

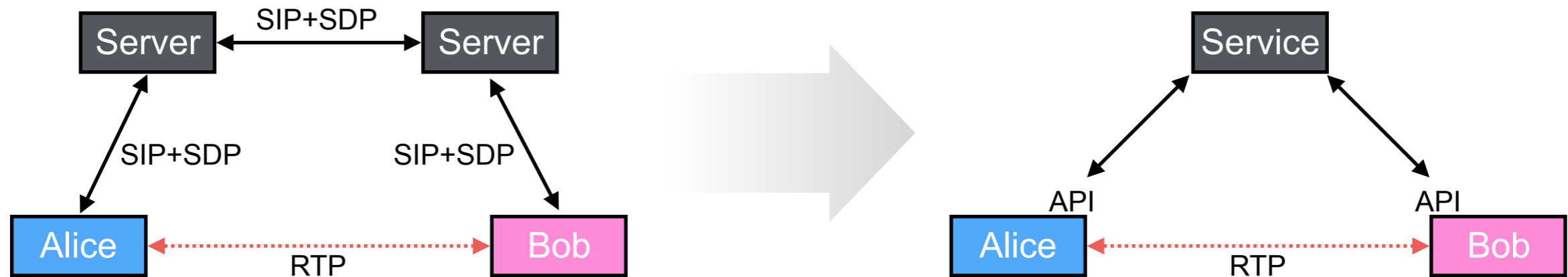


WebRTC: IETF Standards Update

September 2016

Colin Perkins

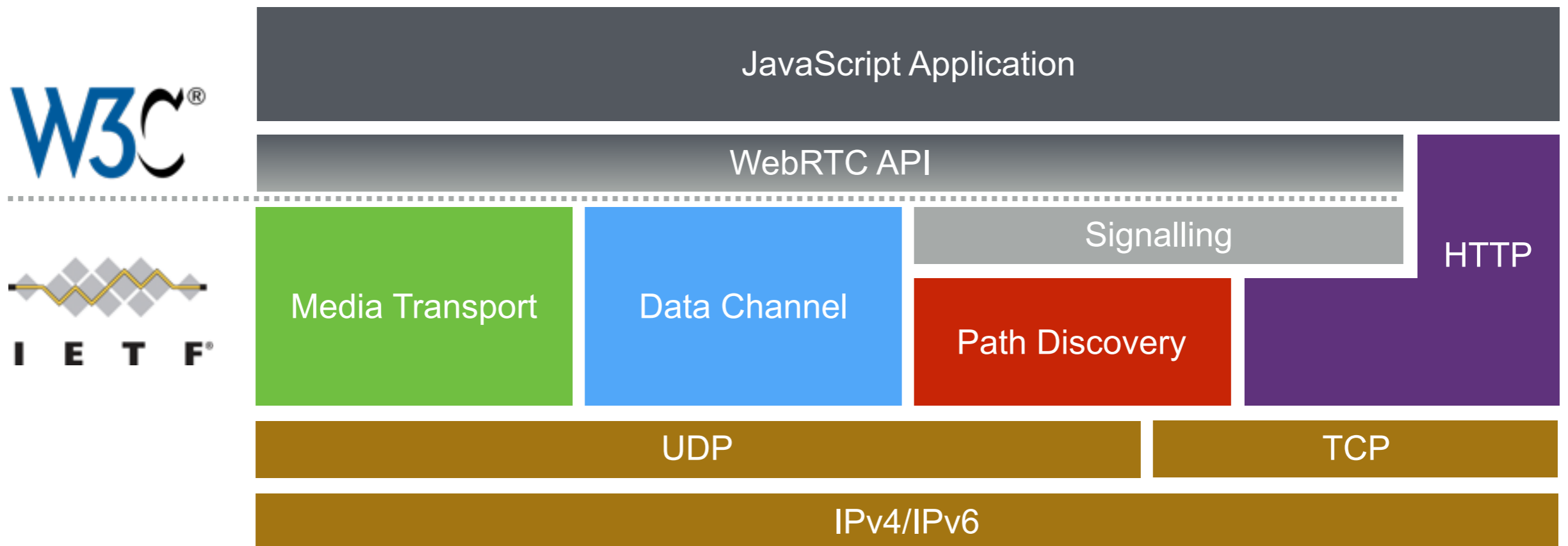
WebRTC Goals



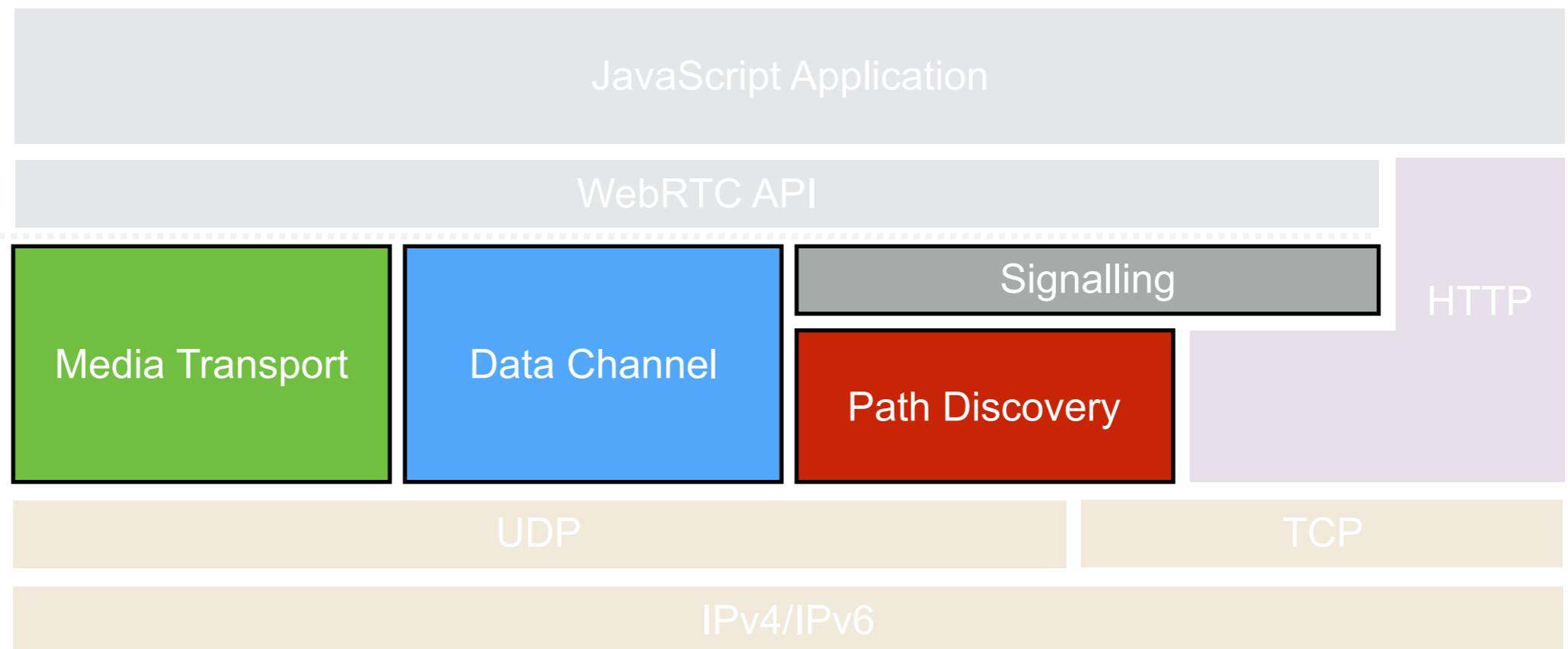
The SIP framework is overly complex and rigid – hinders innovation

Embed standard media stack (RTP, ICE, etc.) into browsers, expose a standard control API rather than a standard signalling protocol – innovate above that API

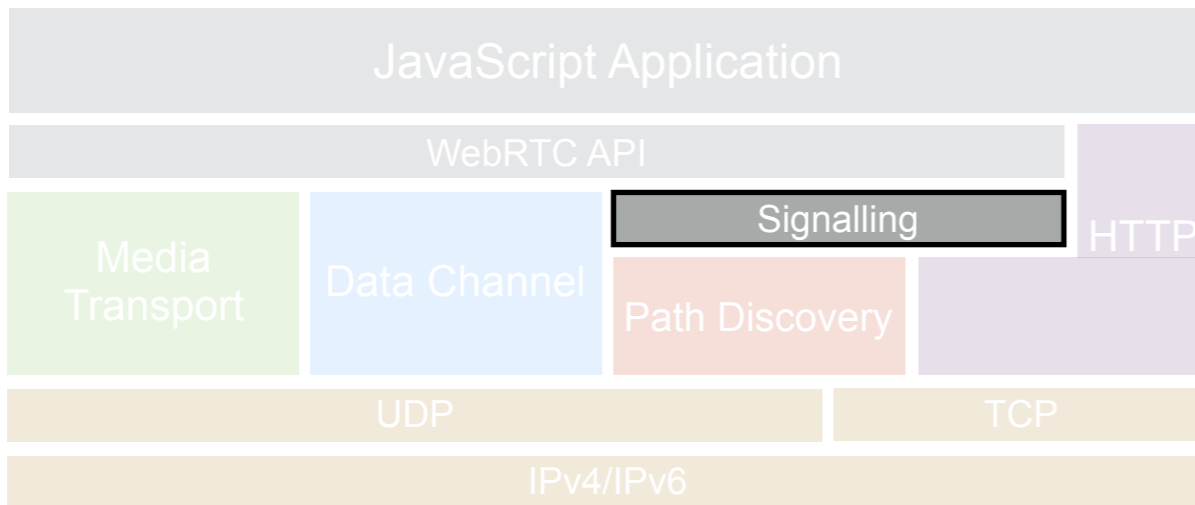
WebRTC



WebRTC in IETF



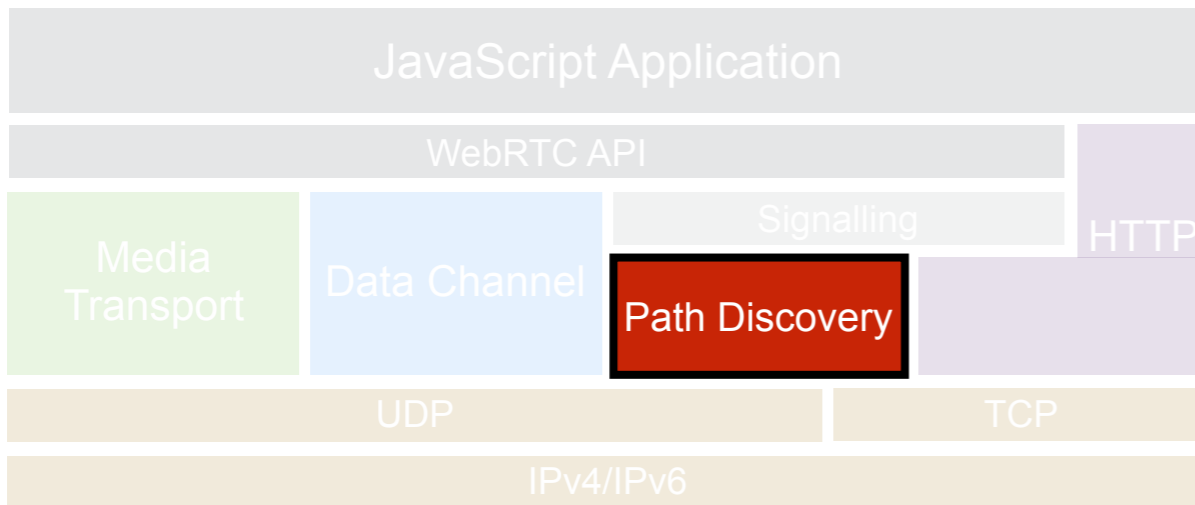
WebRTC in IETF: Signalling



Draft	Status
draft-ietf-rtcweb-use-cases-and-requirements	RFC 7478
draft-ietf-rtcweb-overview	In progress
draft-ietf-rtcweb-security	In progress
draft-ietf-rtcweb-security-arch	In progress
draft-ietf-rtcweb-jsep	In progress
draft-ietf-rtcweb-sdp	In progress
draft-ietf-rtcweb-constraints-registry	In progress
draft-ietf-mmusic-sdp-bundle-negotiation	WG last call
draft-ietf-mmusic-msid	With RFC Editor
draft-ietf-mmusic-sdp-mux-attributes	IESG review
draft-ietf-mmusic-rid	In progress
draft-ietf-mmusic-sdp-simulcast	WG last call
draft-ietf-mmusic-mux-exclusive	With RFC Editor
draft-ietf-mmusic-4572-update	WG last call
draft-ietf-mmusic-dtls-sdp	WG last call

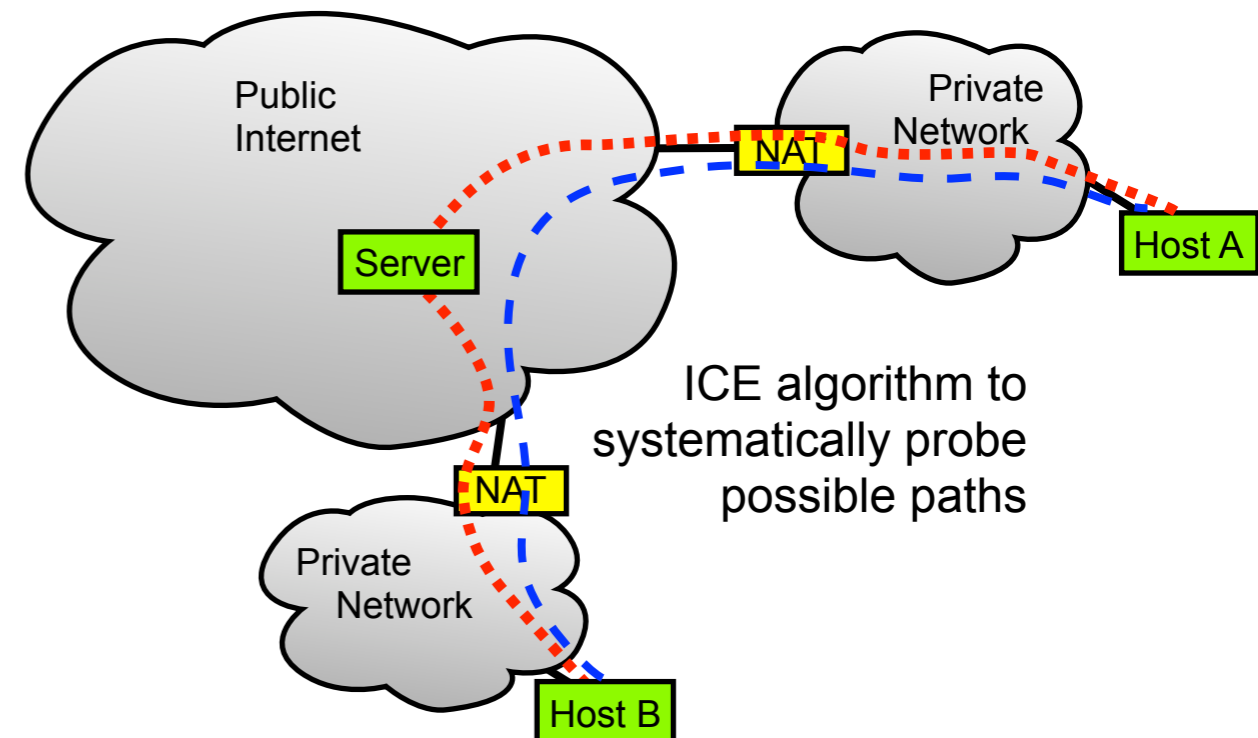
- JSEP and SDP exposed via API
- JSEP extracts SDP offer-answer out into reusable API component
 - SDP not easy to process with JavaScript
 - Extension and modification model poorly specified – simple applications are simple, but over-complicates other scenarios
 - An ORTC-like API might be cleaner?
- SDP BUNDLE extension groups WebRTC traffic on single port:
 - RTP, Data Channel, STUN, DTLS
 - Complexity in identifying m= lines when bundled → msid, rid
 - Complexity in handling bundled attributes, signalling multiplexed flows
- Major issues resolved, but details remain open...

WebRTC in IETF: Path Discovery



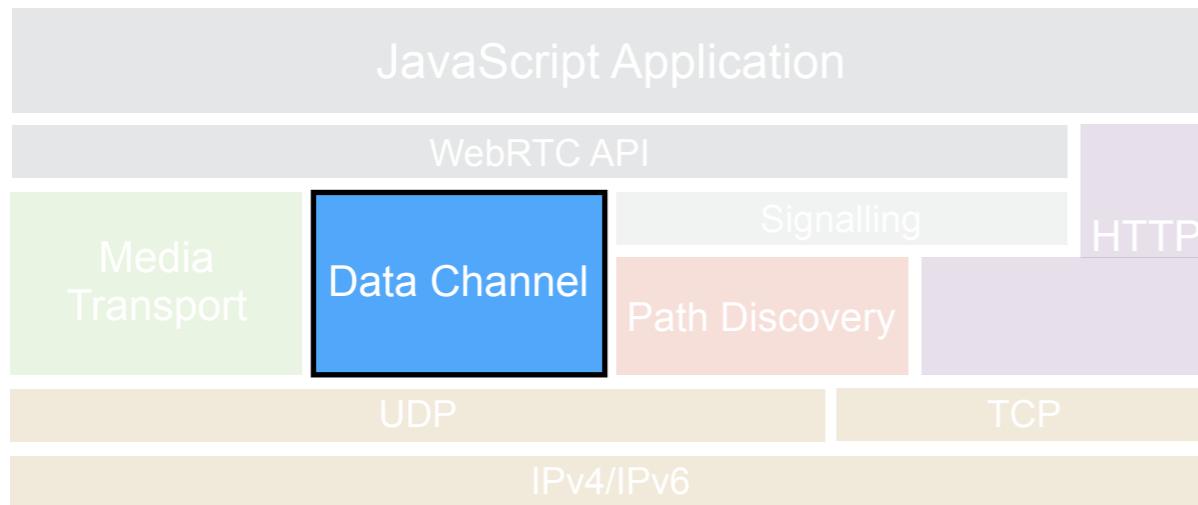
Draft	Status
draft-ietf-rtcweb-transport	Approved
draft-ietf-rtcweb-ip-handling	In progress
draft-ietf-rtcweb-stun-consent-freshness	RFC 7675
draft-ietf-mmusic-sctp-sdp	In progress
draft-ietf-ice-dualstack-fairness	IESG review
draft-ietf-ice-rfc5245bis	In progress
draft-ietf-ice-trickle	In progress
draft-ietf-rtcweb-alpn	With RFC Editor
draft-ietf-tsvwg-rtcweb-qos	With RFC Editor
draft-ietf-mmusic-ice-sip-sdp	In progress
draft-ietf-tram-stunbis	In progress

- STUN and TURN to discover NAT bindings and relay traffic



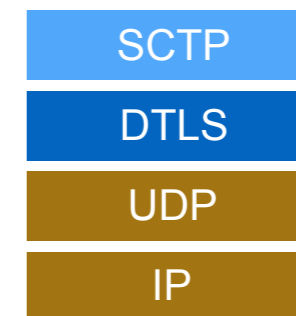
- Privacy concern around local IP address leak resolved
- Ongoing ICE revisions based on deployment experience with SIP

WebRTC in IETF: Data Channel



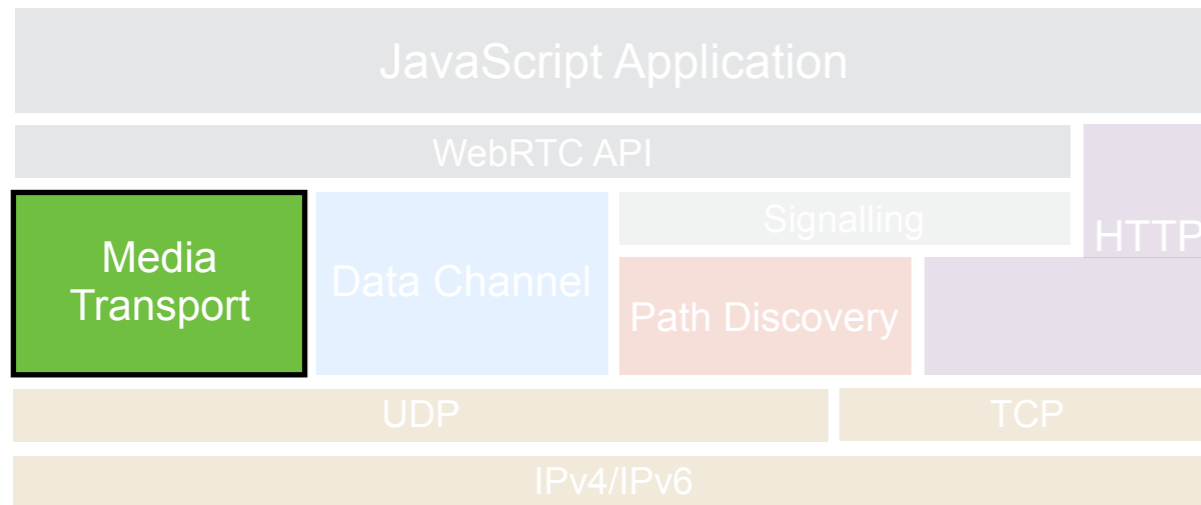
Draft	Status
draft-ietf-rtcweb-data-channel	With RFC Editor
draft-ietf-rtcweb-data-protocol	With RFC Editor
draft-ietf-tsvwg-sctp-dtls-encaps	With RFC Editor
draft-ietf-tsvwg-sctp-ndata	In progress

- Direct peer-to-peer data between browsers; no server involvement
- SCTP in secure UDP tunnel:



- UDP tunnel ensures deployability but prevents SCTP multihoming

WebRTC in IETF: Media Transport

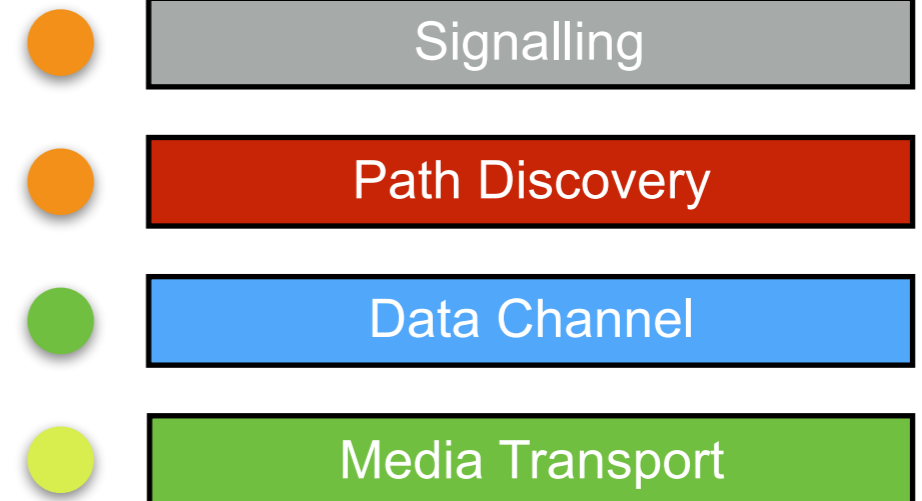


- Audio and video codecs
 - Opus, G.711, and DTMF digits required; AMR recommended
 - H.264 and VP8 required
 - Support for other codecs optional
- Modern RTP and RTCP stack
 - Bundled media on a single UDP port
 - Multiparty multimedia group conferencing – details around multiparty RTP sessions with different media types clarified
 - Secure RTP with DTLS-SRTP handshake
 - Detailed reception quality feedback, with NACK, retransmission, and FEC possible
 - Circuit breaker and congestion control for safe deployment on constrained paths

Draft	Status
draft-ietf-rtcweb-rtp-usage	With RFC Editor
draft-ietf-rtcweb-audio	RFC 7874
draft-ietf-rtcweb-audio-codecs-for-interop	RFC 7875
draft-ietf-rtcweb-video	RFC 7742
draft-ietf-rtcweb-fec	In progress
draft-ietf-avtcore-rtp-circuit-breakers	With RFC Editor
draft-ietf-avtcore-rtp-multi-stream	With RFC Editor
draft-ietf-avtcore-rtp-multi-stream-optimisation	With RFC Editor
draft-ietf-avtcore-multi-media-rtp-session	With RFC Editor
draft-ietf-avtext-rid	IESG review
draft-ietf-avtext-sdes-hdr-ext	RFC 7941
draft-ietf-payload-flexible-fec-scheme	In progress
draft-ietf-avtcore-rfc5761-update	IESG review

WebRTC in IETF: Status Summary

- Media transport and data channel essentially complete
- Path discovery and signalling protocols near completion – resolving details



- Why are the standards taking so long?
 - IPR around choice of mandatory to implement codec
 - Decoupling SDP offer/answer from SIP to form JSEP, and complexity of resulting API interactions
 - Complexity of bundled media: signalling and feature interaction; corner cases around use of RTP and RTCP with multiple simultaneous media types; demultiplexing and QoS with several protocols on a single port
 - Revisions to STUN, TURN, and ICE

Challenges and Future Directions

- How might WebRTC evolve in future?
 - Quality of service support
 - Congestion control
 - ECN and ensuring low latency
 - Multicast and IPTV
 - Relation to new path layer protocols

Challenges and Future Directions

- How might WebRTC evolve in future?

- Quality of service support ————— **Differential QoS on a single UDP flow**

- Congestion control
- ECN and ensuring low latency
- Multicast and IPTV
- Relation to new path layer protocols

Applications set different DSCP code points for the different media types and the data channel, and for different flow priorities

- RFC 7657 and draft-ietf-tsvwg-rtcweb-qos-18

Do QoS-marked flows traverse the network?

- Forwarding behaviour for some DSCP values is implementation defined – unclear what's typical
- DSCP field can be re-written or zeroed at network boundaries
- Networks can discard packets with certain DSCP values due to security or business concerns

Unclear whether QoS support offers any benefits for interdomain use – or indeed, whether it hurts media quality

Challenges and Future Directions

- How might WebRTC evolve in future?

- Quality of service support
- Congestion control
- ECN and ensuring low latency
- Multicast and IPTV
- Relation to new path layer protocols

————— **RTP Circuit Breaker**

New algorithm – does it work in the wide range of scenarios where WebRTC is deployed?

Congestion control for interactive media

Algorithms under development: Google Congestion Control, NADA, SCReAM

- Evaluation at an early stage – unclear any of these are stable in all desired scenarios, or with different types of cross traffic

Generic feedback mechanism under development

- Early work – unclear RTCP feedback can meet the timeliness requirements with reasonable overhead

Initial WebRTC deployments will have evolving congestion control – does this matter?

Challenges and Future Directions

- How might WebRTC evolve in future?

- Quality of service support
- Congestion control
- ECN and ensuring low latency
- Multicast and IPTV
- Relation to new path layer protocols

Explicit Congestion Notification

Desire to move away from loss as congestion signal

- High latency → must fill queue to trigger loss
- Disruptive to user experience

Use of ECN with AQM allows smaller queues

- Requires support from network (CoDel, PIE, ...)
- Requires support from circuit breaker
- Requires support from congestion controller
- Incrementally deployable

IETF L4S and TCP Prague experiments use ECT(1) with radically different congestion control: potentially much lower latency, but disruptive change

- Congestion response: $1/\sqrt{p} \rightarrow 1/p$
- Not interoperable: dual queue AQM required

Response to ECN-CE mark should be less aggressive than response to packet loss

Challenges and Future Directions

- How might WebRTC evolve in future?

- Quality of service support
- Congestion control
- ECN and ensuring low latency
- Multicast and IPTV
- Relation to new path layer protocols

Support for IP Multicast in WebRTC

Two approaches to video streaming:

- HTTP adaptive streaming – browser native format
- Multicast IPTV – designed for managed networks

WebRTC media stack is very similar to the multicast IPTV media stack:

- Missing MPEG-2 codec and payload format
- Missing source-specific multicast support
- Missing rapid channel change extensions

Incremental additions → not complex

Should WebRTC support multicast, so browsers can act as native IPTV clients?

- Better scaling for live streams
- Lower latency

Longer term: media interworking and interoperability?

- Different delivery modes need different encoding
- Hand-off between devices and delivery modes is difficult and non-scalable

Challenges and Future Directions

- How might WebRTC evolve in future?
 - Quality of service support
 - Congestion control
 - ECN and ensuring low latency
 - Multicast and IPTV
 - Relation to new path layer protocols ————— **Substrate protocols and the path layer**

Biggest challenge with WebRTC was making bundled media work

- Significant impact on RTP, congestion control, QoS
- Extremely complex signalling

New work in IETF: SPUD prototype and PLUS BoF

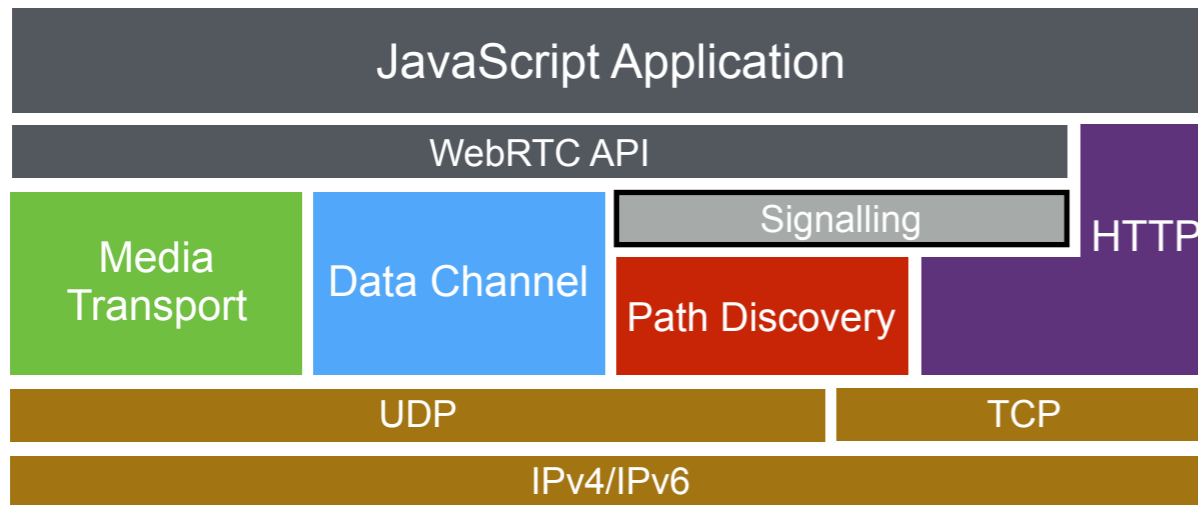
- Common UDP-based substrate layer on which new transport protocols can be run
- A secure path layer, with scope for edge-network communication

Can/should WebRTC migrate to run over this layer?

Challenges and Future Directions

- How might WebRTC evolve in future?
 - Quality of service support
 - Congestion control
 - ECN and ensuring low latency
 - Multicast and IPTV
 - Relation to new path layer protocols
- A transport-oriented viewpoint – what else?
 - Signalling APIs – ORTC vs. SDP-based approaches
 - Simplified JavaScript libraries
 - Monitoring and management tools and interfaces

Conclusions



- WebRTC provide a good baseline – a flexible, evolvable, framework
- Core IETF standards essentially done
- Clear path to evolve the network with lower latency, more adaptive media

Interesting challenges remain, but WebRTC is ready for deployment