### WebRTC Video Processing and Codec Requirements
### draft-ietf-rtcweb-video-05

Abstract

   This specification provides the requirements and considerations for
   WebRTC applications to send and receive video across a network.  It
   specifies the video processing that is required, as well as video
   codecs and their parameters.

Status of This Memo

Copyright Notice

Table of Contents

## 1.  Introduction

   One of the major functions of WebRTC endpoints is the ability to send
   and receive interactive video.  The video might come from a camera, a
   screen recording, a stored file, or some other source.  This
   specification defines how the video is used and discusses special
   considerations for processing the video.  It also covers the video-
   related algorithms WebRTC devices need to support.

   Note that this document only discusses those issues dealing with
   video codec handling.  Issues that are related to transport of media
   streams across the network are specified in
   [I-D.ietf-rtcweb-rtp-usage].

## 2.  Terminology

   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].

## 3.  Pre and Post Processing

   This section provides guidance on pre- or post-processing of video
   streams.

   Unless specified otherwise by the SDP or codec, the color space
   SHOULD be sRGB [SRGB].  For clarity, this the color space indicated
   by codepoint 1 from "ColourPrimaries" as defined in [IEC23001-8].

Unless specified otherwise by the SDP or codec, the video scan
pattern for video codecs is Y'CbCr 4:2:0.

## 3.1.  Camera Source Video

This document imposes no normative requirements on camera capture;
however, implementors are encouraged to take advantage of the
following features, if feasible for their platform:

o  Automatic focus, if applicable for the camera in use

o  Automatic white balance

o  Automatic light level control

o  Dynamic frame rate for video capture based on actual encoding in
   use (e.g., if encoding at 15 fps due to bandwidth constraints, low
   light conditions, or application settings, the camera will ideally
   capture at 15 fps rather than a higher rate).

## 3.2.  Screen Source Video

If the video source is some portion of a computer screen (e.g.,
desktop or application sharing), then the considerations in this
section also apply.

Because screen-sourced video can change resolution (due to, e.g.,
window resizing and similar operations), WebRTC video recipients MUST
be prepared to handle mid-stream resolution changes in a way that
preserves their utility.  Precise handling (e.g., resizing the
element a video is rendered in versus scaling down the received
stream; decisions around letter/pillarboxing) is left to the
discretion of the application.

Note that the default video scan format (Y'CbCr 4:2:0) is known to be
less than optimal for the representation of screen content produced
by most systems in use at the time of this document's publication,
which generally use RGB with at least 24 bits per sample.  In the
future, it may be advisable to use video codecs optimized for screen
content for the representation of this type of content.

Additionally, attention is drawn to the requirements in
[I-D.ietf-rtcweb-security-arch] section 5.2 and the considerations in
[I-D.ietf-rtcweb-security] section 4.1.1.

## 4.  Stream Orientation

In some circumstances - and notably those involving mobile devices - the orientation of the camera may not match the orientation used by the encoder.  Of more importance, the orientation may change over the course of a call, requiring the receiver to change the orientation in which it renders the stream.

While the sender may elect to simply change the pre-encoding orientation of frames, this may not be practical or efficient (in particular, in cases where the interface to the camera returns pre-compressed video frames).  Note that the potential for this behavior adds another set of circumstances under which the resolution of a screen might change in the middle of a video stream, in addition to those mentioned under "Screen Sourced Video," above.

To accommodate these circumstances, RTCWEB implementations that can generate media in orientations other than the default MUST support generating the R0 and R1 bits of the Coordination of Video Orientation (CVO) mechanism described in section 7.4.5 of [TS26.114], and MUST send them for all orientations when the peer indicates support for the mechanism.  They MAY support sending the other bits in the CVO extension, including the higher-resolution rotation bits. All implementations SHOULD support interpretation of the R0 and R1 bits, and MAY support the other CVO bits.

Further, some codecs support in-band signaling of orientation (for example, the SEI "Display Orientation" messages in H.264 and H.265). If CVO has been negotiated, then the sender MUST NOT make use of such codec-specific mechanisms.  However, when support for CVO is not signaled in the SDP, then such implementations MAY make use of the codec-specific mechanisms instead.

## 5.  Mandatory to Implement Video Codec

For the definitions of "WebRTC Browser," "WebRTC Non-Browser", and "WebRTC-Compatible Endpoint" as they are used in this section, please refer to [I-D.ietf-rtcweb-overview].

WebRTC Browsers MUST implement the VP8 video codec as described in [RFC6386] and H.264 Constrained Baseline as described in [H264].

WebRTC Non-Browsers that support transmitting and/or receiving video MUST implement the VP8 video codec as described in [RFC6386] and H.264 Constrained Baseline as described in [H264].

   To promote the use of non-royalty bearing video codecs, participants in the RTCWEB working group, and any successor working groups in the IETF, intend to monitor the evolving licensing landscape as it pertains to the two mandatory-to-

implement codecs.  If compelling evidence arises that one of the
codecs is available for use on a royalty-free basis, the working
group plans to revisit the question of which codecs are required
for Non-Browsers, with the intention being that the royalty-free
codec will remain mandatory to implement, and the other will
become optional.

These provisions apply to WebRTC Non-Browsers only.  There is no
plan to revisit the codecs required for WebRTC Browsers.

"WebRTC-compatible endpoints" are free to implement any video codecs
they see fit.  This follows logically from the definition of "WebRTC-
compatible endpoint."  It is, of course, advisable to implement at
least one of the video codecs that is mandated for WebRTC Browsers,
and implementors are encouraged to do so.

## 6.  Codec-Specific Considerations

SDP allows for codec-independent indication of preferred video
resolutions using the mechanism described in [RFC6236].  WebRTC
endpoints MAY send an "a=imageattr" attribute to indicate the maximum
resolution they wish to receive.  Senders SHOULD interpret and honor
this attribute by limiting the encoded resolution to the indicated
maximum size, as the receiver may not be capable of handling higher
resolutions.

Additionally, codecs may include codec-specific means of signaling
maximum receiver abilities with regards to resolution, frame rate,
and bitrate.

Unless otherwise signaled in SDP, recipients of video streams MUST be
able to decode video at a rate of at least 20 fps at a resolution of
at least 320x240.  These values are selected based on the
recommendations in [HSUP1].

Encoders are encouraged to support encoding media with at least the
same resolution and frame rates cited above.

### 6.1.  VP8

For the VP8 codec, defined in [RFC6386], endpoints MUST support the
payload formats defined in [I-D.ietf-payload-vp8].

In addition to the [RFC6236] mechanism, VP8 encoders MUST limit the
streams they send to conform to the values indicated by receivers in
the corresponding max-fr and max-fs SDP attributes.

**6.2**.  **H.264**

   For the [H264] codec, endpoints MUST support the payload formats
   defined in [RFC6184].  In addition, they MUST support Constrained
   Baseline Profile Level 1.2, and they SHOULD support H.264 Constrained
   High Profile Level 1.3.

   Implementations of the H.264 codec have utilized a wide variety of
   optional parameters.  To improve interoperability the following
   parameter settings are specified:

   packetization-mode:  Packetization-mode 1 MUST be supported.  Other
      modes MAY be negotiated and used.

   profile-level-id:  Implementations MUST include this parameter within
      SDP and MUST interpret it when receiving it.

   max-mbps, max-smbps, max-fs, max-cpb, max-dpb, and max-br:  These

      parameters allow the implementation to specify that they can
      support certain features of H.264 at higher rates and values than
      those signalled by their level (set with profile-level-id).
      Implementations MAY include these parameters in their SDP, but
      SHOULD interpret them when receiving them, allowing them to send
      the highest quality of video possible.

   sprop-parameter-sets:  H.264 allows sequence and picture information
      to be sent both in-band, and out-of-band.  WebRTC implementations
      MUST signal this information in-band.  This means that WebRTC
      implementations MUST NOT include this parameter in the SDP they
      generate.

   H.264 codecs MAY send and MUST support proper interpretation of SEI
   "filler payload" and "full frame freeze" messages.  "Full frame
   freeze" messages are used in video switching MCUs, to ensure a stable
   decoded displayed picture while switching among various input
   streams.

   When the use of the video orientation (CVO) RTP header extension is
   not signaled as part of the SDP, H.264 implementations MAY send and
   SHOULD support proper interpretation of Display Orientation SEI
   messages.

   Implementations MAY send and act upon "User data registered by Rec.
   ITU-T T.35" and "User data unregistered" messages.  Even if they do
   not act on them, implementations MUST be prepared to receive such
   messages without any ill effects.

Unless otherwise signaled, implementations that use H.264 MUST encode
and decode pixels with a implied 1:1 (square) aspect ratio.

## 7.  Security Considerations

This specification does not introduce any new mechanisms or security
concerns beyond what the other documents it references.  In WebRTC,
video is protected using DTLS/SRTP.  A complete discussion of the
security can be found in [I-D.ietf-rtcweb-security] and
[I-D.ietf-rtcweb-security-arch].  Implementors should consider
whether the use of variable bit rate video codecs are appropriate for
their application, keeping in mind that the degree of inter-frame
change (and, by inference, the amount of motion in the frame) may be
deduced by an eavesdropper based on the video stream's bit rate.

Implementors making use of H.264 are also advised to take careful
note of the "Security Considerations" section of [RFC6184], paying
special regard to the normative requirement pertaining to SEI
messages.

## 8.  IANA Considerations

This document requires no actions from IANA.

## 9.  Acknowledgements

The author would like to thank Gaelle Martin-Cocher, Stephan Wenger,
and Bernard Aboba for their detailed feedback and assistance with
this document.  Thanks to Cullen Jennings for providing text and
review.  This draft includes text from draft-cbran-rtcweb-codec.

## 10.  References

### 10.1.  Normative 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>.

[HSUP1]     ITU-T Recommendation H.Sup1, "Application profile - Sign
            language and lip-reading real-time conversation using low
            bit rate video communication", May 1999,
            <http://www.itu.int/rec/T-REC-H.Sup1>.

[I-D.ietf-payload-vp8]
            Westin, P., Lundin, H., Glover, M., Uberti, J., and F.
            Galligan, "RTP Payload Format for VP8 Video", draft-ietf-
            payload-vp8-11 (work in progress), February 2014.

[I-D.ietf-rtcweb-overview]
          Alvestrand, H., "Overview: Real Time Protocols for
          Browser-based Applications", draft-ietf-rtcweb-overview-12
          (work in progress), October 2014.

[IEC23001-8]
          ISO/IEC 23001-8:2013/DCOR1, "Coding independent media
          description code points", 2013, <http://standards.iso.org/
          ittf/PubliclyAvailableStandards/
          c062088_ISO_IEC_23001-8_2013.zip>.

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

[RFC6184]  Wang, Y.-K., Even, R., Kristensen, T., and R. Jesup, "RTP
          Payload Format for H.264 Video", RFC 6184, May 2011.

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

[RFC6386]  Bankoski, J., Koleszar, J., Quillio, L., Salonen, J.,
          Wilkins, P., and Y. Xu, "VP8 Data Format and Decoding
          Guide", RFC 6386, November 2011.

[SRGB]     IEC 61966-2-1, "Multimedia systems and equipment - Colour
          measurement and management - Part 2-1: Colour management -
          Default RGB colour space - sRGB.", October 1999, <http://
          www.colour.org/tc8-05/Docs/colorspace/61966-2-1.pdf>.

[TS26.114]
          3GPP TS 26.114 V12.8.0, "3rd Generation Partnership
          Project; Technical Specification Group Services and System
          Aspects; IP Multimedia Subsystem (IMS); Multimedia
          Telephony; Media handling and interaction (Release 12)",
          December 2014, <http://www.3gpp.org/DynaReport/26114.htm>.

## 10.2.  Informative References

[I-D.ietf-rtcweb-rtp-usage]
          Perkins, C., Westerlund, M., and J. Ott, "Web Real-Time
          Communication (WebRTC): Media Transport and Use of RTP",
          draft-ietf-rtcweb-rtp-usage-06 (work in progress),
          February 2013.

[I-D.ietf-rtcweb-security-arch]
          Rescorla, E., "WebRTC Security Architecture", draft-ietf-
          rtcweb-security-arch-09 (work in progress), February 2014.

   [I-D.ietf-rtcweb-security]
              Rescorla, E., "Security Considerations for WebRTC", draft-
              ietf-rtcweb-security-06 (work in progress), January 2014.

Author's Address

   Adam Roach
   Mozilla
   \
   Dallas
   US

   Phone: +1 650 903 0800 x863
   Email: adam@nostrum.com