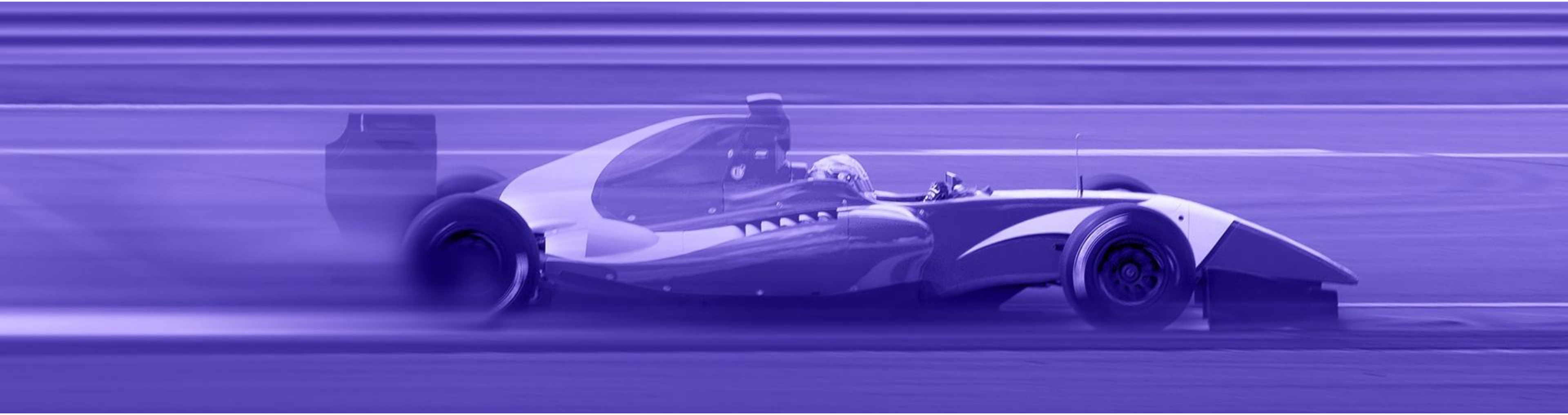
millicast



It's time to WHIP WebRTC into shape

WebRTC

AV1

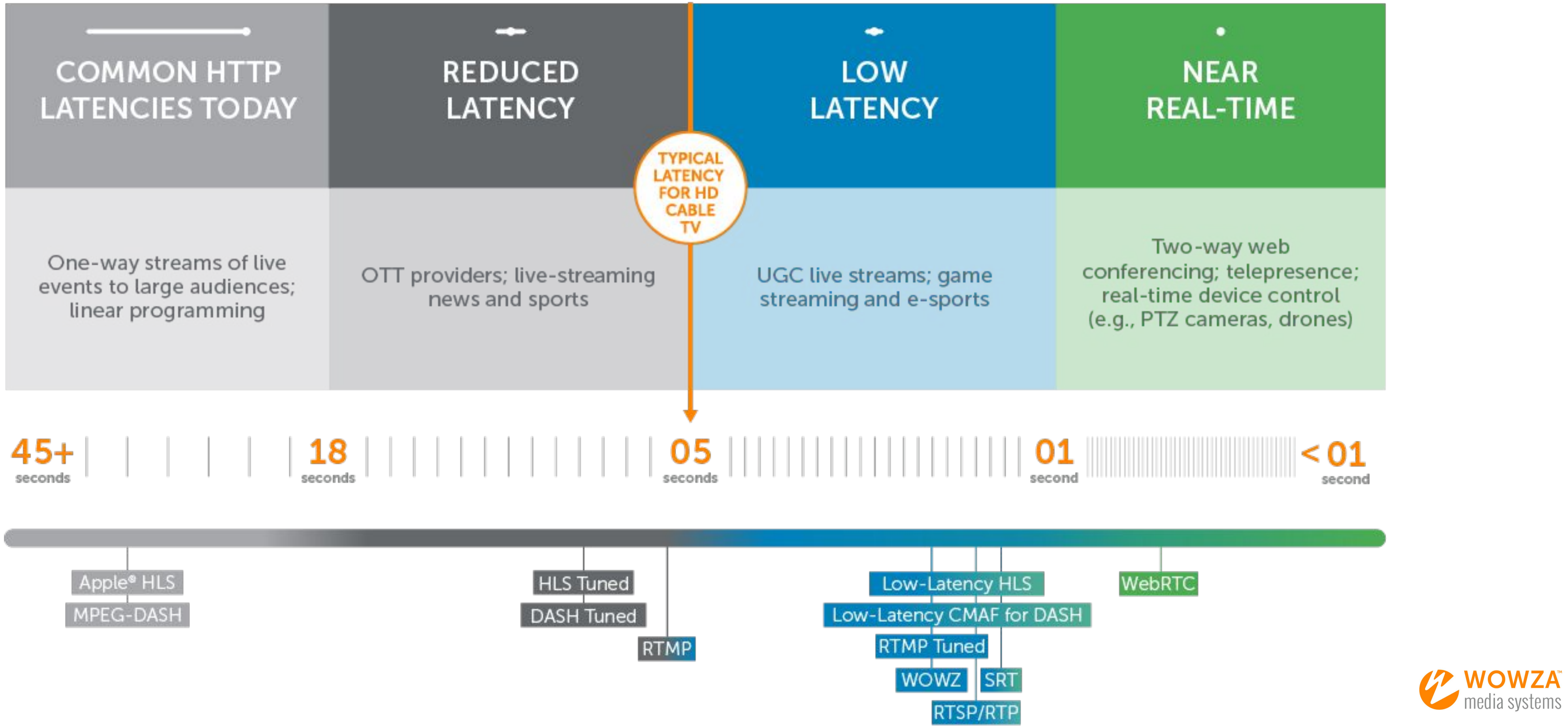
WHY AREN'T WE USING WEBRTC?

NEGATIVE PERCEPTION

- Because it was focused on voIP and Peer-to-Peer use cases at launch
- It's limited to a few concurrent viewers and doesn't scale
- It's associated with poor "web" quality, not for broadcast
- It requires "coding" to use

WE CARE ABOUT LATENCY

AGAIN



WEBRTC WAS MADE FOR THIS

RTC = REAL-TIME

WebRTC

INTERACTIVE

1-WAY

2-WAYS

DYNAMIC UPGRADE

REAL-TIME

200-500 MILLISECONDS

ACROSS THE GLOBE

HIGH QUALITY

WEB QUALITY

BROADCAST QUALITY

(10-12 BITS HDR 4:4:4 VIDEO)

(SURROUND SOUND)

WEB SCALE

MILLIONS OF VIEWERS

PER STREAM

THOUSANDS OF

CONCURRENT STREAMS

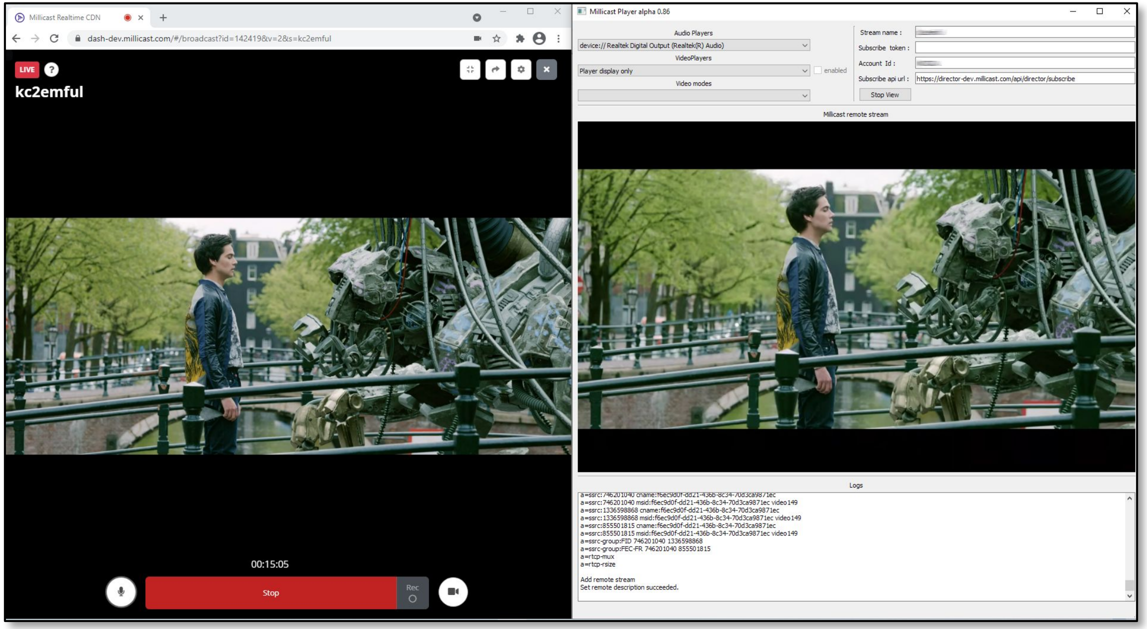
SECURE

TRANSPORT ENCRYPTION

END-TO-END ENCRYPTION

REAL-TIME AV1 SVC

CHROME CANARY M90



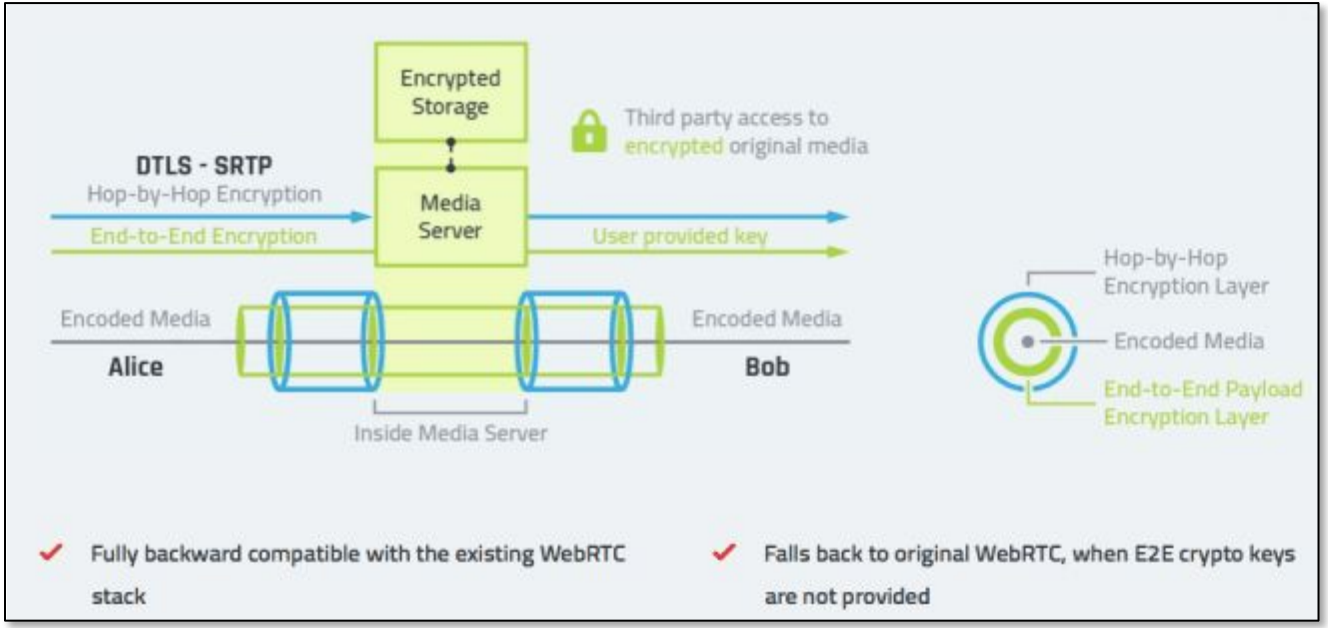
GOOGLE.COM/CHROME/CANARY/

NO RE-ENCODE
ABR SENDER-SIDE

WEBRTCBYDRALEX.COM/INDEX.PHP/2021/01/29/REAL-TIME-AV1-SVC-UNLEASHING-THE-TRUE-POWER-OF-WEBRTC/

TRUE END-TO-END ENCRYPTION

SECURE FRAMES (SFRAME)



WEBRTCBYDRALEX.COM/INDEX.PHP/2020/03/30/SECURE-FRAMES-SFRAMES-END-TO-END-MEDIA-ENCRYPTION-WITH-WEBRTC-NOW-IN-CHROME/

I WANT "REAL-TIME"

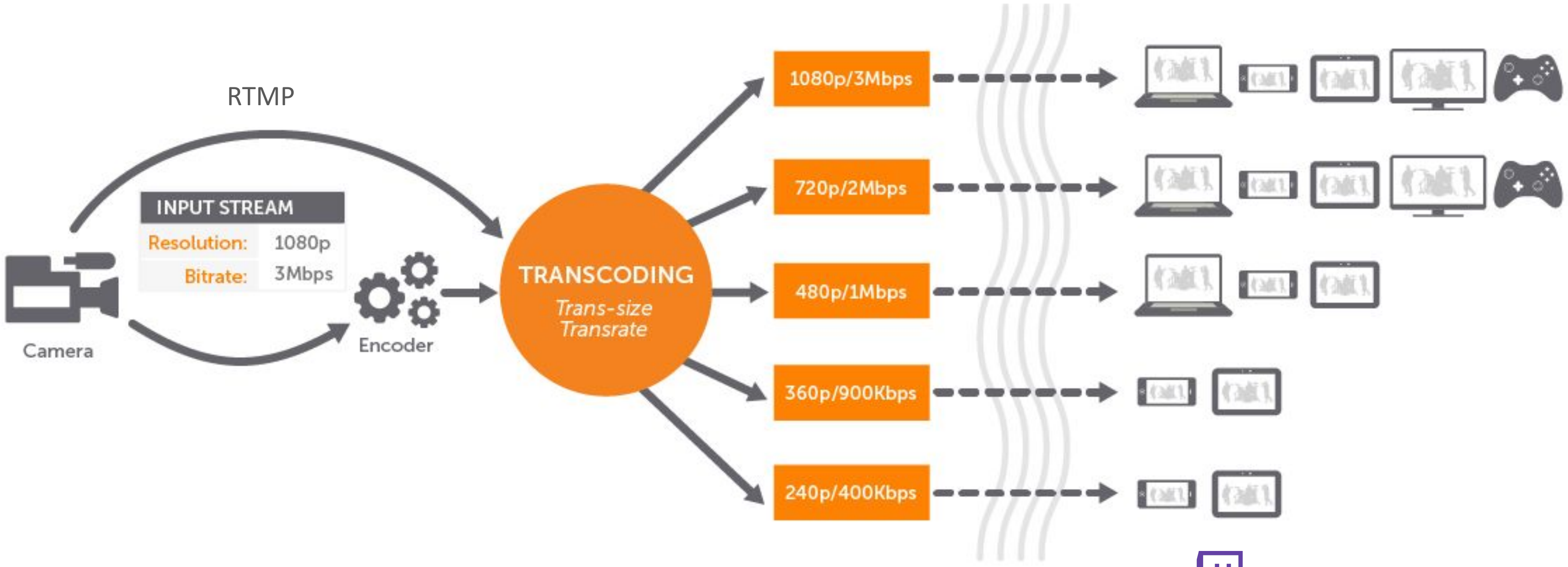
BUT CAN I BRING MY TOYS?

Open Broadcaster Software

Wirecast

vMix

RTMP IS STILL UBIQUITOUS
FOR LIVE STREAMING



MUX · YouTube Live · vimeo · livestream · twitch · WOWZA media systems

WE NEED BROADCAST-QUALITY FEATURES
WITH CONSUMER-GRADE WORKFLOWS

SVC

NETWORK RESILIENCE

E2EE

CONTENT PROTECTION

HDR 10

BROADCAST QUALITY

AV1

REAL-TIME ENCODING

WHIP (WEBRTC HTTP INGESTION PROTOCOL)

Problem to solve

- WebRTC is the best media transport protocol for real-time streaming.
- While other media transport could be used for ingest, using webrtc for both ingest and delivery allows:
  - Working on browsers.
  - Avoiding protocol translation, which adds delay and implementation complexity.
  - Avoiding transcoding by sharing common codecs.
  - Using webrtc features end to end.
- However, there is no standard signalling protocol available to pair with it:
  - SIP or XMPP are not designed to be used in broadcasting/streaming services.
  - RTSP, which is based on RTP is not compatible with WebRTC SDP offer/answer model
- Consequences:
  - Each WebRTC streaming services requires implementing a custom ad-hoc protocol.

We need a reference signalling protocol.

WHIP (WEBRTC HTTP INGESTION PROTOCOL)

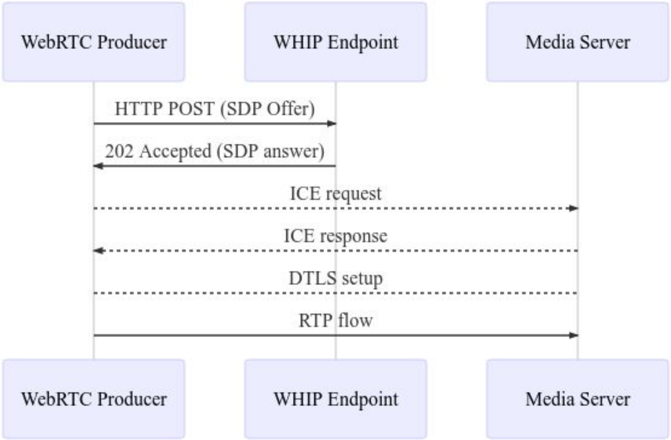
Requirements

- Must be simple to implement, as easy to use as current RTMP URI.
- Support the specific ingest use case, which is a subset of webrtc possible use cases:
  - Only needs to support unidirectional flows.
  - Server is assumed to not be behind NAT (having a public IP or deployed in same private network as publisher)
  - No need to support renegotiations.
- Fully compliant with WebRTC and RTCWEB specs for the given use case.
- Must support authentication.
- Usable both in web browsers and in native encoders.
- Lower the requirements on both hardware encoders and broadcasting tools by reducing optionalities.
- Supports load balancing and redirections.

WHIP (WEBRTC HTTP INGESTION PROTOCOL)

The solution

- HTTP POST for exchanging and SDP O/A.
- Connection state is controlled by ICE/DTLS states:
  - ICE consent freshness [RFC7675] be used to detect abrupt disconnection.
  - DTLS teardown for session termination by either side.
- Authentication and authorization is supported by the Authorization HTTP header with a bearer token as per [RFC6750].
- Support HTTP redirections for load balancing.



WHIP (WEBRTC HTTP INGESTION PROTOCOL)

THE MAGIC BULLET FOR ENCODERS

WHIP is a way to standardize the WebRTC signaling layer and establish the WebRTC connection using a simple HTTP request/response. It’s already in:

millicast

JANVS WEBRTC SERVER

But many still want hardware for physical SDI and HDMI capture:

BirdDog

videon

TERADEK

https://www.ietf.org/archive/id/draft-ietf-wish-whip-00.html

What is still missing in WebRTC for professional media?

AUDIO

- Multiopus is not an official standard, only supported by Chrome and it is hidden.
- NetEQ has issues with music:

  https://fosdem.org/2021/schedule/event/webrtc_musicians/attachments/slides/4601/export/events/attachments/webrtc_musicians/slides/4601/fosdem2021_webrtc_musicians.pdf

- Integration between WebRTC and WebAudio has implementation issues on Chrome:

  https://docs.google.com/presentation/d/1dwgo4N86CriLrRLjVCD5nFDsq_OC8GNALAtCA7e0pIo/edit#slide=id.gecec54b1ef_1_15

- Lack of Webrtc and WebVTT integration.

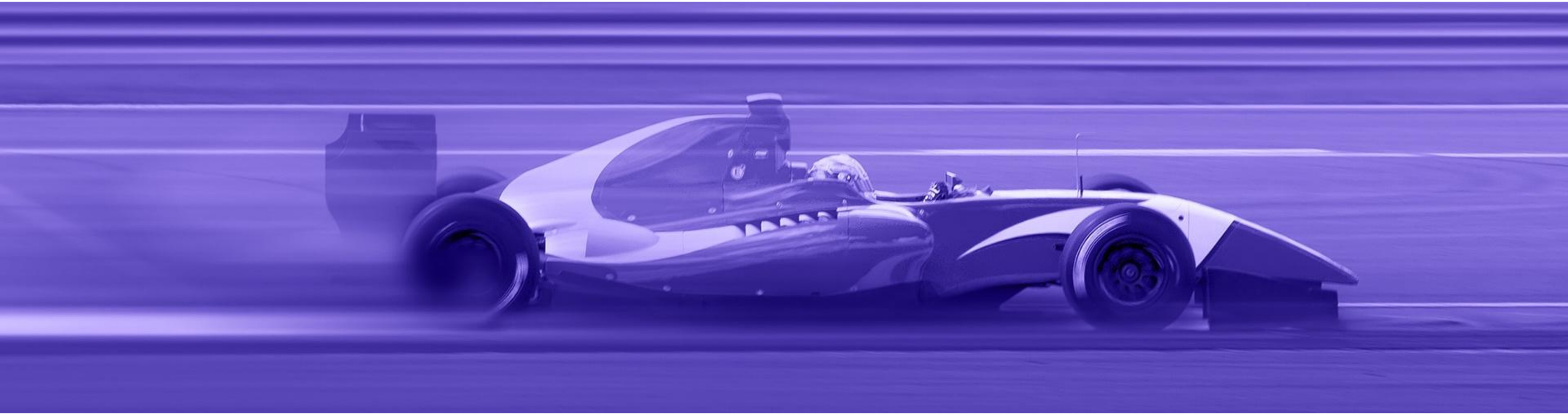
What is still missing in WebRTC for professional media?

VIDEO

- SVC extension only supported by Chrome/Edge as an experimental feature.
- AV1 only supported by Chrome.
- VP9 profile 2 only supported by Chrome/Edge (and only on receiving), experimental support in Safari.
- playoutdelayhint only supported by Chrome /Edge.
- abs-capture-time hidden and stat only supported by Chrome.
- Alpha not supported, but will be in webcodecs.

It's time to WHIP WebRTC into shape

millicast



THANK YOU