

Network Working Group
Internet-Draft
Updates: [4855](#) (if approved)
Intended status: Standards Track
Expires: April 28, 2017

P. Thatcher
Google
M. Zanaty
S. Nandakumar
Cisco Systems
B. Burman
Ericsson
A. Roach
B. Campen
Mozilla
October 25, 2016

RTP Payload Format Restrictions
draft-ietf-mmusic-rid-08

Abstract

In this specification, we define a framework for specifying restrictions on RTP streams in the Session Description Protocol. This framework defines a new "rid" SDP attribute to unambiguously identify the RTP Streams within a RTP Session and restrict the streams' payload format parameters in a codec-agnostic way beyond what is provided with the regular Payload Types.

This specification updates [RFC4855](#) to give additional guidance on choice of Format Parameter (fmt) names, and on their relation to the restrictions defined by this document.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of [BCP 78](#) and [BCP 79](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on April 28, 2017.

Copyright Notice

Copyright (c) 2016 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](http://trustee.ietf.org/license-info) and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1.	Terminology	3
2.	Introduction	3
3.	Key Words for Requirements	4
4.	SDP "a=rid" Media Level Attribute	4
5.	"a=rid" restrictions	6
6.	SDP Offer/Answer Procedures	7
6.1.	Generating the Initial SDP Offer	7
6.2.	Answerer processing the SDP Offer	8
6.2.1.	"a=rid"-unaware Answerer	8
6.2.2.	"a=rid"-aware Answerer	9
6.3.	Generating the SDP Answer	10
6.4.	Offerer Processing of the SDP Answer	10
6.5.	Modifying the Session	12
7.	Use with Declarative SDP	12
8.	Interaction with Other Techniques	12
8.1.	Interaction with VP8 Format Parameters	13
8.1.1.	max-fr - Maximum Framerate	13
8.1.2.	max-fs - Maximum Framesize, in VP8 Macroblocks	13
8.2.	Interaction with H.264 Format Parameters	14
8.2.1.	profile-level-id and max-recv-level - Negotiated Sub-Profile	15
8.2.2.	max-br / MaxBR - Maximum Video Bitrate	15
8.2.3.	max-fs / MaxFS - Maximum Framesize, in H.264 Macroblocks	15
8.2.4.	max-mbps / MaxMBPS - Maximum Macroblock Processing Rate	16
8.2.5.	max-smbps - Maximum Decoded Picture Buffer	16
9.	Format Parameters for Future Payloads	16
10.	Formal Grammar	16
11.	SDP Examples	18
11.1.	Many Bundled Streams using Many Codecs	18

11.2.	Scalable Layers	20
12.	IANA Considerations	20
12.1.	New SDP Media-Level attribute	20
12.2.	Registry for RID-Level Parameters	21
13.	Security Considerations	22
14.	Acknowledgements	23
15.	References	23
15.1.	Normative References	23
15.2.	Informative References	24
	Authors' Addresses	25

[1.](#) Terminology

The terms "Source RTP Stream", "Endpoint", "RTP Session", and "RTP Stream" are used as defined in [\[RFC7656\]](#).

[\[RFC4566\]](#) and [\[RFC3264\]](#) terminology is also used where appropriate.

[2.](#) Introduction

The Payload Type (PT) field in RTP provides a mapping between the RTP payload format and the associated SDP media description. The SDP rtpmap and/or fmpmap attributes are used, for a given PT, to describe the properties of the media that is carried in the RTP payload.

Recent advances in standards have given rise to rich multimedia applications requiring support for multiple RTP Streams within a RTP session [\[I-D.ietf-mmusic-sdp-bundle-negotiation\]](#), [\[I-D.ietf-mmusic-sdp-simulcast\]](#) or having to support a large number of codecs. These demands have unearthed challenges inherent with:

- o The restricted RTP PT space in specifying the various payload configurations,
- o The codec-specific constructs for the payload formats in SDP,
- o Missing or underspecified payload format parameters,
- o Overloading of PTs to indicate not just codec configurations, but individual streams within an RTP session.

To expand on these points: [\[RFC3550\]](#) assigns 7 bits for the PT in the RTP header. However, the assignment of static mapping of RTP payload type numbers to payload formats and multiplexing of RTP with other protocols (such as RTCP) could result in limited number of payload type numbers available for application usage. In scenarios where the number of possible RTP payload configurations exceed the available PT

space within a RTP Session, there is a need for a way to represent the additional restrictions on payload configurations and to effectively map an RTP Stream to its corresponding restrictions. This issue is exacerbated by the increase in techniques - such as simulcast and layered codecs - which introduce additional streams into RTP Sessions.

This specification defines a new SDP framework for restricting Source RTP Streams ([Section 2.1.10 \[RFC7656\]](#)), along with the SDP attributes to restrict payload formats in a codec-agnostic way. This framework can be thought of as a complementary extension to the way the media format parameters are specified in SDP today, via the "a=fmtp" attribute.

The additional restrictions on individual streams are indicated with a new "a=rid" SDP attribute. Note that the restrictions communicated via this attribute only serve to further restrict the parameters that are established on a PT format. They do not relax any existing restrictions.

This specification makes use of the RTP Stream Identifier SDP item defined in [[I-D.ietf-avtext-rid](#)] to provide correlation between the RTP Packets and their format specification in the SDP.

As described in [Section 6.2.1](#), this mechanism achieves backwards compatibility via the normal SDP processing rules, which require unknown a= lines to be ignored. This means that implementations need to be prepared to handle successful offers and answers from other implementations that neither indicate nor honor the restrictions requested by this mechanism.

Further, as described in [Section 6](#) and its subsections, this mechanism achieves extensibility by: (a) having offerers include all supported restrictions in their offer, and (b) having answerers ignore "a=rid" lines that specify unknown restrictions.

[3.](#) Key Words for Requirements

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 [[RFC2119](#)]

[4.](#) SDP "a=rid" Media Level Attribute

This section defines new SDP media-level attribute [[RFC4566](#)], "a=rid", ("restriction identifier") used to communicate a set of restrictions to be applied an identified RTP Stream. Roughly

speaking, this attribute takes the following form (see [Section 10](#) for a formal definition).

```
a=rid:<rid-id> <direction> [pt=<fmt-list>]<restriction>=<value>...
```

An "a=rid" SDP media attribute specifies restrictions defining a unique RTP payload configuration identified via the "rid-id" field. This value binds the restriction to the RTP Stream identified by its RTP Stream Identifier SDES item [[I-D.ietf-avtext-rid](#)]. To be clear, implementations that use the "a=rid" parameter in SDP MUST support the RtpStreamId SDES item described in [[I-D.ietf-avtext-rid](#)]. Such implementations MUST send it for all streams in an m-section that has "a=rid" lines remaining after applying the rules in [Section 6](#) and its subsections.

The "direction" field identifies the directionality of the RTP Stream; it may be either "send" or "recv".

The optional "pt=<fmt-list>" lists one or more PT values that can be used in the associated RTP Stream. If the "a=rid" attribute contains no "pt", then any of the PT values specified in the corresponding "m=" line may be used.

The list of zero or more codec-agnostic restrictions ([Section 5](#)) describe the restrictions that the corresponding RTP Stream will conform to.

This framework MAY be used in combination with the "a=fmtp" SDP attribute for describing the media format parameters for a given RTP Payload Type. In such scenarios, the "a=rid" restrictions ([Section 5](#)) further restrict the equivalent "a=fmtp" attributes.

A given SDP media description MAY have zero or more "a=rid" lines describing various possible RTP payload configurations. A given "rid-id" MUST NOT be repeated in a given media description ("m=" section).

The "a=rid" media attribute MAY be used for any RTP-based media transport. It is not defined for other transports, although other documents may extend its semantics for such transports.

Though the restrictions specified by the "rid" restrictions follow a syntax similar to session-level and media-level parameters, they are defined independently. All "rid" restrictions MUST be registered with IANA, using the registry defined in [Section 12](#).

[Section 10](#) gives a formal Augmented Backus-Naur Form (ABNF) [[RFC5234](#)] grammar for the "rid" attribute. The "a=rid" media attribute is not dependent on charset.

5. "a=rid" restrictions

This section defines the "a=rid" restrictions that can be used to restrict the RTP payload encoding format in a codec-agnostic way.

The following restrictions are intended to apply to video codecs in a codec-independent fashion.

- o max-width, for spatial resolution in pixels. In the case that stream orientation signaling is used to modify the intended display orientation, this attribute refers to the width of the stream when a rotation of zero degrees is encoded.
- o max-height, for spatial resolution in pixels. In the case that stream orientation signaling is used to modify the intended display orientation, this attribute refers to the width of the stream when a rotation of zero degrees is encoded.
- o max-fps, for frame rate in frames per second. For encoders that do not use a fixed framerate for encoding, this value should restrict the minimum amount of time between frames: the time between any two consecutive frames SHOULD NOT be less than 1/max-fps seconds.
- o max-fs, for frame size in pixels per frame. This is the product of frame width and frame height, in pixels, for rectangular frames.
- o max-br, for bit rate in bits per second. The restriction applies to the media payload only, and does not include overhead introduced by other layers (e.g., RTP, UDP, IP, or Ethernet). The exact means of keeping within this limit are left up to the implementation, and instantaneous excursions outside the limit are permissible. For any given one-second sliding window, however, the total number of bits in the payload portion of RTP SHOULD NOT exceed the value specified in "max-br."
- o max-pps, for pixel rate in pixels per second. This value SHOULD be handled identically to max-fps, after performing the following conversion: $\text{max-fps} = \text{max-pps} / (\text{width} * \text{height})$. If the stream resolution changes, this value is recalculated. Due to this recalculation, excursions outside the specified maximum are possible near resolution change boundaries.

- o max-bpp, for maximum number of bits per pixel, calculated as an average of all samples of any given coded picture. This is expressed as a floating point value, with an allowed range of 0.0001 to 48.0. These values MUST be encoded with at most four digits to the right of the decimal point.
- o depend, to identify other streams that the stream depends on. The value is a comma-separated list of rid-ids. These rid-ids identify RTP streams that this stream depends on in order to allow for proper interpretation.

All the restrictions are optional and are subject to negotiation based on the SDP Offer/Answer rules described in [Section 6](#).

This list is intended to be an initial set of restrictions. Future documents may define additional restrictions; see [Section 12.2](#). While this document does not define restrictions for audio codecs or any media types other than video, there is no reason such restrictions should be precluded from definition and registration by other documents.

[Section 10](#) provides formal Augmented Backus-Naur Form (ABNF) [[RFC5234](#)] grammar for each of the "a=rid" restrictions defined in this section.

6. SDP Offer/Answer Procedures

This section describes the SDP Offer/Answer [[RFC3264](#)] procedures when using this framework.

Note that "rid-id" values are only required to be unique within a media section ("m-line"); they do not necessarily need to be unique within an entire RTP session. In traditional usage, each media section is sent on its own unique 5-tuple, which provides an unambiguous scope. Similarly, when using BUNDLE [[I-D.ietf-mmusic-sdp-bundle-negotiation](#)], MID values associate RTP streams uniquely to a single media description.

[6.1](#). Generating the Initial SDP Offer

For each RTP media description in the offer, the offerer MAY choose to include one or more "a=rid" lines to specify a configuration profile for the given set of RTP Payload Types.

In order to construct a given "a=rid" line, the offerer must follow these steps:

1. It MUST generate a "rid-id" that is unique within a media description
2. It MUST set the direction for the "rid-id" to one of "send" or "recv"
3. It MAY include a listing of SDP format tokens (usually corresponding to RTP payload types) allowed to appear in the RTP Stream. Any Payload Types chosen MUST be a valid payload type for the media section (that is, it must be listed on the "m=" line). The order of the listed formats is significant; the alternatives are listed from (left) most preferred to (right) least preferred. When using RID, this preference overrides the normal codec preference as expressed by format type ordering on the "m=" line, using regular SDP rules.
4. The Offerer then chooses zero or more "a=rid" restrictions ([Section 5](#)) to be applied to the RTP Stream, and adds them to the "a=rid" line.
5. If the offerer wishes the answerer to have the ability to specify a restriction, but does not wish to set a value itself, it includes the name of the restriction in the "a=rid" line, but without any indicated value.

Note: If an "a=fmtp" attribute is also used to provide media-format-specific parameters, then the "a=rid" restrictions will further restrict the equivalent "a=fmtp" parameters for the given Payload Type for the specified RTP Stream.

If a given codec would require an "a=fmtp" line when used without "a=rid" then the offer MUST include a valid corresponding "a=fmtp" line even when using "a=rid".

[6.2.](#) Answerer processing the SDP Offer

[6.2.1.](#) "a=rid"-unaware Answerer

If the receiver doesn't support the framework proposed in this specification, the entire "a=rid" line is ignored following the standard [[RFC3264](#)] Offer/Answer rules.

[Section 6.1](#) requires the offer to include a valid "a=fmtp" line for any codecs that otherwise require it (in other words, the "a=rid" line cannot be used to replace "a=fmtp" configuration). As a result, ignoring the "a=rid" line is always guaranteed to result in a valid session description.

6.2.2. "a=rid"-aware Answerer

If the answerer supports the "a=rid" attribute, the following verification steps are executed, in order, for each "a=rid" line in a given media description:

1. The answerer ensures that the "a=rid" line is syntactically well formed. In the case of a syntax error, the "a=rid" line is discarded.
2. Extract the rid-id from the "a=rid" line and verify its uniqueness within a media section. In the case of a duplicate, the entire "a=rid" line, and all "a=rid" lines with rid-ids that duplicate this line, are discarded and **MUST NOT** be included in the SDP Answer.
3. If the "a=rid" line contains a "pt=", the list of payload types is verified against the list of valid payload types for the media section (that is, those listed on the "m=" line). Any PT missing from the "m=" line is discarded from the set of values in the "pt=". If no values are left in the "pt=" parameter after this processing, then the "a=rid" line is discarded.
4. If the "direction" field is "recv", The answerer ensures that "a=rid" restrictions are supported. In the case of an unsupported restriction, the "a=rid" line is discarded.
5. If the "depend" restriction is included, the answerer **MUST** make sure that the listed rid-ids unambiguously match the rid-ids in the SDP offer. Any "a=rid" lines that do not are discarded.
6. The answerer verifies that the restrictions are consistent with at least one of the codecs to be used with the RTP Stream. If the "a=rid" line contains a "pt=", it contains the list of such codecs; otherwise, the list of such codecs is taken from the associated "m=" line. See [Section 8](#) for more detail. If the "a=rid" restrictions are incompatible with the other codec properties for all codecs, then the "a=rid" line is discarded.

Note that the answerer does not need to understand every restriction present in a "send" line: if a stream sender restricts the stream in a way that the receiver does not understand, this causes no issues with interoperability.

6.3. Generating the SDP Answer

Having performed verification of the SDP offer as described in [Section 6.2.2](#), the answerer shall perform the following steps to generate the SDP answer.

For each "a=rid" line:

1. The sense of of the "direction" field is reversed: "send" is changed to "recv", and "recv" is changed to "send".
2. The answerer MAY choose to modify specific "a=rid" restriction value in the answer SDP. In such a case, the modified value MUST be more restricted than the ones specified in the offer. The answer MUST NOT include any restrictions that were not present in the offer.
3. The answerer MUST NOT modify the "rid-id" present in the offer.
4. If the "a=rid" line contains a "pt=", the answerer is allowed to discard one or more media formats from a given "a=rid" line. If the answerer chooses to discard all the media format tokens from an "a=rid" line, the answerer MUST discard the entire "a=rid" line. If the offer did not contain a "pt=" for a given "a=rid" line, then the answer MUST NOT contain a "pt=" in the corresponding line.
5. In cases where the answerer is unable to support the payload configuration specified in a given "a=rid" line in the offer, the answerer MUST discard the corresponding "a=rid" line. This includes situations in which the answerer does not understand one or more of the restrictions in an "a=rid" line with a direction of "recv".

Note: in the case that the answerer uses different PT values to represent a codec than the offerer did, the "a=rid" values in the answer use the PT values that are present in its answer.

6.4. Offerer Processing of the SDP Answer

The offerer SHALL follow these steps when processing the answer:

1. The offerer matches the "a=rid" line in the answer to the "a=rid" line in the offer using the "rid-id". If no matching line can be located in the offer, the "a=rid" line is ignored.
2. If the answer contains any restrictions that were not present in the offer, then the offerer SHALL discard the "a=rid" line.

3. If the restrictions have been changed between the offer and the answer, the offerer MUST ensure that the modifications can be supported; if they cannot, the offerer SHALL discard the "a=rid" line.
4. If the "a=rid" line in the answer contains a "pt=" but the offer did not, the offerer SHALL discard the "a=rid" line.
5. If the "a=rid" line in the answer contains a "pt=" and the offer did as well, the offerer verifies that the list of payload types is a subset of those sent in the corresponding "a=rid" line in the offer. Note that this matching must be performed semantically rather than on literal PT values, as the remote end may not be using symmetric PTs. For the purpose of this comparison: for each PT listed on the "a=rid" line in the answer, the offerer looks up the corresponding "a=rtpmap" and "a=fmtp" lines in the answer. It then searches the list of "pt=" values indicated in the offer, and attempts to find one with an equivalent set of "a=rtpmap" and "a=fmtp" lines in the offer. If all PTs in the answer can be matched, then the "pt=" values pass validation; otherwise, it fails. If this validation fails, the offerer SHALL discard the "a=rid" line. Note that this semantic comparison necessarily requires an understanding of the meaning of codec parameters, rather than a rote byte-wise comparison of their values.
6. If the "a=rid" line contains a "pt=", the offerer verifies that the attribute values provided in the "a=rid" attributes are consistent with the corresponding codecs and their other parameters. See [Section 8](#) for more detail. If the "a=rid" restrictions are incompatible with the other codec properties, then the offerer SHALL discard the "a=rid" line.
7. The offerer verifies that the restrictions are consistent with at least one of the codecs to be used with the RTP Stream. If the "a=rid" line contains a "pt=", it contains the list of such codecs; otherwise, the list of such codecs is taken from the associated "m=" line. See [Section 8](#) for more detail. If the "a=rid" restrictions are incompatible with the other codec properties for all codecs, then the offerer SHALL discard the "a=rid" line.

Any "a=rid" line present in the offer that was not matched by step 1 above has been discarded by the answerer, and does not form part of the negotiated restrictions on an RTP Stream. The offerer MAY still apply any restrictions it indicated in an "a=rid" line with a direction field of "send", but it is not required to do so.

It is important to note that there are several ways in which an offer can contain a media section with "a=rid" lines, but the corresponding media section in the response does not. This includes situations in which the answerer does not support "a=rid" at all, or does not support the indicated restrictions. Under such circumstances, the offerer **MUST** be prepared to receive a media stream to which no restrictions have been applied.

6.5. Modifying the Session

Offers and answers inside an existing session follow the rules for initial session negotiation. Such an offer **MAY** propose a change in the number of RIDs in use. To avoid race conditions with media, any RIDs with proposed changes **SHOULD** use a new ID, rather than re-using one from the previous offer/answer exchange. RIDs without proposed changes **SHOULD** re-use the ID from the previous exchange.

7. Use with Declarative SDP

This document does not define the use of RID in declarative SDP. If concrete use cases for RID in declarative SDP use are identified in the future, we expect that additional specifications will address such use.

8. Interaction with Other Techniques

Historically, a number of other approaches have been defined that allow restricting media streams via SDP. These include:

- o Codec-specific configuration set via format parameters ("a=fmtp"); for example, the H.264 "max-fs" format parameter [[RFC6184](#)]
- o Size restrictions imposed by image attribute attributes ("a=imageattr") [[RFC6236](#)]

When the mechanism described in this document is used in conjunction with these other restricting mechanisms, it is intended to impose additional restrictions beyond those communicated in other techniques.

In an offer, this means that "a=rid" lines, when combined with other restrictions on the media stream, are expected to result in a non-empty union. For example, if image attributes are used to indicate that a PT has a minimum width of 640, then specification of "max-width=320" in an "a=rid" line that is then applied to that PT is nonsensical. According to the rules of [Section 6.2.2](#), this will result in the corresponding "a=rid" line being ignored by the recipient.

In an answer, the "a=rid" lines, when combined with the other restrictions on the media stream, are also expected to result in a non-empty union. If the implementation generating an answer wishes to restrict a property of the stream below that which would be allowed by other parameters (e.g., those specified in "a=fmtp" or "a=imageattr"), its only recourse is to discard the "a=rid" line altogether, as described in [Section 6.3](#). If it instead attempts to restrict the stream beyond what is allowed by other mechanisms, then the offerer will ignore the corresponding "a=rid" line, as described in [Section 6.4](#).

The following subsections demonstrate these interactions using commonly-used video codecs. These descriptions are illustrative of the interaction principles outlined above, and are not normative.

[8.1](#). Interaction with VP8 Format Parameters

[RFC7741] defines two format parameters for the VP8 codec. Both correspond to restrictions on receiver capabilities, and never indicate sending restrictions.

[8.1.1](#). max-fr - Maximum Framerate

The VP8 "max-fr" format parameter corresponds to the "max-fps" restriction defined in this specification. If an RTP sender is generating a stream using a format defined with this format parameter, and the sending restrictions defined via "a=rid" include a "max-fps" parameter, then the sent stream is will conform to the smaller of the two values.

[8.1.2](#). max-fs - Maximum Framesize, in VP8 Macroblocks

The VP8 "max-fs" format parameter corresponds to the "max-fs" restriction defined in this document, by way of a conversion factor of the number of pixels per macroblock (typically 256). If an RTP sender is generating a stream using a format defined with this format parameter, and the sending restrictions defined via "a=rid" include a "max-fs" parameter, then the sent stream will conform to the smaller of the two values; that is, the number of pixels per frame will not exceed:

$$\min(\text{rid_max_fs}, \text{fmtp_max_fs} * \text{macroblock_size})$$

This fmtp parameter also has bearing on the max-height and max-width parameters. [Section 6.1 of \[RFC7741\]](#) requires that the width and height of the frame in macroblocks are also required to be less than $\text{int}(\sqrt{\text{fmtp_max_fs} * 8})$. Accordingly, the maximum width of a transmitted stream will be limited to:


```
min(rid_max_width, int(sqrt(fmt_max_fs * 8)) * macroblock_width)
```

Similarly, the stream's height will be limited to:

```
min(rid_max_height, int(sqrt(fmt_max_fs * 8)) * macroblock_height)
```

8.2. Interaction with H.264 Format Parameters

[RFC6184] defines format parameters for the H.264 video codec. The majority of these parameters do not correspond to codec-independent restrictions:

- o deint-buf-cap
- o in-band-parameter-sets
- o level-asymmetry-allowed
- o max-rcmd-nalu-size
- o max-cpb
- o max-dpb
- o packetization-mode
- o redundant-pic-cap
- o sar-supported
- o sar-understood
- o sprop-deint-buf-req
- o sprop-init-buf-time
- o sprop-interleaving-depth
- o sprop-level-parameter-sets
- o sprop-max-don-diff
- o sprop-parameter-sets
- o use-level-src-parameter-sets

Note that the max-cpb and max-dpb format parameters for H.264 correspond to restrictions on the stream, but they are specific to

the way the H.264 codec operates, and do not have codec-independent equivalents.

The following codec format parameters correspond to restrictions on receiver capabilities, and never indicate sending restrictions.

8.2.1. profile-level-id and max-recv-level - Negotiated Sub-Profile

These parameters include a "level" indicator, which acts as an index into Table A-1 of [H264]. This table contains a number of parameters, several of which correspond to the restrictions defined in this document. [RFC6184] also defines format parameters for the H.264 codec that may increase the maximum values indicated by the negotiated level. The following sections describe the interaction between these parameters and the restrictions defined by this document. In all cases, the H.264 parameters being discussed are the maximum of those indicated by [H264] Table A-1 and those indicated in the corresponding "a=fmtp" line.

8.2.2. max-br / MaxBR - Maximum Video Bitrate

The H.264 "MaxBR" parameter (and its equivalent "max-br" format parameter) corresponds to the "max-bps" restriction defined in this specification, by way of a conversion factor of 1000 or 1200; see [RFC6184] for details regarding which factor gets used under differing circumstances.

If an RTP sender is generating a stream using a format defined with this format parameter, and the sending restrictions defined via "a=rid" include a "max-fps" parameter, then the sent stream is will conform to the smaller of the two values - that is:

$$\min(\text{rid_max_br}, \text{h264_MaxBR} * \text{conversion_factor})$$

8.2.3. max-fs / MaxFS - Maximum Framesize, in H.264 Macroblocks

The H.264 "MaxFs" parameter (and its equivalent "max-fs" format parameter) corresponds roughly to the "max-fs" restriction defined in this document, by way of a conversion factor of 256 (the number of pixels per macroblock).

If an RTP sender is generating a stream using a format defined with this format parameter, and the sending restrictions defined via "a=rid" include a "max-fs" parameter, then the sent stream is will conform to the smaller of the two values - that is:

$$\min(\text{rid_max_fs}, \text{h264_MaxFs} * 256)$$

8.2.4. max-mbps / MaxMBPS - Maximum Macroblock Processing Rate

The H.264 "MaxMBPS" parameter (and its equivalent "max-mbps" format parameter) corresponds roughly to the "max-pps" restriction defined in this document, by way of a conversion factor of 256 (the number of pixels per macroblock).

If an RTP sender is generating a stream using a format defined with this format parameter, and the sending restrictions defined via "a=rid" include a "max-pps" parameter, then the sent stream will conform to the smaller of the two values - that is:

$$\min(\text{rid_max_pps}, \text{h264_MaxMBPS} * 256)$$

8.2.5. max-smbps - Maximum Decoded Picture Buffer

The H.264 "max-smbps" format parameter operates the same way as the "max-mbps" format parameter, under the hypothetical assumption that all macroblocks are static macroblocks. It is handled by applying the conversion factor described in [Section 8.1 of \[RFC6184\]](#), and the result of this conversion is applied as described in [Section 8.2.4](#).

9. Format Parameters for Future Payloads

Registrations of future RTP payload format specifications that define media types that have parameters matching the RID restrictions specified in this memo SHOULD name those parameters in a manner that matches the names of those RID restrictions, and SHOULD explicitly state what media type parameters are restricted by what RID restrictions.

10. Formal Grammar

This section gives a formal Augmented Backus-Naur Form (ABNF) [\[RFC5234\]](#) grammar for each of the new media and "a=rid" attributes defined in this document.

```
rid-syntax      = "a=rid:" rid-id SP rid-dir  
                  [ rid-pt-param-list / rid-param-list ]
```

```
rid-id          = 1*(alpha-numeric / "-" / "_")
```

```
alpha-numeric   = < as defined in {{RFC4566}} >
```

```
rid-dir         = "send" / "recv"
```

```
rid-pt-param-list = SP rid-fmt-list *("; " rid-param)
```



```
rid-param-list    = SP rid-param *("; " rid-param)

rid-fmt-list      = "pt=" fmt *( " ," fmt )

fmt               = < as defined in {{RFC4566}} >

rid-param         = rid-width-param
                  / rid-height-param
                  / rid-fps-param
                  / rid-fs-param
                  / rid-br-param
                  / rid-pps-param
                  / rid-bpp-param
                  / rid-depend-param
                  / rid-param-other

rid-width-param   = "max-width" [ "=" int-param-val ]

rid-height-param  = "max-height" [ "=" int-param-val ]

rid-fps-param     = "max-fps" [ "=" int-param-val ]

rid-fs-param      = "max-fs" [ "=" int-param-val ]

rid-br-param      = "max-br" [ "=" int-param-val ]

rid-pps-param     = "max-pps" [ "=" int-param-val ]

rid-bpp-param     = "max-bpp" [ "=" float-param-val ]

rid-depend-param  = "depend=" rid-list

rid-param-other   = 1*(alpha-numeric / "-") [ "=" param-val ]

rid-list          = rid-id *( " ," rid-id )

int-param-val     = 1*DIGIT

float-param-val   = 1*DIGIT "." 1*DIGIT

param-val         = *( %x20-58 / %x60-7E )
                  ; Any printable character except semicolon
```


11. SDP Examples

Note: see [[I-D.ietf-mmusic-sdp-simulcast](#)] for examples of RID used in simulcast scenarios.

11.1. Many Bundled Streams using Many Codecs

In this scenario, the offerer supports the Opus, G.722, G.711 and DTMF audio codecs, and VP8, VP9, H.264 (CBP/CHP, mode 0/1), H.264-SVC (SCBP/SCHP) and H.265 (MP/M10P) for video. An 8-way video call (to a mixer) is supported (send 1 and receive 7 video streams) by offering 7 video media sections (1 sendrecv at max resolution and 6 recvonly at smaller resolutions), all bundled on the same port, using 3 different resolutions. The resolutions include:

- o 1 receive stream of 720p resolution is offered for the active speaker.
- o 2 receive streams of 360p resolution are offered for the prior 2 active speakers.
- o 4 receive streams of 180p resolution are offered for others in the call.

NOTE: The SDP given below skips a few lines to keep the example short and focused, as indicated by either the "..." or the comments inserted.

The offer for this scenario is shown below.

```
...
m=audio 10000 RTP/SAVPF 96 9 8 0 123
a=rtpmap:96 OPUS/48000
a=rtpmap:9 G722/8000
a=rtpmap:8 PCMA/8000
a=rtpmap:0 PCMU/8000
a=rtpmap:123 telephone-event/8000
a=mid:a1
...
m=video 10000 RTP/SAVPF 98 99 100 101 102 103 104 105 106 107
a=extmap 1 urn:ietf:params:rtp-hdext:sdes:rtp-stream-id
a=rtpmap:98 VP8/90000
a=fmtp:98 max-fs=3600; max-fr=30
a=rtpmap:99 VP9/90000
a=fmtp:99 max-fs=3600; max-fr=30
a=rtpmap:100 H264/90000
a=fmtp:100 profile-level-id=42401f; packetization-mode=0
a=rtpmap:101 H264/90000
```



```
a=fmtp:101 profile-level-id=42401f; packetization-mode=1
a=rtpmap:102 H264/900000
a=fmtp:102 profile-level-id=640c1f; packetization-mode=0
a=rtpmap:103 H264/900000
a=fmtp:103 profile-level-id=640c1f; packetization-mode=1
a=rtpmap:104 H264-SVC/900000
a=fmtp:104 profile-level-id=530c1f
a=rtpmap:105 H264-SVC/900000
a=fmtp:105 profile-level-id=560c1f
a=rtpmap:106 H265/900000
a=fmtp:106 profile-id=1; level-id=93
a=rtpmap:107 H265/900000
a=fmtp:107 profile-id=2; level-id=93
a=sendrecv
a=mid:v1 (max resolution)
a=rid:1 send max-width=1280;max-height=720;max-fps=30
a=rid:2 recv max-width=1280;max-height=720;max-fps=30
...
m=video 10000 RTP/SAVPF 98 99 100 101 102 103 104 105 106 107
a=extmap 1 urn:ietf:params:rtp-hdext:sdes:rtp-stream-id
...same rtpmap/fmtp as above...
a=recvonly
a=mid:v2 (medium resolution)
a=rid:3 recv max-width=640;max-height=360;max-fps=15
...
m=video 10000 RTP/SAVPF 98 99 100 101 102 103 104 105 106 107
a=extmap 1 urn:ietf:params:rtp-hdext:sdes:rtp-stream-id
...same rtpmap/fmtp as above...
a=recvonly
a=mid:v3 (medium resolution)
a=rid:3 recv max-width=640;max-height=360;max-fps=15
...
m=video 10000 RTP/SAVPF 98 99 100 101 102 103 104 105 106 107
a=extmap 1 urn:ietf:params:rtp-hdext:sdes:rtp-stream-id
...same rtpmap/fmtp as above...
a=recvonly
a=mid:v4 (small resolution)
a=rid:4 recv max-width=320;max-height=180;max-fps=15
...
m=video 10000 RTP/SAVPF 98 99 100 101 102 103 104 105 106 107
a=extmap 1 urn:ietf:params:rtp-hdext:sdes:rtp-stream-id
...same rtpmap/fmtp as above...
...same rid:4 as above for mid:v5,v6,v7 (small resolution)...
...
```


11.2. Scalable Layers

Adding scalable layers to a session within a multiparty conference gives a selective forwarding unit (SFU) further flexibility to selectively forward packets from a source that best match the bandwidth and capabilities of diverse receivers. Scalable encodings have dependencies between layers, unlike independent simulcast streams. RIDs can be used to express these dependencies using the "depend" restriction. In the example below, the highest resolution is offered to be sent as 2 scalable temporal layers (using MRST). See [[I-D.ietf-mmusic-sdp-simulcast](#)] for additional detail about simulcast usage.

Offer:

```
...
m=audio ...same as previous example ...
...
m=video ...same as previous example ...
...same rtpmap/fmt as previous example ...
a=sendrecv
a=mid:v1 (max resolution)
a=rid:0 send max-width=1280;max-height=720;max-fps=15
a=rid:1 send max-width=1280;max-height=720;max-fps=30;depend=0
a=rid:2 recv max-width=1280;max-height=720;max-fps=30
a=rid:5 send max-width=640;max-height=360;max-fps=15
a=rid:6 send max-width=320;max-height=180;max-fps=15
a=simulcast: send rid=0;1;5;6 recv rid=2
...
...same m=video sections as previous example for mid:v2-v7...
...
```

12. IANA Considerations

This specification updates [[RFC4855](#)] to give additional guidance on choice of Format Parameter (fmt) names, and on their relation to RID restrictions.

12.1. New SDP Media-Level attribute

This document defines "rid" as SDP media-level attribute. This attribute must be registered by IANA under "Session Description Protocol (SDP) Parameters" under "att-field (media level only)".

The "rid" attribute is used to identify properties of RTP stream with in a RTP Session. Its format is defined in [Section 10](#).

The formal registration information for this attribute follows.

Contact name, email address, and telephone number

IETF MMUSIC Working Group
mmusic@ietf.org
+1 510 492 4080

Attribute name (as it will appear in SDP)

rid

Long-form attribute name in English

Restriction Identifier

Type of attribute (session level, media level, or both)

Media Level

Whether the attribute value is subject to the charset attribute

The attribute is not dependent on charset.

A one-paragraph explanation of the purpose of the attribute

The "rid" SDP attribute is used to to unambiguously identify the RTP Streams within a RTP Session and restrict the streams' payload format parameters in a codec-agnostic way beyond what is provided with the regular Payload Types.

A specification of appropriate attribute values for this attribute

Valid values are defined by the ABNF in [RFCXXXXX]

12.2. Registry for RID-Level Parameters

This specification creates a new IANA registry named "att-field (rid level)" within the SDP parameters registry. The "a=rid" restrictions MUST be registered with IANA and documented under the same rules as for SDP session-level and media-level attributes as specified in [\[RFC4566\]](#).

Parameters for "a=rid" lines that modify the nature of encoded media MUST be of the form that the result of applying the modification to the stream results in a stream that still complies with the other parameters that affect the media. In other words, restrictions always have to restrict the definition to be a subset of what is otherwise allowable, and never expand it.

New restriction registrations are accepted according to the "Specification Required" policy of [\[RFC5226\]](#), provided that the specification includes the following information:

- o contact name, email address, and telephone number
- o restriction name (as it will appear in SDP)
- o long-form restriction name in English
- o whether the restriction value is subject to the charset attribute
- o an explanation of the purpose of the restriction
- o a specification of appropriate attribute values for this restriction
- o an ABNF definition of the restriction

The initial set of "a=rid" restriction names, with definitions in [Section 5](#) of this document, is given below:

Type	SDP Name	Reference
----	-----	-----
att-field	(rid level)	
	max-width	[RFCXXXX]
	max-height	[RFCXXXX]
	max-fps	[RFCXXXX]
	max-fs	[RFCXXXX]
	max-br	[RFCXXXX]
	max-pps	[RFCXXXX]
	max-bpp	[RFCXXXX]
	depend	[RFCXXXX]

It is conceivable that a future document wants to define a RID-level restrictions that contain string values. These extensions need to take care to conform to the ABNF defined for rid-param-other. In particular, this means that such extensions will need to define escaping mechanisms if they want to allow semicolons, unprintable characters, or byte values greater than 127 in the string.

13. Security Considerations

As with most SDP parameters, a failure to provide integrity protection over the "a=rid" attributes provides attackers a way to modify the session in potentially unwanted ways. This could result in an implementation sending greater amounts of data than a recipient

wishes to receive. In general, however, since the "a=rid" attribute can only restrict a stream to be a subset of what is otherwise allowable, modification of the value cannot result in a stream that is of higher bandwidth than would be sent to an implementation that does not support this mechanism.

The actual identifiers used for RIDs are expected to be opaque. As such, they are not expected to contain information that would be sensitive, were it observed by third-parties.

14. Acknowledgements

Many thanks to review from Cullen Jennings, Magnus Westerlund, and Paul Kyzivat. Thanks to Colin Perkins for input on future payload type handing.

15. References

15.1. Normative References

- [I-D.ietf-avtext-rid] Roach, A., Nandakumar, S., and P. Thatcher, "RTP Stream Identifier Source Description (SDS)", [draft-ietf-avtext-rid-09](#) (work in progress), October 2016.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), DOI 10.17487/[RFC2119](#), March 1997, <<http://www.rfc-editor.org/info/rfc2119>>.
- [RFC3264] Rosenberg, J. and H. Schulzrinne, "An Offer/Answer Model with Session Description Protocol (SDP)", [RFC 3264](#), DOI 10.17487/RFC3264, June 2002, <<http://www.rfc-editor.org/info/rfc3264>>.
- [RFC3550] Schulzrinne, H., Casner, S., Frederick, R., and V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications", STD 64, [RFC 3550](#), DOI 10.17487/RFC3550, July 2003, <<http://www.rfc-editor.org/info/rfc3550>>.
- [RFC4566] Handley, M., Jacobson, V., and C. Perkins, "SDP: Session Description Protocol", [RFC 4566](#), DOI 10.17487/RFC4566, July 2006, <<http://www.rfc-editor.org/info/rfc4566>>.
- [RFC4855] Casner, S., "Media Type Registration of RTP Payload Formats", [RFC 4855](#), DOI 10.17487/RFC4855, February 2007, <<http://www.rfc-editor.org/info/rfc4855>>.

- [RFC5234] Crocker, D., Ed. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", STD 68, [RFC 5234](#), DOI 10.17487/[RFC5234](#), January 2008, <<http://www.rfc-editor.org/info/rfc5234>>.

15.2. Informative References

- [H264] ITU-T Recommendation H.264, "Advanced video coding for generic audiovisual services (V9)", February 2014, <<http://www.itu.int/rec/T-REC-H.264-201304-I>>.
- [I-D.ietf-mmusic-sdp-bundle-negotiation] Holmberg, C., Alvestrand, H., and C. Jennings, "Negotiating Media Multiplexing Using the Session Description Protocol (SDP)", [draft-ietf-mmusic-sdp-bundle-negotiation-35](#) (work in progress), October 2016.
- [I-D.ietf-mmusic-sdp-simulcast] Burman, B., Westerlund, M., Nandakumar, S., and M. Zanaty, "Using Simulcast in SDP and RTP Sessions", [draft-ietf-mmusic-sdp-simulcast-05](#) (work in progress), June 2016.
- [RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", [BCP 26](#), [RFC 5226](#), DOI 10.17487/RFC5226, May 2008, <<http://www.rfc-editor.org/info/rfc5226>>.
- [RFC6184] Wang, Y., Even, R., Kristensen, T., and R. Jesup, "RTP Payload Format for H.264 Video", [RFC 6184](#), DOI 10.17487/[RFC6184](#), May 2011, <<http://www.rfc-editor.org/info/rfc6184>>.
- [RFC6236] Johansson, I. and K. Jung, "Negotiation of Generic Image Attributes in the Session Description Protocol (SDP)", [RFC 6236](#), DOI 10.17487/RFC6236, May 2011, <<http://www.rfc-editor.org/info/rfc6236>>.
- [RFC7656] Lennox, J., Gross, K., Nandakumar, S., Salgueiro, G., and B. Burman, Ed., "A Taxonomy of Semantics and Mechanisms for Real-Time Transport Protocol (RTP) Sources", [RFC 7656](#), DOI 10.17487/RFC7656, November 2015, <<http://www.rfc-editor.org/info/rfc7656>>.
- [RFC7741] Westin, P., Lundin, H., Glover, M., Uberti, J., and F. Galligan, "RTP Payload Format for VP8 Video", [RFC 7741](#), DOI 10.17487/RFC7741, March 2016, <<http://www.rfc-editor.org/info/rfc7741>>.

Authors' Addresses

Peter Thatcher
Google

Email: pthatcher@google.com

Mo Zanaty
Cisco Systems

Email: mzanaty@cisco.com

Suhas Nandakumar
Cisco Systems

Email: snandaku@cisco.com

Bo Burman
Ericsson

Email: bo.burman@ericsson.com

Adam Roach
Mozilla

Email: adam@nostrum.com

Byron Campen
Mozilla

Email: bcampen@mozilla.com

