

Media over QUIC Transport: 標準化に向けた貢献

Arch B2 Kota

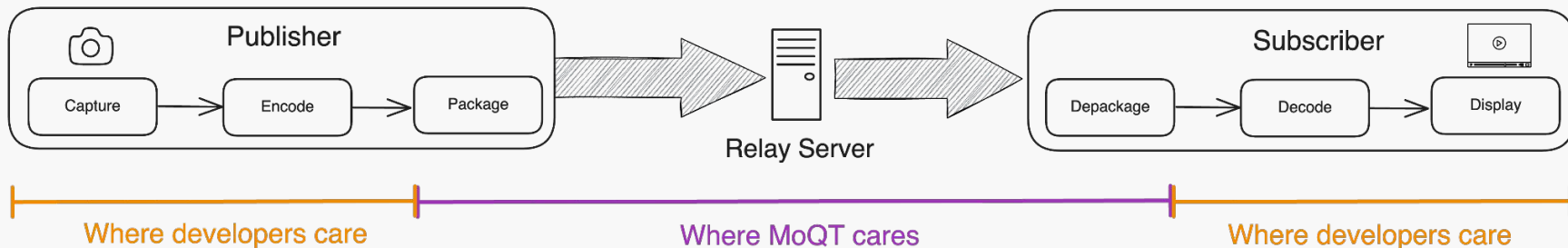
- ライブ配信，オンラインゲームなど多様なユースケースに対応する低遅延ストリーミングプロトコルとして，2023年夏にIETF moq WGにてMedia over QUIC Transport (MoQT) のドラフトが提出され議論されている
- MoQTは個人的に数年前から注目していたプロトコルであり，今期，プロトコルの実装とワーキンググループ内での相互接続テストを行った
- 本WIPではその報告とMoQTに対する今後の取り組み，今後の標準化活動への貢献について述べる

そもそもMedia over QUIC Transportとは

- QUICベースのPub/Sub型ストリーミングプロトコル
- 元々TwitchからWARPという名前で提唱されていたプロトコルの一部が2023年7月にMedia over QUIC Transportという名前でドラフトとして提出された
- ライブストリーミング，オンラインゲーム，ビデオ会議など非常に多様なユースケースを想定して設計されている

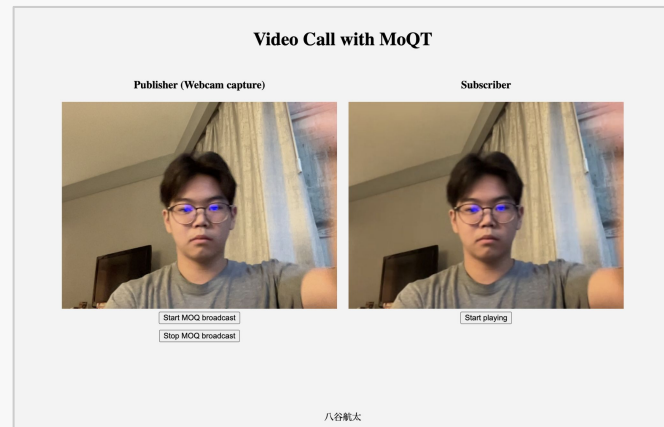
MoQTの特長

- QUICベースであるため、多重化ストリームを利用した配信が可能
 - 例: 映像1フレームに1ストリームを使うことで超低遅延を実現できる
- MoQTオブジェクトのペイロードのエンコード/デコード処理はアプリケーション開発者に委ねられており、自由度が高い
 - 画質/遅延のトレードオフ調整も容易
 - 映像、音声以外のデータも同じプロトコルで伝送できる



MoQTクライアントの実装

- ブラウザ上で動作するMoQT Publisher/Subscriberを実装した
 - MoQTサーバーにMetaのOSS実装を使い、実際にライブ配信を行うWebアプリケーションも実装した
 - E2Eの遅延はネットワーク平常時で140ms程
 - 超低遅延！
 - 公開されているMoQT実装としては6つ目



今学期やったこと

moq WG内での相互接続テスト (interop)

- 先週バンクーバーで行われたIETF120に現地参加
- moq WGのコミッターらと話しinteropをやることになる
- 現地で自前実装のDraft-01からDraft-04へのバージョンアップを行いInteropを実施。成功 🎉
 - Draft-04のInteropが成功したのはこれが2つ目
 - 現在はinteropのテストケース定義に参加している



RG内での今後の計画

- 個人的にはMoQTプロトコルと他プロトコルの比較・評価でTERM，ひいてはGPに望みたいと考えている
- 本来今期のWIPでMoQTとRTMP+HLSの比較を行う予定だったが，RTMP+HLS側の実装に大きな欠陥が見つかり実現しなかった。
 - 来期はこのテーマで再挑戦

標準化活動により深く関わるための計画

- 今回のIETF120で初めてコミッターと会話し、実装について議論した
- 実装についてはある程度貢献できる余地があるものの、仕様自体についての議論に関わるにはまだ知識と経験が足りていないことを痛感した
 - MoQTの仕様を考えるのにMoQTだけ追っているのでは議論にならない
 - 他のストリーミングサービスやベースとなるプロトコル、ネットワークアーキテクチャ自体の知識を深めることが今後の目標