Network Working Group H. Alvestrand
Internet-Draft Google
Intended status: Standards Track April 30, 2012
Expires: November 1, 2012

RTCWEB Resolution Negotiation
draft-alvestrand-rtcweb-resolution-00

Abstract

This draft offers a proposal for a fragment of the SDP usage rules for RTCWEB: Requirements for supporting resolution negotiation.

It proposes to use SDP both for initial and within-call resolution configuration, and suggests that COP should be mentioned as an optional, not mandatory, mechanism.

Requirements Language

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 RFC 2119 [RFC2119].

Table of Contents

1. Introduction

This draft offers a proposal for a fragment of the SDP usage rules for RTCWEB: Requirements for supporting resolution negotiation.

It proposes to use SDP, [RFC6236] in particular, both for initial and within-call resolution configuration, with the "a=recv-ssrc:imageattr" mechanism from [I-D.lennox-mmusic-sdp-source-selection] as a per-stream mechanism, and suggests that Codec Operation Point (COP) specified in [I-D.westerlund-avtext-codec-operation-point] should be mentioned as an optional, not mandatory, mechanism.

2. Requirements

The relevant requirement for video resolution negotiation from the RTCWEB use cases document [I-D.ietf-rtcweb-use-cases-and-requirements] is:

- F25 The browser SHOULD use encoding of streams suitable for the current rendering (e.g. video display size) and SHOULD change parameters if the rendering changes during the session.

The scenarios from which this requirement is derived are:

- 4.2.1 Simple Video Communication Service - user changes size of video during the session.
- 4.2.2 Simple Video Communication Service, NAT/FW that blocks UDP - as above
- 4.2.3 Simple Video Communication Service, global service provider - as above
- 4.2.4 Simple Video Communication Service, enterprise aspects - as above
- 4.2.5 Simple Video Communication Service, access change - bandwidth available changes dramatically during call (wired Ethernet to 3G)
- 4.2.6 Simple Video Communication Service, QoS - as 4.2.5
- 4.2.7 Simple Video Communication Service with sharing - as above
- 4.2.8 Simple video communication service with inter-operator callling - as above

- 4.2.10 Multiparty video communication - user changes size of video

- (4.3.3 Video conferencing system with central server does NOT list F25 as a derived requirement, but notes that "it is important that the delay from when a video stream is selected for display until the video can be displayed is short").

Formulating the requirements in a form more amenable to implementation, there needs to be specified functions that allow the implementation:

- To negotiate a maximum spatial resolution for all videos at call setup time

- To negotiate a maximum temporal resolution ("frame rate") across all videos at call setup time

- To negotiate other parameters as needed to ensure that the sender will not send a stream that the receiver is unable to handle.

- To indicate the desire of the recipient for a particular spatial or temporal resolution on a particular video source, at any given time during the call

- To indicate the intent of the sender to send a video source in a particular spatial or temporal resolution, at any given time during the call

This document does not mention other requirements.

3. Initial negotiation of parameters

We assume that the normal (payload-dependent) procedures for codec negotiation are sufficient to negotiate any codec parameters needed to ensure that the decoder can handle all incoming streams.

After the initial negotiation, the following variables MUST have a known value for each RTP session (represented by one or more m= lines):

- The maximum X-resolution of any handlable video stream

- The maximum Y-resolution of any handlable video stream

- The maximum bitrate in bits per second

o The maximum framerate in frames per second

An RTCWEB client MUST support negotiation of resolution using the "imageattr" attribute, as documented in [RFC6236].

An RTCWEB client MUST support a SAR value of 1.0 (square pixels), and MAY choose to support only the 1.0 value of the "sar" attribute.

The interpretation of the negotiation is that any video stream in the m= line containing the a=imageattr attribute will have a resolution within the bounds established by the negotiation.

An RTCWEB client MUST support negotiation of the "a=framerate" attribute, as specified in [RFC4566] section 6. Note that this is an upper bound on framerate; there is no lower bound negotiated.

These bounds MAY be renegotiated over the course of the call, but MUST NOT be renegotiated to render any currently transmitted video stream out of bounds

These bounds may be supplemented by payload-specific mechanisms, and there is no guarantee that all resolutions within the bounds can be supported.

4. Per-stream declaration of desired video resolution

4.1. SDP-based per-stream declaration

An RTCWEB client MUST support per-SSRC requests for video resolutions, as described in [I-D.lennox-mmusic-sdp-source-selection]. To be precise, it MUST support the a=remote-ssrc:<ssrc> framerate: and a=remote-ssrc:<ssrc>imageattr: attributes.

This satisfies the requirement to indicate the desire of the recipient for a particular spatial or temporal resolution.

We assume that the media sent from a sender to a receiver contains enough information inside the media format to tell what the resolution and framerate is.

The bounds specified for a single stream MUST be within the bounds previously negotiated for the whole session.

This mechanism does not form a negotiation; as specified in the referenced document, it is a declaration by the recipient of what stream formats he desires, and the sender will respond by changing

the video he sends. The sender SHOULD honor the requests by the receiver.

The mere fact that a stream is within the bounds negotiated for the session is not a sufficient condition for guaranteeing that the stream will be accepted; any number of issues, including temporary lack of resources at the recipient. Thus, the sender MUST always be prepared for one or more media streams to be refused by the recipient.

4.2. COP-based per-stream declaration

An RTCWEB client MAY support the COP mechanism [I-D.westerlund-avtext-codec-operation-point] to negotiate the resolution of video within the limits established by the SDP negotiation without the need for additional SDP exchanges.

4.3. Tradeoffs discussion

This section may be deleted before publication as an RFC; its main purpose is to discuss the decision to make the SDP-based mechanism a MUST and the COP-based mechanism a MAY.

Both mechanisms work by having the receiver declare a wish for a resolution, and the sender switching to that resolution. The main differences are:

- In SDP, given the nature of the RTCWEB signalling model, the notification that a change is needed must be sent to the Javascript, which then has to use the createOffer mechanism to create a suitable SDP object, and use whatever mechanism is used for negotiation to send that request to the sender.

- In COP, the decision to signal can possibly be taken either at Javascript level or inside the browser, but once the decision is taken, all further messaging is done by the browser using RTCP packets; Javascript is not involved.

- For SDP, signalling follows the signalling path, which may be via a data channel along the media path, or may be via a completely different mechanism.

- For COP, signalling always follows the media path's return path.

- For SDP, the unbounded nature of the imageattr= specification allows a wide variety of sizes to be requested, including possibly unsuitable ones.

- For COP, the list of alternatives is created explicitly using the Operation Point mechanism.

- For SDP, the signalling transport is (presumably) done using a reliable transport

- For COP, timeout and retransmission must be implemented in the requester.

- For SDP, if imageattr= is already supported, the changes to the parsers involved are small.

- For COP, support involves embedding a completely new functionality set within the RTCP components of the RTP-supporting libraries.

- For SDP, the defining draft specifies some other mechanisms that are not mentioned here, such as "pause".

- For COP, the defining draft specifies some configurations that are not part of the RTCWEB requirements set, such as multicast.

- SDP does not consider the case of substreams for scalable video media.

- COP deos consider configuration of substreams.

- For SDP, an IPR disclosure seeming to assert RF licensing has been made against the defining draft [ipr-ssrc].

- For COP, an IPR disclosure asserting RAND (not RF) licensing has been made against the defining draft, with no assertion on which parts of the draft it applies to. [ipr-cop]

5. Usage considerations

This section notes briefly some of the situations in which resizing might be desirable.

- Change of display window size on screen (window manager resize, for instance)

- Changing the display target between a smaller and a larger window ("large current talker", for instance)

- Retargeting of the display to a different display surface ("attach external monitor", for instance)

- o Temporary CPU or GPU overload due to media stream processing conflicting with other tasks, including handling a large number of media streams

- o Recovery from such overload situations

- o <<< more? >>>

Adaptation to bandwidth changes (congestion control) is NOT included in this set, since a more correct model for this is that it should be detected by the sender and the receiver operating in tandem, and the sender should decide which flows, if any, need their bitrates changed.

Turning video streams off (mute) is also not included; use of "size = 0" has been suggested as one mechanism for video mute, but this proposed mechanism is not addressed in this memo.

6. Relation to WebRTC API constraints

It is intended that the resolution negotiation be influenced by the constraints set by the application of either mandatory or optional constraints at the WebRTC API, as registered in the registry established by [I-D.burnett-rtcweb-constraints-registry].

The following relationships hold for all attributes that the implementation intends to satisfy (note that the constraints listed here have NOT been registered yet):

video-min-height >= value of imageattr y= xyrange lower bound

video-max-height <= value of imageattr y= xyrange upper bound

video-min-framerate is not represented in SDP

video-max-framerate <= value of a=framerate attribute

video-min-aspect-ratio <= value of imageattr "par=" prange lower bound

video-max-aspect-ratio >= value of imageattr "par=" prange upper bound

The implementation is free to increase "min" values or decrease "max" values (make requirements more restrictive) and add "step" in order to fit with its implementation restrictions.

Constraints specified at PeerConnection creation time are reflected as SDP-wide values. Constraints specified when creating a MediaStream or attaching a MediaStream to a PeerConnection are reflected as ssrc-specific values.

The envisioned usage is that the application will not use the values specified by the client directly, but choose the minimum of the specified bounds and the implementation limitations of the browser, adjusted for any odd requirements of the codec or soft/hardware, and choose a representation in the SDP that adequately represents the possible configurations.

7. IANA Considerations

This document makes no request of IANA.

Note to RFC Editor: this section may be removed on publication as an RFC.

8. Security Considerations

All considerations related to normal usage of SDP apply to this memo.

9. Acknowledgements

10. References

10.1. Normative References

[I-D.burnett-rtcweb-constraints-registry]
Burnett, D., "IANA Registry for RTCWeb Media Constraints", draft-burnett-rtcweb-constraints-registry-00 (work in progress), March 2012.

[I-D.ietf-rtcweb-use-cases-and-requirements]
Holmberg, C., Hakansson, S., and G. Eriksson, "Web Real-Time Communication Use-cases and Requirements", draft-ietf-rtcweb-use-cases-and-requirements-06 (work in progress), October 2011.

[I-D.lennox-mmusic-sdp-source-selection]
Lennox, J. and H. Schulzrinne, "Mechanisms for Media Source Selection in the Session Description Protocol (SDP)", draft-lennox-mmusic-sdp-source-selection-03 (work

in progress), January 2012.

[I-D.westerlund-avtext-codec-operation-point]
Westerlund, M., Burman, B., and L. Hamm, "Codec Operation Point RTCP Extension", draft-westerlund-avtext-codec-operation-point-00 (work in progress), March 2012.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC4566] Handley, M., Jacobson, V., and C. Perkins, "SDP: Session Description Protocol", RFC 4566, July 2006.

[RFC6236] Johansson, I. and K. Jung, "Negotiation of Generic Image Attributes in the Session Description Protocol (SDP)", RFC 6236, May 2011.

10.2. Informative References

[ipr-cop] "Telefonaktiebolaget LM Ericsson (publ)'s Statement about IPR related to draft-westerlund-avtext-codec-operation-point-00 - https://datatracker.ietf.org/ipr/1701/", March 2012.

[ipr-ssrc]
"Vidyo, Inc.'s Statement about IPR related to draft-lennox-mmusic-sdp-source-selection-00 - https://datatracker.ietf.org/ipr/1170/".

Author's Address

Harald Alvestrand
Google

Email: harald@alvestrand.no