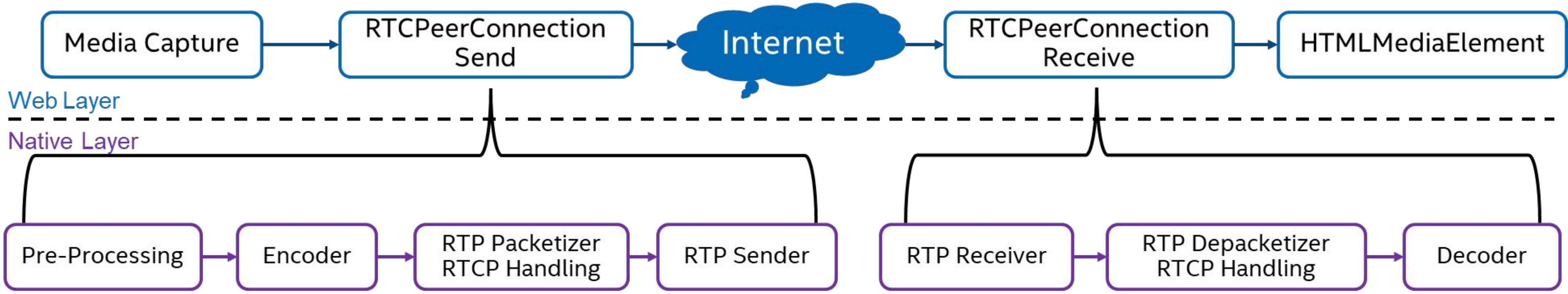
Streaming with WebTransport, WebCodecs and WebAssembly

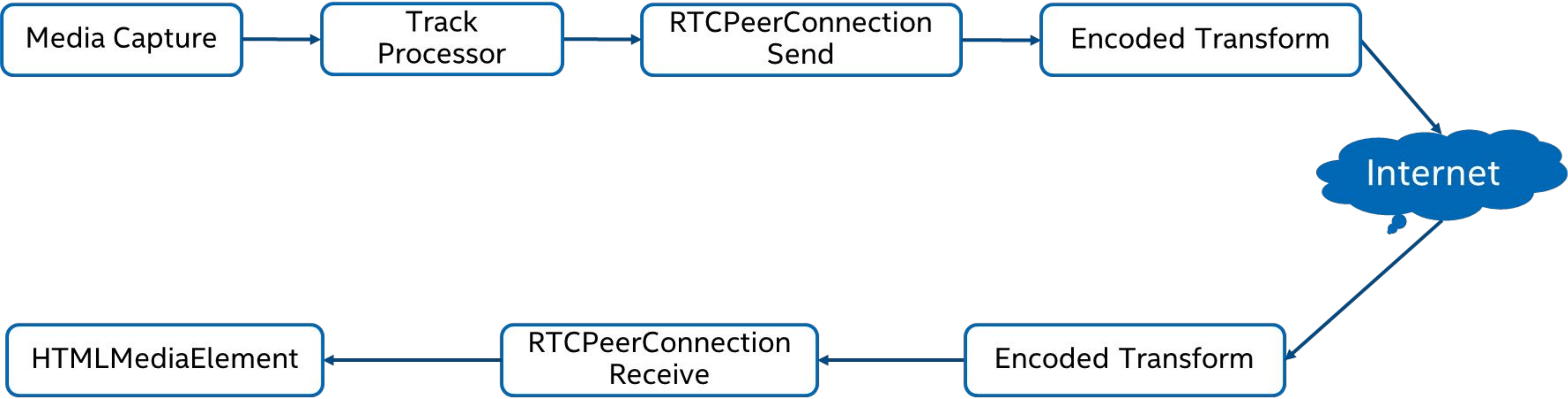
Jianjun Zhu 诸剑俊
jianjun.zhu@intel.com
Intel Web Platform Engineering

WebRTC



Media Capture
RTCPeerConnection Send
Internet
RTCPeerConnection Receive
HTMLMediaElement
Web Layer
Native Layer
Pre-Processing
Encoder
RTP Packetizer RTCP Handling
RTP Sender
RTP Receiver
RTP Depacketizer RTCP Handling
Decoder

WebRTC with Breakout Box and Encoded Transform

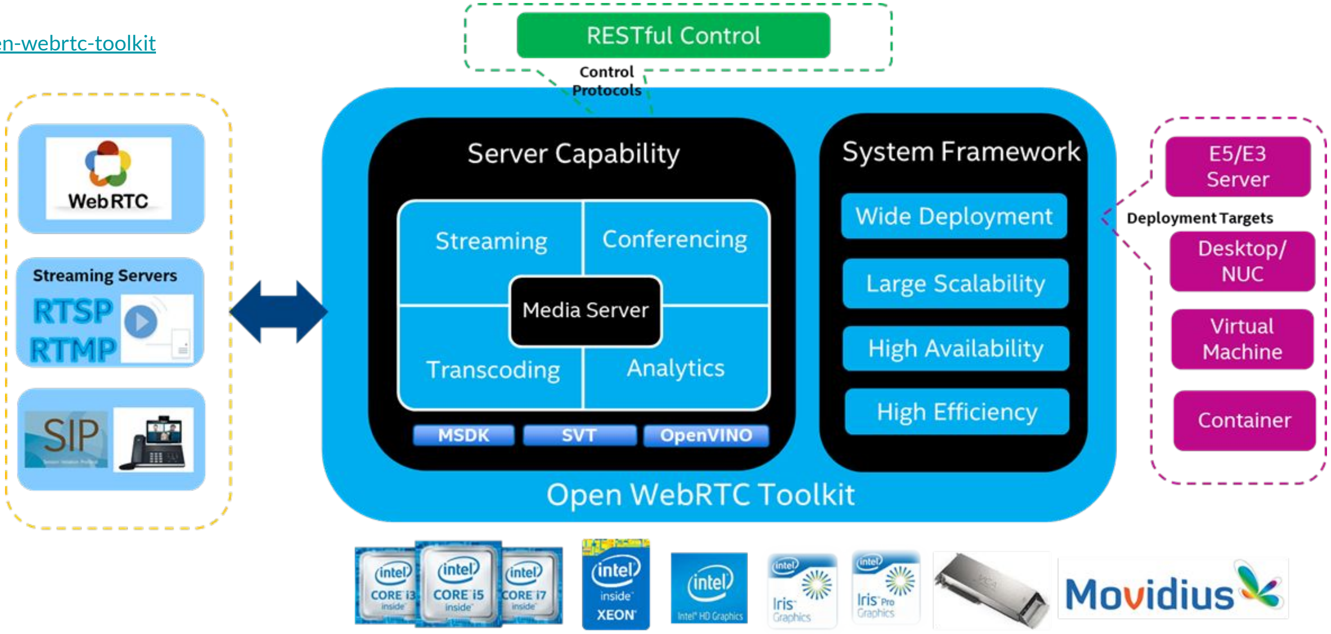


Open WebRTC Toolkit

Open source end-to-end real-time media delivery toolkit based on WebRTC

基于WebRTC的开源端到端 实时媒体传输框架

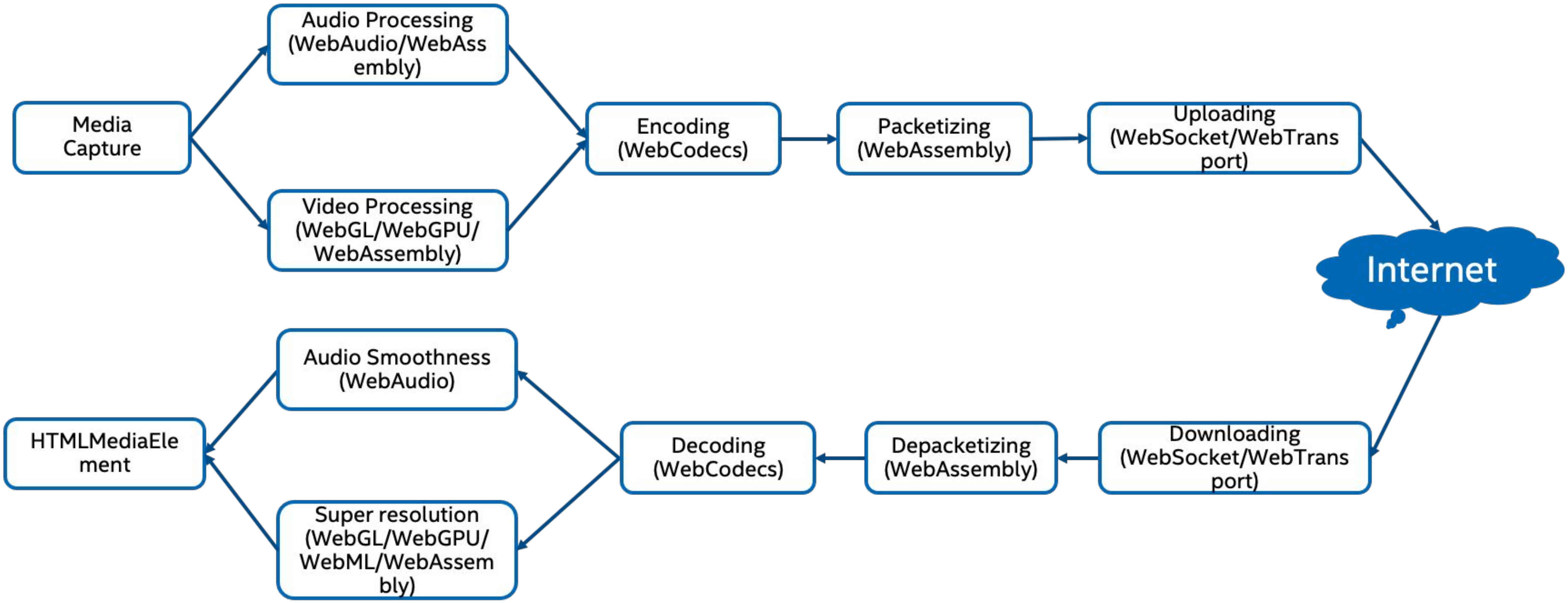
https://github.com/open-webrtc-toolkit



Enhancements? 需要的增强？

- Client / Server architecture 客户端/服务器架构
    - Removing signaling module 移除信令模块
    - Removing ICE module 移除ICE模块
- Custom algorithms, protocols 自定义算法和协议

Customized video conferencing 可自定义的视频会议系统



WebTransport

WebTransport over HTTP/3 over QUIC 基于QUIC和HTTP/3的WebTransport

Supports both ordered/reliable transport and unordered/unreliable transport

支持有序并可靠的 传输和非有序非可靠的 传输

Congestion control 拥塞控制

Multiplexing 多路复用

Only supported by Chrome 目前只有Chrome能支持

Integrate with WebCodecs and other web technologies

需要和WebCodec等其它web技术一起用来构建音 视频系统

* WebTransport discussed here is WebTransport over HTTP/3 这里仅讨论基于HTTP/3的WebTransport
* WebTransport is not a replacement of WebRTC. WebTransport

WebCodecs

Access to built-in media encoders and decoders 访问内置的编解码器

Low latency encoding and decoding 低延迟的编解码

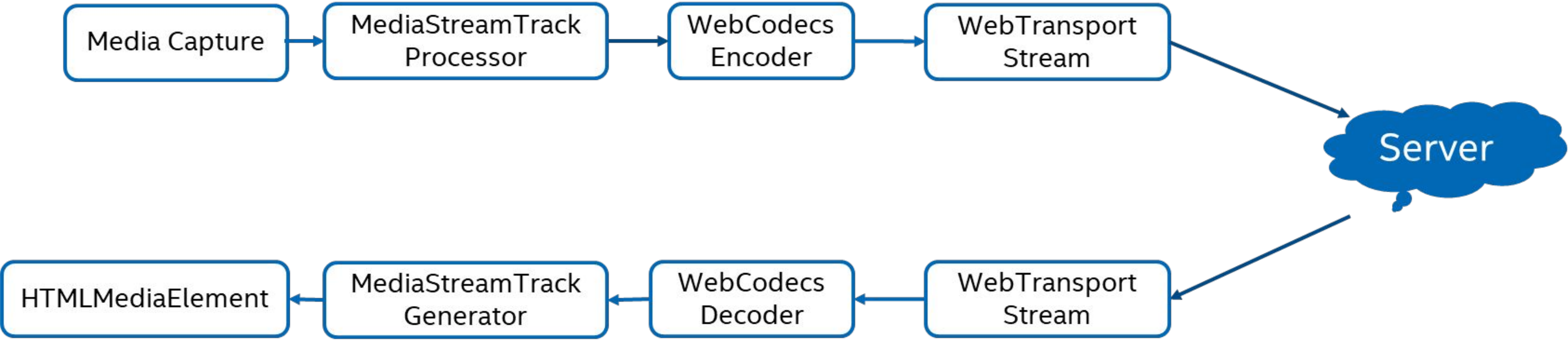
New codecs support 支持新编码标准

Hardware acceleration support 支持硬件加速

WebTransport Stream and Datagram

	WebTransport Stream (ordered/reliable) 有序/可靠	WebTransport Datagram (unordered/unreliable) 非有序/不可靠
Easy to implement 易于开发	✓	
BWE feedback for encoder 编码器接受带宽预测反馈	✓	✓
Deal with packet loss 丢包处理	Recover in transport level, may increase latency 传输层支持丢包恢复，可能会引入延迟	Flexible to use custom algorithms and drop old packets 应用层灵活应用自定义算法或者丢弃过时的数据包

Client Pipeline – WebTransport Stream



WebTransport Stream Latency

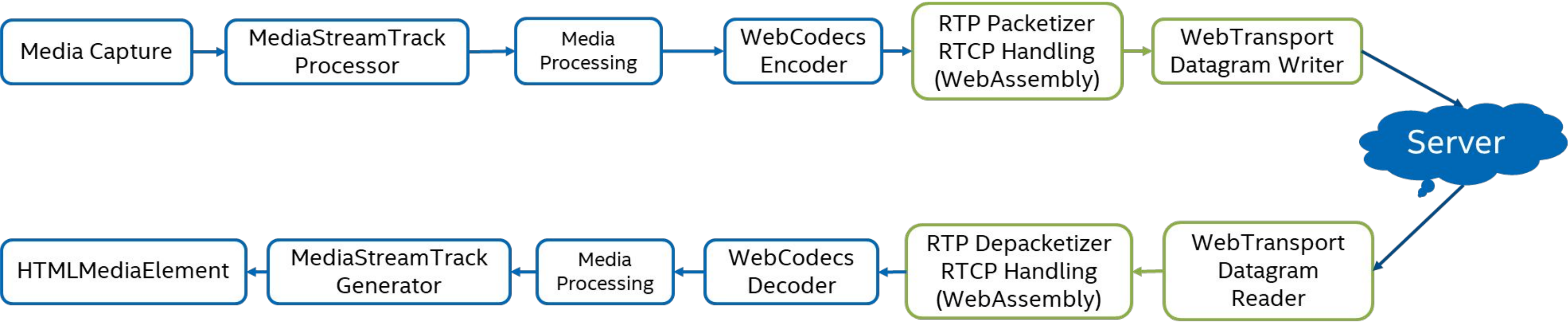
Latency: about 50ms screen to screen latency in a LAN network with packet loss rate < 3%.

延迟:丢包率小于3%的局域网络大约50毫秒屏幕到屏幕延迟

Weak network? 弱网环境?

- Discard outdated packets? 丢弃过时数据包？
- FEC and other techniques from WebRTC? 重用WebRTC中的FEC等技术？

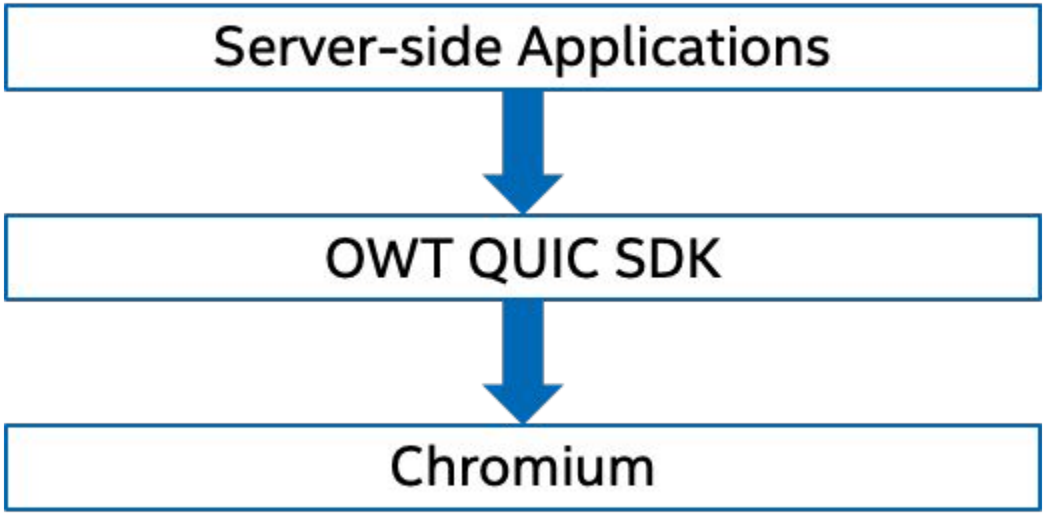
Client Pipeline – WebTransport Datagram



RTP/RTCP

- JavaScript API for WebRTC RTP/RTCP module
- WebAssembly (https://github.com/open-webrtc-toolkit/owt-client-native/pull/566)

Server-side WebTransport Implementation



Send and receive data with WebTransport 需要通过WebTransport发送和接收数据的应用程序

Clean WebTransport APIs without std containers 提供WebTransport API且没有过多的依赖

WebTransport over HTTP/3 implementation 实现基于HTTP/3的WebTransport

Performance optimization

H.264 temporal SVC HW encoding 基于时序的H.264分层编码的硬件加速

VP9 kSVC HW encoding VP9 kSVC的硬件加速

Zero-copy pipeline between media capture and encoder 视频捕捉到编码的0数据拷贝通路

WebCodecs HEVC support in Chromium (both hvcC & annex-b) Chromium中WebCodecs HEVC的支持

WebCodecs multiple encodings latency optimization 多个WebCodecs编码器同时工作的延迟优化

IPC reduction 减少进程间通讯

WebTransport BYOB reader is working in progress 正在实现WebTransport BYOB reader

* SVC: Scalable Video Coding 分层编码
* HEVC: High Efficiency Video Coding, a.k.a.: H.265 高效率视频编码，也称为H.265
* BYOB: Bring your own buffer 自带预先分配的缓存

References

WebTransport

https://www.w3.org/TR/webtransport/

WebCodecs

https://www.w3.org/TR/webcodecs/

Streaming with WebTransport & WebCodecs

https://www.w3.org/2018/12/games-workshop/slides/21-webtransport-webcodecs.pdf

W3C WebTransport TPAC 2021

https://docs.google.com/presentation/d/1SG1FUvn9UKGx_gAqWnCzPHlsIXkDUXctZPs7blfqS5k/edit#slide=id.p