Journal of Zhejiang University SCIENCE A
ISSN 1009-3095 (Print); ISSN 1862-1775 (Online)
www.zju.edu.cn/jzus; www.springerlink.com
E-mail: jzus@zju.edu.cn

# RTP payload format for H.264/SVC scalable video coding

WENGER Stephan, WANG Ye-kui, HANNUKSELA Miska M.

(*Nokia Research Center, Tampere 33721, Finland*)

E-mail: {stephan.wenger; ye-kui.wang; miska.hannuksela}@nokia.com

Received Dec. 3, 2005; revision accepted Feb. 25, 2006

**Abstract:** The scalable extension of H.264/AVC, known as scalable video coding or SVC, is currently the main focus of the Joint Video Team's work. In its present working draft, the higher level syntax of SVC follows the design principles of H.264/AVC. Self-contained network abstraction layer units (NAL units) form natural entities for packetization. The SVC specification is by no means finalized yet, but nevertheless the work towards an optimized RTP payload format has already started. RFC 3984, the RTP payload specification for H.264/AVC has been taken as a starting point, but it became quickly clear that the scalable features of SVC require adaptation in at least the areas of capability/operation point signaling and documentation of the extended NAL unit header. This paper first gives an overview of the history of scalable video coding, and then reviews the video coding layer (VCL) and NAL of the latest SVC draft specification. Finally, it discusses different aspects of the draft SVC RTP payload format, including the design criteria, use cases, signaling and payload structure.

**Key words:** H.264, Advanced video coding (AVC), Scalable video coding (SVC), Scalability, Real-time transport protocol (RTP), Packetization

**doi:**10.1631/jzus.2006.A0657 **Document code:** A **CLC number:** TN919.8

## INTRODUCTION

Scalability in video coding and transmission, as a concept, is known for many decades. As one of the earliest examples, Jones' paper from 1979 (Jones, 1979) can be cited, which Cliff Reader, in his well known white paper "History of Video Compression" (Reader, 2002), describes as "[…] Jones exploited the frequency property of the transform to reduce resolution in the changed areas by transmitting only subsets of the transform coefficients. This was a kind of early form of transform-domain spatial scalability."

In international standards, temporal scalability was first introduced in MPEG-1 (ISO/IEC JTC1, 1992) with the B picture concept. MPEG-1's B pictures share the two unrelated properties of bi-directional prediction and their non-reference nature. The reason for this design choice is neither immediately intuitive, nor fully explained in any publication we are aware of. However, it is commonly believed that the complexity of the B-picture decoding process—B pictures are roughly twice as expensive to decode as P pictures—has led to concerns about the complexity provisioning of the decoding. As B pictures are never used for reference, a cycle-starving decoder can choose to skip their decoding, thereby staying within its cycle limit. By coincidence, MPEG created a scalable video coding scheme in which the B pictures can form a temporal enhancement layer, when being transported in an independent transport channel.

MPEG-2 (ITU-T and ISO/IEC JTC1, 1994) retained this property of B-picture based temporal scalability, but added, in its rarely (if ever) used scalable profile, two more forms of scalability: spatial and SNR. When spatial scalability is in use, an enhancement layer employs for reference the reconstructed pictures of a base layer, coded at a lower spatial resolution. SNR scalability uses a similar mechanism, except that the resolutions of base and enhancement layers are identical. Most commonly, an SNR enhancement layer uses finer quantization to improve the fidelity of the reconstructed enhancement picture. All above mentioned technologies are commonly categorized as coarse granularity scalability (CGS), as the granularity of scalability lies only in the form of decoding/skipping of all bits of a given enhancement layer (and all enhancement layers that use the skipped enhancement layer for reference).

MPEG-4 (ISO/IEC JTC1, 1998) retains all the mentioned CGS mechanisms, and adds a concept known as Fine Granularity Scalability (Li, 2001). FGS allows the reconstruction of slice data where a certain number of bits at the end of each slice may be missing. The most commonly cited reason for removing bits in FGS enhancement layers is the need to adhere to congestion control principles in IP networks. However, current congestion control algorithms require bandwidth adaptation in a TCP friendly way, which overstretches the ability of bandwidth adaptation of MPEG-4's FGS mechanisms by far. Consequently, the profile in question is not in any significant practical use.

When the Video Coding Experts Group, formally known as ITU-T SG16 Q.6, discussed the requirements for what became later H.264/AVC, scalability was explicitly excluded. Perhaps the main reason for this design choice was that, at the time, no proposals for a radically new scalability mechanism were available that overcame the shortcoming of existing scalable video coding technologies. Clearly, at the time it was felt that the lack of success of scalability as a concept would not warrant the design effort. This rationale was followed once again two years later after the formation of the Joint Video Team in December 2001 and until mid 2004. Therefore, H.264/AVC, as specified today, does not explicitly mention scalable video coding, nor offers coding tools specifically designed to support scalability. The concepts of non-reference pictures (including non-reference B pictures) and sub-sequences (Tian *et al*., 2005a), however, implicitly allow for temporal scalability utilizing various picture types.

With the finalization of H.264/AVC version 1, and the work on the fidelity range extensions coming to a close, the focus of the research and standardization community, once again, focused on scalable video coding. However, this time the committee set very high hurdles for the acceptance of proposals. A scalable mechanism is only considered for standardization when it has no, or only a very small negative impact on the coding efficiency, and offers reasonable complexity. Any negative impact on the coding efficiency must be well balanced against positive aspects for at least one use case. These constraints led to concepts not previously found in standards, e.g. the proposed single loop decoding. In January 2005, a first working draft of the SVC extensions was created (Joint Video Team, 2005a). This draft (and its successors) is commonly known as SVC. Procedurally, SVC is planned to take the form of Annex G to ITU-T Rec. H.264 and ISO/IEC 14496-10, and is planned to be backward compatible with the existing versions of H.264/AVC. In particular, the SVC base layer should be an H.264/AVC compliant stream. Technical maturity will be reached perhaps around mid 2006, for which timeframe the ISO/IEC Committee Draft is expected.

In the following Section 2, we first present the SVC design principles followed by the design criteria of the SVC RTP payload format in Section 3. The use cases, signaling and payload structure of the RTP payload format are discussed in Sections 4 to 7, respectively. The special handling of FGS data is sketched in Section 8. Finally Section 9 draws the conclusions.

## SVC DESIGN PRINCIPLES

### Video coding layer

A side note to start: the term "Layer" in Video Coding Layer and Network abstraction layer refers to a conceptual distinction, and is closely related to syntax layers (block, macroblock, slice, … layers). It should not be confused with base and enhancement layers.

At the time of writing, the most recent draft of SVC is available for public review in (Joint Video Team, 2005b), and the associated test model can be found in (Joint Video Team, 2005c). The latter contains two separate documents, one of which is what comes closest to tutorial/introductory information to the SVC concepts and is recommended for an initial study. The SVC working draft alone is already more than 130 pages in size, by no means complete, and non-explanatory in nature.

As previous scalable technologies, SVC also distinguishes between a base layer and an enhancement layer. In the SVC case, the base layer is anticipated to conform to a non-scalable profile of H.264/AVC. The enhancement layers conform to the SVC specification.

A scalable video bitstream contains the non-scalable base layer and one or more enhancement layers. An enhancement layer may enhance the temporal resolution (i.e. the frame rate), the spatial resolution, or the quality of the video content represented by the lower layer or part of it. According to the SVC draft specification, the scalable layers can be aggregated to a single transport stream, or transported independently.

The concepts of video coding layer (VCL) and network abstraction layer (NAL) are inherited from H.264/AVC. The VCL contains the signal processing functionality of the codec; mechanisms such as transform, quantization, entropy coding, intra prediction, motion-compensated inter prediction, loop filter, and (as a new addition) inter-layer prediction. A coded picture of a base or enhancement layer consists of one or more slices. The network abstraction layer (NAL) encapsulates each slice generated by the VCL into typical NAL units.

Each SVC layer is formed by NAL units, representing the coded video bits of the layer. SVC specifies the decoding order of these NAL units. An RTP (Schulzrinne *et al.*, 2003) stream carrying only one layer would carry NAL units belonging to that layer only. An RTP stream carrying a complete scalable video bit stream would carry NAL units of a base layer and one or more enhancement layers. It is currently under discussion whether the SVC payload format needs to support other operation points, such as carrying NAL units belonging to more than one enhancement layers without a base layer.

In some cases, the bit rate of a given enhancement layer can be reduced by truncating bits from individual NAL units. Truncation leads to a graceful degradation of the reproduced enhancement layer's video quality. This concept is known as fine-grained (granularity) scalability (FGS).

In its latest draft, SVC inherits H.264/AVC's temporal scalability mechanisms unchanged. A so-called hierarchical B pictures coding structure is used. Earlier versions of the SVC draft specification supported a motion-compensated temporal filtering (MCTF) update step (Ohm, 2005). In October 2005, it was decided to withdraw that step from the coding loop, as it was felt to be sufficient to apply it in the form of a pre-filter on the encoder side.

SVC supports the other CGS concepts, namely spatial and SNR enhancement layers, in a form roughly comparable to earlier standards. Three additional features, previously unknown, shall at least be mentioned briefly:

Inter layer prediction: Intra texture, motion and residual data can be predicted from layers other than the currently reconstructed layer and the next layer in the hierarchy. In other words, it is possible to use information from layers "far away" in the hierarchy for prediction.

Single-loop decoding: Traditionally, each reconstruction of each base and enhancement layer requires a full run of a decoder loop—the compressed video data is reconstructed into the spatial domain before higher enhancement layers can be reproduced. In SVC, when restricting the bitstream somewhat by constraining the Intra prediction, single-loop decoding is possible. Here, coded data of the base layer and all applicable enhancement layers are considered jointly. The decoder needs to implement motion compensation and full picture reconstruction only for the scalable layer desired for playback, hence the decoding complexity is greatly reduced (Schwarz *et al.*, 2005).

Cropping and zooming: The spatial scalability has been generalized to enable the base layer to be a cropped and zoomed version of the enhancement layer.

The quantization and entropy coding modules have been designed to provide FGS capability. In a coding mode known as progressive refinement, successive refinements of the transform coefficients are encoded by repeatedly decreasing the quantization step size and applying a "cyclic" entropy coding akin to sub-bitplane coding.

## Network abstraction layer

H.264/AVC's network abstraction layer key properties are retained in SVC, and therefore the forthcoming RTP payload format specification can follow the same design principles as RFC 3984 (Wenger *et al.*, 2005), the RTP payload format for H.264/AVC video. In particular:

(1) NAL units form the basic structure of an SVC bit stream. They are independently processable

entities, and, in most cases, should be carried as a single transport unit (packet) by the underlying transport infrastructure. Please see Section 5.8 of RFC3984 for exceptions. The relevance of each NAL units for the decoding process is indicated by the easily parseable NRI bit field; see below.

(2) The Parameter Set concept is used to carry most information pertaining to more than one NAL unit. Enhancement layers may share the same picture or sequence parameter sets as other base or enhancement layers, or may refer to different ones.

(3) In an RTP environment, framing is implemented by using the natural RTP packet boundaries, or payload-internal mechanisms. No Annex B bit stream is formed, and no start codes are required.

An SVC NAL unit consists of a header of one, two or three bytes, and the payload byte string. The header indicates the type of the NAL unit, the (potential) presence of bit errors or syntax violations in the NAL unit payload, and information on the relative importance of the NAL unit for the decoding process. Optionally, when the header is three bytes in size, scalable layer decoding dependency information is also included.

The NAL unit header is designed to co-serve as the payload header of an RTP payload format. The syntax and semantics of the NAL unit header, as specified in (Joint Video Team, 2005b), are summarized below.

The first octet of the NAL unit header shares the syntax with the one presented in H.264/AVC and RFC 3984, as shown in Fig.1.

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

**Fig.1 NAL unit header (first octet)**

The *F*, or *forbidden_zero_bit* was included to support wireline/wireless gateways. The H.264 specification declares a value of 1 as a syntax violation. Instead of disposing of NAL units with known bit errors that occurred on a wireless link, a gateway may indicate the presence of errors utilizing this bit, and forwarding the erroneous NAL unit in an RTP packet.

*NRI*, or *nal_ref_idc* signals the relative importance of that NAL unit. A value of 00 indicates that the content of the NAL unit is not used to reconstruct reference pictures for inter picture prediction. Such NAL units can be discarded without risking the integrity of the reference pictures in the same layer. Values greater than 00 indicate that the decoding of the NAL unit is required to maintain the integrity of the reference pictures. For SVC enhancement layers, a slice or slice data partitioning NAL unit with an NRI value of 11 indicates that it belongs to a key picture. A key picture can be understood as the first picture in decoding order within each group of pictures (GOP). Note that in SVC the key picture may either be intra-coded or inter-coded using the previous key picture for inter prediction reference.

The *nal_unit_type* specifies the NAL unit payload type as defined in Table 7-1 of (Joint Video Team, 2005b). For a reference of all currently defined NAL unit types and their semantics, please refer to Section 7.4.1 in (Joint Video Team, 2005b).

Common NAL unit types are such as those for IDR and non-IDR Slices, (in-band) sequence and picture parameter sets, data partitions of various types, and a few other data structures of lower importance. In H.264/AVC, the NAL unit types 20 and 21 (among others) were reserved for future extensions. SVC uses these two NAL unit types and indicates the presence of one more octet (as shown in Fig.2) that is helpful from a transport viewpoint.

```
+ − − − − − − − − − − − − − − − − +
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
+ − + − + − + − + − + − + − + − +
|          PRID         | D | E |
+ − − − − − − − − − − − − − − − − +
```

**Fig.2 NAL unit header (second octet)**

The *simple_priority_id* (*PRID*) carries a priority identifier for the NAL unit. When the *extension_flag* is equal to 0, *PRID* is used for inferring the values of *dependency_id*, *temporal_level*, and *quality_level*.

A value of 1 in the *discardable_flag* (*D*) indicates that the decoding of the NAL unit is not required for the decoding process of NAL units of “higher” layers. Therefore, an NAL unit with *D*=1 can be discarded without risking the integrity of higher scalable layers.

The *extension_flag* (*E*), when set, signals the presence of a third octet in the NAL unit header, as shown in Fig.3.

```
+---------------+
|0|1|2|3|4|5|6|7|
+-+-+-+-+-+-+-+-+
|  TL  | DID | QL |
+---------------+
```

**Fig.3 NAL unit header (third octet)**

This octet adds more dependency information. Specifically, the *temporal_level* (*TL*) is used to indicate temporal scalability layer. A layer consisting of NAL units that carry pictures with a smaller *TL* value has a lower frame rate. The *dependency_id* (*DID*) field can be used to indicate the inter-layer coding dependency hierarchy. At any temporal location, a picture of a lower *DID* value may be used for inter-layer prediction for coding of a picture with a higher *DID* value.

The *quality_level* (*QL*) indicates the FGS layer hierarchy. At any temporal location and with identical *dependency_id* value, an FGS picture with *quality_level* value equal to *QL* uses the FGS picture or base quality picture (the non-FGS picture when *QL*-1 =0) with *quality_level* value equal to *QL*-1 for inter-layer prediction. When *QL* is larger than 0, the NAL unit contains an FGS slice or its part.

## DESIGN CRITERIA OF THE PAYLAOD FORMAT

The SVC RTP payload draft, as presented in (Wenger and Wang, 2005), does not contain a detailed discussion on the requirements. Nevertheless, the authors were led by a number of constraints that may explain the design choices made. Some of these constraints are:

1. Backward compatibility with RFC 3984

As SVC is a backward compatible extension of H.264/AVC, the same should be the case for its packetization. In particular, it should be possible—or even required—to transport the base layer utilizing RFC 3984. Only if this is achieved, RFC 3984-aware legacy devices are still capable of utilizing an SVC base layer in an RTP environment.

2. Re-use of RFC 3984's mechanisms wherever possible

Many person-years of standardization and implementation work have been allocated to the design of RFC 3984. Re-using these results has positive commercial implications, as experience with the packetization and its technical and commercial environment is already available.

3. Focus on the use case of Internet transport

Please note that we did not write "over IP", but "Internet". The distinction is both technical and procedural. The Internet, currently and in most cases, is considered a best-effort IP network offering end-to-end connectivity. Congestion control and security are major problems and have to be addressed by every specification under IETF control. Intelligence in the network is not considered a desirable feature; whenever possible, intelligence should be implemented at the endpoints.

As it is true for most RTP payload specifications that are published in the form of an RFC, compromises have to be made between the support of all operation points the media standard committee considers useful, and the constraints of the IETF standardization environment—even if this may rule out a few use cases either side considers valuable. The current SVC payload Internet Draft already takes most of the mentioned constraints into account and, therefore, covers only a subset of the use cases anticipated by MPEG (ISO/IEC MPEG, 2005) and VCEG (ITU-T VCEG, 2005).

## SCALABLE VIDEO OVER THE INTERNET

From the time of the development of IP multicast around 1988, and the first experiments of what became the MBONE in 1992, researches have viewed scalable media codecs as valuable complements to IP multicast. However, it took until 1996 and McCanne's famous paper (Jacobson *et al*., 1996) to describe a design that actually makes use of the features of both technologies. This paper, and similar (mostly unpublished) work by other research groups, triggered a flurry of activities towards scalability beyond that available in MPEG-1 and MPEG-2, and specifically lead to H.263's Annex O scalability mechanisms (ITU-T, 1998) (The MPEG standards were, at the time, not considered a valuable operation point for Internet-based video as they were perceived requiring too much bandwidth). However, to the best of our knowledge, even those activities did not lead to significant product deployments.

We see three main use cases for scalable video over the Internet:

(1) Multicast/broadcast of video data to receivers with heterogeneous connectivity, following McCanne's concept.

(2) Multicast/Broadcast on the server side, but aggregation middleboxes in the network, in order to avoid network address translator (NAT) and firewall problems.

(3) Low complexity bandwidth adaptation in servers for pre-coded content.

Fig.4 depicts use case 1. A server carries one base layer and three enhancement layers, forming a hierarchy. Terminals T1, T2, and T3 are connected to the server through the Internet, over links that allow for certain maximum bit rates. The capacity of the links and the bit rate demands of the streams are illustrated by the line width of the connections—the wider a line, the higher the bit rate. The end-to-end capacity is a function of both the connectivity of the endpoint and the congestion of each link. Therefore, the picture should be viewed as a snapshot of a configuration at a given time—the connectivity of each terminal may change frequently with the changes of the congestion situation. Note also that for the sake of simplicity, we assume uncongested links from the server to the backbone.

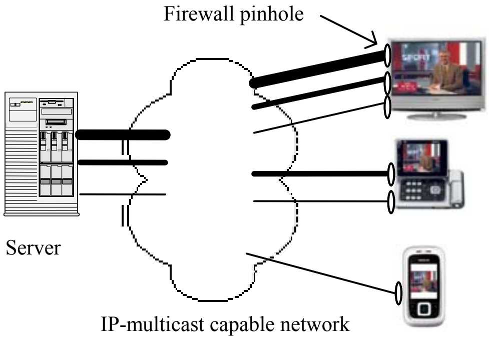

**Fig.4 Receiver driven layered multicast**

According to McCanne's concepts, each layer is transported in its own IP multicast group and terminals subscribe to layers utilizing IP multicast mechanisms, namely IGMP (Fenner, 1997). This implies that one terminal may have to subscribe to many IP multicast groups for the best possible quality. While, considering Internet technologies in their purest form, this is not a problem and actually desirable, practical constraints, namely the existence of NATs and Firewalls, make such an approach only feasible in certain academic and research environments. System administrators are often very reluctant to open pinholes in their firewalls, as both the security risk (especially in case of a user-maintained endpoint) and the administrative effort increases. For mass deployment to end users, today, it is almost essential to reduce the number of pinholes in a firewall to the absolute minimum—ideally to a single one.

This line of thought leads to a scenario as depicted in Fig.5. As the server, in most cases, will be a professionally maintained device, it is reasonable to assume that its administrators have control over the firewall and can open as many pinholes as required. Therefore, the server sends to multiple IP multicast groups, each carrying a single layer. Close to the edge of the network, a middlebox is used to aggregate the content of potentially more than one multicast group into a single RTP stream carrying one or more layers. Only for that RTP stream, a pinhole has to be opened in a firewall. Physically, we expect middleboxes of this kind to be co-located with wireless access gateways and similar entities.

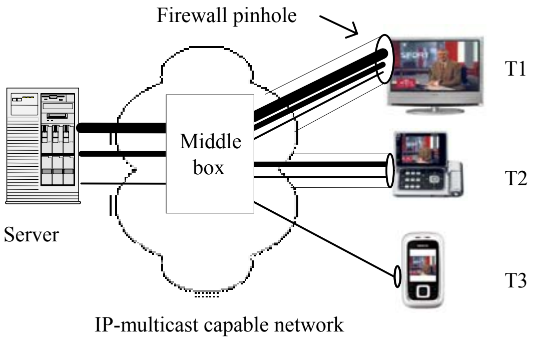

**Fig.5 Middlebox receives layered IP multicast and generates single layered bit stream for each receiver**

The advantage of such a topology is reduced server and core network load, and reduced server complexity, as the server does not need to generate and simulcast multiple full representations.

In order to fulfill its role, the middlebox has to be aware of the RTP details, its payload format, and the signaling. It also needs to be located inside the security context of the session, so to be able to parse the payload header (which is encrypted, according to SRTP (Baugher *et al*., 2004)). In short, these middleboxes are not simple network layer routers that receive configuration information through IGMP, but need to be signaling aware application layer devices of high sophistication.

An RTP session requires per RFC 3550 a continuous numbering space for the RTP sequence numbers. Furthermore, the RTP payload type is negotiated for each session. Finally, RTP timestamps are required to be initialized to random offsets at session startup, so as to minimize the risk of breaking encryption. For all these (and a number of other) reasons, the middleboxes logically terminate the incoming RTP streams (from the server) and generate new and independent outgoing RTP sessions (to the terminal). Terminating RTP sessions is a feature of an RTP mixer.

On the other hand, according to RFC 3550, RTP mixers "mix" content of several (unrelated) RTP sessions. Clearly, this is not the case for a simple forming of an aggregated layered stream, as key properties of the stream are retained. Because of that, the middlebox has to be seen as an RTP translator. What we envision is that the middlebox shares properties of both mixers and translators, and implements neither fully.

RTP translators, traditionally, operate on video streams by transcoding, which previously was mostly implemented by reconstruction of the incoming stream to the pixel domain, and re-encoding. The goal of using scalability is to avoid this pixel-domains transcoding step, and so, to avoid the quality loss and the cycle complexity in the middlebox. Indeed, this is possible. However, from an RTP standpoint, nevertheless, incoming sessions are terminated and a new session with modified content is created, and transcoding occurs.

SVC is designed to facilitate the aggregation of multiple layers into a single bit stream. More concretely, the extended NAL unit headers contain sufficient information to allow the association of each NAL unit with its layer. Please see Section 2.2. The step of forming a multi-layered bit stream from various available layers can be described as arranging NAL units from the various incoming layers into a certain order. No bit stream reconstruction is required and all pieces of information necessary are either placed in fixed-length bit fields in the NAL unit header, or are part of the signaling. Therefore, the "transcoding" process is very lightweight.

It is necessary to require the middlebox to be sufficiently media aware to parse certain parts of the bit stream. Whether these parts are documented in an RTP payload format or in the media specification itself, is a problem of documentation, but not of concept or implementation.

A third scenario is presented in Fig.6. Here, the terminals are connected directly to the server, utilizing only a single transport address for video (IP address and port number). For each terminal, the server composes a bit stream tailored to the terminal's needs, by aggregating NAL units of appropriate layers.

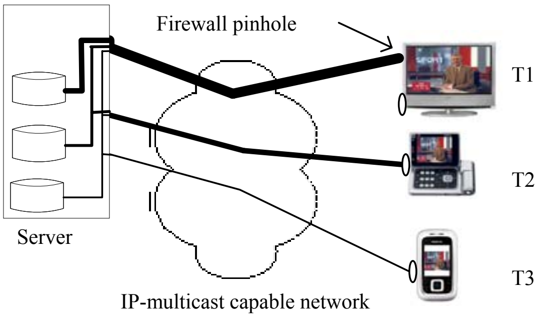

**Fig.6 Server combines single bit stream for each terminal from individually stored layers**

Video data is stored in layered form, and from the files containing the layers only those NAL units are packetized to what a specific terminal requires. This architecture saves resources on the server, namely the disk space that would be required to keep many different representations.

Please note that a comparable functionality could also be implemented if a reasonable RTP multiplex scheme were available. However, as the work towards RTP multiplexers is either stalled (Handley, 1998) or tailored towards the multiplexing of many streams of itdentical properties e.g. in trunking gateways (Thompson e*t al*., 2005), it is suggested that payload specific support is required.

## SIGNALING

In this paper we define "signaling" as all traffic of administrative information that is related to scalable stream transport but is not directly related to the carriage of NAL units.

A number of signaling protocol families are currently in use that support RTP for the media transport. Of those, we restrict the following discussion to the "native" IETF protocols, namely the Session Initiation Protocol (Rosenberg *et al.*, 2002) and the Real-Time Streaming Protocol (Schulzrinne *et al.*, 1998). Both have in common the representation format for the signaling messages, according to the Session Description Protocol RFC 2327 (Handley and Jacobson, 1998) and its forthcoming successor (Handley *et al.*, 2005). SDP, in its current form, has only very rudimentary support for scalable media. In particular, according to RFC 2327 it is possible to express on the session level that certain directly adjacent transport addresses (in terms of IP multicast address values) carry RTP sessions that, together, form scalable media. However, it is not possible to express different media types for adjacent transport addresses. Instead, all scalable layers have to share the same media description, which disallows, for example, different scalable layers conforming to more than one profile or level.

The acknowledged lack of generic signaling support has triggered discussions in the IETF, whether such support should be implemented in a payload specific or in a generic way.

**Generic signaling**

One early example of a "generic" solution would perhaps be the one discussed in (Vitali and Fumagalli, 2005). While many aspects of this draft, in our opinion, have major shortcomings, it could form a base for discussions towards generic signaling. Similarly, Li's draft on the binding of an FEC RTP session with the media session the FEC projects (Li, 2005) could be a starting point.

One problem of a "generic" solution is to find the attributes different (current and future) scalable media formats have in common. Examples for these attributes include: number of scalable layers, layer dependency information (which is trivial for hierarchical layering schemes, but could also be phrased toward non hierarchical layering types or even multiple description coding), and the properties of each layer (frame rate, bit rate, frame size, profile & level, decoding dependency information, initial parameter sets, FGS capable, sub-picture layer or not, sub-region layer or not). This SVC-specific list already suggests that it will be very difficult to identify a generic mechanism that is sufficiently flexible to support all current layered coding schemes, and still be manageable in terms of document complexity. However, it may be possible to extract a likely common subset of attributes for a generic solution.

**Payload specific signaling**

In its current form, the SVC RTP payload draft (Wenger and Wang, 2005) follows the other possible avenue, which is a payload specific solution. The draft suggests that each layer (or group of layers, the draft is not specific in this regard) be transported in its own RTP session, which is announced/negotiated as an independent SDP media description. From an SDP and higher protocol viewpoint, all these descriptions appear to be independent media streams. Their relationship is described in a single SDP attribute, which carries a binary description of the layering structure in BASE64 format. The content of this attribute is not accessible to non-SVC-aware mechanisms.

When writing the draft, we felt that such an extreme payload dependency—not only for the payload format itself, but also for the signaling—offers flexibility and also advantages in the standardization process coordination that would outweigh the disadvantage of not being understandable by non-SVC-aware signaling devices. However, during IETF65 this view was challenged by not a few participants, and therefore newer versions of the draft will likely include some form of layer dependency signaling that does not require knowledge of the SVC bit-oriented syntax. Alternatively, this signaling could be "outsourced" to an independent generic signaling support mechanism, as already discussed.

## MULTI-LAYER SUPPORT IN A SINGLE RTP SESSION

If each layer would always be transported in its own RTP session, the signaling support as mentioned in the previous session would perhaps be sufficient to implement all CGS functionalities of SVC. We believe that at least the use cases presented in Section 4 do not require further mechanisms, and that is why (Wenger and Wang, 2005) does not include more mechanisms. However, on mailing lists and during public and private discussions it was suggested that the support of carrying multiple layers in a single RTP session is also of value. In that case, it may be beneficial to have the properties of each NAL unit (including its layer dependencies) available for middleboxes such as the ones suggested in use case 2.

As those middleboxes necessarily have to be aware of both signaling and (at least from a standardization viewpoint) of the payload itself, it is perhaps not necessary to define new codepoints for the support of encapsulating NAL units from multiple SVC layers into a single RTP packet in the payload format. Based on our current knowledge, we suggest that JVT defines these codepoints in an easily parseable format, and the payload specification references that information in a similar spirit as the H.264/AVC payload format specification reference the NAL unit header syntax and semantics. As mentioned before, this is more a question of document structure than of the technology itself.

## NAL UNIT AGGREGATION AND FRAGMENTATION

The H.264/AVC RTP payload specification contains mechanisms to aggregate more than one NAL unit into a single RTP packet, and to split overly large NAL unit into multiple RTP packets. Two fundamentally different modes of operation are supported: in non-interleaved mode, the NAL units can only be aggregated in decoding order, whereas in interleaved mode, NAL units belonging to multiple pictures can be interleaved. The former mode is intended to avoid excessive RTP/UDP/IP header overhead that would result when encapsulating small NAL units in single NAL unit packets, whereas the latter mode is an error resilience tool and also allows for a number of sophisticated transport scenarios, such as those described in (Schierl *et al*., 2005) and (Tian *et al*., 2005b).

What the aggregation packets of both non-interleaved and interleaved mode have in common is that they include, in their respective payload headers, a “summary” of the information on the NAL unit headers carried in the aggregation packet. For example, according to RFC 3984, the value of the NRI field in the RTP payload header for an aggregation packet “[…] MUST be the maximum of all the NAL units carried in the aggregation packet”. As discussed in Section 2.2, SVC’s NAL unit header can be up to three octets long. It has been suggested that the header of an aggregation packet needs to reflect these additional octets. Implementing this would require additional aggregation packet types. Similar arguments could perhaps be made for the Fragmentation Unit packet types.

Once more, we believe that the difficulty of defining more packet types can most easily be overcome by disallowing more than one layer in one RTP session. Alternatively, it could be possible to require that aggregation only be performed with NAL units belonging to a single layer. Finally, it could be argued that any devices that wish to re-arrange RTP packets must necessarily be media aware and therefore can be required to look into the media data themselves. Hence, no support from a payload viewpoint is required. This is the reason why (Wenger and Wang, 2005) does not include new aggregation and fragmentation packets.

## FINE GRANULAR SCALABILITY

SVC includes Fine Granular Scalability in the form of FGS slices. Without losing generality, any number of FGS slice octets can be removed, starting at the end of the slice, as long as the slice header is kept intact. There is no explicit, easily accessible bit field available that carries the size of the slice header —a mechanism intended to prune an FGS slice must parse the slice header.

For the sake of simplicity, we assume henceforth that the slice is carried in one NAL unit, and we use the term FGS-NAL unit for a single slice carried in a single NAL unit.

Obviously, pruning FGS NAL units can only be performed in a middlebox that is fully media aware and inside the security context. Therefore, we believe it is reasonable to require the middlebox to parse the slice header of an FGS slice when it wishes to reduce the sending bit rate. However, this task would be the only one that requires bit-oriented processing of syntax not documented in the SVC payload draft. In order to optimize the middlebox design, and to simplify documentation, it is possible to trade overhead for an explicit signaling of an FGS slice header size against middlebox complexity. Discussions on this subject are ongoing.

## CONCLUSION AND OUTLOOK

SVC appears to be the most promising attempt for a scalable video coding discussed over the last decade. One main target transport mechanism is RTP. We reviewed the current, early draft of an RTP payload specification for SVC. After presenting some background information, four major points of discussion related to the payload specification were presented. More specifically, it is currently unclear how the SDP signaling support can be implemented, whether there is a need for carrying more than one SVC layer in a given RTP session, if SVC-specific aggregation and fragmentation mechanisms are required and how they can be implemented, and whether optimization support for fine granularity scalability should be implemented.

We encourage all readers to contribute to the development of the SVC RTP payload specification, by commenting on the forthcoming-01 version of the Internet Draft and postings to the AVT mailing list (please see http://www.ietf.org/html.charters/avt-charter. html for logistics).

## References

Baugher, M., McGrew, D., Naslund, M., Carrara, E., Norrman, K., 2004. The Secure Real-time Transport Protocol (SRTP). RFC 3711, available from http://www.ietf.org/rfc/rfc3711.txt.

Fenner, W., 1997. Internet Group Management Protocol, Version 2. RFC 2236, available from http://www.ietf.org/rfc/rfc2236.txt.

Handley, M., 1998. GeRM: Generic RTP Multiplexing. Internet Draft, Work in Progress, Expired. Available from http://www.ietf.org/proceedings/98dec/I-D/draft-ietf-avt-germ-00.txt.

Handley, M., Jacobson, V., 1998. SDP: Session Description Protocol. RFC 2327, available from http://www.ietf.org/rfc/rfc2327.txt.

Handley, M., Jacobson, V., Perkins, C., 2005. SDP: Session Description Protocol. Internet Draft, Work in Progress. Available from http://www.ietf.org/internet-drafts/draft-ietf-mmusic-sdp-new-25.txt.

ISO/IEC JTC1, 1992. Information Technology—Coding of Audio-visual Objects, Part 2: Visual. ISO/IEC 11176-2 (MPEG-1).

ISO/IEC JTC1, 1998. Information Technology—Coding of Audio-visual Objects, Part 2: Visual. ISO/IEC 14496-2 (MPEG-4).

ISO/IEC MPEG, 2005. SVC Requirements Specified by MPEG. JVT-N026, available from http://ftp3.itu.ch/av-arch/jvt-site/2005_01_HongKong/jvt-n026.doc.

ITU-T, 1998. Video Coding for Low Bitrate Communication. ITU-T Recommendation H.263, Version 2.

ITU-T, 2003. Advanced Video Coding for Generic Audio-visual Services. ITU-T Recommendation H.264.

ITU-T and ISO/IEC JTC1, 1994. Generic Coding of Moving Pictures and Associated Audio Information, Part 2: Video. ITU-T Recommendation H.262, ISO/IEC 13818-2 (MPEG-2).

ITU-T VCEG, 2005. SVC Requirements Specified by VCEG. JVT-N027, available from http://ftp3.itu.ch/av-arch/jvt-site/2005_01_HongKong/jvt-n027.doc.

Jacobson, V., McCanne, S., Vetterli, M., 1996. Receiver-Driven Layered Multicast. Proc. of ACM SIGCOMM'96. Stanford, CA, p.117-130.

Joint Video Team, 2005a. Scalable Video Coding—Working Draft 1. Available from http://ftp3.itu.ch/av-arch/jvt-site/2005_01_HongKong/JV T-N020d1.zip.

Joint Video Team, 2005b. Scalable Video Coding—Working Draft 4. Available from http://ftp3.itu.ch/av-arch/jvt-site/2005_10_Nice/JVT-Q201d1.zip.

Joint Video Team, 2005c. Joint Scalable Video Model—JSVM-4. Available from http://ftp3.itu.ch/av-arch/jvt-site/2005_10_Nice/JVT-Q202.zip.

Jones, H.W., 1979. A comparison of theoretical and experimental video compression designs. *IEEE Trans. on Electromag. Compat.*, **21**(1):50-56.

Li, W., 2001. Overview of fine granular scalability in MPEG-4 video standard. *IEEE CSVT*, **11**(3):301-317.

Li, A., 2005. FEC Grouping Semantics in SDP. Internet Draft, Work in Progress. Available from http://www.ietf.org/internet-drafts/draft-ietf-mmusic-fec-grouping-02.txt.

Ohm, J.R., 2005. Advances in scalable video coding. *Proceeding of the IEEE*, **93**(1):42-56.

Reader, C., 2002. History of Video Compression. Draft Version 2.0, JVT-D-068, available from http://ftp3.itu.ch/av-arch/jvt-site/2002_07_Klagenfurt/JVT-D068.doc.

Rosenberg, J., Schulzrinne, H., Gamarilllo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., Schooler, E., 2002. SIP: Session Initiation Protocol. RFC 3261, available from http://www.ietf.org/rfc/rfc3261.txt.

Schierl, T., Wiegand, T., Kampmann, M., 2005. 3GPP Compliant Adaptive Wireless Video Streaming Using AVC. Proc. of IEEE International Conference on Image Processing.

Schulzrinne, H., Rao, A., Lanphier, R., 1998. Real Time Streaming Protocol. RFC2326, available from http://

www.ietf.org/rfc/rfc2326.txt.

Schulzrinne, H., Casner, S., Frederick, R., Jacobson, V., 2003. RTP: A Transport Protocol for Real-Time Applications. RFC 3550, STD 64, available from http://www.ietf.org/rfc/rfc3550.txt.

Schwarz, H., Hinz, T., Marpe, D., Wiegand, T., 2005. Constrained Inter-Layer Prediction for Single-Loop Decoding in Spatial Scalability. Proc. of IEEE International Conference on Image Processing.

Thompson, B., Koren, T., Wing, D., 2005. Tunneling Multiplexed Compressed RTP (TCRTP). RFC 4170, available from http://www.ietf.org/rfc/rfc4170.txt.

Tian, D., Hannuksela, M.M., Gabbouj, M., 2005a. Subsequence Video Coding for Improved Temporal Scalability. Proc. of IEEE Int. Symposium on Circuits and Systems (ISCAS).

Tian, D., Malamal Vadakital, V.K., Hannuksela, M.M., Wenger, S., Gabbouj, M., 2005b. Improved H.264/AVC Video Broadcast/Multicast. Proc. of Visual Communications and Image Processing (VCIP).

Vitali, A., Fumagalli, M., 2005. Standard-compatible Multiple-Description Coding (MDC) and Layered Coding (LC) of Audio/Video Streams. Internet Draft, Work in Progress. Available from http://www.ietf.org/internet-drafts/draft-vtali-ietf-avt-mdc-lc-00.txt.

Wenger, S., Wang, Y.K., 2005. RTP Payload Format for SVC Video. Internet Draft, Work in Progress. Available from http://www.ietf.org/internet-drafts/draft-wenger-avt-rtp-svc-00.txt.

Wenger, S., Hannuksela, M.M., Stockhammer, T., Westerlund, M., Singer, D., 2005. RTP Payload Format for H.264 Video. RFC 3984, available from http://www.ietf.org/rfc/rfc3984.txt.