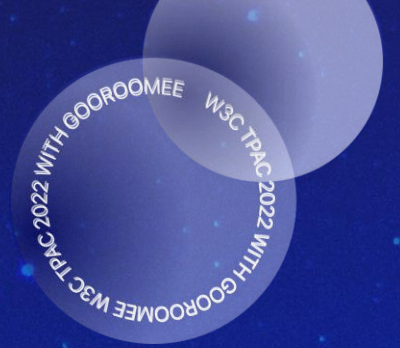
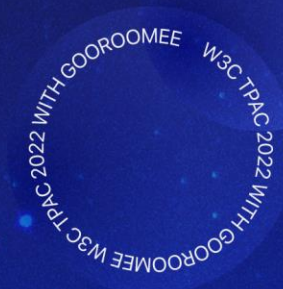


W3C TPAC 2022 투아보기

부제: 웹 서비스를 성장 시키는 HTML5 표준과 활용 바로알기





▶ Session #4.

석윤루미 WebRTC & WebTransport 표준 동향

WebRTC

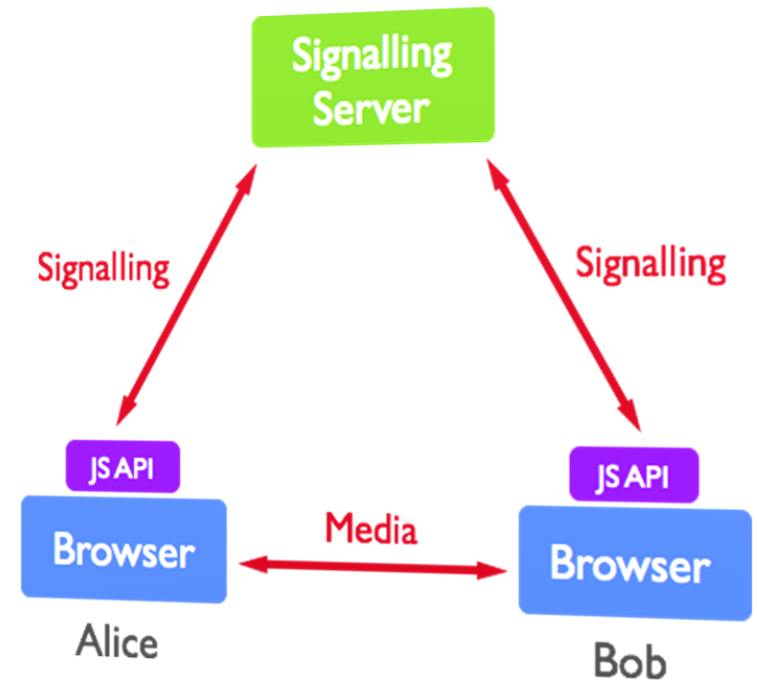
- WebRTC란?
 - 개요
 - 아키텍처와 연결
- WebRTC 표준 동향
 - WebRTC In W3C TPAC 2022
- WebRTC 활용
- WebRTC 정리

WebTransport

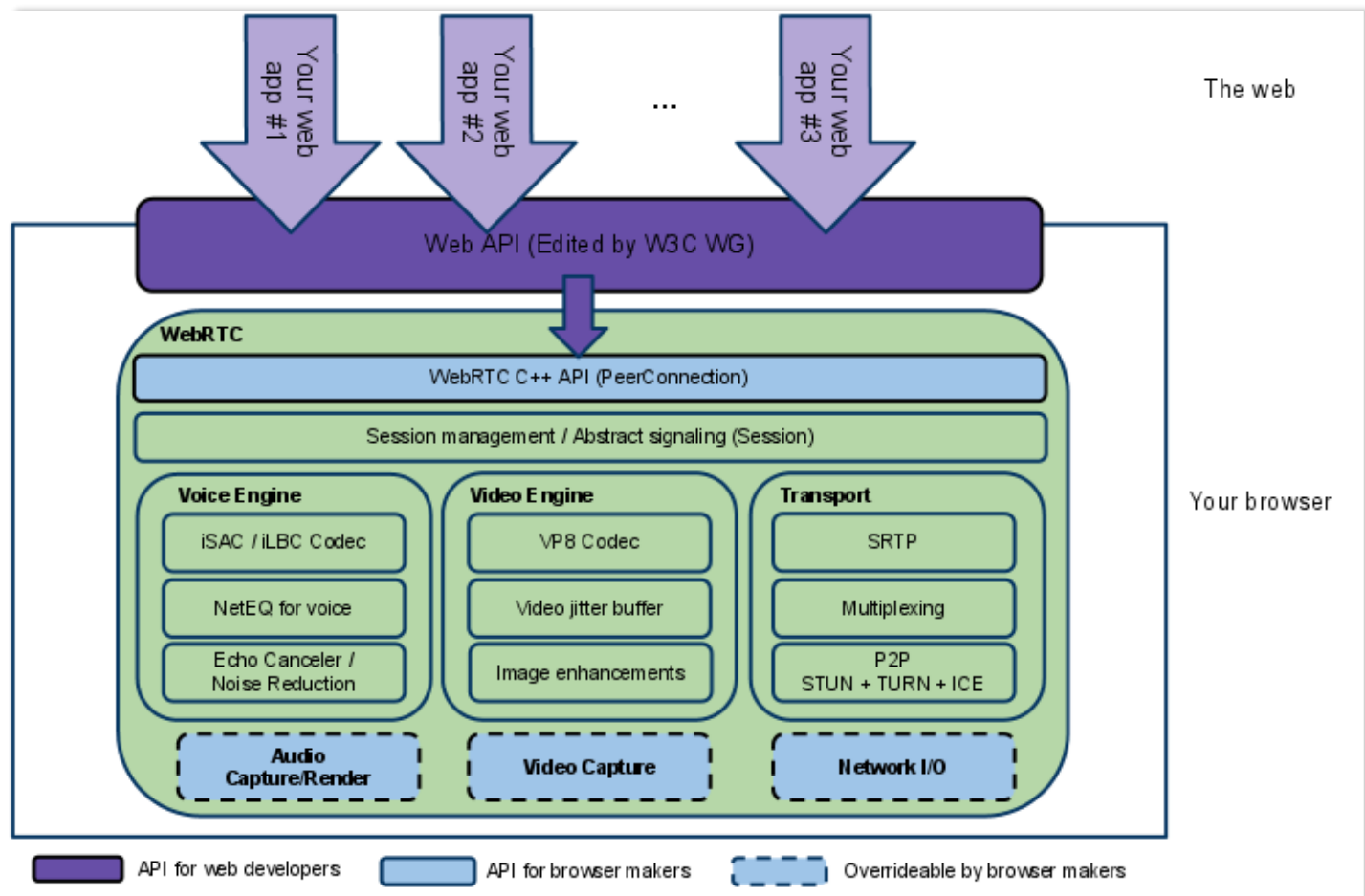
- WebTransport란?
 - 개요
 - 프로토콜
 - WebSocket과 비교
 - API
- WebTransport 표준 동향
 - WebTransport In W3C TPAC 2022
- WebTransport 정리

WebRTC란?

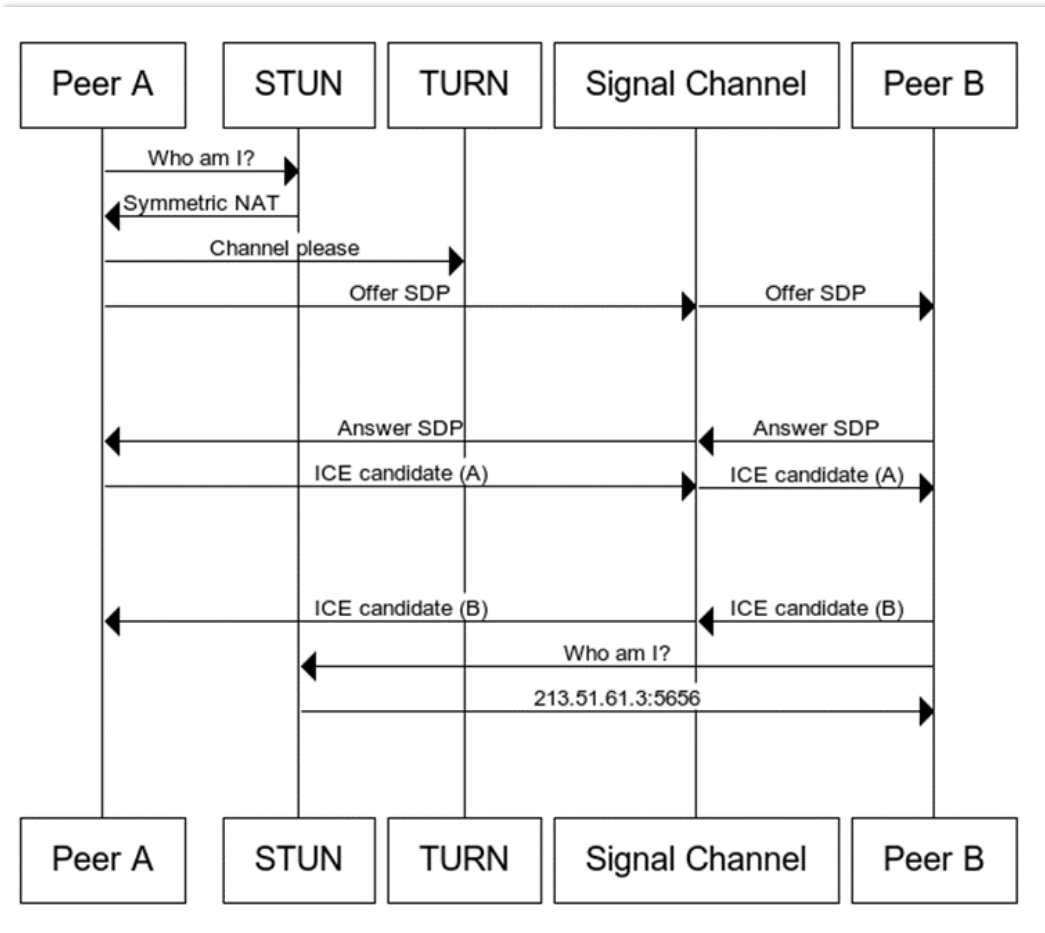
- ✓ **WebRTC**(Web Real-Time Communication)은 웹 어플리케이션과 사이트가
중간자 없이 브라우저 간에 오디오나 영상 미디어를 포착하고 마음대로 스트림할 뿐 아니라,
임의의 데이터도 교환할 수 있도록 하는 기
- ✓ WebRTC를 구성하는 일련의 표준들은 플러그인이나
제 3자 소프트웨어 설치 없이 종단 간 데이터 공유와 다자 화상통화를 가능하게 함



WebRTC - 아키텍처와 연결



WebRTC - 아키텍처와 연결



WebRTC 표준 동향 - WebRTC In W3C TPAC 2022

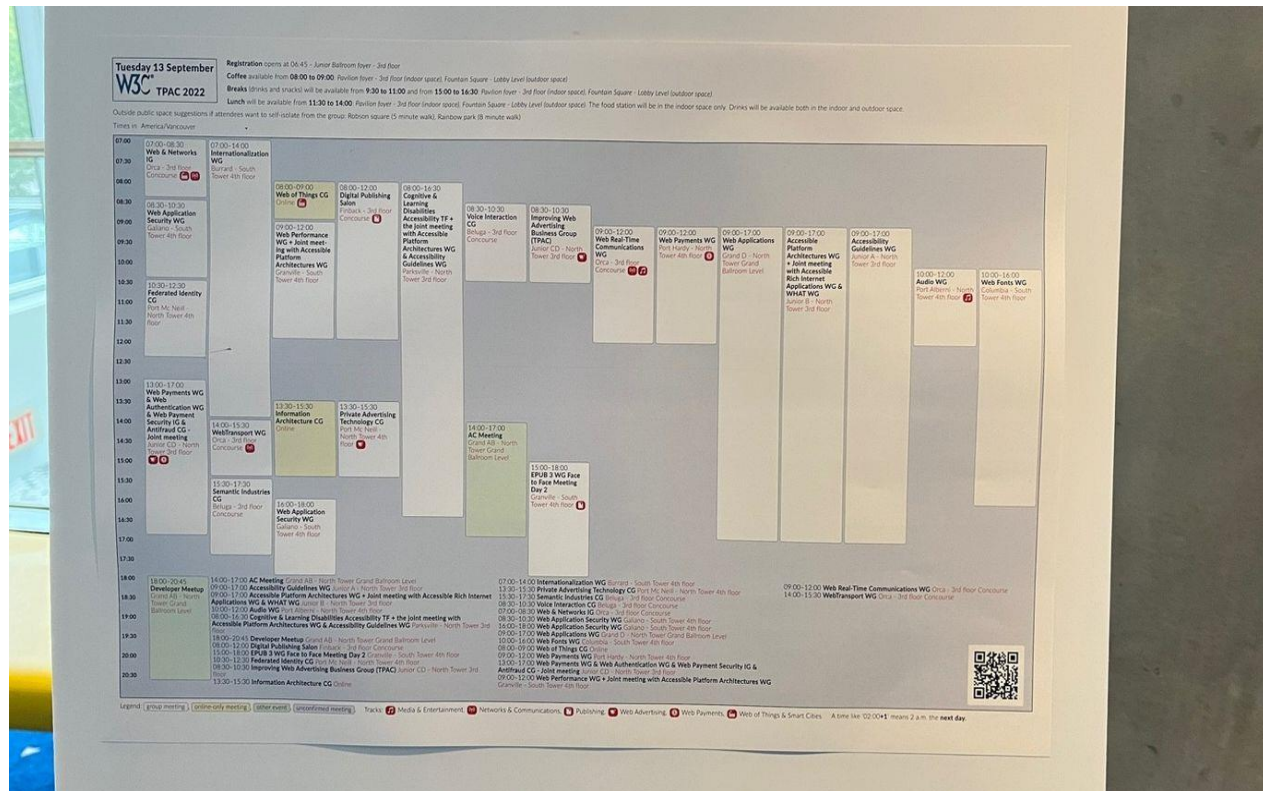
기술 그룹이란?

W3C의 목적에 맞게 (웹 기술 표준화) 각각의 웹 기술을 표준화하기 위해 생긴 그룹
그룹의 주제는 굉장히 다양하며 끝에 표시되는 단어에 따라 그룹들의 성격 또한 다름

- Working Group: result, software, test suits, 다른 그룹의 result를 논의하는 등을 하는 그룹
 - **Web Real-Time Communications Working Group(WG)**, Audio Working Group(WG)
- Interest Group: 잠재적인 웹 기술과 정책을 평가하고자 하는, idea를 교환하기 위한 그룹
 - Web Payment Security Interest Group(IG)
- Community Group: 미래의 표준화를 위해 누구나 참여해서 공유할 수 있는 커뮤니티 그룹
 - Web Extensions Community Group(CG)

WebRTC 표준 동향 - WebRTC In W3C TPAC 2022

W3C TPAC 2022



Verifiable Credentials Working Group

Web Application Security Working Group

Web Applications Working Group

Web Authentication Working Group

Web Editing Working Group

Web Fonts Working Group

Web Machine Learning Working Group

Web of Things Working Group

Web Payments Working Group

Web Performance Working Group

Web Real-Time Communications Working Group

WebAssembly Working Group

WebTransport Working Group

WebRTC 표준 동향 - WebRTC In W3C TPAC 2022

Web Real-Time Communications Working Group

웹 브라우저에서 실시간 통신을 가능하게 하는 client-side APIs를 정의

- peer간 데이터 송수신
- peer간 연결 구축(방화벽/NAT traversal 포함)

다른 그룹들이 맡기 전까지 표준화하는 APIs

- 하드웨어 디바이스 조회 (e.g. camera, microphone, speakers)
- 하드웨어 디바이스에서 media 캡처 (e.g. camera and microphone, output devices such as a screen),
- media streams의 인코딩이나 다른 processing
- media streams의 데이터 접근
- media streams의 디코딩이나 다른 processing(echo canceling, stream synchronization, and a number of other functions)
- 화면과 음성 output 장치를 통해 전달

WebRTC 표준 동향 - WebRTC In W3C TPAC 2022

표준화하고 있는 작업 List (WebRTC-charter)

- Identify for WebRTC 1.0 (2018-09-27)
- WebRTC Priority Control API (2021-03-18)
- MediaStream Image Capture (2021-04-06)
- MediaStream Recording (2021-06-04)
- MediaStreamTrack Content Hints (2021-07-22)
- Identifiers for WebRTC's Statistics API (2021-11-10)
- Media Capture from DOM Elements (2021-11-15)
- WebRTC Encoded Transform (2022-01-30)
- MediaStreamTrack Insertable Media Processing using Streams (2022-02-10)
- Audio Output Devices API (2022-02-10)
- Scalable Video Coding(SVC) Extension for WebRTC (2022-03-03)
- Media Capture and Streams (2022-03-10)
- Screen Capture (2022-03-17)
- Region Capture (2022-05-31)
- Capture Handle - Bootstrapping Collaboration when ScreenSharing (2022-05-19)
- Viewport Capture (2022-05-24)
- The Capture-Handle Actions Mechanism (작업 중)
- Media Capture Automation (작업 중)

WebRTC 표준 동향 - WebRTC In W3C TPAC 2022

현재 하고 있는 것들

Media Capture

- new features for capture - gaze correction, face detection
- new features for capture handling - crop, select, prefer
- Challenging the model
 - integration with WebCodecs
 - Stream transforms of media (unencoded and encoded)

Encoded Transform

- Congestion Control
 - AR/VR 대용량 데이터

WebRTC 표준 동향 - WebRTC In W3C TPAC 2022

Low latency streaming

- Game streaming (Section 3.2.1)
 - ultra-low latency (< 100 ms)
- Low latency broadcast (Section 3.2.2)
 - webinars, classes, events, “town hall” meetings
 - large scale (1M participants) and low latency (< 1 second)
 - CDN/eCDN support for scalability and cost control

WebRTC 표준 동향 - WebRTC In W3C TPAC 2022

Next generation Web Media APIs

- Capture
 - Media Capture and Streams Extensions
 - Mediacapture-transform
- Encode/decode
 - WebCodecs
 - Media Source Extension 2
- Transport
 - WebTransport(HTTP/3 over QUIC)
 - WebRTC data channel in Workers(SCTP/DTLS/UDP)
- Framework
 - WHATWG Streams
 - Web Assembly

WebRTC 표준 동향 - WebRTC In W3C TPAC 2022

WebRTC - 사용사례

코로나 19 팬데믹 동안 시장에 보편화된 서비스들(WebRTC)

- Podcasting
- Video conferencing
- Video streaming services (live, video-on-demand)
- Game streaming
- IoT devices (doorbells/security, exercise equipment, robots, smart speakers)
- Large scale webinars, classes, “town hall” meetings (100K+ viewers)
- Online events (auctions, conferences, sporting events, concerts)

TPAC 2021 이후에 추가

- Section 3.2: Low latency broadcast
- Section 3.3: Internet of Things
- Section 3.4: Decentralized messaging
- Section 3.9: Reduced complexity signaling

WebRTC 활용

Mesh

Connections:	4 10
Uplink:	4 mbps
Downlink:	4 mbps
Total:	20 mbps



MCU

Connections:	1 5
Uplink:	1 mbps
Downlink:	1 mbps
Total:	10 mbps



SFU

Connections:	5 25
Uplink:	1 mbps
Downlink:	4 mbps
Total:	25 mbps



WebRTC 활용

WebRTC in 구루미

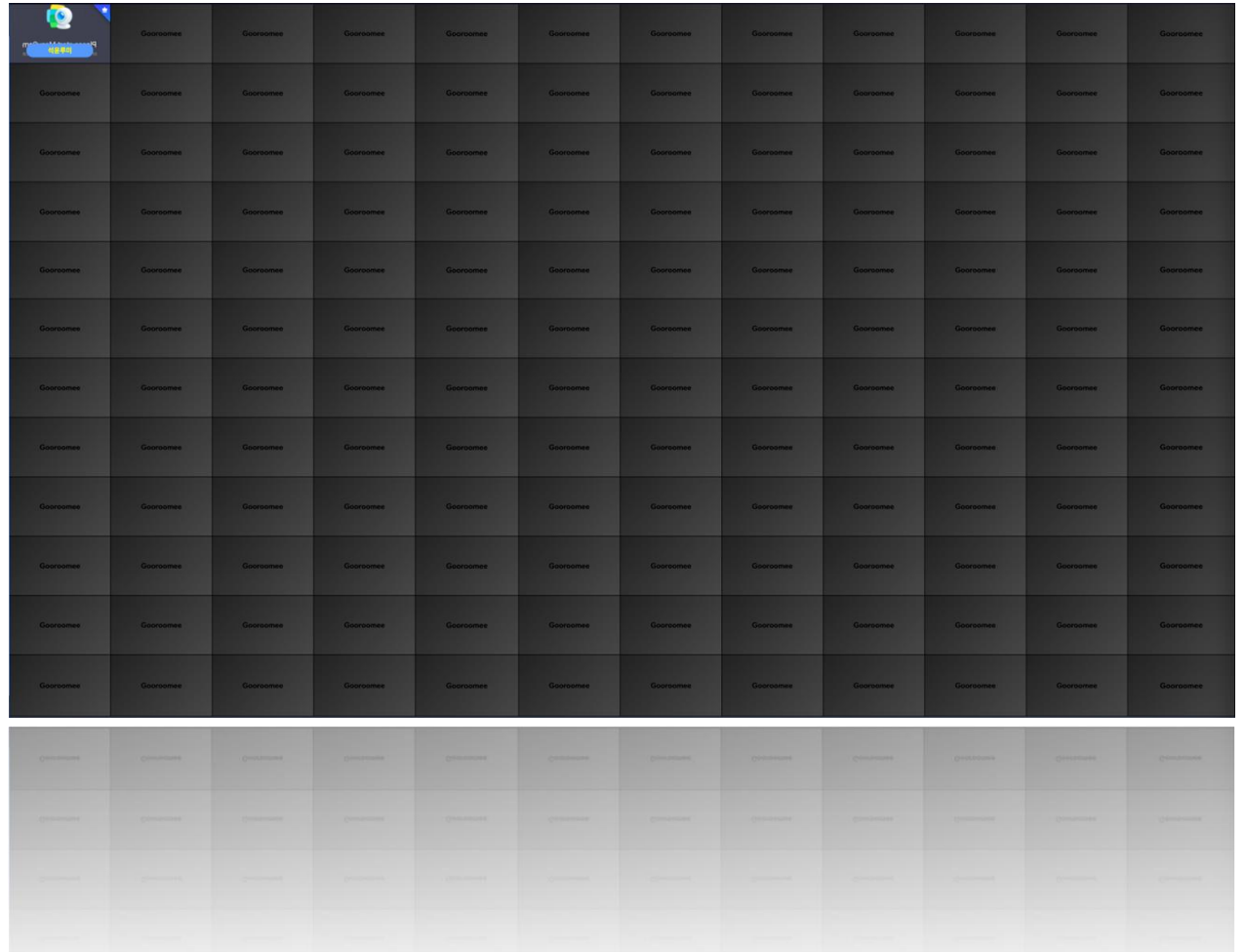
- WebRTC 기반 미디어 서버를 자체 개발 (SaaS)
- 기기 종류에 관계 없이 다자 화상통화 가능
- P2P, SFU, MCU 유연한 전환이 가능
- 기존 화상통화 장비와도 연동이 가능
- 최대 FullHD 영상 화질 제공

→ 위의 모든 기능을 구루미 플랫폼에서 제공하는 API 로 누구나 이용 가능

WebRTC 활용

WebRTC In 구루미

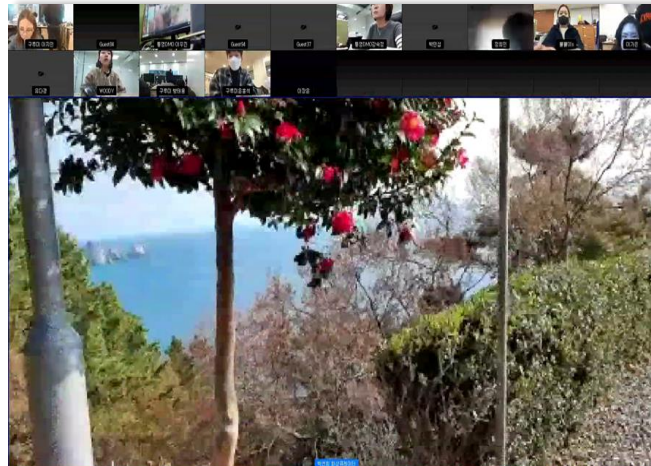
미디어 서버의 MCU 기술을 활용한
최대 200인 레이아웃 제공



WebRTC 활용

WebRTC In 구루미

화상통화 서비스를 넘어,
공연 예술, 이벤트, 라이프 로깅, 인터뷰 등
다양한 용도로 활용



WebRTC 활용

WebRTC In 구루미

구루미 캠퍼터디



WebRTC 정리

- gaze correction, face detection, crop 등 다양한 Media capture 제공
- AR/VR을 대비한 대용량 데이터 처리
- game streaming, ultra low latency
- low latency broadcast
- ...

일반적인 화상통화를 넘어

미디어 캡처에 대한 발전, 대용량 데이터 처리, 초 지연성 방송을 할 수 있는 기술로 방향을 잡아 성장

WebRTC

- WebRTC란?
 - 개요
 - 아키텍처와 연결
- WebRTC 표준 동향
 - WebRTC In W3C TPAC 2022
- WebRTC 활용
- WebRTC 정리

WebTransport

- WebTransport란?
 - 개요
 - 프로토콜
 - WebSocket과 비교
 - API
- WebTransport 표준 동향
 - WebTransport In W3C TPAC 2022
- WebTransport 정리

WebTransport란?

WebTransport는 HTTP/3 프로토콜을 양방향 전송으로 사용하는 웹 API
(HTTP/2 에서도 사용 가능, but HTTP/3에서 성능보단 부족)

- 서버와의 통신에서 low latency 제공
- reliable and unreliable, ordered and unordered, client-server and p2p, data and media
- WebSocket 과 동일한 보안 properties 적용 (TLS, server-controlled origin policy)

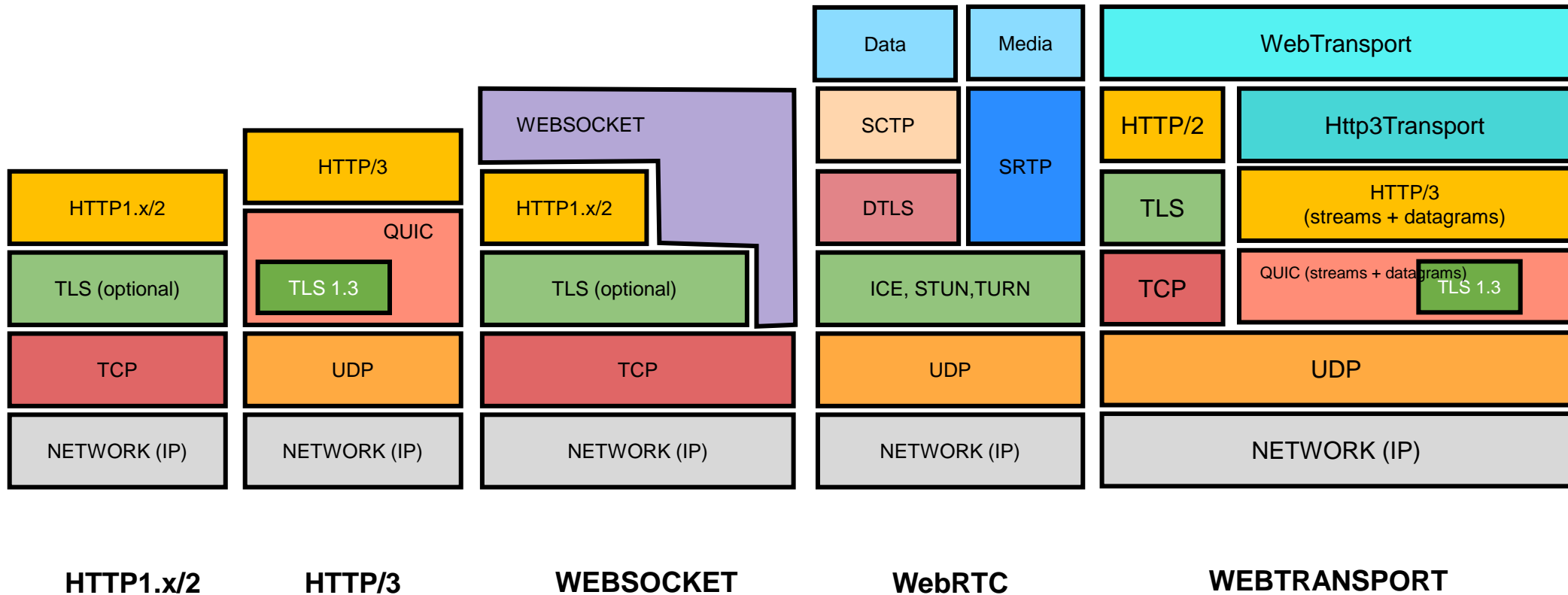
WebSocket 대체?

- WebSocket의 정렬된 메시지 스트림은 WebTransport stream api으로 대체 가능
- datagram api 또는 여러개의 stream API를 사용하면 WebSocket의 HOL(Head-of-line) blocking 발생 안함
- QUIC handshake TLS를 통해 TCP 연결보다 새 연결을 설정할 때 성능 이점

WebRTC DataChannel 대체?

- 클라이언트 - 서버 연결 일 경우 대체 가능
- WebTransport API는 WebRTC와 달리 Web Worker내에서 지원

WebTransport란? - 프로토콜과 API



WebTransport란? - WebSocket과 비교

WebSocket

- WebApplication에서 대표적인 Client - Server 통신 프로토콜
- HTTP1.x/2 기반 (TCP)
 - 암호화(TLS), 흐름 제어(TCP), 혼잡 제어(TCP) 등이 다른 계층에서 동작
 - 4RTT(Round-Trip-Time) , 3RTT
- 한개의 정렬된 스트림 메시지만 지원 - HOL (head-of-line) blocking

WebTransport

- HTTP/3 기반 (UDP)
 - 흐름제어, 혼잡 제어 등이 Application Layer 계층에서 동작
 - 1RTT, 0RTT
- 여러가지 타입의 송수신 API를 제공
 - Datagram
 - UnidirectionalStream
 - BidirectionalStream

WebTransport란? - WebSocket과 비교

WebSocket's message-framed semantics

```
socket.send(
```

"Hello World!"

```
socket.send(
```

"How are you?"

Receiver "Hello World!"

"How are you?"

(framed + ordered = blocking)

WebTransport's long-lived stream semantics

```
writer.write(utf8.encode(    ));
```

"Hello World!"

```
writer.write(utf8.encode(    ));
```

"How are you?"

Receiver:

"Hello World!How are you?"

(stream of unframed ordered data)

WebTransport's stream semantics

```
for (const msg of [    ,    ]) {
```

```
  const writable = await wt.createUnidirectionalStream();
```

```
  const encoder = new TextEncoderStream("utf-8");
```

```
  const writer = writable.pipeThrough(encoder).getWriter();
```

```
  await writer.write(msg);
```

```
  await writer.close();
```

```
}
```

Receiver "How are you?"

"Hello World!"

(in-parallel streams arrive unordered)

WebTransport란? - API

Connection

```
async function createWebTransport(url) {  
  const wt= new WebTransport(url);  
  
  wt.closed.then(()=>{})  
    .catch((error)=> {});  
  
  await wt.ready;  
  return wt  
}
```

WebTransport란? - API

Datagram API

```
async function sendDatagrams(url, datagrams) {  
  const wt = createWebTransport(url);  
  const writer = wt.datagrams.writable.getWriter();  
  for (const datagram of datagrams) {  
    await writer.ready;  
    writer.write(datagram).catch(() => {});  
  }  
}
```

```
async function receiveDatagrams(url) {  
  const wt = createWebTransport(url);  
  const reader = wt.datagrams.readable.getReader();  
  while (reader){  
    const {value, done} = await reader.read();  
    if(done) { break; }  
  }  
}
```

WebTransport란? - API

UnidirectionalStream API

```
async function sendStreams(url, datas) {  
  const wt = createWebTransport(url);  
  const writer = wt.createUnidirectionalStream.writable.getWriter();  
  for (const data of datas) {  
    writer.write(data).catch(() => {});  
  }  
  await writer.close();  
}
```

```
async function receiveStreams(url) { // Server  
  const wt = createWebTransport(url);  
  const reader = wt.createUnidirectionalStream.readable.getReader();  
  while (reader){  
    const {value, done} = await reader.read();  
    if(done) { break; }  
  }  
  reader.closed.then(()=>{})  
    .catch((error)=>{})  
}
```

WebTransport 표준 동향 - WebTransport In W3C TPAC 2022

WebTransport Working Group

HTTP/3 protocol 에서 동작하는 클라이언트-서버 간의 양방향 프로토콜 API 정의

표준화하고 있는 작업

WebTransport API

연관된 W3C 그룹들

- WebRTC Working Group
- Media Working Group
- Web and Networks Interest Group
- Web Application Security Working Group
- ...

WebTransport 표준 동향 - WebTransport In W3C TPAC 2022

2021년 TPAC 이후 작업한 리스트

- Block ports on fetch's bad ports list
- Add packetLost to WebTransportStats
- Add WebTransport constructor for reliability
- Add read-only property for WebTransport
- ...

논의중인 이슈

- Datagram vs stream & relative stream prioritization
- Stats for congestion control and bandwidth estimation

WebTransport 정리

- video conferencing & telephony applications
- cloud gaming streaming
- low latency & live media delivery
- pubsub model
- ...

WebTransport로 UDP 기반의 WebSocket 및 HOL blocking이 없는 WebSocket을 지향하는 모든 분야에서 사용 가능한 프로토콜

Thank You



Gooroom

※ 본 문서의 저작권은 (주)구루미에 있으며, 제공 되는 자료는 수정이 불가능합니다.

홈페이지 biz.gooroom.com
도입문의 sales@gooroom.com
대표전화 1833-9229

W3C TPAC 2022 WITH GOOROOMEE