RTCWEB Video Codec Requirements

draft-ietf-rtcweb-video-02

Honolulu, HI, USA

Thursday, November 12th, 2014

Change Summary (1/2)

- Clarified that transport issues are out of scope, removed all transport discussions.
- Proposes sRGB as default colorspace.
- Changes camera behaviors to suggestions.
- Clarifies that screen-source video may result in resolution changes.
- Proposes CVO to convey camera orientation.

Change Summary (2/2)

- Adds SHOULD requirement for RFC6236 indication of receive resolutions
- Adds MUST requirement for 10fps @ 320x240 *unless* otherwise signaled
- Makes codec-specific parameters like max-fr and max-fs mandatory to honor.
- Specifies H.264 packetization-mode 1, mandates use of profile-level-id

Open Issue 1: sRGB

- Not worth spending mic time on – if this doesn’t make you happy, propose something else on the list.

Open Issue 2: Screen Source Video Metadata

- Do we want to define additional metadata to indicate whether a stream is sourced from a camera versus a screen capture?

- This would allow the receiving party to tune, e. g., output filters.

Open Issue 3: Orientation Indication

- Currently says **SHOULD** send and **SHOULD** interpret CVO, at least with 90° increments.
- **MAY** support 5.625° increments (note: these are compatible with 90° increments)
- Is this the right normative level?

Open Issue 4: VP8 Filter Support

- It has been asserted that VP8 already requires support of “bilinear” and “none” filters.
- Can someone send a citation to the list?

Open Issue 5: H.264 SEI

- Do we need to require the handling of specific SEI messages?
- One example that has been raised is freeze-frame messages. Do we need to mandate support?
- Are there other messages we should care about?

Open Issue 6: MTI Video Codec

- Hold on a second.
- This deserves its own cover slide.

RTCWEB Mandatory-to-Implement Video Codec

Honolulu, HI, USA

Thursday, November 12th, 2014

How We're Going to Discuss This

- I'm going to present the "novel plan" that I posted to the list.
- First, we'll allow people to ask clarifying questions only.
  - **You may express assent to have me throw something at you by coming to the microphone during this period to express support for or opposition to the plan.**
- Once we're all clear on the plan, I'm leaving the front of the room, and the discussion of merits can begin.

Definitions (from draft-ietf-rtcweb-overview)

- A **WebRTC User Agent** (also called a WebRTC UA or a **WebRTC browser**) is something that conforms to both the protocol specification and the Javascript API defined [in the W3C WebRTC and Media Capture specs].
- A **WebRTC [Non-Browser]**[1] is something that conforms to the protocol specification, but does not claim to implement the Javascript API.
- A **WebRTC-compatible endpoint** is an endpoint that is capable of successfully communicating with a WebRTC endpoint, but may fail to meet some requirements of a WebRTC endpoint.

[1] Called a "WebRTC Device" in draft-ietf-rtcweb-overview-12

Just to get this out of the way

- "*WebRTC-compatible endpoints*" are free to implement any video codecs they see fit, if any.
- This follows logically from our current definition of "*WebRTC-compatible endpoint*," and, as such, is simply an observation rather than part of the proposal.

Novel Plan

1. WebRTC **User Agents (Browsers)** MUST implement both VP8 and H.264.
2. WebRTC **Devices (Non-Browsers)** MUST implement both VP8 and H.264. If compelling evidence arises that one of the codecs is available for use on a royalty-free basis, such as all IPR declarations known for the codec being of (IETF) Royalty-Free or (ISO) type 1, the IETF will change this normative statement to indicate that only that codec is required.

“Let no one think that flexibility and a predisposition to compromise is a sign of weakness or a sell-out.”

Paul Kagame

We Need to Solve This

- The industry is stalled and this is one of the reasons
- We are all united – “make rtcweb successful”
- No decision sends a message to app developers to - wait

Something for Everyone

Community	Whats in it for them
App developers	Maximum flexibility – browsers will work with whatever tech you may already have or are constrained to use
Open source community	Path towards a pure RF situation – some workarounds today (openh264) or hardware
VP8 Technology proponents	Validation of importance of VP8, adoption as MTI
Folks with H.264 gear	Ability to interoperate their existing products with browsers and webRTC devices

The Trajectory is Towards “Both”

- Firefox now doing both
- Many chipsets doing both (demos)

My Ask

Let’s make rtcweb successful together

MTI Codec - Nov 2014

As seen from a Google viewpoint

Short version

+1

Our WebRTC Goals

- Interoperable, high quality real time comms
- Ability to do Royalty-Free implementations

Our Context

- Uncertainty about MTI is Bad for WebRTC
  - Users want interoperability
  - Users want to deploy Real Soon
  - Uncertainty makes them choose other solutions
- We need to get this settled

This Seems To Work

- Interoperability across all WebRTC endpoints
- RF technology incorporated
- Path towards a pure RF situation
- Compatibility with a wide set of non-WebRTC devices (“WebRTC compatible”)

We Can Live With This

- After 3+ years, this seems the closest we've been to a consensus
- If the WG can find consensus for 2 MTI codecs, we will accept that decision
- We believe this is a compromise we are able to live with.

The Great codec Compromise

- Process:
  - Question to get a sense of who’ll participate in the consensus process
  - Question to ferret out new/unresolved technical issues
  - Questions

The Great codec Compromise

Please stand (or signal in the jabber chat) if you will be part of that consensus process for this question. If you're here to read email or watch the show, we want to know that your sitting throughout this isn't expressing opinions for the consensus process.

The Great codec Compromise

Q#1: If you support adding the original text as proposed by Adam to the draft please hum now.

Q#2: If you do not support adding the original text as proposed by Adam to the draft please hum now.