(DON)是荷载结构中的一个域
或一个获得变量指示NAL单元的解码顺序。
不按解码顺序传输的例子和原理以及DON的使用见13节。

传输和解码顺序的耦合由OPTIONAL sprop-interleaving-depth
MIME参数控制, 见下。当OPTIONAL sprop-interleaving-depth

MIME 参数的值等于0 (明确或缺省)
或者外部手段不允许传输NAL单元顺序不同于他们的解码顺序, NAL单元的
MIME 参数的值等于0 (明确或缺省) 或者外部手段不允许传输NAL单元顺序不同
MIME参数的值大于0或者传输NAL单元
与解码序号不一致通过外部手段被允许时,

- 在MTAP16/MTAP24中的NAL单元顺序不要求是NAL单元的解码顺序
- 在两个连续包中的STAP-B,

MTAP,FU解嵌套产生的NAL单元序号不要求是NAL单元解码序号。

用于单NAL单元包 STAP-A和FU-A的RTP荷载结构不包含DON.
STAP-B, FU-B结构包含DON, MTAP结构允许推导DON象5.7.2指定的一样.

注释:档FU-A出现在交错方式,后边总跟一个FU-B, 他设置自己的DON.

注释:
一个传输器想封装单个NAL单元每个包并且传输包不按照他们的解码顺序, 可以使用ST
AP-B包类型。

在单个NAL单元打包方式, NAL单元的传输顺序,由RTP顺序号确定,
必须和他们的NAL单元解码序号一致。
在非交错打包方式中,
在单NAL单元包,STAP-A,FU-A中NAL单元的传输顺序必须和他们的NAL单元解码顺序一致
.

在一个STAP中的NAL单元必须按照他们的NAL单元解码顺序出现。因此, 解码顺序首先
由STAP隐含顺序提供, 第二
通过RTP序号提供(对于STAPs, FUs, 单个NAL unit包之间的)。

对于运送在STAP-B,
MTAP以及FU-B开始的一些列分片单元中的NAL单元的DON值的信令在5.7.1, 5.7.2,
指定5.8。

传输顺序中的NAL单元的第一个DON值可以设置成任何值, DON值的范围是0到65535。到
达最大值后, DON的值回绕到0.

包含在STAP-B,
MTAP,或FU-B开始的一系列分片单元中的两个NAL单元的解码顺序按照如下确定:
DON(i)是索引为i传输顺序的解码顺序号. 函数don_diff(m,n)定义如下:

```
If DON(m) == DON(n), don_diff(m,n) = 0

If (DON(m) < DON(n) and DON(n) - DON(m) < 32768),
don_diff(m,n) = DON(n) - DON(m)

If (DON(m) > DON(n) and DON(m) - DON(n) >= 32768),
don_diff(m,n) = 65536 - DON(m) + DON(n)

If (DON(m) < DON(n) and DON(n) - DON(m) >= 32768),
don_diff(m,n) = - (DON(m) + 65536 - DON(n))

If (DON(m) > DON(n) and DON(m) - DON(n) < 32768),
don_diff(m,n) = - (DON(m) - DON(n))
```

don_diff(m,n)正值指示具有传输顺序n的NAL单元解码顺序跟在具有传输顺序m的NAL单
元后面。 don_diff(m,n)等于0

指示NAL单元解码顺序号可以按照任何NAL单元优先。don_diff(m,n)的负值指示索引为
n的NAL单元解码序号先于索引为
m的NAL单元。

DON相关域的值(DON, DONB, and DOND;
5.7)必须使得上面指定的DON的值确定的解码器顺序号符合NAL单元解码序号。

如果两个NAL解码单元顺序的NAL单元交换，新的顺序号不符合NAL单元解码顺序，NAL单元不可以有相同的DON值．如果

在一个NAL单元流中两个连续NAL单元的序号交换并且新的序号仍符合NAL单元解码顺序号，NAL解码单元可以有相同的
DON值。例如：当使用的视频编码profile允许任意分片顺序，
一个编码图像的所有编码片的NAL单元可以有相同的DON
值。因此，相同DON值的
NAL单元可以按照任何顺序解码,有不同DON值的NAL单元应该按照上面指定的顺序传递给解码器。
当两个连续的NAL单元解码顺序的NAL单元有不同的DON值，
第二个NAL单元的DON应该是第一个NAL单元的DON值加1。

解包过程恢复NAL单元解码的例子在第7部分给出。

注：
接收者不应该预测两个解码顺序号连续的NAL的DON值的绝对差等于1,甚至在没有错误的传输过程。
没有要求增加1,就像关联DON的值到NAL单元的时间一样，
不可能知道所有NAL单元是否分发给接收者。例如：
一个网关可以不转发非引用的编码的NAL片或SEI NAL
单元，当需要转发的网络带宽不足时。；另外的例子：

现场广播被预先编码的内容不时的打断，如广告。预先编码的第一个内帧图像事先传送使得接收端准备可用。

当传送第一个内帧时，发送者不能精确知道在解码顺序后的第一个内帧前，有多少NAL单元被编码。因此，预编码

片断的第一个内帧的DON值不得不估算，当他们传送时,因此DON中可能产生空隙。

5.6. 单个NAL单元包

定义在此的单个NAL单元包必须只包含一个类型定义在[1]中的NAL单元。这意味聚合包和分片单元不可以用在单个NAL

单元包中。一个封装单个NAL单元包到RTP的NAL单元流的RTP序号必须符合NAL单元的解码顺序。单个NAL单元包的结构
显示在图2。

注：NAL单元的第一字节和RTP荷载头第一个字节重合。

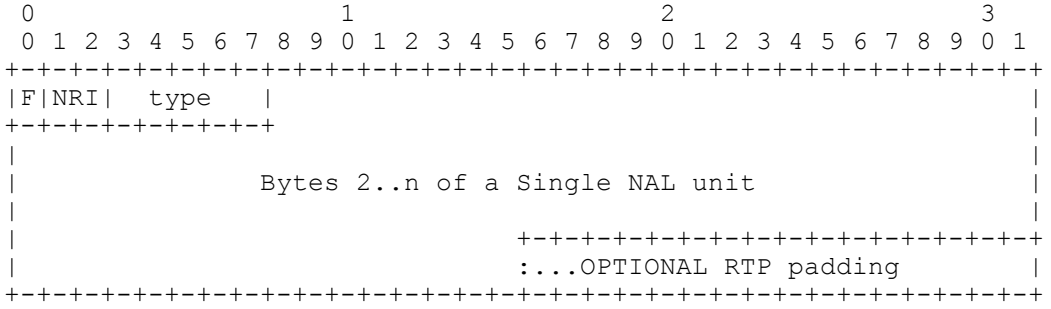


Figure 2. 单个NAL单元包的RTP荷载格式。

5.7. 聚合包

聚合包是本荷载规范的NAL单元聚合安排。本计划的引入是反映两个主要目标网络差异巨大的MTU:
有线IP网络(MTU 通常被以太网的MTU限制; 大约1500 字节),
基于无线通信系统的IP或非IP (ITU-T
5.
H.324/M)网络，它的优先传输最大单元是254或更少。为了阻止连个世界媒体的转换以及避免不必要的打包
负担，引入聚合单元安排。

本规范定义了两类聚合包:

o 单时间聚合包(STAP): 聚合相同NALU时间的NAL单元。两类STAP被定义,
一类不包括DON (STAP-A)另一类包括DON (STAP-B).

o 多时间聚合包(MTAP): 聚合具有差异NALU时间的NAL单元. 两个MTAP被定义,
差别在 NAL单元时戳位移长度不同。

词语NALU-时间被定义成如果NAL单元被传输他自己的RTP包中时RTP的时戳。

运送在一个聚合包中的每个NAL单元封装在一个聚合单元中。参见下面四个不同聚合单元和他们的特性。

聚合包的RTP荷载格式的结构见图3。

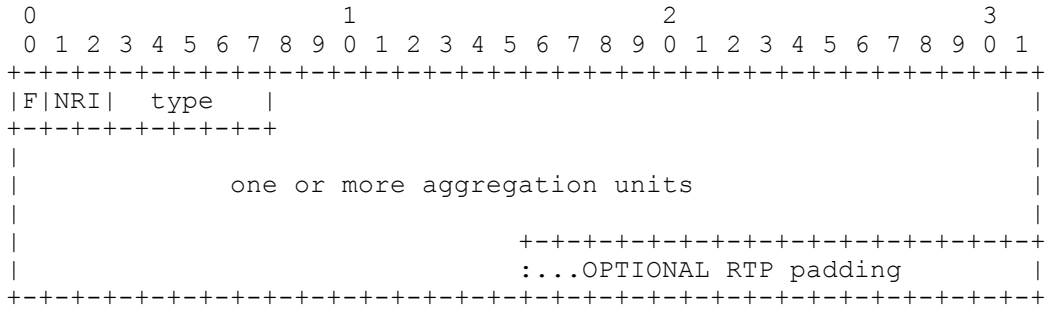


图 3. 聚合包的RTP荷载格式。

MTAPs, STAPs公用以下打包规则:RTP时戳必须设置为被聚合NAL单元中最早NALU时间。
NAL单元类型的类型域必须被设置成
适当的值,像表4描述的一样.
如果聚合NAL单元的F位是0, F位必须清除,否则, 则必须被设置。
NRI的值必须是运送在聚合包中NAL单元的最大值。

表 4. STAPs和MTAPs的类型域

Type	Packet	时戳位移域长度（位）	DON相关的域(DON, DONB, DOND）是否存在
24	STAP-A	0	no
25	STAP-B	0	yes
26	MTAP16	16	yes
27	MTAP24	24	yes

RTP头的marker位设置为聚合包中最后NAL单元如果单独封装在RTP传输中对应Marker位

的值。

聚合包的荷载由一个或多个聚合单元组成。见5.7.1，5.7.2四个不同类型的聚合单元。一个包聚合包可以运送必要多的
聚合单元；但是，
聚合包中整个数据显然必须适合于一个IP包,并且大小应该选择使得结果的IP包比MTU小。一个聚合包

不可以包含5.8中指定的分片单元。聚合包不可以嵌套;即，一个聚合包包含另一个聚合包。

5.7.1. 单时间聚合包

单时刻聚合包(STAP)应该用于当聚合在一起的NAL单元共享相同的NALU时刻。STAP-A荷载不包括DON，至少包含一个单时刻聚合单元
见图4. STAP-B荷载包含一个16位的无符号解码顺序号(DON)
(网络字节序)紧跟至少一个单时刻聚合单元。见图5.

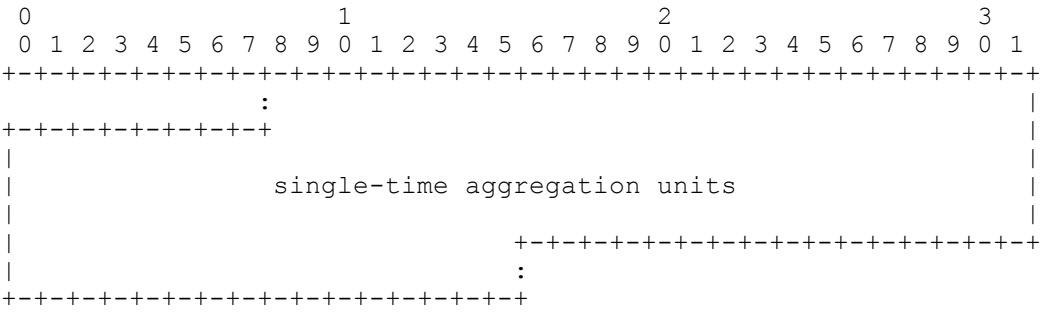


图 4. STAP-A荷载格式

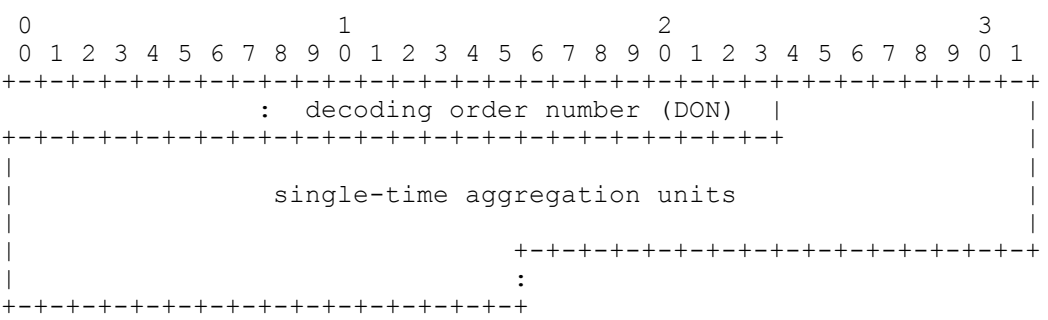


图 5. STAP-B 荷载格式

DON域指定STAP-B传输顺序中第一个NAL单元的DON值.
对每个后续出现在STAP-B中的NAL单元，它的DON值等于
(STAP-B中前一个NAL的DON值+1)%65535，%是取模运算。

单时刻聚合单元有一个16位无符号大小信息（网络字节序），他指示后续NAL单元的大小（以字节为单位）(不包括
这两个字节,但包括NAL单元类型字节),后面紧跟NAL单元本身,
包括它的NAL单元类型字节. 单时刻聚合单元在RTP荷载
中是字节对齐的,单可以不是32位字边界对齐。图6 表示单时刻聚合单元的结构。

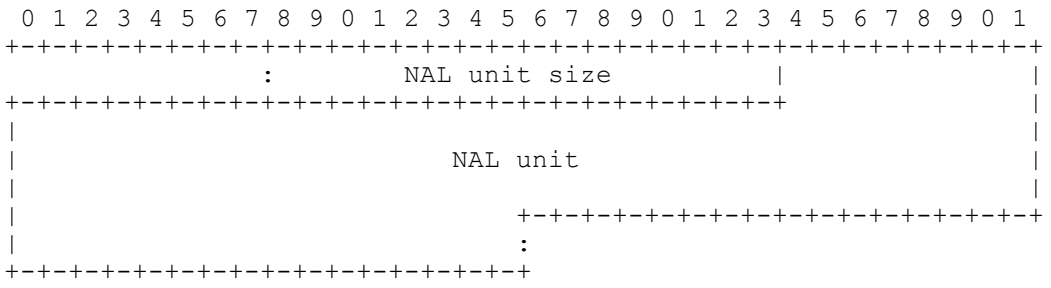


图 6. 单时刻聚合单元的结构

图 7表示一个例子--一个RTP包包含一个STAP-A. STAP包含两个单时刻聚合单元，在图中用1，2标记。

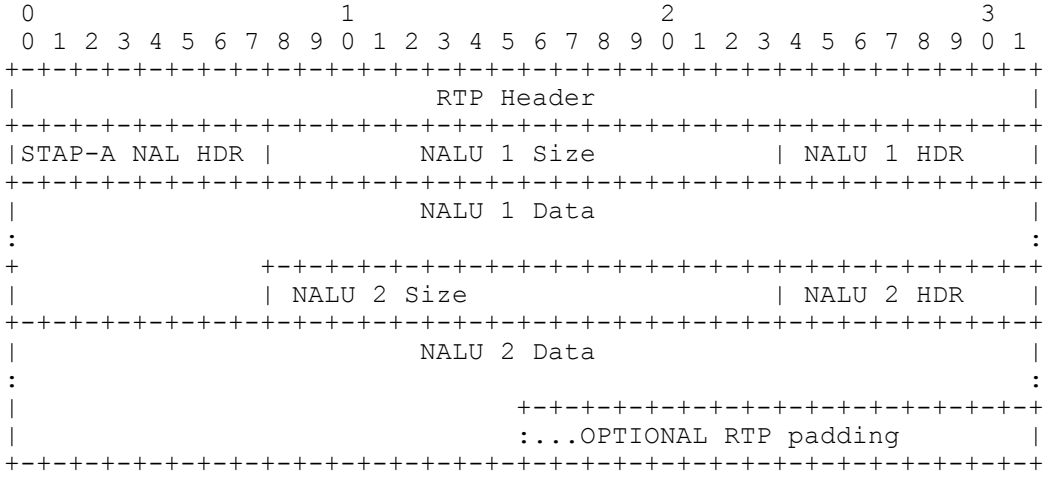


图 7. RTP包包含一个STAP-A. STAP包含两个单时刻聚合单元

图 8 表示一个RTP包包含一个STAP-B. STAP包含两个单时刻聚合单元，用1，2标记。

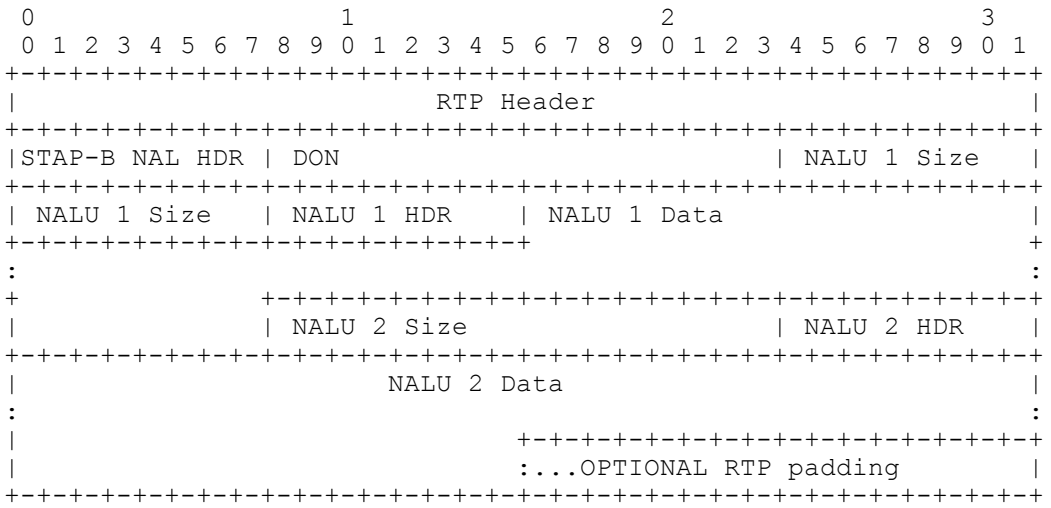


图 8. 一个RTP包包含一个STAP-B. STAP包含两个单时刻聚合单元例子

5.7.2. 多时刻聚合包(MTAPs)

多时刻聚合包的NAL单元荷载有16位的无符号解码顺序号基址(DONB)
(网络字节序）以及一个或多个多时刻聚合单元，如
图9表示。DONB 必须包含MTAP中NAL单元的第一个NAL的DON的值。

注释:NAL解码顺序中的第一个NAL单元不必要是封装在MTAP中的第一个NAL单元。

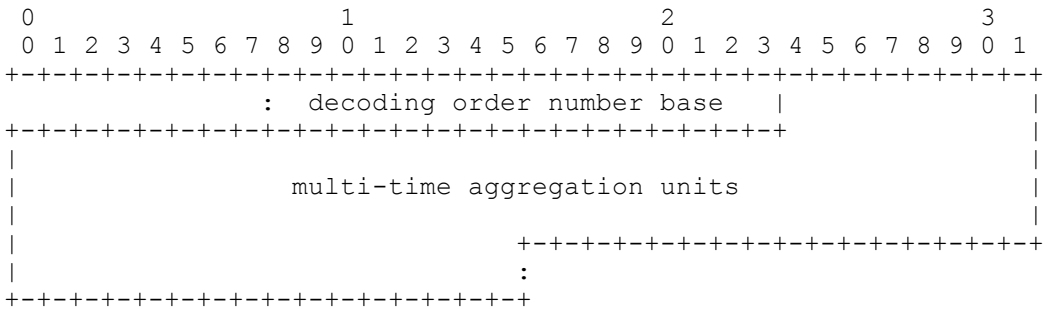


图 9. MTAP的NAL单元荷载格式

本规范定义两个不同多时刻聚合单元。两个都有16位的无符号大小信息用于后续NAL单元(网络字节序),一个8位无符号解码序号
差值(DOND), 和n位 (网络字节序) 时戳位移(TS
5.7.2. 多时刻聚合包(MTAPs)赜?6位的无符号解码顺序号基址(DONB) (网络字
/MTAP24): 时戳位移越大, MTAP的灵活性越大, 但是负担也越大。

MTAP16/MTAP24多时刻聚合单元的结构分别在图 10
, 11表示。一个包中的聚合单元的开始/结束不要求位于32位的边界。
跟随NAL单元的DON 等于(DONB + DOND) % 65536, %代表取摸操作.
本文没有指定MTAP内的NAL单元如何排序，但大多数
情况，应该使用NAL单元解码顺序。

时戳位移域必须设置成等于以下公式的值：如果NALU-time大于等于包的RTP时戳,则时戳位移等于(NALU-time - 包的RTP时戳).
如果NALU-time小于包的RTP时戳,则时戳位移等于 NALU-time + (2^32 -
包的RTP时戳).

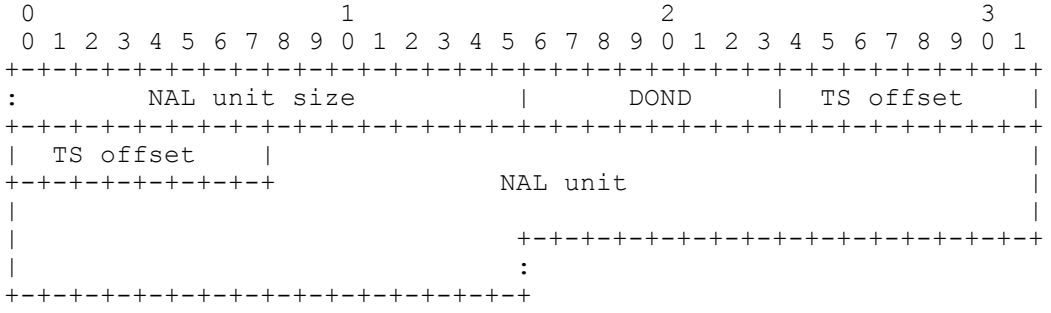
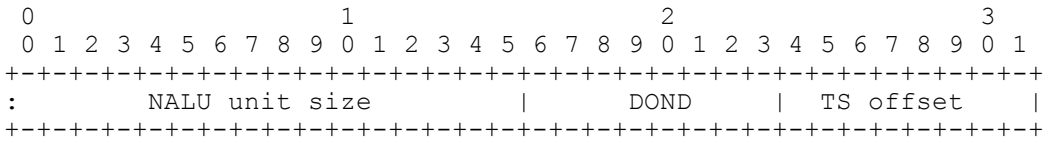


图 10. MTAP16多时刻聚合单元



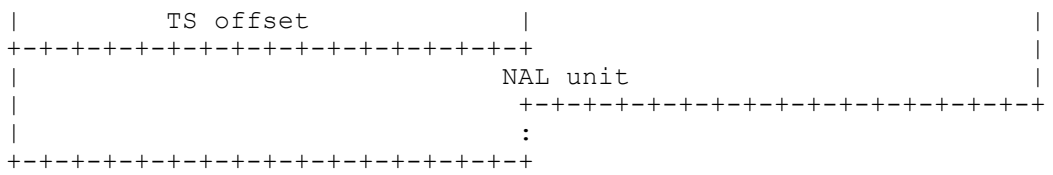


图 11. MTAP24多时刻聚合单元

一个MTAP中的最早的聚合单元时戳位移必须为0。因此，
MTAP的RTP时戳和最早NALU-time相同.

注释:
最早多时刻聚合单元是MTAP中所有聚合单元的扩展RTP时戳中的最小者，如果聚合单元封装在单个NAL单元包中。

扩展时戳是有多于32位的时戳，有能力计算时戳域的饶回,因此时戳如果绕回能够确定时戳的最小值。这样的"最早"聚合

单元可以不是封装在MTAP中的第一个聚合单元，最早NAL单元不必和NAL解码顺序的第一个NAL单元相同。

图 12
表示一个例子，一个RTP包包含一个多时刻MTAP16类型的聚合包，包括两个多时刻聚合单元，分别用1，2标记。

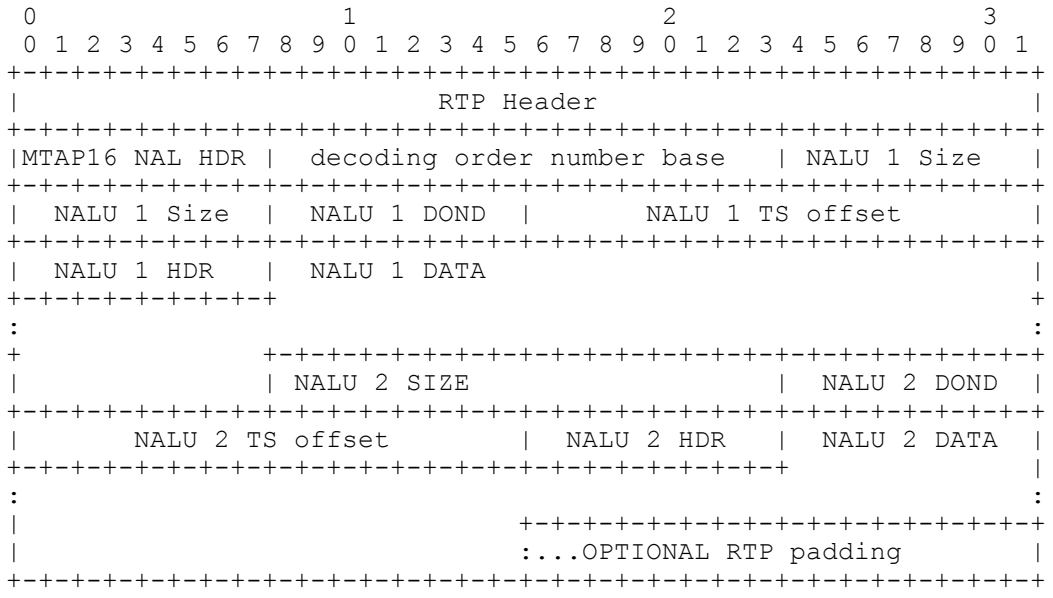
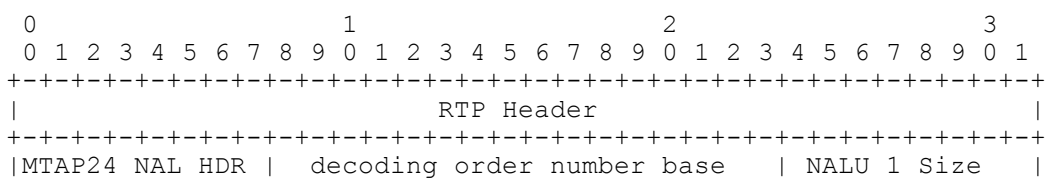


图 12.
一个RTP包包含一个多时刻MTAP16类型的聚合包，包括两个多时刻聚合单元

图 13
表示一个例子，一个RTP包包含一个多时刻MTAP24类型的聚合包，包括两个多时刻聚合单元，分别用1，2标记。



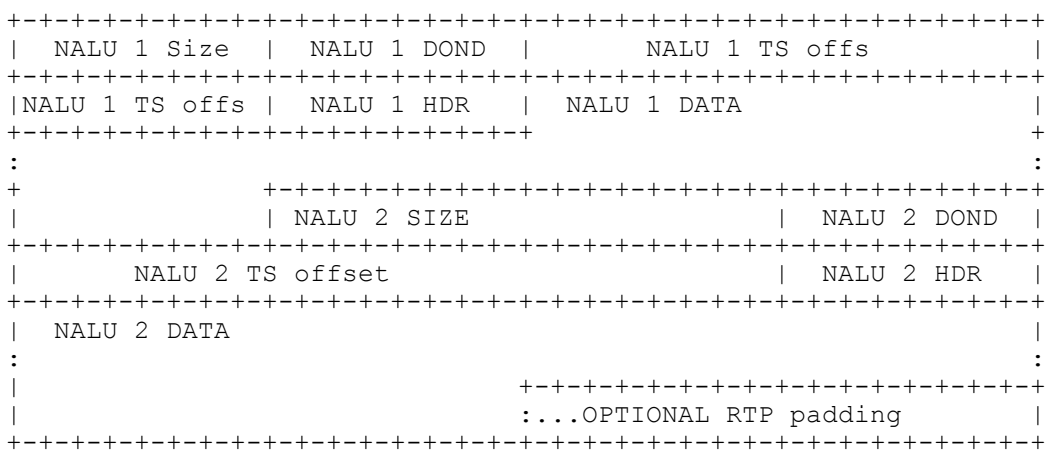


图 13. RTP包包含一个多时刻MTAP24类型的聚合包，包括两个多时刻聚合单元

5.8. 分片单元 (FUs)

本荷载类型允许分片一个NAL单元到几个RTP包中。在应用层这样做比依赖于底层（IP）的分片有以下好处：

o
荷载格式有能力传输NAL单元大于64K字节的单元通过IPv4网络，或许存在预编码的视频,特别在高清格式 (
每个图像的分片数目有限制，导致每个图像的NAL单元数目的限制,
从而导致大的 NAL单元).

- 分派机制允许分片单个图像并且采用一般向前的纠错像12.5描述的那样.

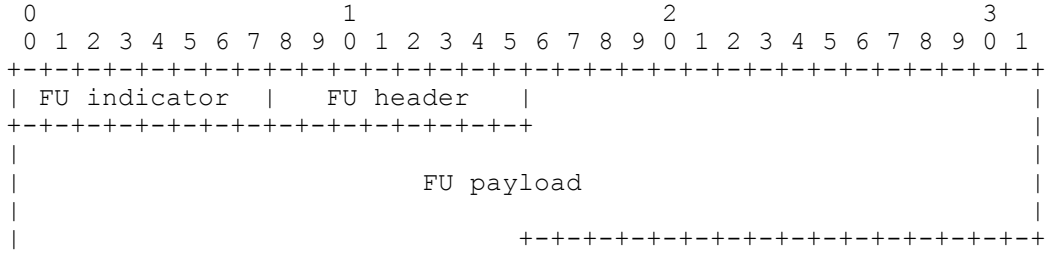
分片只定义于单个NAL单元不用于任何聚合包。NAL单元的一个分片由整数个连续NAL单元字节组成. 每个NAL单元字节

必须正好是该NAL单元一个分片的一部分。相同NAL单元的分片必须使用递增的RTP序号连续顺序发送(第一和最后分片之间
没有其他的RTP包）。相似，NAL单元必须按照RTP顺序号的顺序装配。

当一个NAL单元被分片运送在分片单元(FUs)中时，被引用为分片NAL单元。STAPs,MTAP不可以被分片。 FUs不可以嵌套。
即，一个FU 不可以包含另一个FU.

运送FU的RTP时戳被设置成分片NAL单元的NALU时刻.

图 14
表示FU-A的RTP荷载格式。FU-A由1字节的分片单元指示，1字节的分片单元头，和分片单元荷载组成。



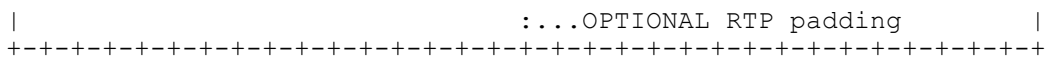


图 14. FU-A的RTP荷载格式

图 15 表示FU-B的RTP荷载格式.
FU-B由1字节的分片单元指示，1字节的分片单元头，和解码顺序号（DON）
以及分片单元荷载组成。

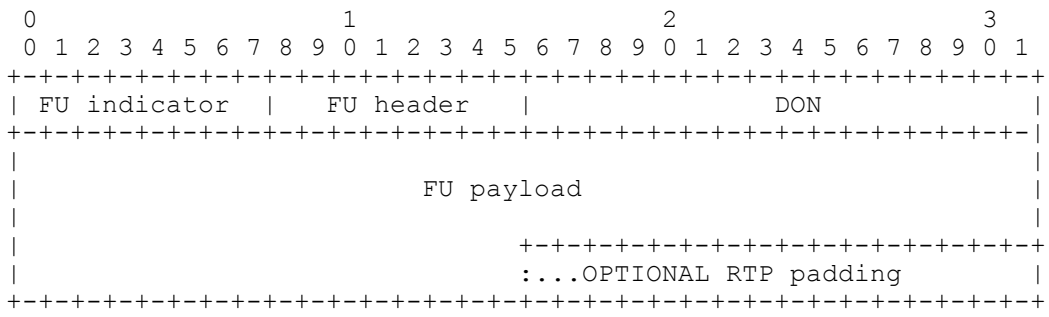
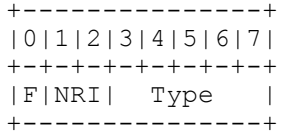


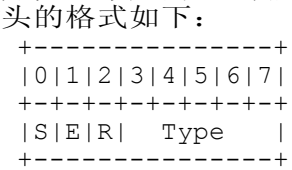
图 15. FU-B的RTP荷载格式

对于分片NAL单元的第一个分片如果用于交错打包方式，则必须使用NAL单元类型FU-B
。NAL单元类型FU-B MUST不可以
用于其他情况。换句话,
在交错打包方式,每个被分片的NALU，FU-B作为第一个分片,后面跟随的是一个或多个F
U-A分片.

FU指示字节有以下格式：



FU指示字节的类型域的28，29表示FU-A和FU-B。F的使用在5。3描述。NRI域的值必须
根据分片NAL单元的NRI域的值设置。
FU头的格式如下：



S: 1 bit

当设置成1,开始位指示分片NAL单元的开始。当跟随的FU荷载不是分片NAL单元荷载的
开始，开始位设为0。
E: 1 bit
当设置成1，结束位指示分片NAL单元的结束，即,
荷载的最后字节也是分片NAL单元的最后一个字节。当跟随的
FU荷载不是分片NAL单元的最后分片,结束位设置为0。
R: 1 bit
保留位必须设置为0，接收者必须忽略该位。

Type: 5 bits
NAL单元荷载类型定义在[1]的表7-1.

FU-B中DON的值的选择在5.5已经描述.

注:
FU-B中的DON域允许网关分片NAL单元到FU-B而不用组织进来的NAL单元到NAL单元解码顺序。

一个分片单元不可以传输在一个FU中; 即,
开始位和结束位不可以被同时设置在同一个FU头中。

FU荷载由分片NAL单元的荷载分片组成, 使得如果连续FU的分片单元荷载顺序连接,可以重构分片NAL单元的荷载。

NAL单元分片的类型字节不包括, 就像在分片单元荷载中一样,但是分片单元的NAL单元的类型信息运送FU指示字节
的F和NRI域以及FU头的类型域。一个FU荷载可以有任意字节也可以为空。

注释:
空的FUs允许减少某类发送者在几乎无丢失环境中的延迟。这些发送者特点是他们的NALU完全产生前, 可以打

包NALU分片,因此,在NALU大小未知之前。如果零长度分片不被允许,发送者不得不产生至少一位数据在当前分片被发送
前. 由于H.264的特性,
有时几个宏快占据0位, 这是不希望的并且增加延迟。但是,
(潜在)使用0长度的NALU应该仔细
权衡增加NALU丢失的风险, 因为增加了传输包。

如果一个分片单元丢失,接收者应该丢弃后续的所有分片单元对应于相同分片NAL单元的传输顺序的分片。

终端或MANE中的接收者可以聚合前一个NAL单元的n-1分片到一个(不完全的)
NAL单元,甚至分片n没有接收到. 这种情况下,
NAL单元的forbidden_zero_bit必须被设置成1指示语法违背.

6. 打包规则

打包方式在5.2节介绍. 对于多于一个打包方式的公共打包规则在6.1节指定.
单个NAL单元方式
的打包规则, 非交错方式, 交错方式的打包规则分别在6.2, 6.3,6.4节指定。

6.1. 公共打包规则

不管使用那种打包方式, 所有发送者必须遵守以下打包规则:

o
属于同一编码图像(共享相同RTP时戳值)的编码NAL单元片断或者编码数据分区NAL单元片断可以
按照定义在[1]中的应用Profile允许的任何顺序发送;
但是,对于延迟敏感的系统,他们应该按照

他们原始编码顺序发送, 以减少延迟。注意: 编码顺序不必要是扫描顺序, 而是NAL包对RTP协议
栈可用的顺序。

o 参数集根据8.4节给定的规则和建议处理。

o MANEs
不可以重复任何NAL单元, 除了顺序或图像参数集NAL单元,同样本文或者H.264规范也没有提供

手段识别重复的NAL单元。顺序和图像参数集NAL单元可以重复使得他们的纠错接收更

可靠，但是,任何

这样的重复不可以影响任何活动顺序或图像参数集的内容。重复应该在应用层进行，不应通过复制RTP
包进行（相同序号）。

使用非交错方式和交错方式的发送者必须遵守以下打包规则：

o
MANEs可以转换单个NAL单元包到一个聚合包,转换一个聚合包到几个单个NAL单元包,或在RTP转换器中混合
两个概念。RTP转换器至少应该考虑如下参数：路径MTU大小，
不平等的保护机制(即,根据RFC 2733通过

基于包的FEC,特别对于顺序和图像参数集NAL单元以及编码片断数据分区NAL单元），系统可以忍受的延迟
以及接收者缓冲能力。
注：RTP转换器要求按照每个RFC3550处理RTCP.

6.2. 单个NAL单元模式

本方式应用在OPTIONAL打包方式MIME参数值等于0,不包含打包方式,或者没有外部手段指示其他的打包方式的时候。
所有的接收者必须支持本方式。它主要用于低延迟应用（和使用ITU-T H.241建议兼容的系统）。(见12.1节).
只有单个NAL单元包可以用在这种方式。STAPs, MTAPs, and FUs
不可以使用。单个NAL单元的传输顺序必须和NAL
解码顺序一致。

6.3. 非交错方式

本方式应用在OPTIONAL打包方式MIME参数值等于1或者改方式被外部的手段打开时。本方式应该被支持。它主要用于
低延迟应用。本方式只允许单个NAL单元包, STAP-As, FU-As包。STAP-Bs, MTAPs,FU-Bs不可以使用。NAL单元的传输
顺序必须和NAL单元解码顺序一致。

6.4. 交错方式

可
本方式应用在OPTIONAL打包方式MIME参数值等于2或者改方式被外部的手段打开时。有些接收者可以支持本方式。
可以使用 STAP-Bs, MTAPs, FU-As,FU-Bs。STAP-As
和单个NAL单元包不可以使用。包和NAL单元传输顺序的限制
在5.5节指定。

7. 打包过程 (信息)

打包过程是实现相关的。因此,下面的描述应该被看成合适实现的例子。其他的方案也可以使用。相关描述算法的优化

也是可能的。7.1演示单个NAL单元和非交错打包方式的打包过程,7.2描述交错方式的打包过程。7.3 包括附加的封装
指导对于智能接收者。

所有相关于缓冲区管理正常的RTP机制也适用。特别的,重复的过期的RTP包(由RTP序号/时戳指示)被删除。 为了确定

精确的解码时间，如可能的延迟因素也被允许为了正确的流之间的同步。

7.1. 单个NAL单元和非交错方式

接收者包括一个接收缓冲区以补偿传输延迟和抖动。接收者存储进来的包按照接收顺序在接收缓冲区中。包被解封装

按照RTP序号的顺序。如果封装包是一个单个NAL单元包,包含在包中的NAL单元直接传递给解码器。如果解封装的包是
一个STAP-AI,
包含在包中的NAL单元按照他们在包中的封装顺序传递给解码器。如果解封装包是一个FU-A，所有的分
片NAL单元单分片连接在一起传递给解码器。

信息:
如果解码器支持任意分片顺序,编码的图像片可以按照任意顺序传送给解码器而不管他们的接收传送顺序。

7.2. 交错方式

这些打包规则后面的一般概念是重新排序NAL单元从传输顺序到NAL单元解码顺序。

接收者包括一个接收缓冲区以补偿传输延迟抖动以及重新排序包从传输顺序到NAL单元解码顺序。本部分,接收者操作

的描述假设没有传输延迟抖动。为了和实际的差异，一个接收缓冲区也用于补偿传输延迟抖动,接收者者本部分调用
解交错缓冲区。接收者应该准备传输延迟抖动;即,
或者保留单独的缓冲区用于传输延迟抖动缓冲和解交错缓冲或者
使用接收缓冲用于传输延迟抖动和解交错。而且,
接收者应该考虑传输延迟抖动在缓冲区操作时，即,在开始解码和
回放前增加缓冲区。

本部分组织如下: 7.2.1 描述如何计算交错缓冲区的大小.
7.2.2指定接收过程如何组织接收到的NAL单元到NAL解码顺序。

7.2.1. 解交错缓冲区的大小

当 SDP Offer/Answer 模型或其他任何能力交换过程被使用时,
接收流的属性应该使得接收者的能力不被超过。
在 SDP Offer/Answer 摸型行中,
接收者可以指示它的能力以分配一个解交错缓冲区使用deintbuf-cap MIME 参数。
发送者指示解交错缓冲区大小的要求使用sprop-deint-buf-req MIME参数.
因此，推荐设置解交错缓冲区大小（字节数目）
等于或大于sprop-deint-buf-req MIME 参数指定的值. 参见 8.1
得到更多信息关于 deint-buf-cap和sprop-deint-buf-req
MIME参数，8.2.2 关于他们在SDP Offer/Answer模型中的使用。

在会话建立中一个公布的会话描述被使用,sprop-deint-buf-req
MIME参数指定交错缓冲大小的要求。因此，推荐
设置解交错缓冲区大小（字节位单位）等于或大于sprop-deint-buf-req MIME
参数的值.

7.2.2. 解交错过程

在接收者中有两个缓冲状态:
初始缓冲和正在播放缓冲。初始缓冲发生在RTP会话被初始化时。初始缓冲后,解码和播放
开始了，使用缓冲-播放模型。

不管缓冲的状态,接收者存储进来的NAL单元按照接收顺序,在解交错缓冲区中。聚合包的 NAL单元存储在单个解交错缓冲区中
DON的值被计算为所有NAL单元存储。

描述在下面的接收操作需要以下的函数常数帮助:

o 函数AbsDON在8.1指定.

o 函数don_diff在 5.5 指定.

o 常数 N 是 OPTIONAL sprop-interleaving-depth MIME 类型参数的值(8.1)加1.

初始缓冲持续直到以下条件完成:

o 在解交错缓冲区中有 N VCL NAL单元。

o 如果sprop-max-don-diff存在, don_diff(m,n)大于sprop-max-don-diff的值, 其中 n 对应所有接收到
的NAL单元中最大AbsDON值的NAL单元, m
对应所有接收到的NAL单元中最小AbsDON值的NAL单元。

o 初始缓冲区已经持续时间等于或大于 OPTIONAL sprop-init-buf-time MIME 参数指定的值.

要从解交错缓冲区删除的NAL单元的确定如下:

o 如果解交错缓冲区包含至少N 个VCL
NAL单元,NAL单元被从解交错缓冲区移出传递给解码器按照下面指定
的次序直到缓冲区中包含N-1 VCL NAL 单元。

o 如果sprop-max-don-diff存在, 所有的NAL单元
m, 他们的don_diff(m,n)大于sprop-max-don-diff的从解交错
缓冲区移出传送给解码器按照下面指定的顺序。在此, n
不管缓冲的状态,接收者存储进来的NAL单元按照接

NAL单元传递给解码器的顺序指定如下:

o 让PDON是一个变量RTP会话开始时初始化为0。

o 对于每个关联DON的NAL单元,
按如下计算一个DON距离。如果NAL单元的DON大于PDON的值, DON距离等于DON-PDON.
否则DON距离等于 65535 - PDON + DON + 1.

o
NAL单元分发给解码器按照DON距离递增的顺序。如果几个NAL单元有相同的DON距离, 则他们可以按照任意顺序递交给解码器.

o 当一定数目的NAL单元传递给解码器,
PDON的值设置为传送给解码器最后一个NAL单元的DON值。

7.3. 附加打包规则

以下附加打包规则可用于实现一个可操作的H.264打包器:

o 智能RTP接收者 (即在网关中) 可以识别丢失的编码片断数据分区A (DPAs). 如果发现丢失的DPA,网关可以决定不发送

对应的编码片断数据分区B和C,因为对于H.264解码器他们的信息是无意义的。这样通

过丢弃无用的包而不用分析复杂
的位流，一个MANE可以减少网络负担。

o 智能RTP接收者(即在网关中）可以识别丢失的FU. 如果发现丢失一个FU, 网关可以决定不发送同一个分片NAL的后续FU

因为对于H.264解码器他们的信息是无意义的.这样通过丢弃无用的包而不用分析复杂的位流，一个MANE可以减少网络负担。

o
不得不丢弃包或NALU的智能接收者应该首先丢弃所有NAL单元类型中NRI值等于0的包/NALU. 这样最小化用户体验的影响并

保持参考图像完整。如果更多的包不得不被丢弃,则NRI值低的包应该在NRI值高的前面被丢弃。但是,丢弃任何NRI值大于
0的包可能导致解码器飘移应该被避免。

8. 荷载格式参数

This section specifies the parameters that MAY be used to select optional features of the payload format and certain features of the bitstream. The parameters are specified here as part of the MIME subtype registration for the ITU-T H.264 | ISO/IEC 14496-10 codec. A mapping of the parameters into the Session Description Protocol (SDP) [5] is also provided for applications that use SDP. Equivalent parameters could be defined elsewhere for use with control protocols that do not use MIME or SDP.

Some parameters provide a receiver with the properties of the stream that will be sent. The name of all these parameters starts with "sprop" for stream properties. Some of these "sprop" parameters are limited by other payload or codec configuration parameters. For example, the sprop-parameter-sets parameter is constrained by the profile-level-id parameter. The media sender selects all "sprop" parameters rather than the receiver. This uncommon characteristic of the "sprop" parameters may not be compatible with some signaling protocol concepts, in which case the use of these parameters SHOULD be avoided.

8.1. MIME Registration

The MIME subtype for the ITU-T H.264 | ISO/IEC 14496-10 codec is allocated from the IETF tree.

The receiver MUST ignore any unspecified parameter.

Media Type name: video

Media subtype name: H264

Required parameters: none

OPTIONAL parameters:

profile-level-id:

A base16 [6] (hexadecimal) representation of the following three bytes in the sequence parameter set NAL unit specified in [1]: 1) profile_idc, 2) a byte herein referred to as profile-iop, composed of the values of constraint_set0_flag, constraint_set1_flag, constraint_set2_flag, and reserved_zero_5bits in bit-significance order, starting from the most significant bit, and 3) level_idc. Note that reserved_zero_5bits is required to be equal to 0 in [1], but other values for it may be specified in the future by ITU-T or ISO/IEC.

If the profile-level-id parameter is used to indicate properties of a NAL unit stream, it indicates the profile and level that a decoder has to support in order to comply with [1] when it decodes the stream. The profile-iop byte indicates whether the NAL unit stream also obeys all constraints of the indicated profiles as follows. If bit 7 (the most significant bit), bit 6, or bit 5 of profile-iop is equal to 1, all constraints of the Baseline profile, the Main profile, or the Extended profile, respectively, are obeyed in the NAL unit stream.

If the profile-level-id parameter is used for capability exchange or session setup procedure, it indicates the profile that the codec supports and the highest level supported for the signaled profile. The profile-iop byte indicates whether the codec has additional limitations whereby only the common subset of the algorithmic features and limitations of the profiles signaled with the profile-iop byte and of the profile indicated by profile_idc is supported by the codec. For example, if a codec supports only the common subset of the coding tools of the Baseline profile and the Main profile at level 2.1 and below, the profile-level-id becomes 42E015, in which 42 stands for the Baseline profile, E0 indicates that only the common subset for all profiles is supported, and 15 indicates level 2.1.

Informative note: Capability exchange and session setup procedures should provide means to list the capabilities for each supported codec profile separately. For example, the one-of-N codec selection procedure of the SDP Offer/Answer model can be used (section 10.2 of [7]).

If no profile-level-id is present, the Baseline Profile without additional constraints at Level 1 MUST be implied.

max-mbps, max-fs, max-cpb, max-dpb, and max-br:
These parameters MAY be used to signal the capabilities of a receiver implementation. These parameters MUST NOT be used for any other purpose. The profile-level-id parameter MUST be present in the same receiver capability description that contains any of these parameters. The level conveyed in the value of the profile-level-id parameter MUST be such that the receiver is fully capable of supporting. max-mbps, max-fs, max-cpb, max-dpb, and max-br MAY be used to indicate capabilities of the receiver that extend the required capabilities of the signaled level, as specified below.

When more than one parameter from the set (max-mbps, max-fs, max-cpb, max-dpb, max-br) is present, the receiver MUST support all signaled capabilities simultaneously. For example, if both max-mbps and max-br are present, the signaled level with the extension of both the frame rate and bit rate is supported. That is, the receiver is able to decode NAL unit streams in which the macroblock processing rate is up to max-mbps (inclusive), the bit rate is up to max-br (inclusive), the coded picture buffer size is derived as specified in the semantics of the max-br parameter below, and other properties comply with the level specified in the value of the profile-level-id parameter.

A receiver MUST NOT signal values of max-mbps, max-fs, max-cpb, max-dpb, and max-br that meet the requirements of a higher level,

referred to as level A herein, compared to the level specified in the value of the profile-level-id parameter, if the receiver can support all the properties of level A.

Informative note: When the OPTIONAL MIME type parameters are used to signal the properties of a NAL unit stream, max-mbps, max-fs, max-cpb, max-dpb, and max-br are not present, and the value of profile-level-id must always be such that the NAL unit stream complies fully with the specified profile and level.

max-mbps: The value of max-mbps is an integer indicating the maximum macroblock processing rate in units of macroblocks per second. The max-mbps parameter signals that the receiver is capable of decoding video at a higher rate than is required by the signaled level conveyed in the value of the profile-level-id parameter. When max-mbps is signaled, the receiver MUST be able to decode NAL unit streams that conform to the signaled level, with the exception that the MaxMBPS value in Table A-1 of [1] for the signaled level is replaced with the value of max-mbps. The value of max-mbps MUST be greater than or equal to the value of MaxMBPS for the level given in Table A-1 of [1]. Senders MAY use this knowledge to send pictures of a given size at a higher picture rate than is indicated in the signaled level.

max-fs: The value of max-fs is an integer indicating the maximum frame size in units of macroblocks. The max-fs parameter signals that the receiver is capable of decoding larger picture sizes than are required by the signaled level conveyed in the value of the profile-level-id parameter. When max-fs is signaled, the receiver MUST be able to decode NAL unit streams that conform to the signaled level, with the exception that the MaxFS value in Table A-1 of [1] for the signaled level is replaced with the value of max-fs. The value of max-fs MUST be greater than or equal to the value of MaxFS for the level given in Table A-1 of [1]. Senders MAY use this knowledge to send larger pictures at a

proportionally lower frame rate than is indicated in the signaled level.

max-cpb The value of max-cpb is an integer indicating the maximum coded picture buffer size in units of 1000 bits for the VCL HRD parameters (see A.3.1 item i of [1]) and in units of 1200 bits for the NAL HRD parameters (see A.3.1 item j of [1]). The max-cpb parameter signals that the receiver has more memory than the minimum amount of coded picture buffer memory required by the signaled level conveyed in the value of the profile-level-id parameter. When max-cpb is signaled, the receiver MUST be able to decode NAL unit streams that conform to the signaled level, with the exception that the MaxCPB value in Table A-1 of [1] for the signaled level is replaced with the value of max-cpb. The value of max-cpb MUST be greater than or equal to the value of MaxCPB for the level given in Table A-1 of [1]. Senders MAY use this knowledge to construct coded video streams with greater variation of bit rate than can be achieved with the MaxCPB value in Table A-1 of [1].

Informative note: The coded picture buffer is used in the hypothetical reference decoder (Annex C) of H.264. The use of the hypothetical reference decoder is recommended in H.264 encoders to verify that the produced bitstream conforms to the standard and to control the output bitrate. Thus, the coded picture buffer is conceptually independent of any other potential buffers in the receiver, including de-interleaving and de-jitter buffers. The coded picture buffer need not be implemented in decoders as specified in Annex C of H.264, but rather standard-compliant decoders can have any buffering arrangements provided that they can decode standard-compliant bitstreams. Thus, in practice, the input buffer for video decoder can be integrated with de-interleaving and de-jitter buffers of the receiver.

max-dpb: The value of max-dpb is an integer indicating the maximum decoded picture buffer size in units of 1024 bytes. The max-dpb parameter signals that the receiver has more memory than the minimum amount of decoded picture buffer memory required by the signaled level conveyed in the value of the profile-level-id parameter. When max-dpb is signaled, the receiver MUST be able to decode NAL unit streams that conform to the signaled level, with the exception that the MaxDPB value in Table A-1 of [1] for the signaled level is replaced with the value of max-dpb. Consequently, a receiver that signals max-dpb MUST be capable of storing the following number of decoded frames, complementary field pairs, and non-paired fields in its decoded picture buffer:

Min(1024 * max-dpb / (PicWidthInMbs * FrameHeightInMbs * 256 * ChromaFormatFactor), 16)

PicWidthInMbs, FrameHeightInMbs, and ChromaFormatFactor are defined in [1].

The value of max-dpb MUST be greater than or equal to the value of MaxDPB for the level given in Table A-1 of [1]. Senders MAY use this knowledge to construct coded video streams with improved compression.

Informative note: This parameter was added primarily to complement a similar codepoint in the ITU-T Recommendation H.245, so as to facilitate signaling gateway designs. The decoded picture buffer stores reconstructed samples and is a property of the video decoder only. There is no relationship between the size of the decoded picture buffer and the buffers used in RTP, especially de-interleaving and de-jitter

max-br: The value of max-br is an integer indicating the maximum video bit rate in units of 1000 bits per second for the VCL HRD parameters (see A.3.1 item i of [1]) and in units of 1200 bits

per second for the NAL HRD parameters (see A.3.1 item j of [1]).

The max-br parameter signals that the video decoder of the receiver is capable of decoding video at a higher bit rate than is required by the signaled level conveyed in the value of the profile-level-id parameter. The value of max-br MUST be greater than or equal to the value of MaxBR for the level given in Table A-1 of [1].

When max-br is signaled, the video codec of the receiver MUST be able to decode NAL unit streams that conform to the signaled level, conveyed in the profile-level-id parameter, with the following exceptions in the limits specified by the level:

- The value of max-br replaces the MaxBR value of the signaled level (in Table A-1 of [1]).
- When the max-cpb parameter is not present, the result of the following formula replaces the value of MaxCPB in Table A-1 of [1]: (MaxCPB of the signaled level) * max-br / (MaxBR of the signaled level).

For example, if a receiver signals capability for Level 1.2 with max-br equal to 1550, this indicates a maximum video bitrate of 1550 kbits/sec for VCL HRD parameters, a maximum video bitrate of 1860 kbits/sec for NAL HRD parameters, and a CPB size of 4036458 bits (1550000 / 384000 * 1000 * 1000).

The value of max-br MUST be greater than or equal to the value MaxBR for the signaled level given in Table A-1 of [1].

Senders MAY use this knowledge to send higher bitrate video as allowed in the level definition of Annex A of H.264, to achieve improved video quality.

Informative note: This parameter was added primarily to complement a similar codepoint in the ITU-T Recommendation H.245, so as to facilitate signaling gateway designs. No assumption can be made from the value of

this parameter that the network is capable of handling such bit rates at any given time. In particular, no conclusion can be drawn that the signaled bit rate is possible under congestion control constraints.

redundant-pic-cap:
This parameter signals the capabilities of a receiver implementation. When equal to 0, the parameter indicates that the receiver makes no attempt to use redundant coded pictures to correct incorrectly decoded primary coded pictures. When equal to 0, the receiver is not capable of using redundant slices; therefore, a sender SHOULD avoid sending redundant slices to save bandwidth. When equal to 1, the receiver is capable of decoding any such redundant slice that covers a corrupted area in a primary decoded picture (at least partly), and therefore a sender MAY send redundant slices. When the parameter is not present, then a value of 0 MUST be used for redundant-pic-cap. When present, the value of redundant-pic-cap MUST be either 0 or 1.

When the profile-level-id parameter is present in the same capability signaling as the redundant-pic-cap parameter, and the profile indicated in profile-level-id is such that it disallows the use of redundant coded pictures (e.g., Main Profile), the value of redundant-pic-cap MUST be equal to 0. When a receiver indicates redundant-pic-cap equal to 0, the received stream SHOULD NOT contain redundant coded pictures.

Informative note: Even if redundant-pic-cap is equal to 0, the decoder is able to ignore redundant codec pictures provided that the decoder supports such a profile (Baseline, Extended) in which redundant coded pictures are allowed.

Informative note: Even if redundant-pic-cap is equal to 1, the receiver may also choose other error concealment strategies to

replace or complement decoding of redundant slices.

sprop-parameter-sets:
This parameter MAY be used to convey any sequence and picture parameter set NAL units (herein referred to as the initial parameter set NAL units) that MUST precede any other NAL units in decoding order. The parameter MUST NOT be used to indicate codec capability in any capability exchange procedure. The value of the parameter is the base64 [6] representation of the initial parameter set NAL units as specified in sections 7.3.2.1 and 7.3.2.2 of [1]. The parameter sets are conveyed in decoding order, and no framing of the parameter set NAL units takes place. A comma is used to separate any pair of parameter sets in the list. Note that the number of bytes in a parameter set NAL unit is typically less than 10, but a picture parameter set NAL unit can contain several hundreds of bytes.

Informative note: When several payload types are offered in the SDP Offer/Answer model, each with its own sprop-parameter-sets parameter, then the receiver cannot assume that those parameter sets do not use conflicting storage locations (i.e., identical values of parameter set identifiers). Therefore, a receiver should double-buffer all sprop-parameter-sets and make them available to the decoder instance that decodes a certain payload type.

parameter-add: This parameter MAY be used to signal whether the receiver of this parameter is allowed to add parameter sets in its signaling response using the sprop-parameter-sets MIME parameter. The value of this parameter is either 0 or 1. 0 is equal to false; i.e., it is not allowed to add parameter sets. 1 is equal to true; i.e., it is allowed to add parameter sets. If the parameter is not present, its value MUST be 1.

packetization-mode:
This parameter signals the properties of an RTP payload type or the capabilities of a receiver implementation. Only a single configuration point can be indicated; thus, when capabilities to support more than one packetization-mode are declared, multiple configuration points (RTP payload types) must be used.

When the value of packetization-mode is equal to 0 or packetization-mode is not present, the single NAL mode, as defined in section 6.2 of RFC 3984, MUST be used. This mode is in use in standards using ITU-T Recommendation H.241 [15] (see section 12.1). When the value of packetization-mode is equal to 1, the non-interleaved mode, as defined in section 6.3 of RFC 3984, MUST be used. When the value of packetization-mode is equal to 2, the interleaved mode, as defined in section 6.4 of RFC 3984, MUST be used. The value of packetization mode MUST be an integer in the range of 0 to 2, inclusive.

sprop-interleaving-depth:
This parameter MUST NOT be present when packetization-mode is not present or the value of packetization-mode is equal to 0 or 1. This parameter MUST be present when the value of packetization-mode is equal to 2.

This parameter signals the properties of a NAL unit stream. It specifies the maximum number of VCL NAL units that precede any VCL NAL unit in the NAL unit stream in transmission order and follow the VCL NAL unit in decoding order. Consequently, it is guaranteed that receivers can reconstruct NAL unit decoding order when the buffer size for NAL unit decoding order recovery is at least the value of sprop-interleaving-depth + 1 in terms of VCL NAL units.

The value of sprop-interleaving-depth MUST be an integer in the range of 0 to 32767, inclusive.

sprop-deint-buf-req:
This parameter MUST NOT be present when packetization-mode is not present or the value of packetization-mode is equal to 0 or 1. It MUST be present when the value of packetization-mode is equal to 2.

sprop-deint-buf-req signals the required size of the deinterleaving buffer for the NAL unit stream. The value of the parameter MUST be greater than or equal to the maximum buffer occupancy (in units of bytes) required in such a deinterleaving buffer that is specified in section 7.2 of RFC 3984. It is guaranteed that receivers can perform the deinterleaving of interleaved NAL units into NAL unit decoding order, when the deinterleaving buffer size is at least the value of sprop-deint-buf-req in terms of bytes.

The value of sprop-deint-buf-req MUST be an integer in the range of 0 to 4294967295, inclusive.

Informative note: sprop-deint-buf-req indicates the required size of the deinterleaving buffer only. When network jitter can occur, an appropriately sized jitter buffer has to be provisioned for as well.

deint-buf-cap: This parameter signals the capabilities of a receiver implementation and indicates the amount of deinterleaving buffer space in units of bytes that the receiver has available for reconstructing the NAL unit decoding order. A receiver is able to handle any stream for which the value of the sprop-deint-buf-req parameter is smaller than or equal to this parameter.

If the parameter is not present, then a value of 0 MUST be used for deint-buf-cap. The value of deint-buf-cap MUST be an integer in the range of 0 to 4294967295, inclusive.

Informative note: deint-buf-cap indicates the maximum possible size of the deinterleaving buffer of the receiver only.

When network jitter can occur, an appropriately sized jitter buffer has to be provisioned for as well.

sprop-init-buf-time:
This parameter MAY be used to signal the properties of a NAL unit stream. The parameter MUST NOT be present, if the value of packetization-mode is equal to 0 or 1.

The parameter signals the initial buffering time that a receiver MUST buffer before starting decoding to recover the NAL unit decoding order from the transmission order. The parameter is the maximum value of (transmission time of a NAL unit - decoding time of the NAL unit), assuming reliable and instantaneous transmission, the same timeline for transmission and decoding, and that decoding starts when the first packet arrives.

An example of specifying the value of sprop-init-buf-time follows. A NAL unit stream is sent in the following interleaved order, in which the value corresponds to the decoding time and the transmission order is from left to right:

0 2 1 3 5 4 6 8 7 ...

Assuming a steady transmission rate of NAL units, the transmission times are:

0 1 2 3 4 5 6 7 8 ...

Subtracting the decoding time from the transmission time column-wise results in the following series:

0 -1 1 0 -1 1 0 -1 1 ...

Thus, in terms of intervals of NAL unit transmission times, the value of sprop-init-buf-time in this example is 1.

The parameter is coded as a non-negative base10 integer representation in clock ticks of a 90-kHz clock. If the parameter is not present, then no initial buffering time value is defined. Otherwise the value of sprop-init-buf-time MUST be an integer in the range of 0 to 4294967295, inclusive.

In addition to the signaled sprop-init-buf-time, receivers SHOULD take into account the transmission delay jitter buffering, including buffering for the delay jitter caused by mixers, translators, gateways, proxies, traffic-shapers, and other network elements.

sprop-max-don-diff:
This parameter MAY be used to signal the properties of a NAL unit stream. It MUST NOT be used to signal transmitter or receiver or codec capabilities. The parameter MUST NOT be present if the value of packetization-mode is equal to 0 or 1. sprop-max-don-diff is an integer in the range of 0 to 32767, inclusive. If sprop-max-don-diff is not present, the value of the parameter is unspecified. sprop-max-don-diff is calculated as follows:

sprop-max-don-diff = max{AbsDON(i) - AbsDON(j)},
for any i and any j>i,

where i and j indicate the index of the NAL unit in the transmission order and AbsDON denotes a decoding order number of the NAL unit that does not wrap around to 0 after 65535. In other words, AbsDON is calculated as follows: Let m and n be consecutive NAL units in transmission order. For the very first NAL unit in transmission order (whose index is 0), AbsDON(0) = DON(0). For other NAL units, AbsDON is calculated as follows:

If DON(m) == DON(n), AbsDON(n) = AbsDON(m)

If (DON(m) < DON(n) and DON(n) - DON(m) < 32768),
AbsDON(n) = AbsDON(m) + DON(n) - DON(m)

If (DON(m) > DON(n) and DON(m) - DON(n) >= 32768),
AbsDON(n) = AbsDON(m) + 65536 - DON(m) + DON(n)

If (DON(m) < DON(n) and DON(n) - DON(m) >= 32768),

AbsDON(n) = AbsDON(m) - (DON(m) + 65536 - DON(n))

If (DON(m) > DON(n) and DON(m) - DON(n) < 32768),
AbsDON(n) = AbsDON(m) - (DON(m) - DON(n))

where DON(i) is the decoding order number of the NAL unit having index i in the transmission order. The decoding order number is specified in section 5.5 of RFC 3984.

Informative note: Receivers may use sprop-max-don-diff to trigger which NAL units in the receiver buffer can be passed to the decoder.

max-rcmd-nalu-size:

This parameter MAY be used to signal the capabilities of a receiver. The parameter MUST NOT be used for any other purposes. The value of the parameter indicates the largest NALU size in bytes that the receiver can handle efficiently. The parameter value is a recommendation, not a strict upper boundary. The sender MAY create larger NALUs but must be aware that the handling of these may come at a higher cost than NALUs conforming to the limitation.

The value of max-rcmd-nalu-size MUST be an integer in the range of 0 to 4294967295, inclusive. If this parameter is not specified, no known limitation to the NALU size exists. Senders still have to consider the MTU size available between the sender and the receiver and SHOULD run MTU discovery for this purpose.

This parameter is motivated by, for example, an IP to H.223 video telephony gateway, where NALUs smaller than the H.223 transport data

unit will be more efficient. A gateway may terminate IP; thus, MTU discovery will normally not work beyond the gateway.

Informative note: Setting this parameter to a lower than necessary value may have a negative impact.

Encoding considerations:
This type is only defined for transfer via RTP (RFC 3550).

A file format of H.264/AVC video is defined in [29]. This definition is utilized by other file formats, such as the 3GPP multimedia file format (MIME type video/3gpp) [30] or the MP4 file format (MIME type video/mp4).

Security considerations:
See section 9 of RFC 3984.

Public specification:
Please refer to RFC 3984 and its section 15.

Additional information:
None

File extensions: none
Macintosh file type code: none
Object identifier or OID: none

Person & email address to contact for further information:
stewe@stewe.org

Intended usage: COMMON

Author:
stewe@stewe.org
Change controller:
IETF Audio/Video Transport working group delegated from the IESG.

8.2. SDP Parameters

8.2.1. Mapping of MIME Parameters to SDP

The MIME media type video/H264 string is mapped to fields in the Session Description Protocol (SDP) [5] as follows:

- The media name in the "m=" line of SDP MUST be video.
- The encoding name in the "a=rtpmap" line of SDP MUST be H264 (the MIME subtype).
- The clock rate in the "a=rtpmap" line MUST be 90000.
- The OPTIONAL parameters "profile-level-id", "max-mbps", "max-fs", "max-cpb", "max-dpb", "max-br", "redundant-pic-cap", "sprop-parameter-sets", "parameter-add", "packetization-mode", "sprop-interleaving-depth", "deint-buf-cap", "sprop-deint-buf-req", "sprop-init-buf-time", "sprop-max-don-diff", and "max-rcmd-nalu-size", when present, MUST be included in the "a=fmtp" line of SDP. These parameters are expressed as a MIME media type string, in the form of a semicolon separated list of parameter=value pairs.

An example of media representation in SDP is as follows (Baseline Profile, Level 3.0, some of the constraints of the Main profile may not be obeyed):

```
m=video 49170 RTP/AVP 98
a=rtpmap:98 H264/90000
a=fmtp:98 profile-level-id=42A01E;
          sprop-parameter-sets=Z0IACpZTBYmI,aMljiA==
```

8.2.2. Usage with the SDP Offer/Answer Model

When H.264 is offered over RTP using SDP in an Offer/Answer model [7] for negotiation for unicast usage, the following limitations and rules apply:

- The parameters identifying a media format configuration for H.264 are "profile-level-id", "packetization-mode", and, if required by "packetization-mode", "sprop-deint-buf-req". These three parameters MUST be used symmetrically; i.e., the answerer MUST either maintain all configuration parameters or remove the media format (payload type) completely, if one or more of the parameter values are not supported.

Informative note: The requirement for symmetric use applies only for the above three parameters and not for the other stream properties and capability parameters.

To simplify handling and matching of these configurations, the same RTP payload type number used in the offer SHOULD also be used in the answer, as specified in [7]. An answer MUST NOT contain a payload type number used in the offer unless the configuration ("profile-level-id", "packetization-mode", and, if present, "sprop-deint-buf-req") is the same as in the offer.

Informative note: An offerer, when receiving the answer, has to compare payload types not declared in the offer based on media type (i.e., video/h264) and the above three parameters with any payload types it has already declared, in order to determine whether the configuration in question is new or equivalent to a configuration already offered.

- The parameters "sprop-parameter-sets", "sprop-deint-buf-req", "sprop-interleaving-depth", "sprop-max-don-diff", and "sprop-init-buf-time" describe the properties of the NAL unit stream that the offerer or answerer is sending for this media format configuration. This differs from the normal usage of the Offer/Answer parameters: normally such parameters declare the properties of the stream that the offerer or the answerer is able to receive. When dealing with H.264, the offerer assumes that the answerer will be able to receive media encoded using the configuration being offered.

  Informative note: The above parameters apply for any stream sent by the declaring entity with the same configuration; i.e., they are dependent on their source. Rather then being bound to the payload type, the values may have to be applied to another payload type when being sent, as they apply for the configuration.

- The capability parameters ("max-mbps", "max-fs", "max-cpb", "max-dpb", "max-br", ,"redundant-pic-cap", "max-rcmd-nalu-size") MAY be used to declare further capabilities. Their interpretation depends on the direction attribute. When the direction attribute is sendonly, then the parameters describe the limits of the RTP packets and the NAL unit stream that the sender is capable of producing. When the direction attribute is sendrecv or recvonly, then the parameters describe the limitations of what the receiver accepts.

- As specified above, an offerer has to include the size of the deinterleaving buffer in the offer for an interleaved H.264 stream. To enable the offerer and answerer to inform each other about their capabilities for deinterleaving buffering, both parties are RECOMMENDED to include "deint-buf-cap". This information MAY be used when the value for "sprop-deint-buf-req" is selected in a second round of offer and answer. For interleaved streams, it is also RECOMMENDED to consider offering multiple payload types with different buffering requirements when the capabilities of the receiver are unknown.

- The "sprop-parameter-sets" parameter is used as described above. In addition, an answerer MUST maintain all parameter sets received in the offer in its answer. Depending on the value of the "parameter-add" parameter, different rules apply: If "parameter-add" is false (0), the answer MUST NOT add any additional parameter sets. If "parameter-add" is true (1), the answerer, in its answer, MAY add additional parameter sets to the "sprop-parameter-sets" parameter. The answerer MUST also, independent of the value of "parameter-add", accept to receive a video stream using the sprop-parameter-sets it declared in the answer.

  Informative note: care must be taken when parameter sets are added not to cause overwriting of already transmitted parameter sets by using conflicting parameter set identifiers.

For streams being delivered over multicast, the following rules apply in addition:

- The stream properties parameters ("sprop-parameter-sets", "sprop-deint-buf-req", "sprop-interleaving-depth", "sprop-max-don-diff", and "sprop-init-buf-time") MUST NOT be changed by the answerer. Thus, a payload type can either be accepted unaltered or removed.

- The receiver capability parameters "max-mbps", "max-fs", "max-cpb", "max-dpb", "max-br", and "max-rcmd-nalu-size" MUST be supported by the answerer for all streams declared as sendrecv or recvonly; otherwise, one of the following actions MUST be performed: the media format is removed, or the session rejected.

- The receiver capability parameter redundant-pic-cap SHOULD be supported by the answerer for all streams declared as sendrecv or recvonly as follows: The answerer SHOULD NOT include redundant coded pictures in the transmitted stream if the offerer indicated redundant-pic-cap equal to 0. Otherwise (when redundant_pic_cap is equal to 1), it is beyond the scope of this memo to recommend how the answerer should use redundant coded pictures.

Below are the complete lists of how the different parameters shall be interpreted in the different combinations of offer or answer and direction attribute.

- In offers and answers for which "a=sendrecv" or no direction attribute is used, or in offers and answers for which "a=recvonly" is used, the following interpretation of the parameters MUST be used.

  Declaring actual configuration or properties for receiving:

  - profile-level-id
  - packetization-mode

  Declaring actual properties of the stream to be sent (applicable only when "a=sendrecv" or no direction attribute is used):

  - sprop-deint-buf-req
  - sprop-interleaving-depth
  - sprop-parameter-sets
  - sprop-max-don-diff
  - sprop-init-buf-time

  Declaring receiver implementation capabilities:

  - max-mbps
  - max-fs
  - max-cpb
  - max-dpb
  - max-br
  - redundant-pic-cap
  - deint-buf-cap
  - max-rcmd-nalu-size

  Declaring how Offer/Answer negotiation shall be performed:

  - parameter-add

- In an offer or answer for which the direction attribute "a=sendonly" is included for the media stream, the following interpretation of the parameters MUST be used:

  Declaring actual configuration and properties of stream proposed to be sent:

  - profile-level-id
  - packetization-mode
  - sprop-deint-buf-req

- sprop-max-don-diff
- sprop-init-buf-time
- sprop-parameter-sets
- sprop-interleaving-depth

Declaring the capabilities of the sender when it receives a stream:

- max-mbps
- max-fs
- max-cpb
- max-dpb
- max-br
- redundant-pic-cap
- deint-buf-cap
- max-rcmd-nalu-size

Declaring how Offer/Answer negotiation shall be performed:

- parameter-add

Furthermore, the following considerations are necessary:

o Parameters used for declaring receiver capabilities are in general downgradable; i.e., they express the upper limit for a sender's possible behavior. Thus a sender MAY select to set its encoder using only lower/lesser or equal values of these parameters. "sprop-parameter-sets" MUST NOT be used in a sender's declaration of its capabilities, as the limits of the values that are carried inside the parameter sets are implicit with the profile and level used.

o Parameters declaring a configuration point are not downgradable, with the exception of the level part of the "profile-level-id" parameter. This expresses values a receiver expects to be used and must be used verbatim on the sender side.

o When a sender's capabilities are declared, and non-downgradable parameters are used in this declaration, then these parameters express a configuration that is acceptable. In order to achieve high interoperability levels, it is often advisable to offer multiple alternative configurations; e.g., for the packetization mode. It is impossible to offer multiple configurations in a single payload type. Thus, when multiple configuration offers are made, each offer requires its own RTP payload type associated with the offer.

- o A receiver SHOULD understand all MIME parameters, even if it only supports a subset of the payload format's functionality. This ensures that a receiver is capable of understanding when an offer to receive media can be downgraded to what is supported by the receiver of the offer.

- o An answerer MAY extend the offer with additional media format configurations. However, to enable their usage, in most cases a second offer is required from the offerer to provide the stream properties parameters that the media sender will use. This also has the effect that the offerer has to be able to receive this media format configuration, not only to send it.

- o If an offerer wishes to have non-symmetric capabilities between sending and receiving, the offerer has to offer different RTP sessions; i.e., different media lines declared as "recvonly" and "sendonly", respectively. This may have further implications on the system.

8.2.3. Usage in Declarative Session Descriptions

When H.264 over RTP is offered with SDP in a declarative style, as in RTSP [27] or SAP [28], the following considerations are necessary.

- o All parameters capable of indicating the properties of both a NAL unit stream and a receiver are used to indicate the properties of a NAL unit stream. For example, in this case, the parameter "profile-level-id" declares the values used by the stream, instead of the capabilities of the sender. This results in that the following interpretation of the parameters MUST be used:

  Declaring actual configuration or properties:

  - profile-level-id
  - sprop-parameter-sets
  - packetization-mode
  - sprop-interleaving-depth
  - sprop-deint-buf-req
  - sprop-max-don-diff
  - sprop-init-buf-time

Not usable:

- max-mbps
- max-fs
- max-cpb
- max-dpb
- max-br
- redundant-pic-cap
- max-rcmd-nalu-size
- parameter-add
- deint-buf-cap

o A receiver of the SDP is required to support all parameters and values of the parameters provided; otherwise, the receiver MUST reject (RTSP) or not participate in (SAP) the session. It falls on the creator of the session to use values that are expected to be supported by the receiving application.

8.3. Examples

A SIP Offer/Answer exchange wherein both parties are expected to both send and receive could look like the following. Only the media codec specific parts of the SDP are shown. Some lines are wrapped due to text constraints.

Offerer -> Answer SDP message:

```
m=video 49170 RTP/AVP 100 99 98
a=rtpmap:98 H264/90000
a=fmtp:98 profile-level-id=42A01E; packetization-mode=0;
          sprop-parameter-sets=Z0IACpZTBYmI,aMljiA==
a=rtpmap:99 H264/90000
a=fmtp:99 profile-level-id=42A01E; packetization-mode=1;
          sprop-parameter-sets=Z0IACpZTBYmI,aMljiA==
a=rtpmap:100 H264/90000
a=fmtp:100 profile-level-id=42A01E; packetization-mode=2;
           sprop-parameter-sets=Z0IACpZTBYmI,aMljiA==;
           sprop-interleaving-depth=45; sprop-deint-buf-req=64000;
           sprop-init-buf-time=102478; deint-buf-cap=128000
```

The above offer presents the same codec configuration in three different packetization formats. PT 98 represents single NALU mode, PT 99 non-interleaved mode; PT 100 indicates the interleaved mode. In the interleaved mode case, the interleaving parameters that the offerer would use if the answer indicates support for PT 100 are also included. In all three cases the parameter "sprop-parameter-sets" conveys the initial parameter sets that are required for the answerer when receiving a stream from the offerer when this configuration

(profile-level-id and packetization mode) is accepted. Note that the value for "sprop-parameter-sets", although identical in the example above, could be different for each payload type.

Answerer -> Offerer SDP message:

```
m=video 49170 RTP/AVP 100 99 97
a=rtpmap:97 H264/90000
a=fmtp:97 profile-level-id=42A01E; packetization-mode=0;
          sprop-parameter-sets=Z0IACpZTBYmI,aMljiA==,As0DEWlsIOp==,
          KyzFGleR
a=rtpmap:99 H264/90000
a=fmtp:99 profile-level-id=42A01E; packetization-mode=1;
          sprop-parameter-sets=Z0IACpZTBYmI,aMljiA==,As0DEWlsIOp==,
          KyzFGleR; max-rcmd-nalu-size=3980
a=rtpmap:100 H264/90000
a=fmtp:100 profile-level-id=42A01E; packetization-mode=2;
          sprop-parameter-sets=Z0IACpZTBYmI,aMljiA==,As0DEWlsIOp==,
          KyzFGleR; sprop-interleaving-depth=60;
          sprop-deint-buf-req=86000; sprop-init-buf-time=156320;
          deint-buf-cap=128000; max-rcmd-nalu-size=3980
```

As the Offer/Answer negotiation covers both sending and receiving streams, an offer indicates the exact parameters for what the offerer is willing to receive, whereas the answer indicates the same for what the answerer accepts to receive. In this case the offerer declared that it is willing to receive payload type 98. The answerer accepts this by declaring a equivalent payload type 97; i.e., it has identical values for the three parameters "profile-level-id", packetization-mode, and "sprop-deint-buf-req". This has the following implications for both the offerer and the answerer concerning the parameters that declare properties. The offerer initially declared a certain value of the "sprop-parameter-sets" in the payload definition for PT=98. However, as the answerer accepted this as PT=97, the values of "sprop-parameter-sets" in PT=98 must now be used instead when the offerer sends PT=97. Similarly, when the answerer sends PT=98 to the offerer, it has to use the properties parameters it declared in PT=97.

The answerer also accepts the reception of the two configurations that payload types 99 and 100 represent. It provides the initial parameter sets for the answerer-to-offerer direction, and for buffering related parameters that it will use to send the payload types. It also provides the offerer with its memory limit for deinterleaving operations by providing a "deint-buf-cap" parameter. This is only useful if the offerer decides on making a second offer, where it can take the new value into account. The "max-rcmd-nalu-size" indicates that the answerer can efficiently process NALUs up to

the size of 3980 bytes. However, there is no guarantee that the network supports this size.

Please note that the parameter sets in the above example do not represent a legal operation point of an H.264 codec. The base64 strings are only used for illustration.

8.4. Parameter Set Considerations

The H.264 parameter sets are a fundamental part of the video codec and vital to its operation; see section 1.2. Due to their characteristics and their importance for the decoding process, lost or erroneously transmitted parameter sets can hardly be concealed locally at the receiver. A reference to a corrupt parameter set has normally fatal results to the decoding process. Corruption could occur, for example, due to the erroneous transmission or loss of a parameter set data structure, but also due to the untimely transmission of a parameter set update. Therefore, the following recommendations are provided as a guideline for the implementer of the RTP sender.

Parameter set NALUs can be transported using three different principles:

A. Using a session control protocol (out-of-band) prior to the actual RTP session.

B. Using a session control protocol (out-of-band) during an ongoing RTP session.

C. Within the RTP stream in the payload (in-band) during an ongoing RTP session.

It is necessary to implement principles A and B within a session control protocol. SIP and SDP can be used as described in the SDP Offer/Answer model and in the previous sections of this memo. This section contains guidelines on how principles A and B must be implemented within session control protocols. It is independent of the particular protocol used. Principle C is supported by the RTP payload format defined in this specification.

The picture and sequence parameter set NALUs SHOULD NOT be transmitted in the RTP payload unless reliable transport is provided for RTP, as a loss of a parameter set of either type will likely prevent decoding of a considerable portion of the corresponding RTP

stream. Thus, the transmission of parameter sets using a reliable session control protocol (i.e., usage of principle A or B above) is RECOMMENDED.

In the rest of the section it is assumed that out-of-band signaling provides reliable transport of parameter set NALUs and that in-band transport does not. If in-band signaling of parameter sets is used, the sender SHOULD take the error characteristics into account and use mechanisms to provide a high probability for delivering the parameter sets correctly. Mechanisms that increase the probability for a correct reception include packet repetition, FEC, and retransmission. The use of an unreliable, out-of-band control protocol has similar disadvantages as the in-band signaling (possible loss) and, in addition, may also lead to difficulties in the synchronization (see below). Therefore, it is NOT RECOMMENDED.

Parameter sets MAY be added or updated during the lifetime of a session using principles B and C. It is required that parameter sets are present at the decoder prior to the NAL units that refer to them. Updating or adding of parameter sets can result in further problems, and therefore the following recommendations should be considered.

- When parameter sets are added or updated, principle C is vulnerable to transmission errors as described above, and therefore principle B is RECOMMENDED.

- When parameter sets are added or updated, care SHOULD be taken to ensure that any parameter set is delivered prior to its usage. It is common that no synchronization is present between out-of-band signaling and in-band traffic. If out-of-band signaling is used, it is RECOMMENDED that a sender does not start sending NALUs requiring the updated parameter sets prior to acknowledgement of delivery from the signaling protocol.

- When parameter sets are updated, the following synchronization issue should be taken into account. When overwriting a parameter set at the receiver, the sender has to ensure that the parameter set in question is not needed by any NALU present in the network or receiver buffers. Otherwise, decoding with a wrong parameter set may occur. To lessen this problem, it is RECOMMENDED either to overwrite only those parameter sets that have not been used for a sufficiently long time (to ensure that all related NALUs have been consumed), or to add a new parameter set instead (which may have negative consequences for the efficiency of the video coding).

- When new parameter sets are added, previously unused parameter set identifiers are used. This avoids the problem identified in the

previous paragraph. However, in a multiparty session, unless a synchronized control protocol is used, there is a risk that multiple entities try to add different parameter sets for the same identifier, which has to be avoided.

- Adding or modifying parameter sets by using both principles B and C in the same RTP session may lead to inconsistencies of the parameter sets because of the lack of synchronization between the control and the RTP channel. Therefore, principles B and C MUST NOT both be used in the same session unless sufficient synchronization can be provided.

In some scenarios (e.g., when only the subset of this payload format specification corresponding to H.241 is used), it is not possible to employ out-of-band parameter set transmission. In this case,

synchronization with the non-parameter-set-data in the bitstream is implicit, but the possibility of a loss has to be taken into account. The loss probability should be reduced using the mechanisms discussed above.

- When parameter sets are initially provided using principle A and then later added or updated in-band (principle C), there is a risk associated with updating the parameter sets delivered out-of-band. If receivers miss some in-band updates (for example, because of a loss or a late tune-in), those receivers attempt to decode the bitstream using out-dated parameters. It is RECOMMENDED that parameter set IDs be partitioned between the out-of-band and in-band parameter sets.

To allow for maximum flexibility and best performance from the H.264 coder, it is recommended, if possible, to allow any sender to add its own parameter sets to be used in a session. Setting the "parameter-add" parameter to false should only be done in cases where the session topology prevents a participant to add its own parameter sets.

9. Security Considerations

RTP packets using the payload format defined in this specification are subject to the security considerations discussed in the RTP specification [4], and in any appropriate RTP profile (for example, [16]). This implies that confidentiality of the media streams is achieved by encryption; for example, through the application of SRTP [26]. Because the data compression used with this payload format is applied end-to-end, any encryption needs to be performed after compression.

A potential denial-of-service threat exists for data encodings using compression techniques that have non-uniform receiver-end computational load. The attacker can inject pathological datagrams into the stream that are complex to decode and that cause the receiver to be overloaded. H.264 is particularly vulnerable to such attacks, as it is extremely simple to generate datagrams containing NAL units that affect the decoding process of many future NAL units. Therefore, the usage of data origin authentication and data integrity protection of at least the RTP packet is RECOMMENDED; for example, with SRTP [26].

Note that the appropriate mechanism to ensure confidentiality and integrity of RTP packets and their payloads is very dependent on the application and on the transport and signaling protocols employed. Thus, although SRTP is given as an example above, other possible choices exist.

Decoders MUST exercise caution with respect to the handling of user data SEI messages, particularly if they contain active elements, and MUST restrict their domain of applicability to the presentation containing the stream.

End-to-End security with either authentication, integrity or confidentiality protection will prevent a MANE from performing media-aware operations other than discarding complete packets. And in the case of confidentiality protection it will even be prevented from performing discarding of packets in a media aware way. To allow any MANE to perform its operations, it will be required to be a trusted entity which is included in the security context establishment.

10. Congestion Control

Congestion control for RTP SHALL be used in accordance with RFC 3550 [4], and with any applicable RTP profile; e.g., RFC 3551 [16]. An additional requirement if best-effort service is being used is: users of this payload format MUST monitor packet loss to ensure that the packet loss rate is within acceptable parameters. Packet loss is considered acceptable if a TCP flow across the same network path, and experiencing the same network conditions, would achieve an average throughput, measured on a reasonable timescale, that is not less than the RTP flow is achieving. This condition can be satisfied by implementing congestion control mechanisms to adapt the transmission rate (or the number of layers subscribed for a layered multicast session), or by arranging for a receiver to leave the session if the loss rate is unacceptably high.

The bit rate adaptation necessary for obeying the congestion control principle is easily achievable when real-time encoding is used. However, when pre-encoded content is being transmitted, bandwidth adaptation requires the availability of more than one coded representation of the same content, at different bit rates, or the existence of non-reference pictures or sub-sequences [22] in the bitstream. The switching between the different representations can normally be performed in the same RTP session; e.g., by employing a concept known as SI/SP slices of the Extended Profile, or by switching streams at IDR picture boundaries. Only when non-downgradable parameters (such as the profile part of the profile/level ID) are required to be changed does it become necessary to terminate and re-start the media stream. This may be accomplished by using a different RTP payload type.

MANEs MAY follow the suggestions outlined in section 7.3 and remove certain unusable packets from the packet stream when that stream was damaged due to previous packet losses. This can help reduce the network load in certain special cases.

11. IANA Consideration

IANA has registered one new MIME type; see section 8.1.

12. Informative Appendix: Application Examples

This payload specification is very flexible in its use, in order to cover the extremely wide application space anticipated for H.264. However, this great flexibility also makes it difficult for an implementer to decide on a reasonable packetization scheme. Some information on how to apply this specification to real-world scenarios is likely to appear in the form of academic publications and a test model software and description in the near future. However, some preliminary usage scenarios are described here as well.

12.1. Video Telephony according to ITU-T Recommendation H.241 Annex A

H.323-based video telephony systems that use H.264 as an optional video compression scheme are required to support H.241 Annex A [15] as a packetization scheme. The packetization mechanism defined in this Annex is technically identical with a small subset of this specification.

When a system operates according to H.241 Annex A, parameter set NAL units are sent in-band. Only Single NAL unit packets are used. Many such systems are not sending IDR pictures regularly, but only when required by user interaction or by control protocol means; e.g., when switching between video channels in a Multipoint Control Unit or for error recovery requested by feedback.

12.2. Video Telephony, No Slice Data Partitioning, No NAL Unit Aggregation

The RTP part of this scheme is implemented and tested (though not the control-protocol part; see below).

In most real-world video telephony applications, picture parameters such as picture size or optional modes never change during the lifetime of a connection. Therefore, all necessary parameter sets (usually only one) are sent as a side effect of the capability exchange/announcement process, e.g., according to the SDP syntax specified in section 8.2 of this document. As all necessary parameter set information is established before the RTP session starts, there is no need for sending any parameter set NAL units. Slice data partitioning is not used, either. Thus, the RTP packet stream basically consists of NAL units that carry single coded slices.

The encoder chooses the size of coded slice NAL units so that they offer the best performance. Often, this is done by adapting the coded slice size to the MTU size of the IP network. For small

picture sizes, this may result in a one-picture-per-one-packet strategy. Intra refresh algorithms clean up the loss of packets and the resulting drift-related artifacts.

12.3. Video Telephony, Interleaved Packetization Using NAL Unit Aggregation

This scheme allows better error concealment and is used in H.263 based designs using RFC 2429 packetization [10]. It has been implemented, and good results were reported [12].

The VCL encoder codes the source picture so that all macroblocks (MBs) of one MB line are assigned to one slice. All slices with even MB row addresses are combined into one STAP, and all slices with odd MB row addresses into another. Those STAPs are transmitted as RTP packets. The establishment of the parameter sets is performed as discussed above.

Note that the use of STAPs is essential here, as the high number of individual slices (18 for a CIF picture) would lead to unacceptably high IP/UDP/RTP header overhead (unless the source coding tool FMO is used, which is not assumed in this scenario). Furthermore, some wireless video transmission systems, such as H.324M and the IP-based video telephony specified in 3GPP, are likely to use relatively small transport packet size. For example, a typical MTU size of H.223 AL3 SDU is around 100 bytes [17]. Coding individual slices according to this packetization scheme provides further advantage in communication between wired and wireless networks, as individual slices are likely to be smaller than the preferred maximum packet size of wireless systems. Consequently, a gateway can convert the STAPs used in a wired network into several RTP packets with only one NAL unit, which are preferred in a wireless network, and vice versa.

12.4. Video Telephony with Data Partitioning

This scheme has been implemented and has been shown to offer good performance, especially at higher packet loss rates [12].

Data Partitioning is known to be useful only when some form of unequal error protection is available. Normally, in single-session RTP environments, even error characteristics are assumed; i.e., the packet loss probability of all packets of the session is the same statistically. However, there are means to reduce the packet loss probability of individual packets in an RTP session. A FEC packet according to RFC 2733 [18], for example, specifies which media packets are associated with the FEC packet.

In all cases, the incurred overhead is substantial but is in the same order of magnitude as the number of bits that have otherwise been spent for intra information. However, this mechanism does not add any delay to the system.

Again, the complete parameter set establishment is performed through control protocol means.

12.5. Video Telephony or Streaming with FUs and Forward Error Correction

This scheme has been implemented and has been shown to provide good performance, especially at higher packet loss rates [19].

The most efficient means to combat packet losses for scenarios where retransmissions are not applicable is forward error correction (FEC). Although application layer, end-to-end use of FEC is often less efficient than an FEC-based protection of individual links (especially when links of different characteristics are in the transmission path), application layer, end-to-end FEC is unavoidable in some scenarios. RFC 2733 [18] provides means to use generic, application layer, end-to-end FEC in packet-loss environments. A binary forward error correcting code is generated by applying the XOR operation to the bits at the same bit position in different packets. The binary code can be specified by the parameters (n,k) in which k is the number of information packets used in the connection and n is the total number of packets generated for k information packets; i.e., n-k parity packets are generated for k information packets.

When a code is used with parameters (n,k) within the RFC 2733 framework, the following properties are well known:

a) If applied over one RTP packet, RFC 2733 provides only packet repetition.

b) RFC 2733 is most bit rate efficient if XOR-connected packets have equal length.

c) At the same packet loss probability p and for a fixed k, the greater the value of n is, the smaller the residual error probability becomes. For example, for a packet loss probability of 10%, k=1, and n=2, the residual error probability is about 1%, whereas for n=3, the residual error probability is about 0.1%.

d) At the same packet loss probability p and for a fixed code rate k/n, the greater the value of n is, the smaller the residual error probability becomes. For example, at a packet loss probability of p=10%, k=1 and n=2, the residual error rate is about 1%, whereas

for an extended Golay code with k=12 and n=24, the residual error rate is about 0.01%.

For applying RFC 2733 in combination with H.264 baseline coded video without using FUs, several options might be considered:

1) The video encoder produces NAL units for which each video frame is coded in a single slice. Applying FEC, one could use a simple code; e.g., (n=2, k=1). That is, each NAL unit would basically just be repeated. The disadvantage is obviously the bad code performance according to d), above, and the low flexibility, as only (n, k=1) codes can be used.

2) The video encoder produces NAL units for which each video frame is encoded in one or more consecutive slices. Applying FEC, one could use a better code, e.g., (n=24, k=12), over a sequence of NAL units. Depending on the number of RTP packets per frame, a loss may introduce a significant delay, which is reduced when more RTP packets are used per frame. Packets of completely different length might also be connected, which decreases bit rate efficiency according to b), above. However, with some care and for slices of 1kb or larger, similar length (100-200 bytes difference) may be produced, which will not lower the bit efficiency catastrophically.

3) The video encoder produces NAL units, for which a certain frame contains k slices of possibly almost equal length. Then, applying FEC, a better code, e.g., (n=24, k=12), can be used over the sequence of NAL units for each frame. The delay compared to that of 2), above, may be reduced, but several disadvantages are obvious. First, the coding efficiency of the encoded video is lowered significantly, as slice-structured coding reduces intra-frame prediction and additional slice overhead is necessary. Second, pre-encoded content or, when operating over a gateway, the video is usually not appropriately coded with k slices such that FEC can be applied. Finally, the encoding of video producing k slices of equal length is not straightforward and might require more than one encoding pass.

Many of the mentioned disadvantages can be avoided by applying FUs in combination with FEC. Each NAL unit can be split into any number of FUs of basically equal length; therefore, FEC with a reasonable k and n can be applied, even if the encoder made no effort to produce slices of equal length. For example, a coded slice NAL unit containing an entire frame can be split to k FUs, and a parity check code (n=k+1, k) can be applied. However, this has the disadvantage

that unless all created fragments can be recovered, the whole slice will be lost. Thus a larger section is lost than would be if the frame had been split into several slices.

The presented technique makes it possible to achieve good transmission error tolerance, even if no additional source coding layer redundancy (such as periodic intra frames) is present. Consequently, the same coded video sequence can be used to achieve the maximum compression efficiency and quality over error-free transmission and for transmission over error-prone networks. Furthermore, the technique allows the application of FEC to pre-encoded sequences without adding delay. In this case, pre-encoded sequences that are not encoded for error-prone networks can still be transmitted almost reliably without adding extensive delays. In addition, FUs of equal length result in a bit rate efficient use of RFC 2733.

If the error probability depends on the length of the transmitted packet (e.g., in case of mobile transmission [14]), the benefits of applying FUs with FEC are even more obvious. Basically, the flexibility of the size of FUs allows appropriate FEC to be applied for each NAL unit and unequal error protection of NAL units.

When FUs and FEC are used, the incurred overhead is substantial but is in the same order of magnitude as the number of bits that have to be spent for intra-coded macroblocks if no FEC is applied. In [19], it was shown that the overall performance of the FEC-based approach enhanced quality when using the same error rate and same overall bit rate, including the overhead.

12.6. Low Bit-Rate Streaming

This scheme has been implemented with H.263 and non-standard RTP packetization and has given good results [20]. There is no technical reason why similarly good results could not be achievable with H.264.

In today's Internet streaming, some of the offered bit rates are relatively low in order to allow terminals with dial-up modems to access the content. In wired IP networks, relatively large packets, say 500 - 1500 bytes, are preferred to smaller and more frequently occurring packets in order to reduce network congestion. Moreover, use of large packets decreases the amount of RTP/UDP/IP header overhead. For low bit-rate video, the use of large packets means that sometimes up to few pictures should be encapsulated in one packet.

However, loss of a packet including many coded pictures would have drastic consequences for visual quality, as there is practically no other way to conceal a loss of an entire picture than to repeat the previous one. One way to construct relatively large packets and maintain possibilities for successful loss concealment is to construct MTAPs that contain interleaved slices from several pictures. An MTAP should not contain spatially adjacent slices from the same picture or spatially overlapping slices from any picture. If a packet is lost, it is likely that a lost slice is surrounded by spatially adjacent slices of the same picture and spatially corresponding slices of the temporally previous and succeeding pictures. Consequently, concealment of the lost slice is likely to be relatively successful.

12.7. Robust Packet Scheduling in Video Streaming

Robust packet scheduling has been implemented with MPEG-4 Part 2 and simulated in a wireless streaming environment [21]. There is no technical reason why similar or better results could not be achievable with H.264.

Streaming clients typically have a receiver buffer that is capable of storing a relatively large amount of data. Initially, when a streaming session is established, a client does not start playing the stream back immediately. Rather, it typically buffers the incoming data for a few seconds. This buffering helps maintain continuous playback, as, in case of occasional increased transmission delays or network throughput drops, the client can decode and play buffered data. Otherwise, without initial buffering, the client has to freeze the display, stop decoding, and wait for incoming data. The buffering is also necessary for either automatic or selective retransmission in any protocol level. If any part of a picture is lost, a retransmission mechanism may be used to resend the lost data. If the retransmitted data is received before its scheduled decoding or playback time, the loss is recovered perfectly. Coded pictures can be ranked according to their importance in the subjective quality of the decoded sequence. For example, non-reference pictures, such as conventional B pictures, are subjectively least important, as their absence does not affect decoding of any other pictures. In addition to non-reference pictures, the ITU-T H.264 | ISO/IEC 14496-10 standard includes a temporal scalability method called sub-sequences [22]. Subjective ranking can also be made on coded slice data partition or slice group basis. Coded slices and coded slice data partitions that are subjectively the most important can be sent earlier than their decoding order indicates, whereas coded slices and coded slice data partitions that are subjectively the least important can be sent later than their natural coding order indicates. Consequently, any retransmitted parts of the most important slices

and coded slice data partitions are more likely to be received before their scheduled decoding or playback time compared to the least important slices and slice data partitions.

13. Informative Appendix: Rationale for Decoding Order Number

13.1. Introduction

The Decoding Order Number (DON) concept was introduced mainly to enable efficient multi-picture slice interleaving (see section 12.6) and robust packet scheduling (see section 12.7). In both of these applications, NAL units are transmitted out of decoding order. DON indicates the decoding order of NAL units and should be used in the receiver to recover the decoding order. Example use cases for efficient multi-picture slice interleaving and for robust packet scheduling are given in sections 13.2 and 13.3, respectively. Section 13.4 describes the benefits of the DON concept in error resiliency achieved by redundant coded pictures. Section 13.5 summarizes considered alternatives to DON and justifies why DON was chosen to this RTP payload specification.

13.2. Example of Multi-Picture Slice Interleaving

An example of multi-picture slice interleaving follows. A subset of a coded video sequence is depicted below in output order. R denotes a reference picture, N denotes a non-reference picture, and the number indicates a relative output time.

```
... R1 N2 R3 N4 R5 ...
```

The decoding order of these pictures from left to right is as follows:

```
... R1 R3 N2 R5 N4 ...
```

The NAL units of pictures R1, R3, N2, R5, and N4 are marked with a DON equal to 1, 2, 3, 4, and 5, respectively.

Each reference picture consists of three slice groups that are scattered as follows (a number denotes the slice group number for each macroblock in a QCIF frame):

```
0 1 2 0 1 2 0 1 2 0 1
2 0 1 2 0 1 2 0 1 2 0
1 2 0 1 2 0 1 2 0 1 2
0 1 2 0 1 2 0 1 2 0 1
2 0 1 2 0 1 2 0 1 2 0
1 2 0 1 2 0 1 2 0 1 2
0 1 2 0 1 2 0 1 2 0 1
2 0 1 2 0 1 2 0 1 2 0
1 2 0 1 2 0 1 2 0 1 2
```

For the sake of simplicity, we assume that all the macroblocks of a slice group are included in one slice. Three MTAPs are constructed from three consecutive reference pictures so that each MTAP contains three aggregation units, each of which contains all the macroblocks from one slice group. The first MTAP contains slice group 0 of picture R1, slice group 1 of picture R3, and slice group 2 of picture R5. The second MTAP contains slice group 1 of picture R1, slice group 2 of picture R3, and slice group 0 of picture R5. The third MTAP contains slice group 2 of picture R1, slice group 0 of picture R3, and slice group 1 of picture R5. Each non-reference picture is encapsulated into an STAP-B.

Consequently, the transmission order of NAL units is the following:

```
R1, slice group 0, DON 1, carried in MTAP,   RTP SN: N
R3, slice group 1, DON 2, carried in MTAP,   RTP SN: N
R5, slice group 2, DON 4, carried in MTAP,   RTP SN: N
R1, slice group 1, DON 1, carried in MTAP,   RTP SN: N+1
R3, slice group 2, DON 2, carried in MTAP,   RTP SN: N+1
R5, slice group 0, DON 4, carried in MTAP,   RTP SN: N+1
R1, slice group 2, DON 1, carried in MTAP,   RTP SN: N+2
R3, slice group 1, DON 2, carried in MTAP,   RTP SN: N+2
R5, slice group 0, DON 4, carried in MTAP,   RTP SN: N+2
N2,                DON 3, carried in STAP-B, RTP SN: N+3
N4,                DON 5, carried in STAP-B, RTP SN: N+4
```

The receiver is able to organize the NAL units back in decoding order based on the value of DON associated with each NAL unit.

If one of the MTAPs is lost, the spatially adjacent and temporally co-located macroblocks are received and can be used to conceal the loss efficiently. If one of the STAPs is lost, the effect of the loss does not propagate temporally.

13.3. Example of Robust Packet Scheduling

An example of robust packet scheduling follows. The communication system used in the example consists of the following components in the order that the video is processed from source to sink:

- o camera and capturing
- o pre-encoding buffer
- o encoder
- o encoded picture buffer
- o transmitter
- o transmission channel
- o receiver
- o receiver buffer
- o decoder
- o decoded picture buffer
- o display

The video communication system used in the example operates as follows. Note that processing of the video stream happens gradually and at the same time in all components of the system. The source video sequence is shot and captured to a pre-encoding buffer. The pre-encoding buffer can be used to order pictures from sampling order to encoding order or to analyze multiple uncompressed frames for bit rate control purposes, for example. In some cases, the pre-encoding buffer may not exist; instead, the sampled pictures are encoded right away. The encoder encodes pictures from the pre-encoding buffer and stores the output; i.e., coded pictures, to the encoded picture buffer. The transmitter encapsulates the coded pictures from the encoded picture buffer to transmission packets and sends them to a receiver through a transmission channel. The receiver stores the received packets to the receiver buffer. The receiver buffering process typically includes buffering for transmission delay jitter. The receiver buffer can also be used to recover correct decoding order of coded data. The decoder reads coded data from the receiver buffer and produces decoded pictures as output into the decoded picture buffer. The decoded picture buffer is used to recover the output (or display) order of pictures. Finally, pictures are displayed.

In the following example figures, I denotes an IDR picture, R denotes a reference picture, N denotes a non-reference picture, and the number after I, R, or N indicates the sampling time relative to the previous IDR picture in decoding order. Values below the sequence of pictures indicate scaled system clock timestamps. The system clock is initialized arbitrarily in this example, and time runs from left to right. Each I, R, and N picture is mapped into the same timeline compared to the previous processing step, if any, assuming that

encoding, transmission, and decoding take no time. Thus, events happening at the same time are located in the same column throughout all example figures.

A subset of a sequence of coded pictures is depicted below in sampling order.

```
...  N58 N59 I00 N01 N02 R03 N04 N05 R06 ... N58 N59 I00 N01 ...
... --|---|---|---|---|---|---|---|---|- ... -|---|---|---|- ...
...  58  59  60  61  62  63  64  65  66  ... 128 129 130 131 ...
```

Figure 16. Sequence of pictures in sampling order

The sampled pictures are buffered in the pre-encoding buffer to arrange them in encoding order. In this example, we assume that the non-reference pictures are predicted from both the previous and the next reference picture in output order, except for the non-reference pictures immediately preceding an IDR picture, which are predicted only from the previous reference picture in output order. Thus, the pre-encoding buffer has to contain at least two pictures, and the buffering causes a delay of two picture intervals. The output of the pre-encoding buffering process and the encoding (and decoding) order of the pictures are as follows:

```
... N58 N59 I00 R03 N01 N02 R06 N04 N05 ...
... -|---|---|---|---|---|---|---|---|- ...
... 60  61  62  63  64  65  66  67  68  ...
```

Figure 17. Re-ordered pictures in the pre-encoding buffer

The encoder or the transmitter can set the value of DON for each picture to a value of DON for the previous picture in decoding order plus one.

For the sake of simplicity, let us assume that:

- the frame rate of the sequence is constant,
- each picture consists of only one slice,
- each slice is encapsulated in a single NAL unit packet,
- there is no transmission delay, and
- pictures are transmitted at constant intervals (that is, 1 / frame rate).

When pictures are transmitted in decoding order, they are received as follows:

```
... N58 N59 I00 R03 N01 N02 R06 N04 N05 ...
... -|---|---|---|---|---|---|---|---|- ...
... 60  61  62  63  64  65  66  67  68  ...
```

Figure 18. Received pictures in decoding order

The OPTIONAL sprop-interleaving-depth MIME type parameter is set to 0, as the transmission (or reception) order is identical to the decoding order.

The decoder has to buffer for one picture interval initially in its decoded picture buffer to organize pictures from decoding order to output order as depicted below:

```
... N58 N59 I00 N01 N02 R03 N04 N05 R06 ...
... -|---|---|---|---|---|---|---|---|- ...
... 61  62  63  64  65  66  67  68  69  ...
```

Figure 19. Output order

The amount of required initial buffering in the decoded picture buffer can be signaled in the buffering period SEI message or with the num_reorder_frames syntax element of H.264 video usability information. num_reorder_frames indicates the maximum number of frames, complementary field pairs, or non-paired fields that precede any frame, complementary field pair, or non-paired field in the sequence in decoding order and that follow it in output order. For the sake of simplicity, we assume that num_reorder_frames is used to indicate the initial buffer in the decoded picture buffer. In this example, num_reorder_frames is equal to 1.

It can be observed that if the IDR picture I00 is lost during transmission and a retransmission request is issued when the value of the system clock is 62, there is one picture interval of time (until the system clock reaches timestamp 63) to receive the retransmitted IDR picture I00.

Let us then assume that IDR pictures are transmitted two frame intervals earlier than their decoding position; i.e., the pictures are transmitted as follows:

```
...  I00 N58 N59 R03 N01 N02 R06 N04 N05 ...
... --|---|---|---|---|---|---|---|---|- ...
...  62  63  64  65  66  67  68  69  70  ...
```

Figure 20. Interleaving: Early IDR pictures in sending order

The OPTIONAL sprop-interleaving-depth MIME type parameter is set equal to 1 according to its definition. (The value of sprop-interleaving-depth in this example can be derived as follows: Picture I00 is the only picture preceding picture N58 or N59 in transmission order and following it in decoding order. Except for pictures I00, N58, and N59, the transmission order is the same as the decoding order of pictures. As a coded picture is encapsulated into exactly one NAL unit, the value of sprop-interleaving-depth is equal to the maximum number of pictures preceding any picture in transmission order and following the picture in decoding order.)

The receiver buffering process contains two pictures at a time according to the value of the sprop-interleaving-depth parameter and orders pictures from the reception order to the correct decoding order based on the value of DON associated with each picture. The output of the receiver buffering process is as follows:

```
... N58 N59 I00 R03 N01 N02 R06 N04 N05 ...
... -|---|---|---|---|---|---|---|---|- ...
... 63  64  65  66  67  68  69  70  71  ...
```

Figure 21. Interleaving: Receiver buffer

Again, an initial buffering delay of one picture interval is needed to organize pictures from decoding order to output order, as depicted below:

```
... N58 N59 I00 N01 N02 R03 N04 N05 ...
... -|---|---|---|---|---|---|---|---|- ...
... 64  65  66  67  68  69  70  71  ...
```

Figure 22. Interleaving: Receiver buffer after reordering

Note that the maximum delay that IDR pictures can undergo during transmission, including possible application, transport, or link layer retransmission, is equal to three picture intervals. Thus, the

loss resiliency of IDR pictures is improved in systems supporting retransmission compared to the case in which pictures were transmitted in their decoding order.

13.4. Robust Transmission Scheduling of Redundant Coded Slices

A redundant coded picture is a coded representation of a picture or a part of a picture that is not used in the decoding process if the corresponding primary coded picture is correctly decoded. There should be no noticeable difference between any area of the decoded primary picture and a corresponding area that would result from application of the H.264 decoding process for any redundant picture in the same access unit. A redundant coded slice is a coded slice that is a part of a redundant coded picture.

Redundant coded pictures can be used to provide unequal error protection in error-prone video transmission. If a primary coded representation of a picture is decoded incorrectly, a corresponding redundant coded picture can be decoded. Examples of applications and coding techniques using the redundant codec picture feature include the video redundancy coding [23] and the protection of "key pictures" in multicast streaming [24].

One property of many error-prone video communications systems is that transmission errors are often bursty. Therefore, they may affect more than one consecutive transmission packets in transmission order. In low bit-rate video communication, it is relatively common that an entire coded picture can be encapsulated into one transmission packet. Consequently, a primary coded picture and the corresponding redundant coded pictures may be transmitted in consecutive packets in transmission order. To make the transmission scheme more tolerant of bursty transmission errors, it is beneficial to transmit the primary coded picture and redundant coded picture separated by more than a single packet. The DON concept enables this.

13.5. Remarks on Other Design Possibilities

The slice header syntax structure of the H.264 coding standard contains the frame_num syntax element that can indicate the decoding order of coded frames. However, the usage of the frame_num syntax element is not feasible or desirable to recover the decoding order, due to the following reasons:

- The receiver is required to parse at least one slice header per coded picture (before passing the coded data to the decoder).

- Coded slices from multiple coded video sequences cannot be interleaved, as the frame number syntax element is reset to 0 in each IDR picture.

- The coded fields of a complementary field pair share the same value of the frame_num syntax element. Thus, the decoding order of the coded fields of a complementary field pair cannot be recovered based on the frame_num syntax element or any other syntax element of the H.264 coding syntax.

The RTP payload format for transport of MPEG-4 elementary streams [25] enables interleaving of access units and transmission of multiple access units in the same RTP packet. An access unit is specified in the H.264 coding standard to comprise all NAL units associated with a primary coded picture according to subclause 7.4.1.2 of [1]. Consequently, slices of different pictures cannot be interleaved, and the multi-picture slice interleaving technique (see section 12.6) for improved error resilience cannot be used.

14. Acknowledgements

The authors thank Roni Even, Dave Lindbergh, Philippe Gentric, Gonzalo Camarillo, Gary Sullivan, Joerg Ott, and Colin Perkins for careful review.

15. References

15.1. Normative References

[1] ITU-T Recommendation H.264, "Advanced video coding for generic audiovisual services", May 2003.

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

[3] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

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

[5] Handley, M. and V. Jacobson, "SDP: Session Description Protocol", RFC 2327, April 1998.

[6] Josefsson, S., "The Base16, Base32, and Base64 Data Encodings", RFC 3548, July 2003.

[7] Rosenberg, J. and H. Schulzrinne, "An Offer/Answer Model with Session Description Protocol (SDP)", RFC 3264, June 2002.

15.2. Informative References

[8] "Draft ITU-T Recommendation and Final Draft International Standard of Joint Video Specification (ITU-T Rec. H.264 | ISO/IEC 14496-10 AVC)", available from http://ftp3.itu.int/av-arch/jvt-site/2003_03_Pattaya/JVT-G050r1.zip, May 2003.

[9] Luthra, A., Sullivan, G.J., and T. Wiegand (eds.), Special Issue on H.264/AVC. IEEE Transactions on Circuits and Systems on Video Technology, July 2003.

[10] Bormann, C., Cline, L., Deisher, G., Gardos, T., Maciocco, C., Newell, D., Ott, J., Sullivan, G., Wenger, S., and C. Zhu, "RTP Payload Format for the 1998 Version of ITU-T Rec. H.263 Video (H.263+)", RFC 2429, October 1998.

[11] ISO/IEC IS 14496-2.

[12] Wenger, S., "H.26L over IP", IEEE Transaction on Circuits and Systems for Video technology, Vol. 13, No. 7, July 2003.

[13] Wenger, S., "H.26L over IP: The IP Network Adaptation Layer", Proceedings Packet Video Workshop 02, April 2002.

[14] Stockhammer, T., Hannuksela, M.M., and S. Wenger, "H.26L/JVT Coding Network Abstraction Layer and IP-based Transport" in Proc. ICIP 2002, Rochester, NY, September 2002.

[15] ITU-T Recommendation H.241, "Extended video procedures and control signals for H.300 series terminals", 2004.

[16] Schulzrinne, H. and S. Casner, "RTP Profile for Audio and Video Conferences with Minimal Control", STD 65, RFC 3551, July 2003.

[17] ITU-T Recommendation H.223, "Multiplexing protocol for low bit rate multimedia communication", July 2001.

[18] Rosenberg, J. and H. Schulzrinne, "An RTP Payload Format for Generic Forward Error Correction", RFC 2733, December 1999.

[19] Stockhammer, T., Wiegand, T., Oelbaum, T., and F. Obermeier, "Video Coding and Transport Layer Techniques for H.264/AVC-Based Transmission over Packet-Lossy Networks", IEEE International Conference on Image Processing (ICIP 2003), Barcelona, Spain, September 2003.

[20] Varsa, V. and M. Karczewicz, "Slice interleaving in compressed video packetization", Packet Video Workshop 2000.

[21] Kang, S.H. and A. Zakhor, "Packet scheduling algorithm for wireless video streaming," International Packet Video Workshop 2002.

[22] Hannuksela, M.M., "Enhanced concept of GOP", JVT-B042, available http://ftp3.itu.int/av-arch/video-site/0201_Gen/JVT-B042.doc, January 2002.

[23] Wenger, S., "Video Redundancy Coding in H.263+", 1997 International Workshop on Audio-Visual Services over Packet Networks, September 1997.

[24] Wang, Y.-K., Hannuksela, M.M., and M. Gabbouj, "Error Resilient Video Coding Using Unequally Protected Key Pictures", in Proc. International Workshop VLBV03, September 2003.

[25] van der Meer, J., Mackie, D., Swaminathan, V., Singer, D., and P. Gentric, "RTP Payload Format for Transport of MPEG-4 Elementary Streams", RFC 3640, November 2003.

[26] Baugher, M., McGrew, D., Naslund, M., Carrara, E., and K. Norrman, "The Secure Real-time Transport Protocol (SRTP)", RFC 3711, March 2004.

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

[28] Handley, M., Perkins, C., and E. Whelan, "Session Announcement Protocol", RFC 2974, October 2000.

[29] ISO/IEC 14496-15: "Information technology - Coding of audio-visual objects - Part 15: Advanced Video Coding (AVC) file format".

[30] Castagno, R. and D. Singer, "MIME Type Registrations for 3rd Generation Partnership Project (3GPP) Multimedia files", RFC 3839, July 2004.

Authors' Addresses

Stephan Wenger
TU Berlin / Teles AG
Franklinstr. 28-29
D-10587 Berlin
Germany

Phone: +49-172-300-0813
EMail: stewe@stewe.org

Miska M. Hannuksela
Nokia Corporation
P.O. Box 100
33721 Tampere
Finland

Phone: +358-7180-73151
EMail: miska.hannuksela@nokia.com

Thomas Stockhammer
Nomor Research
D-83346 Bergen
Germany

Phone: +49-8662-419407
EMail: stockhammer@nomor.de

Magnus Westerlund
Multimedia Technologies
Ericsson Research EAB/TVA/A
Ericsson AB
Torshamsgatan 23
SE-164 80 Stockholm
Sweden

Phone: +46-8-7190000
EMail: magnus.westerlund@ericsson.com

David Singer
QuickTime Engineering
Apple
1 Infinite Loop MS 302-3MT
Cupertino
CA 95014
USA

Phone +1 408 974-3162
EMail: singer@apple.com

Full Copyright Statement

Intellectual Property

Acknowledgement

Funding for the RFC Editor function is currently provided by the Internet Society.