

ハイブリッド CDN-P2P 映像配信の検討 - RTP ストリーム共有における再エンコードフリーなアプローチ -

岡田 浩希[†]
TIS Inc.[†]

吉見 真聡[‡]
TIS Inc.[‡]

1 はじめに

オンライン会議やライブ配信などのリアルタイムな映像配信システムでは、メディアデータの伝送に伴う通信コストが大きな課題となっている。あらかじめ録画しておいた静的なメディアデータを配布するオンデマンドな映像配信では、分割化されたメディアデータを CDN でキャッシュし、データを再利用することで通信コストを削減している。一方で、リアルタイム性を重視した映像配信では、低遅延を維持するために CDN にデータをキャッシュをせず、映像データを直接伝送し続けること求められる。このため、リアルタイムな映像配信システムでは、限られた帯域幅を有効活用するための新たなアプローチが求められている。

リアルタイムなストリーミングデータを効率的に配信する方法の一つとして、受信したデータを他のサーバやデバイスに再度転送する手法がある。ストリーミングデータを配信する CDN では、この手法を応用し、多数の中継サーバを用いてユーザの接続台数を増やしている。本研究では、この伝統的な CDN に加えて、ユーザデバイス同士でデータを共有し合う仕組みを導入する。WebRTC を活用したハイブ

リッド CDN-P2P 映像配信システムにおける、ブラウザ間の直接的な RTP ストリーム共有を検討した。WebRTC の DataChannel を用いてブラウザ間の映像データを直接交換し、伝統的な CDN の役割を部分的に代替する。ユーザ間で映像データを共有することで、CDN の負荷を軽減し、上流回線の通信コストを削減することができる。また、ユーザ間のデータ共有に WebRTC MediaChannel を用いた場合と比較して、再エンコードを必要とせず、リアルタイム性を保ちながら通信処理を分散する。このアプローチは、通信コストの削減が求められるリアルタイム映像配信システムにおいて、新たな効率性と品質向上の可能性を示唆する。

2 CDN-P2P 映像配信システムの実装

2.1 ブラウザ間のデータ共有手法の検討

まず、ブラウザ間でのメディアデータ共有手法を検討する。一般的にブラウザは他のブラウザと直接通信する機能を持たないため、WebRTC を用いてブラウザ間のデータ共有を行う。WebRTC では、ブラウザ間のデータ共有を実現するために、MediaChannel と DataChannel の 2 つのチャンネルを提供している。MediaChannel は、映像や音声などのメディアデータを伝送するためのチャンネルであり、DataChannel は任意のデータを伝送するためのチャンネルである。ブラウザにおける MediaChannel は容易に利用可能な一方で、内

Hybrid CDN-P2P Video Distribution - A No Re-encoding Approach to RTP Stream Sharing

[†] OKADA Hiroki, TIS Inc.

[‡] YOSHIMI Masato, TIS Inc.

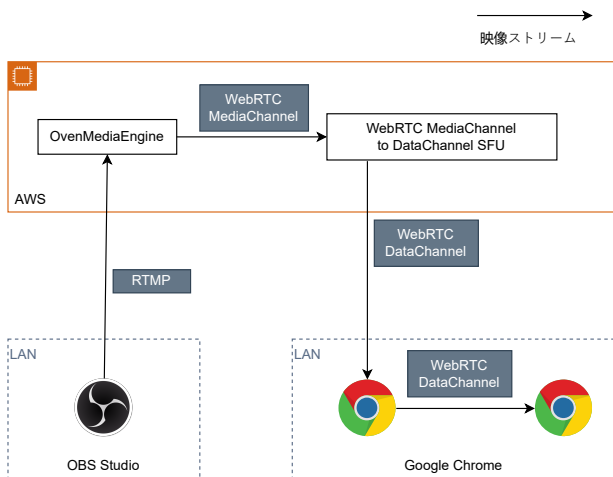


図1 ハイブリッド CDN-P2P 配信システム構成

部処理が隠蔽されており、データの伝送に関する制御が難しい。Encoded Transforms[1] と呼ばれる機能で、受信したエンコードされた状態のメディアデータを取得することは可能だが、現状殆どのブラウザでサポートされていない。一方で、DataChannel は任意のデータを伝送できるため、受け取った映像データを他のブラウザへの再伝送が容易に実現できる。上記の理由から、DataChannel を用いてブラウザ間で任意のデータを共有する仕組みを実装した。

2.2 ハイブリッド CDN-P2P 配信システム

WebRTC を活用したハイブリッド CDN-P2P 映像配信システムを検討する。本稿では、図1に示すような WebRTC 配信サーバ (OvenMediaEngine[2]) とクライアント (ブラウザ) から構成されるシステムを想定する。ブラウザは MediaChannel で映像を受信するとデコードが強制されるため、他のブラウザに転送するためには再エンコードが必要となる。このため、配信サーバから MediaChannel で RTP パケットを受信し、パケットをそのまま DataChannel で再配信する中継サーバを実装した。ブラウザは DataChannel 経由で RTP

パケットを受信するため、再エンコードを行わずに映像データを他のブラウザに転送できる。

2.3 RTP ストリームの再生

次に、ブラウザで RTP ストリームを再生する方法を検討する。ブラウザは RTP ストリームを直接再生する機能を持たないため、RTP パケットからメディアデータを取り出し、デコードして再生する必要がある。メディアデータのデコードは WebCodecs API を用いて実現する。WebCodecs API によってデコードされたメディアデータは、HTMLVideoElement か HTMLCanvasElement で再生が可能である。

再生機能の実現のため、RTP パケットのデパケライズ処理と、WebCodecs API を用いた H.264 映像のデコード処理を実装した。

3 まとめと今後の展望

本研究では WebRTC DataChannel を活用したハイブリッドな CDN-P2P 映像配信システムを検討し、プロトタイプ実装を行った。DataChannel による RTP パケットの配信と WebCodecs API によるデコードと再生処理を実装して実現可能性を示した。今回の実装では、配信サーバからデータ受信しているブラウザを手動で選択して WebRTC 接続処理を行い、ブラウザ間のデータ共有を実現している。今後は、通信コストの低減を目的に、適切な接続先ブラウザを自動選択するアルゴリズムを検討する。

参考文献

- [1] WebRTC encoded transform. <https://www.w3.org/TR/webrtc-encoded-transform/>. Accessed: 2024-1-12.
- [2] OvenMediaEngine: OvenMediaEngine (OME) is a Sub-Second latency live streaming server with Large-Scale and High-Definition. #webrtc #llhls.