# AnchorNet: Bridging Live and Collaborative Streaming with a Unified Architecture

(Operational System Track)

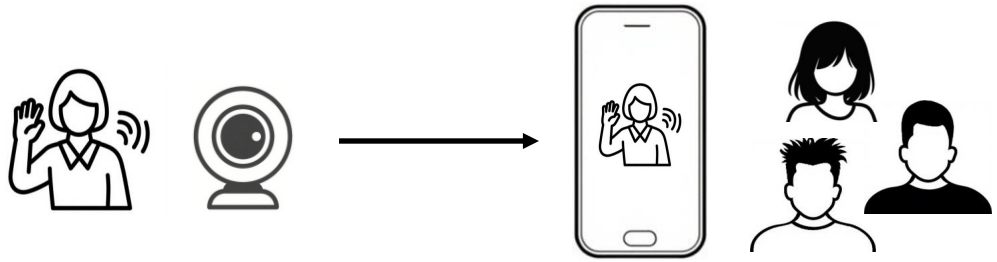
**Tong Meng** Wei Zhang Dong Chen Zhen Wang Quanqing Li
Changqing Yan Chao Yuan Le Zhang Jianxin Kuang Jianlin Xu

(ByteDance)

usenix
THE ADVANCED COMPUTING SYSTEMS ASSOCIATION

USENIX ATC '25

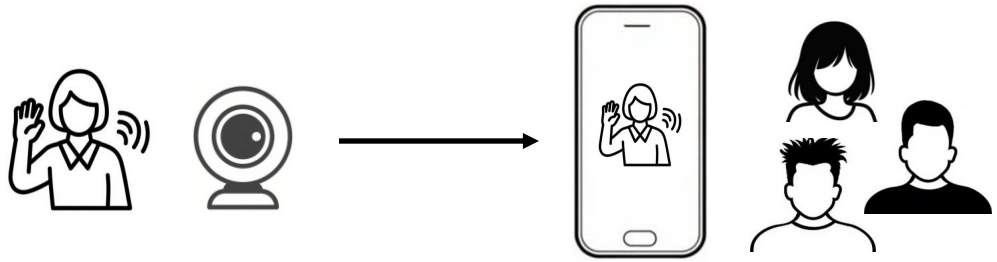
# Two Live Streaming Modes



**Basic Live Streaming**

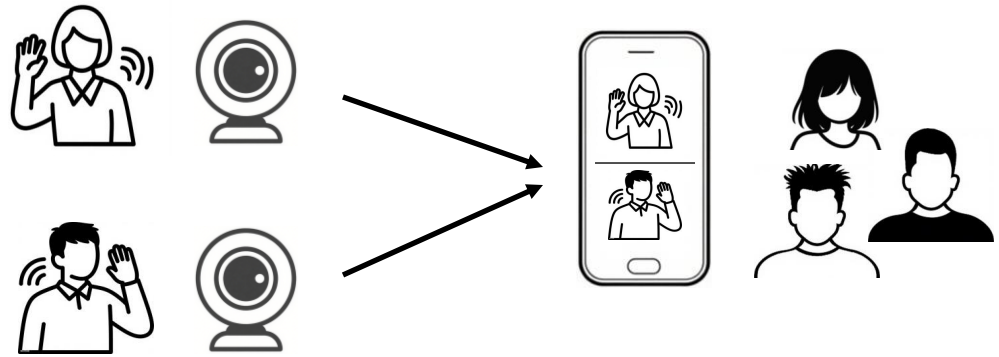
- Channel host as a single broadcaster
- Stream to massive viewers with second-level latency

# Two Live Streaming Modes



**Basic Live Streaming**

- Channel host as a single broadcaster
- Stream to massive viewers with second-level latency



**Collaborative Streaming**

- Multiple collaborative broadcaster(s) chat in real time
- Co-broadcasters’ streams are mixed to a single stream seen by viewers

# Roadmap

- Challenges of supporting an additional streaming mode
- Streaming architecture design of AnchorNet
- Audio splicing techniques for smooth streaming mode switching
- Deployment results

# Roadmap

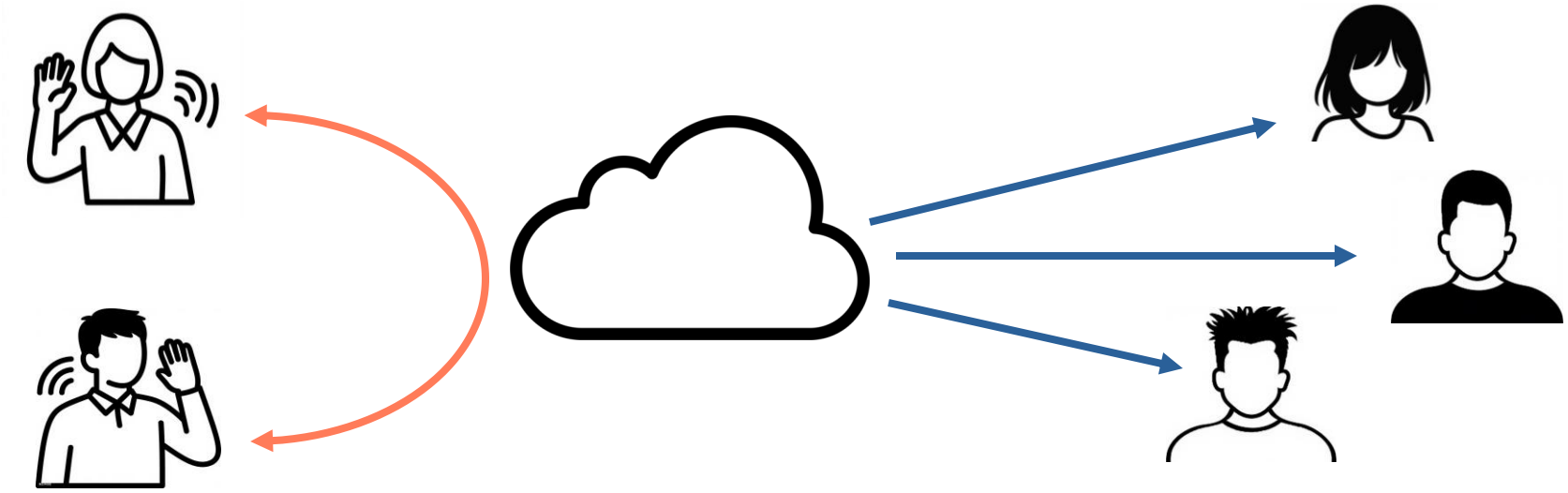
- Challenges of supporting an additional streaming mode

- Streaming architecture design of AnchorNet

- Audio splicing techniques for smooth streaming mode switching

- Deployment results

# Challenges #1: Multiple Protocol Stacks

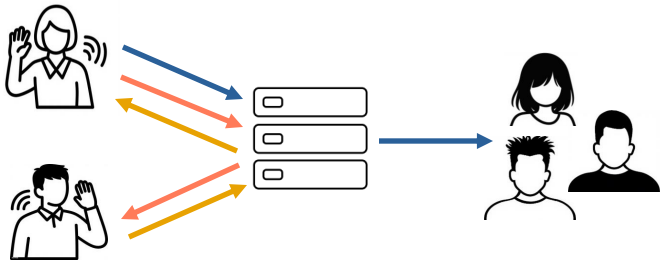


WebRTC protocol suite for real-time communication

HTTP-based CDN distribution for large-scale cost effectiveness

# Challenges #2: Stream Mixing

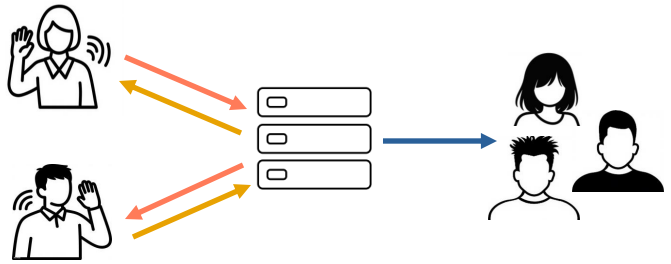
## Broadcaster-Side Mixing



**Cons**

- Higher burden on channel host's device, thus possible barrier on complex scenes
- Higher first-mile congestion, possibly limited mixed bitrate
- Mixed stream delayed by a round trip at the first mile

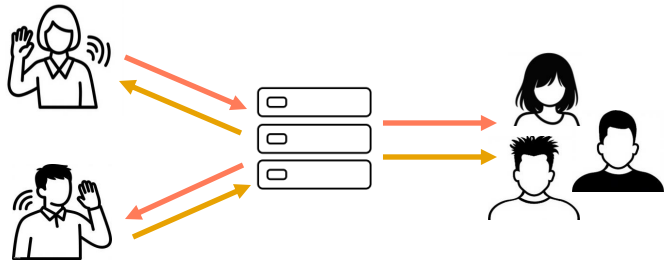
## Server-Side Mixing



**Cons**

- Higher server-side computation overhead

## Viewer-Side Mixing

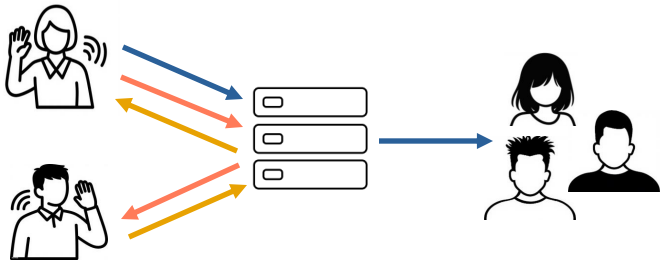


**Cons**

- End-to-end WebRTC stack, higher deployment cost than CDN-based distribution
- More complex viewer-side subscription signaling

# Challenges #2: Stream Mixing

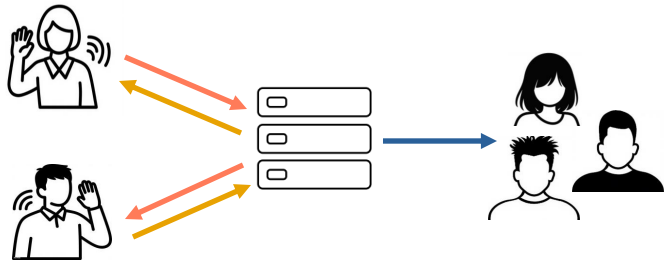
## Broadcaster-Side Mixing



**Cons**

- Higher burden on channel host's device, thus possible barrier on complex scenes
- Higher first-mile congestion, possibly limited mixed bitrate
- Mixed stream delayed by a round trip at the first mile

## Server-Side Mixing

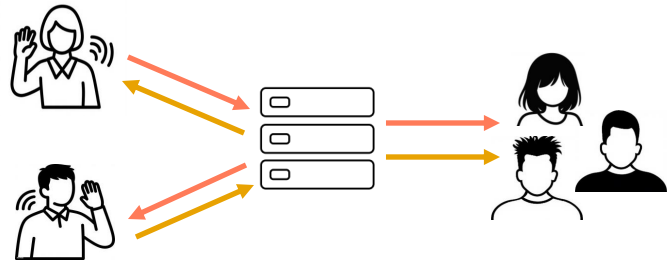


**Cons**

- Higher server-side computation overhead

**Can be amortized by a large number of viewers**

## Viewer-Side Mixing

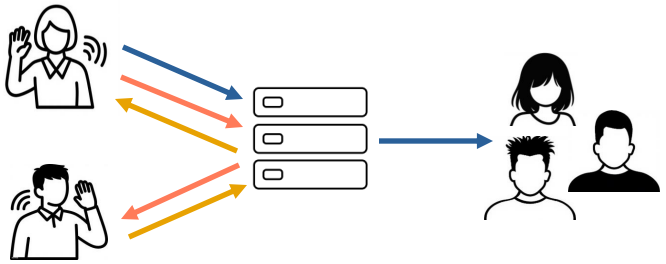


**Cons**

- End-to-end WebRTC stack, higher deployment cost than CDN-based distribution
- More complex viewer-side subscription signaling

# Challenges #2: Stream Mixing

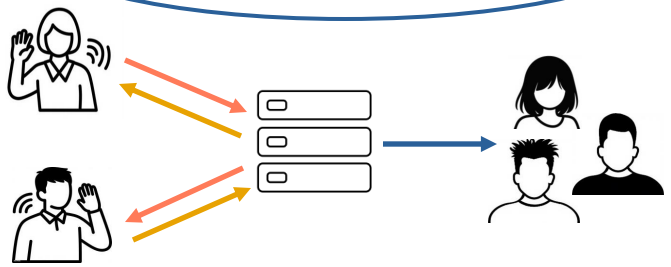
## Broadcaster-Side Mixing



**Cons**

- Higher burden on channel host's device, thus possible barrier on complex scenes
- Higher first-mile congestion, possibly limited mixed bitrate
- Mixed stream delayed by a round trip at the first mile

## Server-Side Mixing

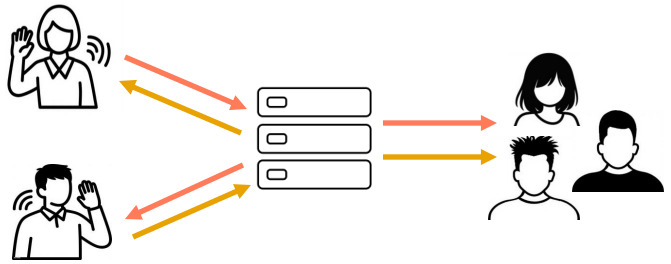


**Cons**

- Higher server-side computation overhead

**Can be amortized by a large number of viewers**

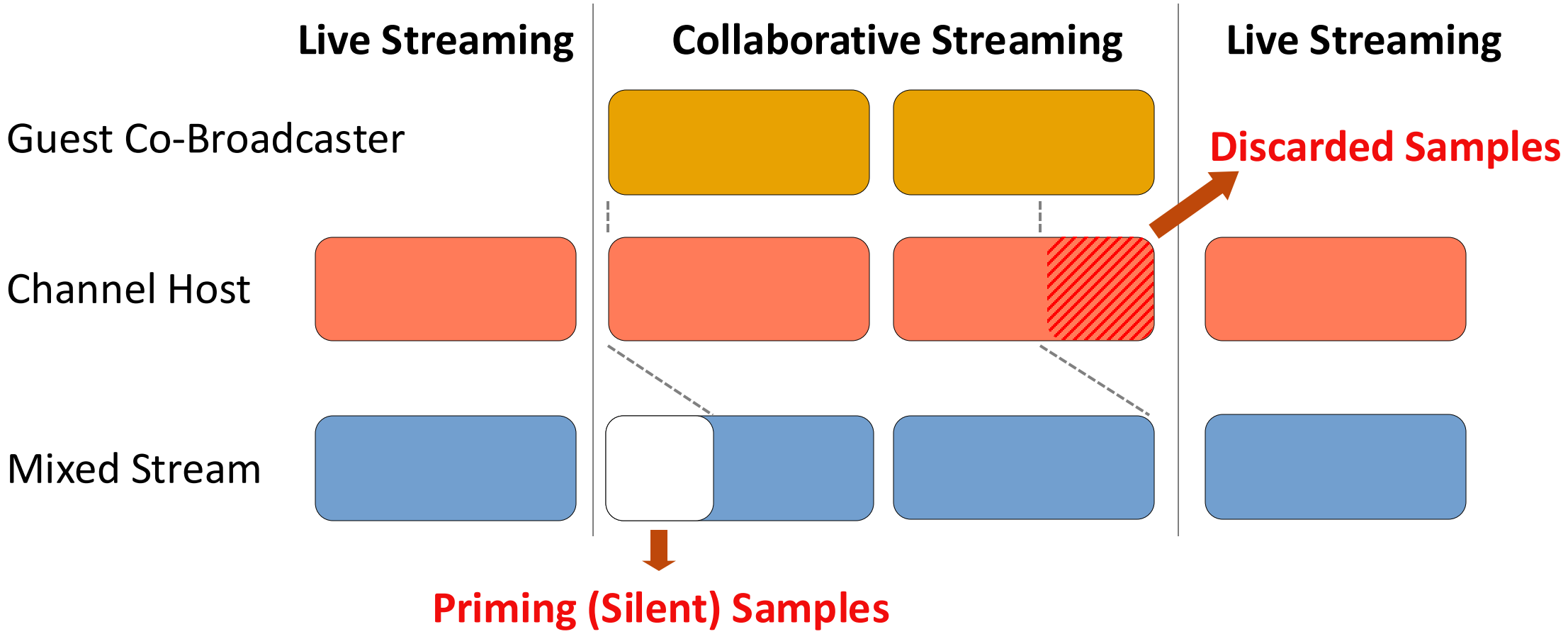
## Viewer-Side Mixing



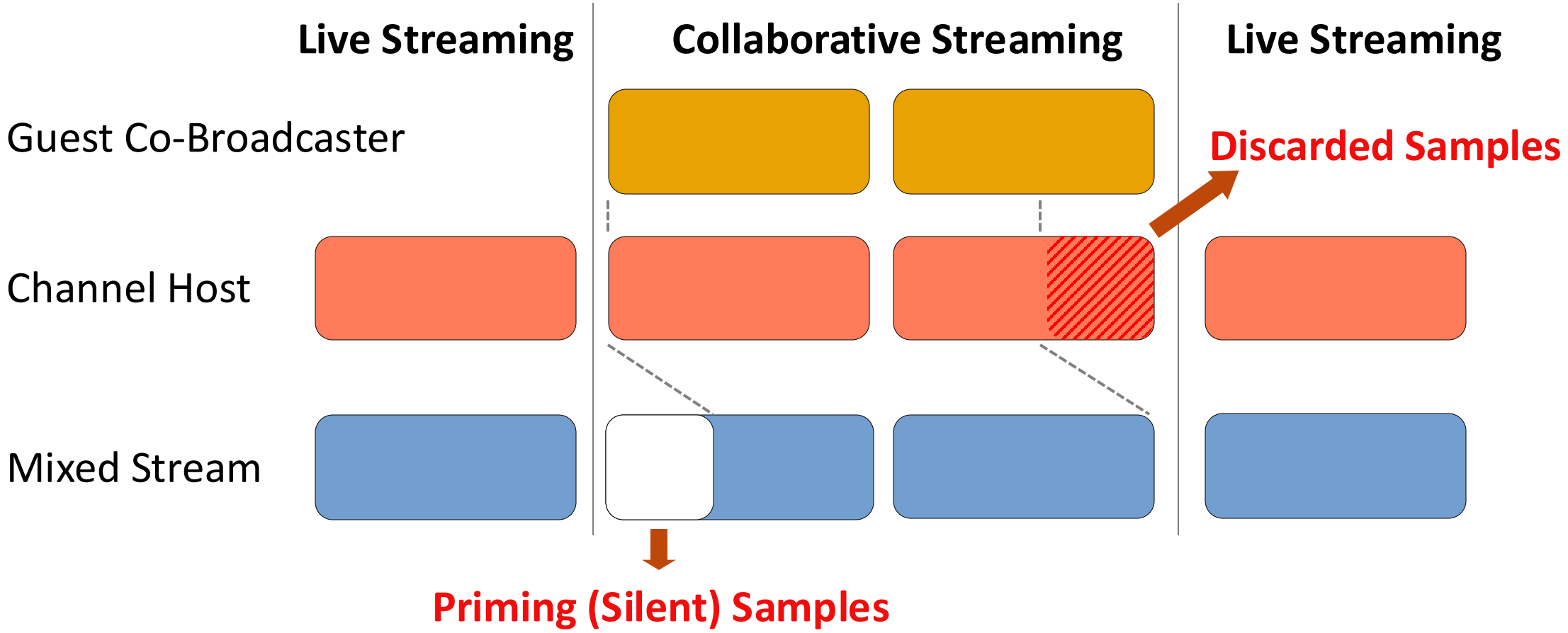
**Cons**

- End-to-end WebRTC stack, higher deployment cost than CDN-based distribution
- More complex viewer-side subscription signaling

# Challenges #3: Audio Stream Splicing

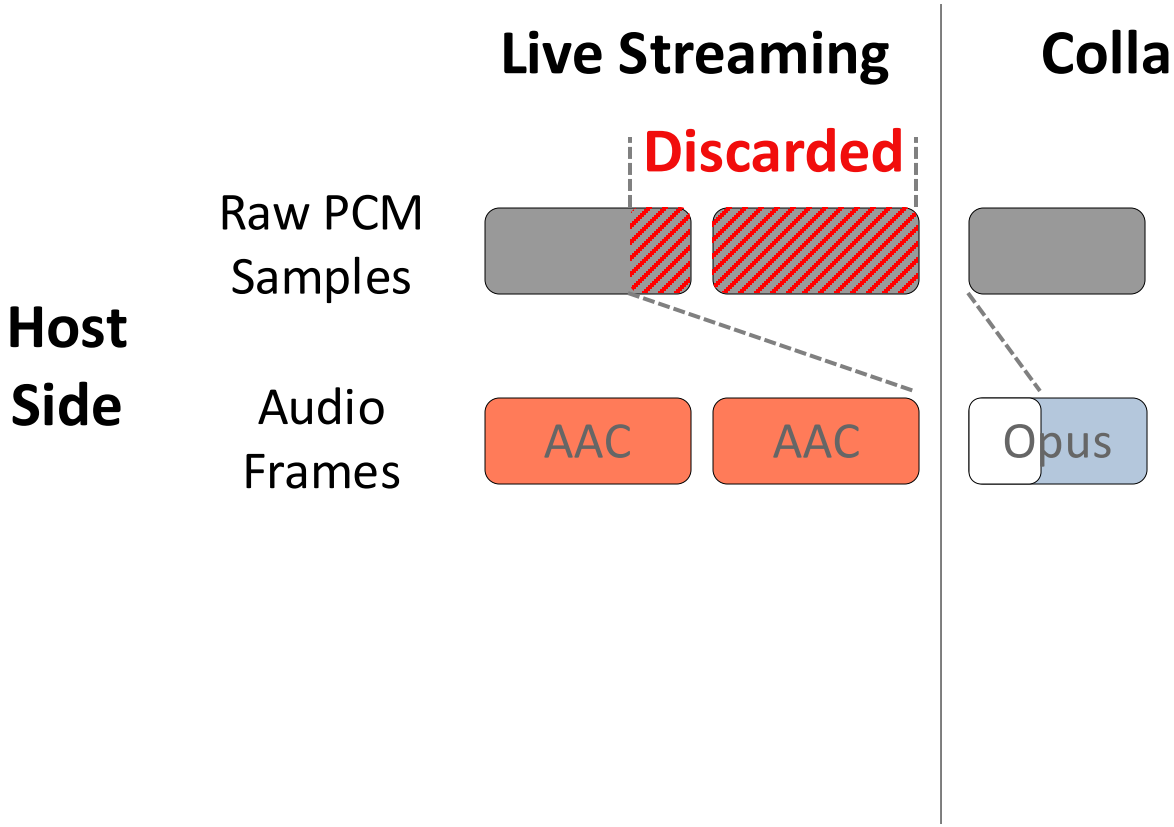


# Challenges #3: Audio Stream Splicing

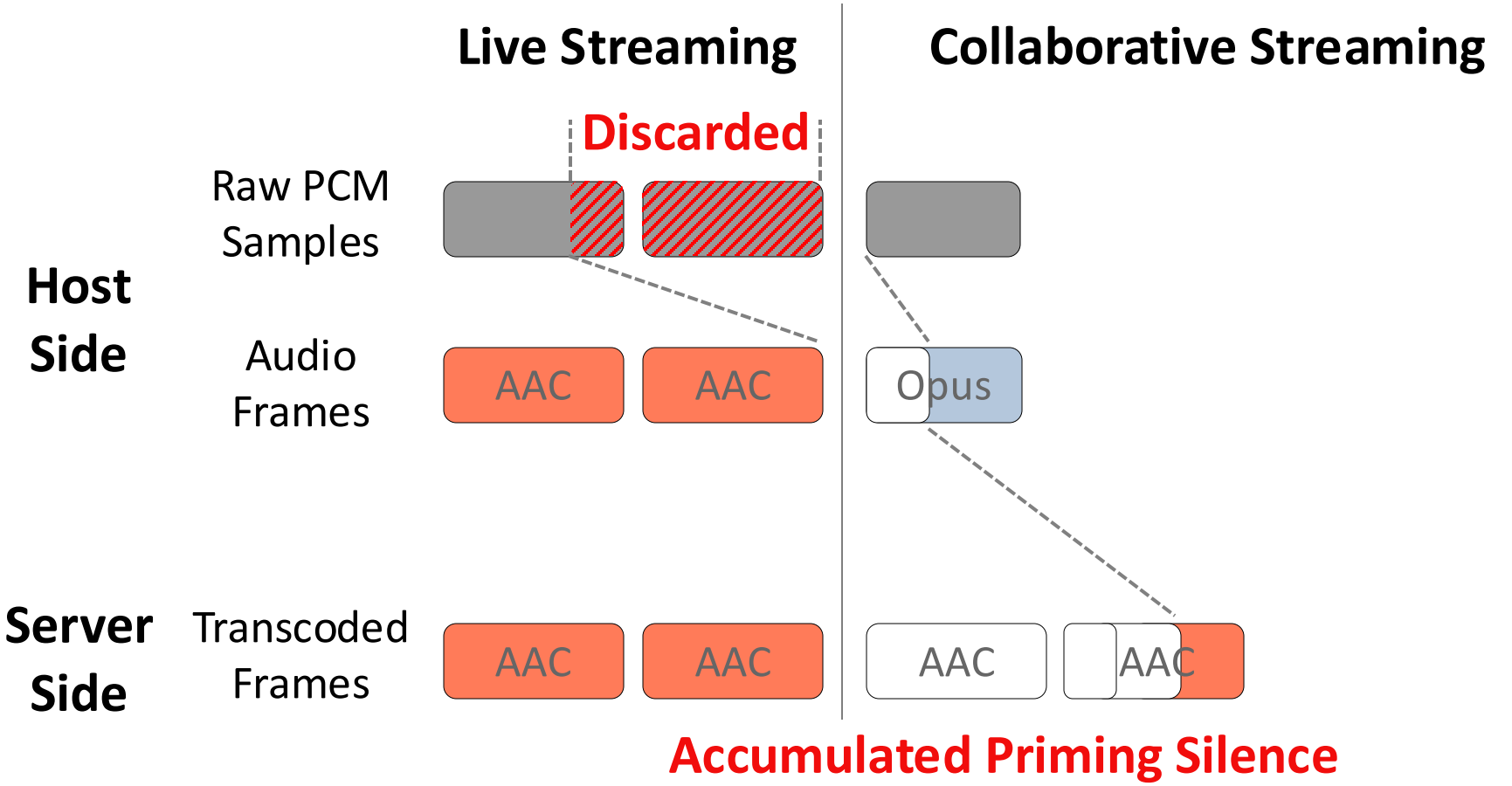



**Audio encoder delay causes audio glitches**

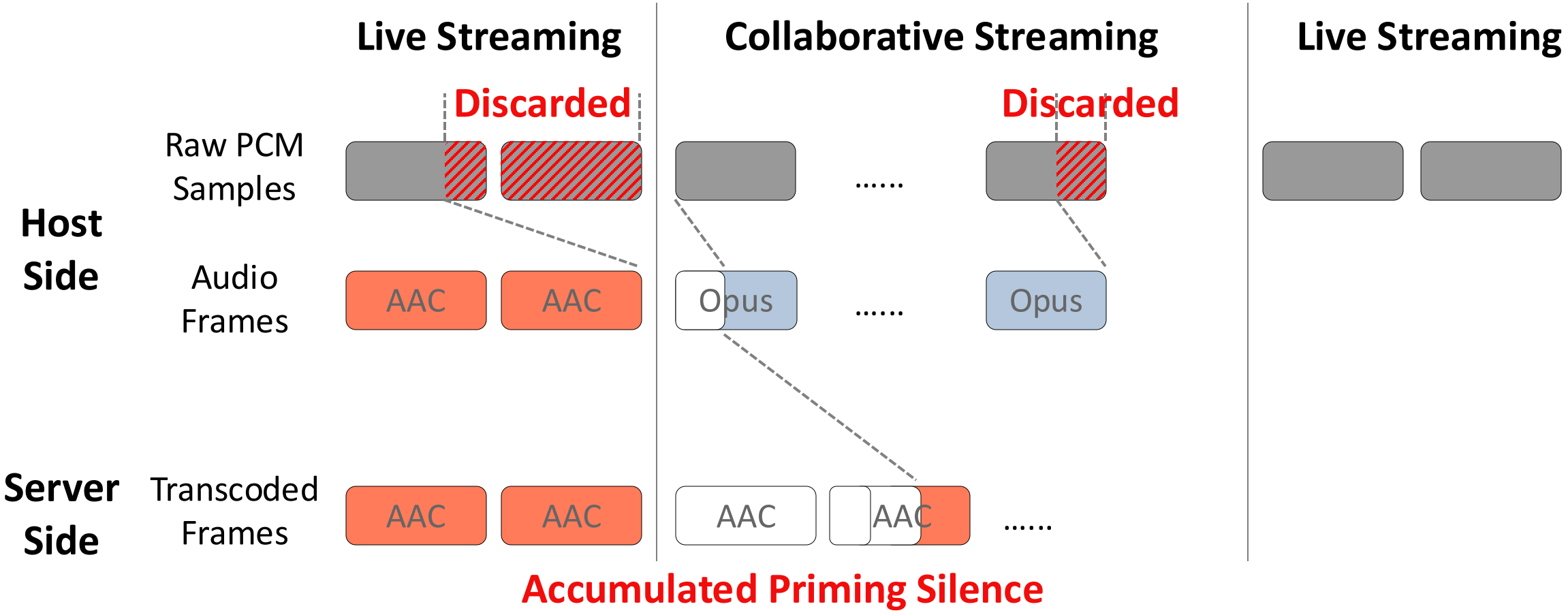
# Challenges #3: Audio Stream Splicing



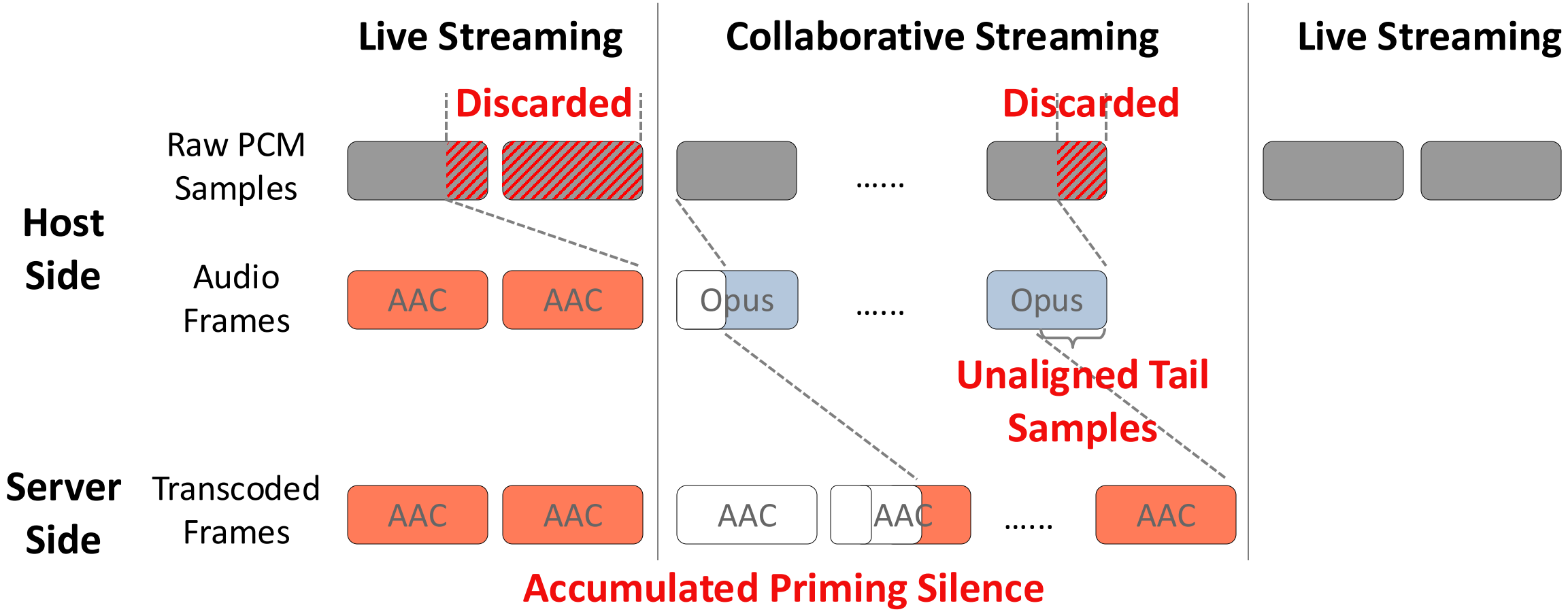
# Challenges #3: Audio Stream Splicing



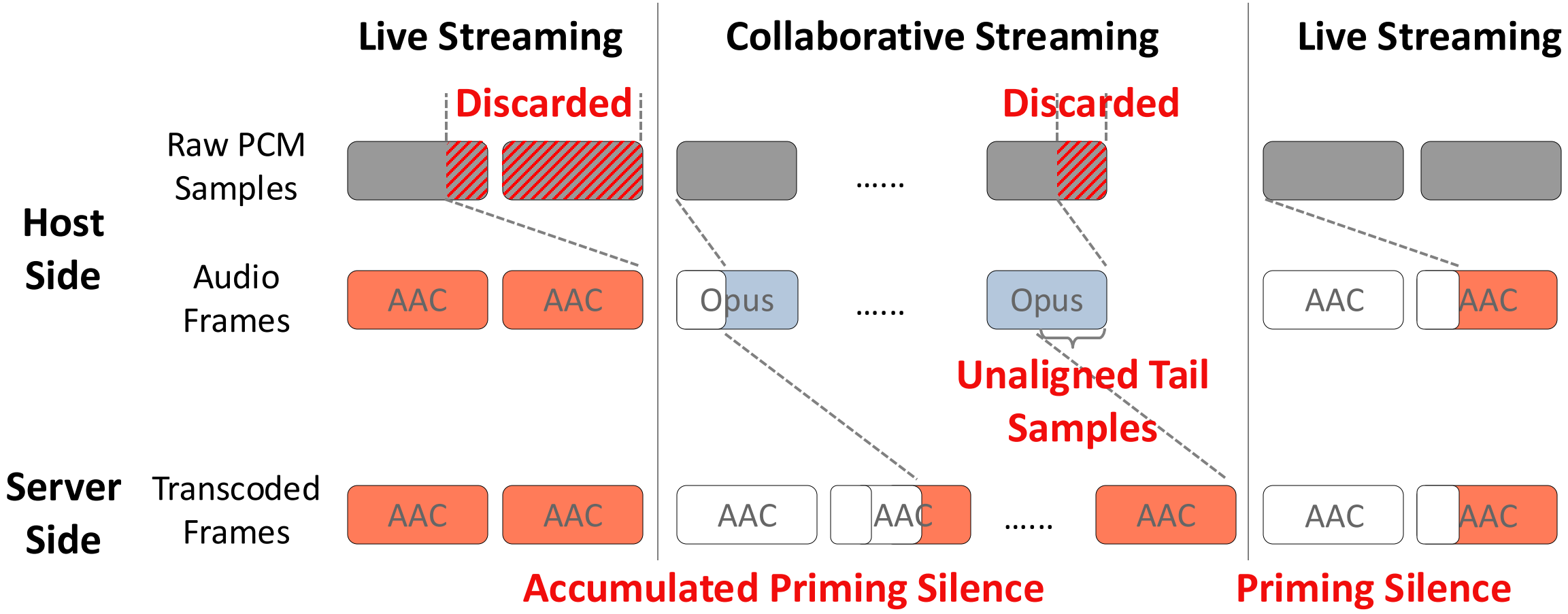
# Challenges #3: Audio Stream Splicing



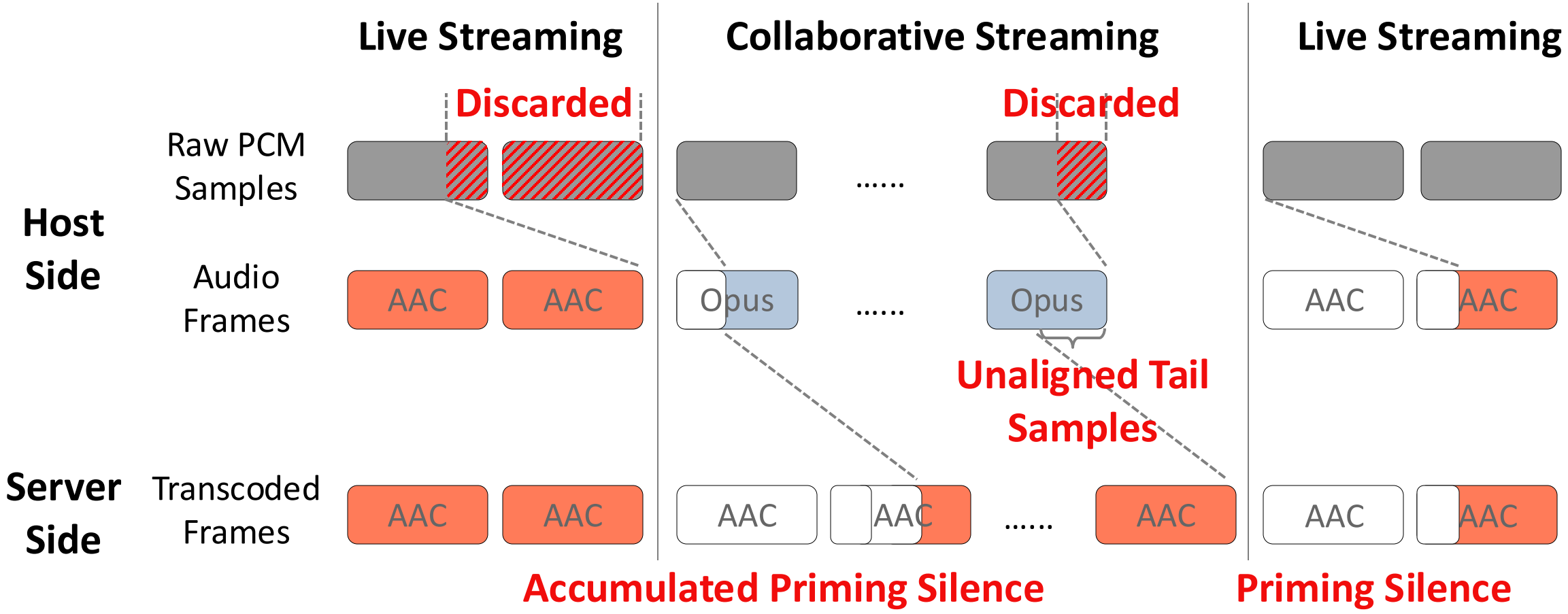
# Challenges #3: Audio Stream Splicing



# Challenges #3: Audio Stream Splicing



# Challenges #3: Audio Stream Splicing

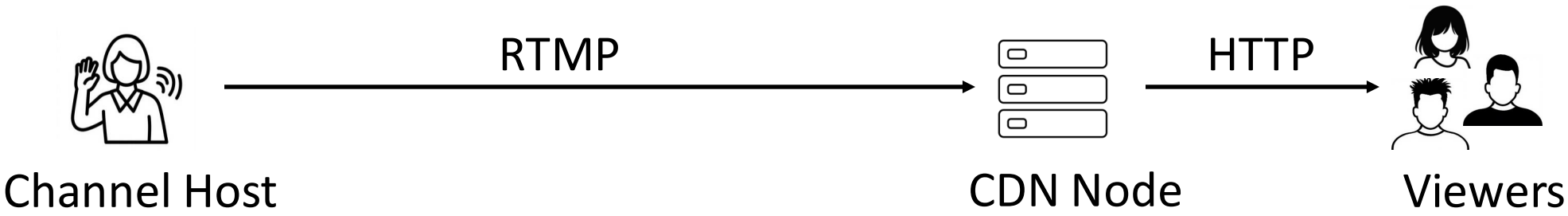



**Different audio codecs in different streaming modes make it worse!**

# Roadmap

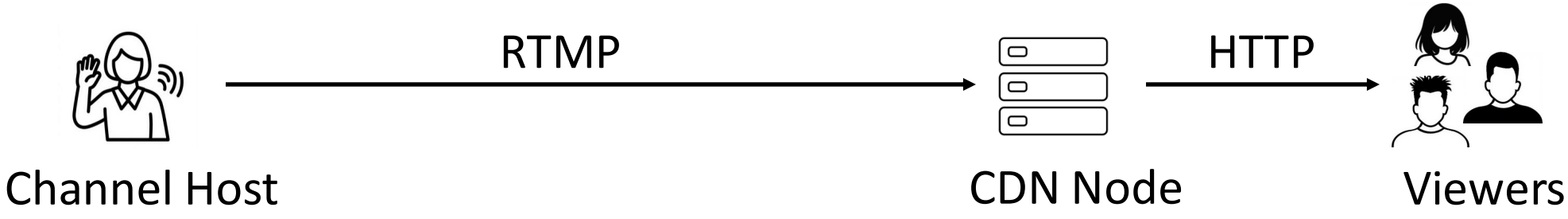
- Challenges of supporting an additional streaming mode
- Streaming architecture design of AnchorNet
- Audio splicing techniques for smooth streaming mode switching
- Deployment results

# Our Legacy Architecture: DualNet



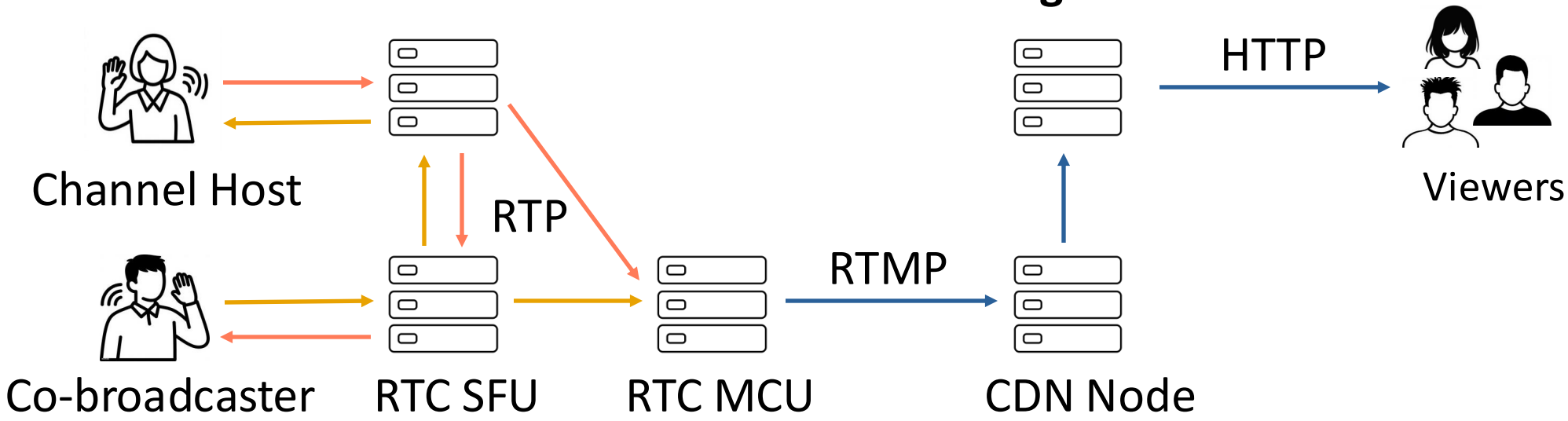

**Live Streaming**

# Our Legacy Architecture: DualNet

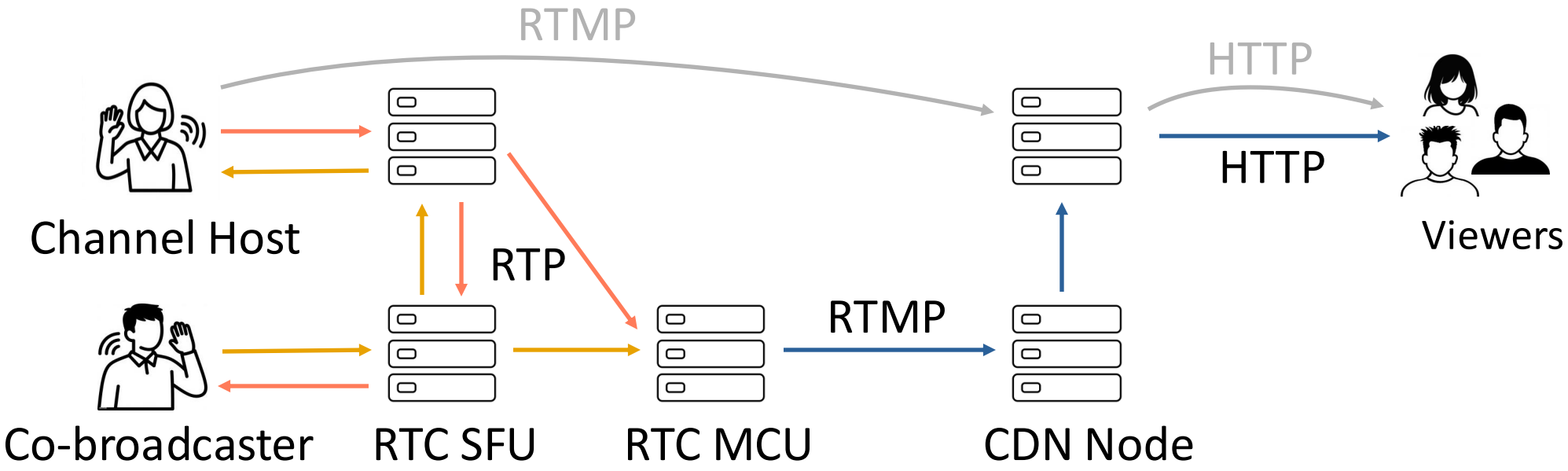



**Live Streaming**

**Collaborative Streaming**



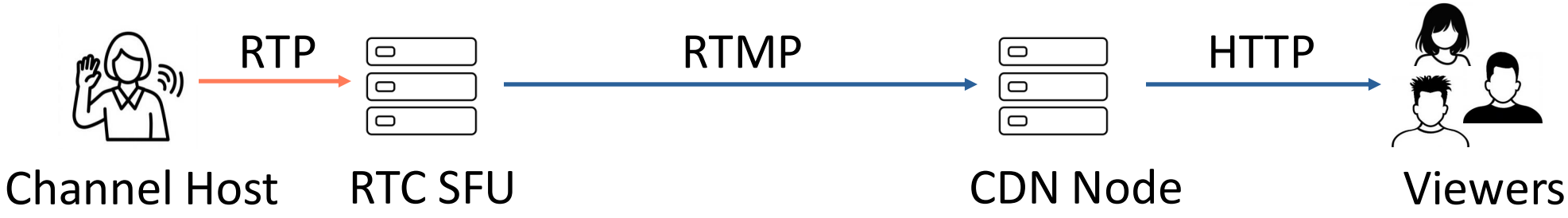
# Problems with DualNet




- CDN ingestion node may change
  - Stream splicing within CDN, involving coordination with 3rd CDN providers
- Implementation ossification

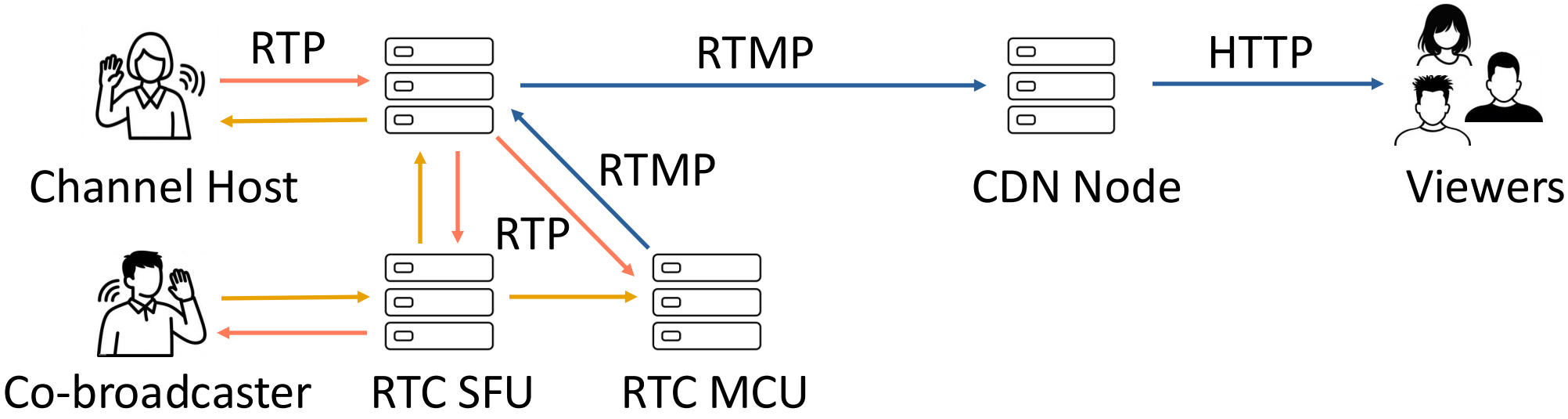
- Different publishing path
  - From host directly vs. relay through RTC servers
  - Switchover delay between two first-mile connections

# Unified AnchorNet Architecture

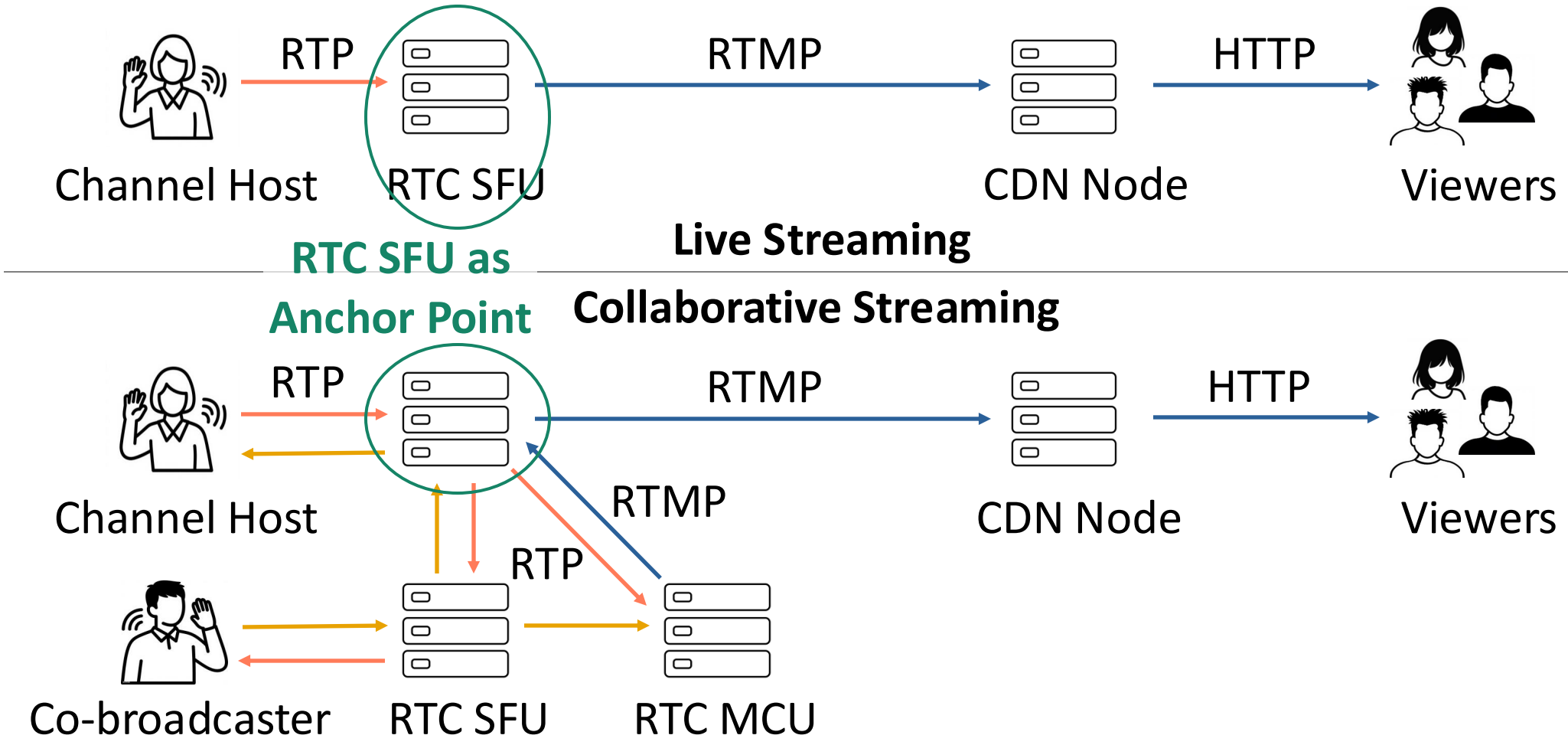



**Live Streaming**

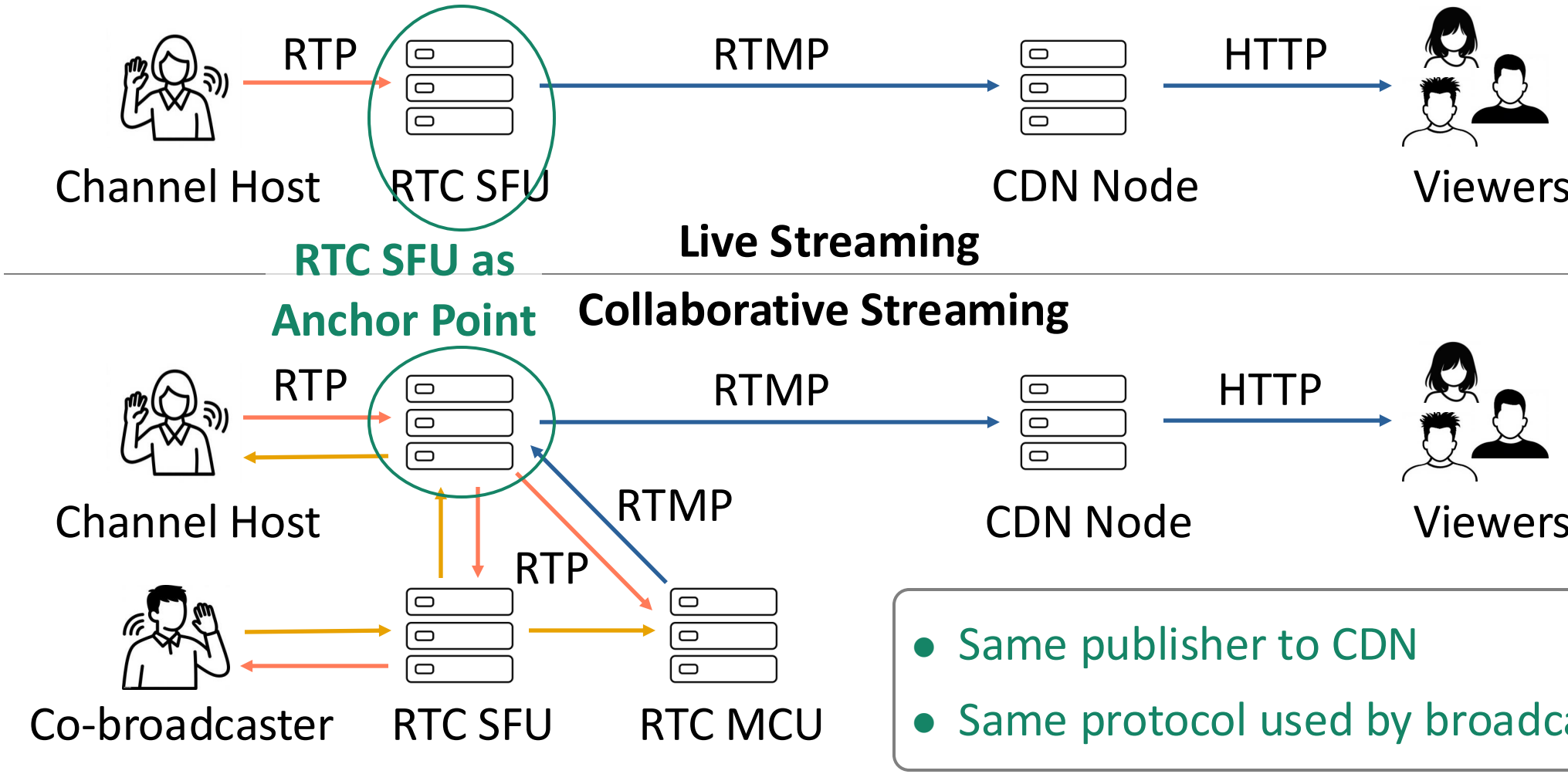
**Collaborative Streaming**



# Unified AnchorNet Architecture



# Unified AnchorNet Architecture

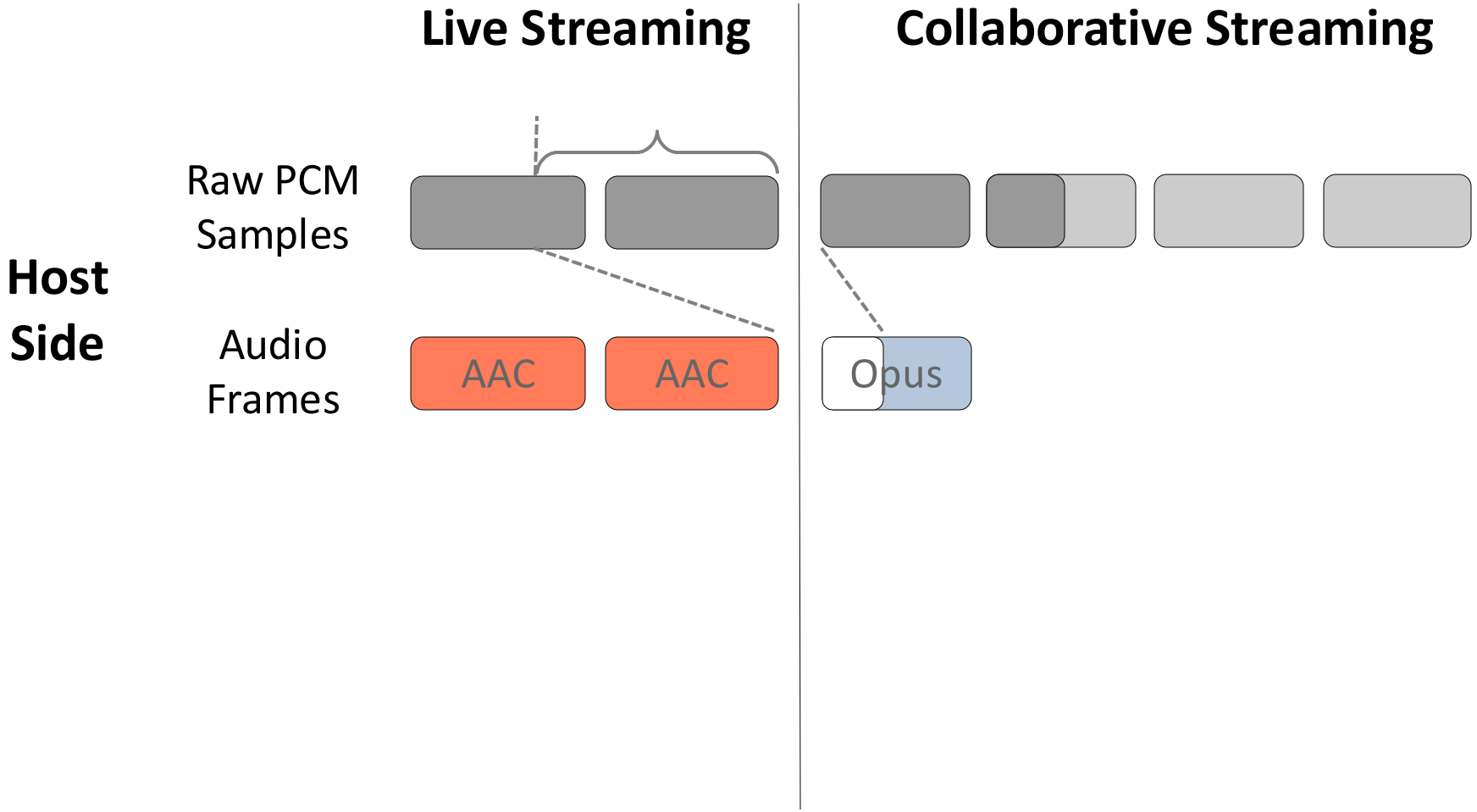



- Same publisher to CDN
- Same protocol used by broadcaster

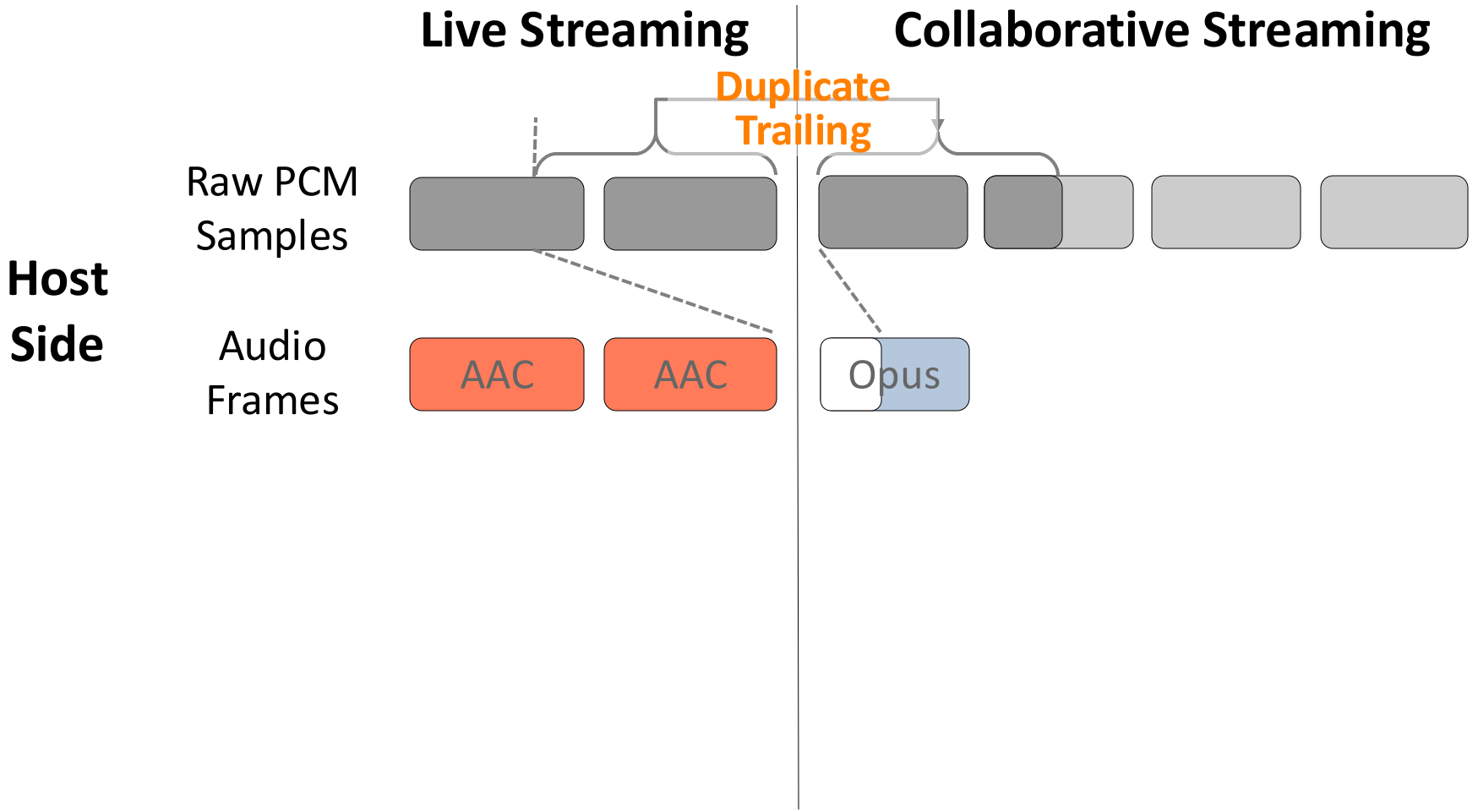
# Roadmap

- Challenges of supporting an additional streaming mode
- Streaming architecture design of AnchorNet
- Audio splicing techniques for smooth streaming mode switching
- Deployment results

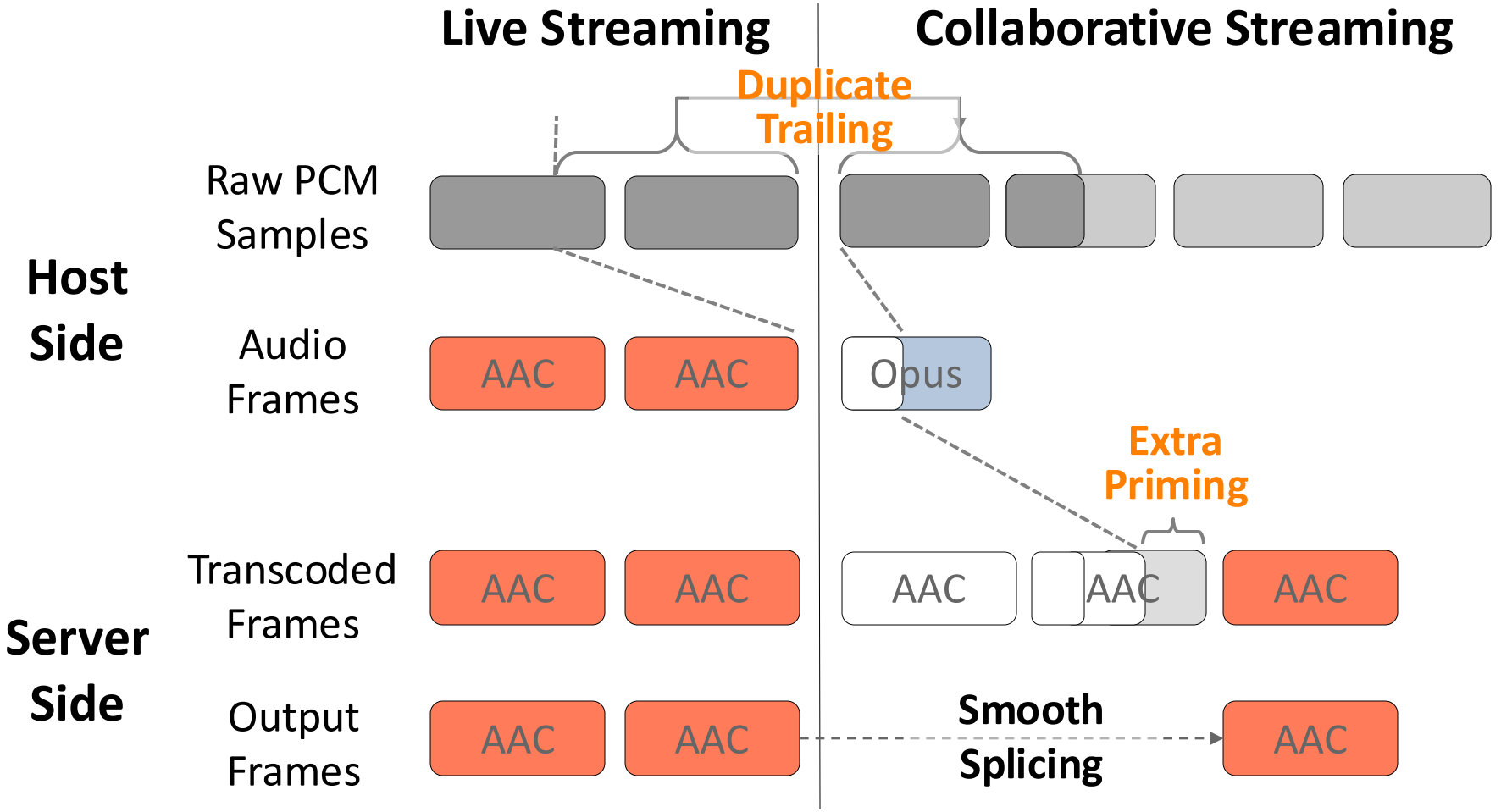
# Audio Sample-Level Stream Splicing



# Audio Sample-Level Stream Splicing



# Audio Sample-Level Stream Splicing



# Audio Sample-Level Stream Splicing

Refer to our paper for details of all four splicing techniques

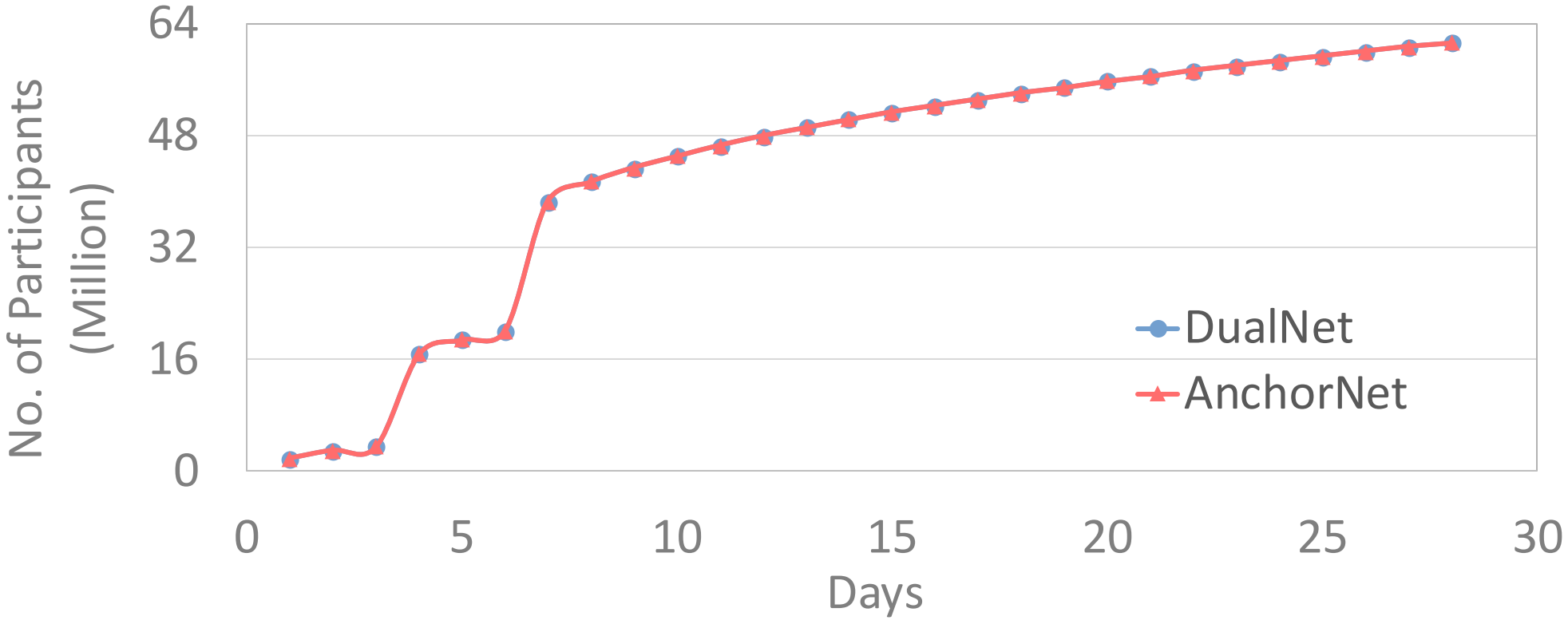
- Extra priming
- Duplicate trailing samples
- Rescale unaligned trailing samples
- Reconstructing overlapped MDCT window

  * used in time to frequency domain transform by many audio encoding algorithms

# Roadmap

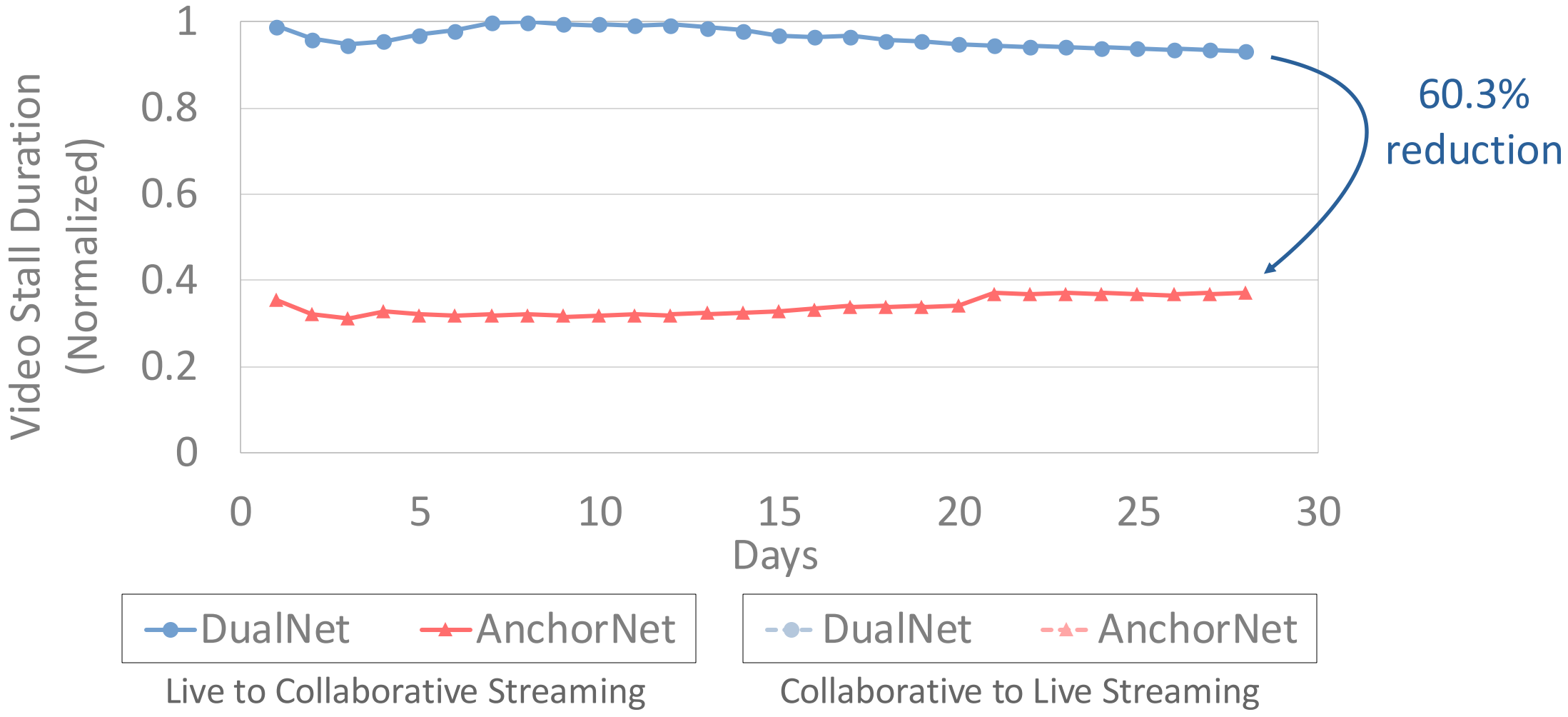
- Challenges of supporting an additional streaming mode
- Streaming architecture design of AnchorNet
- Audio splicing techniques for smooth streaming mode switching
- Deployment results

# Practical A/B Test

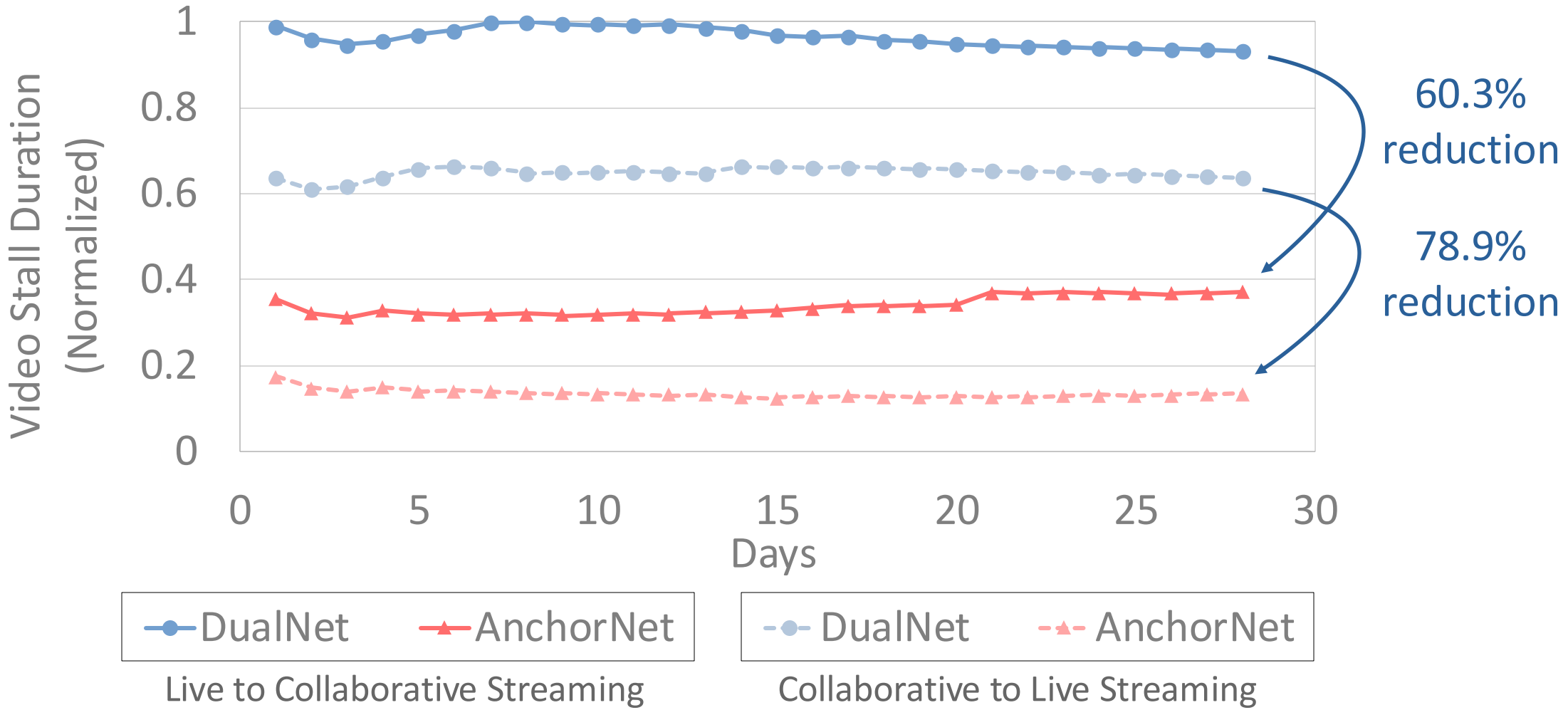



4-week A/B Tests conducted in 2024Q4, involving 10s of millions of users in each group

# Significantly reduced rebuffering



# Significantly reduced rebuffering

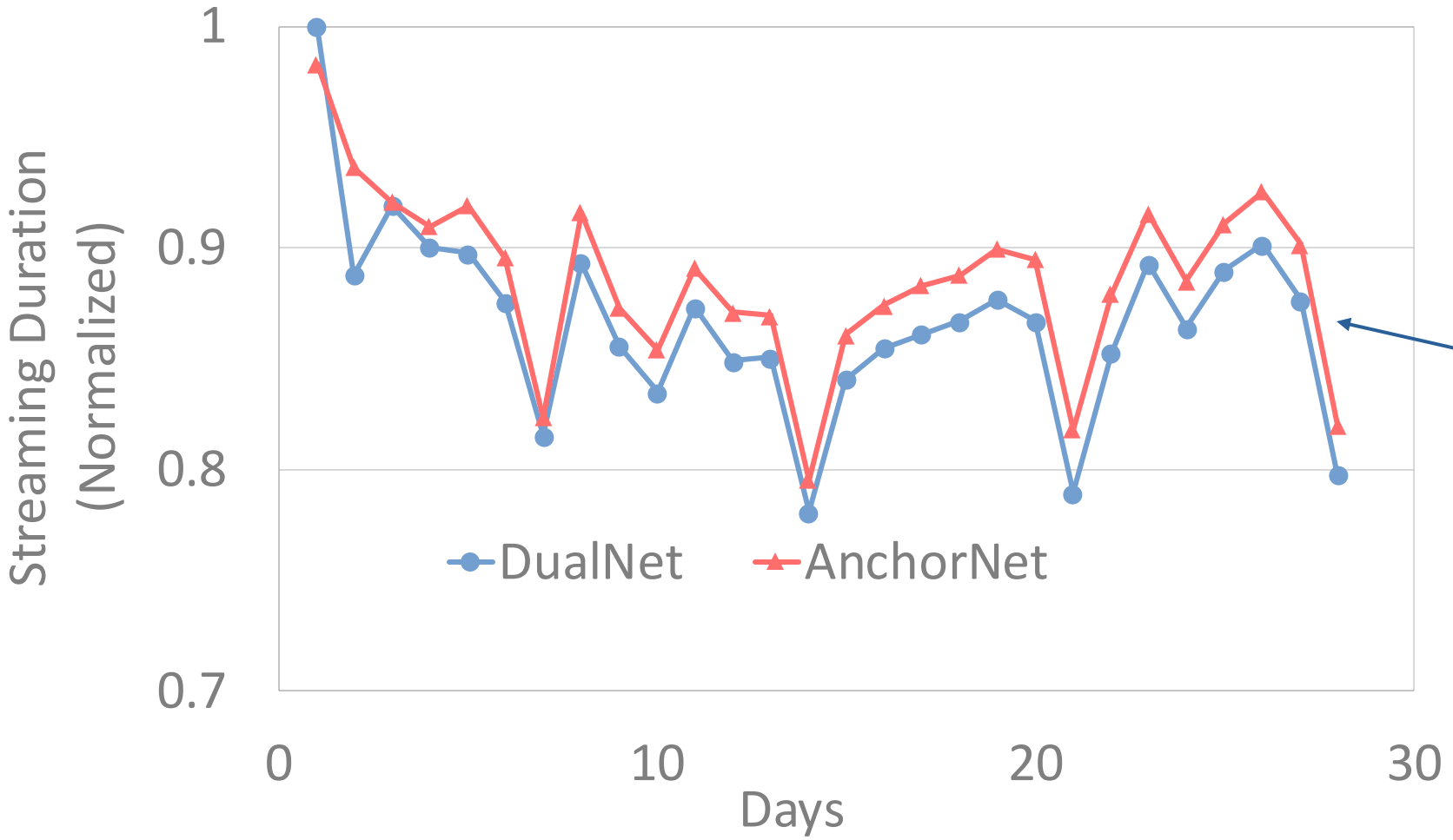


# Increased User Engagement

**Daily Active Time Increments:**

| | |
|---|---|
| Channel Host Streaming: | +0.53% |
| Co-Broadcaster Streaming: | +2.15% |
| Viewer Watching: | +3.83% |

# Increased User Engagement




Daily Active Time:

Channel Host: +0.53%

Co-Broadcaster: +2.15%

Viewer: +3.83%

# Further Optimization

- Currently, RTC MCU servers are mostly centrally deployed. Edge deployment may further optimize end-to-end latency.

- Joint optimization of collaborative stream mixing and CDN transcoding (e.g., for ABR, recorded playback)

Thanks!