WEBRTC

Aarya Tapaswi

TABLE OF CONTENT

Plugins are software components that enhance the functionality of an application by adding specific features or capabilities.

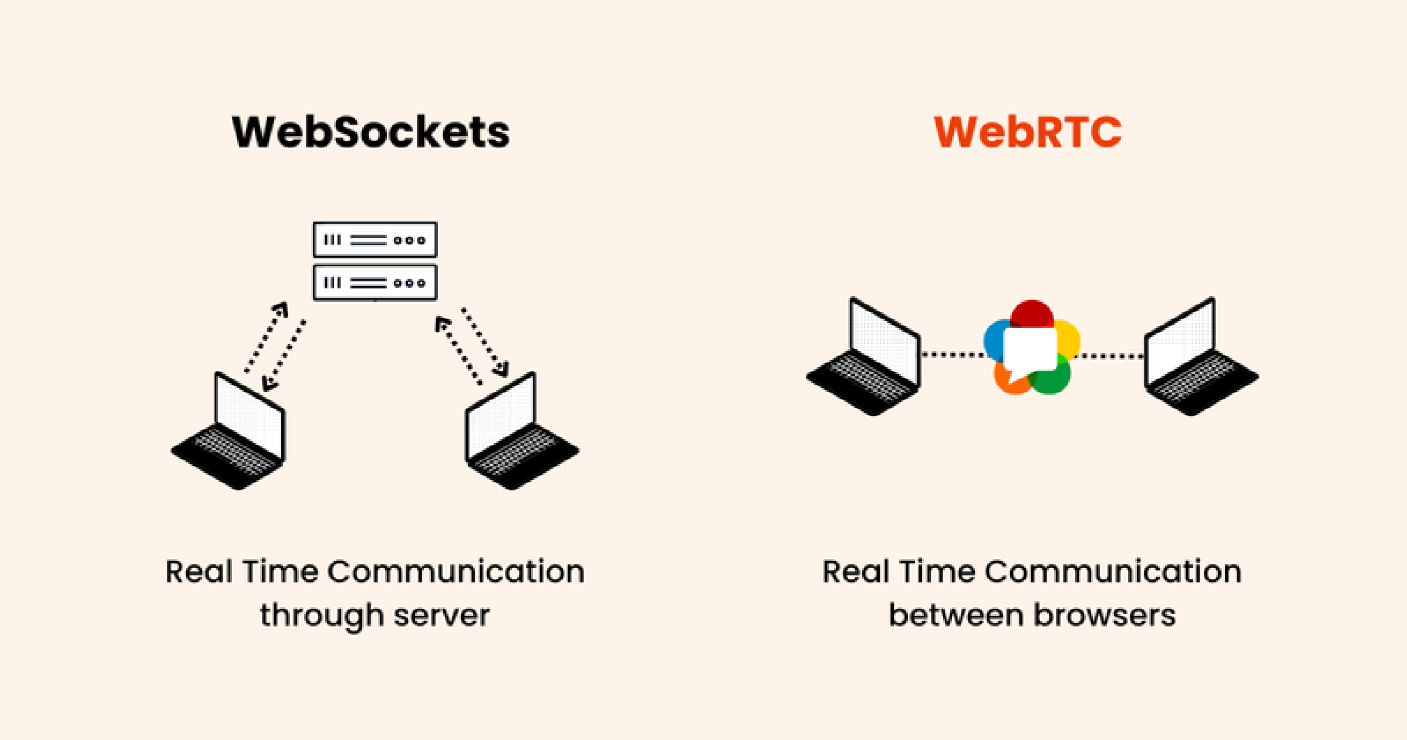
DEFINITION

WebRTC stands for "Web Real-Time Communication." It is an open-source technology and collection of APIs (Application Programming Interfaces) that enable real-time communication directly between web browsers and devices, without the need for plugins or additional software.

INTRODUCTION

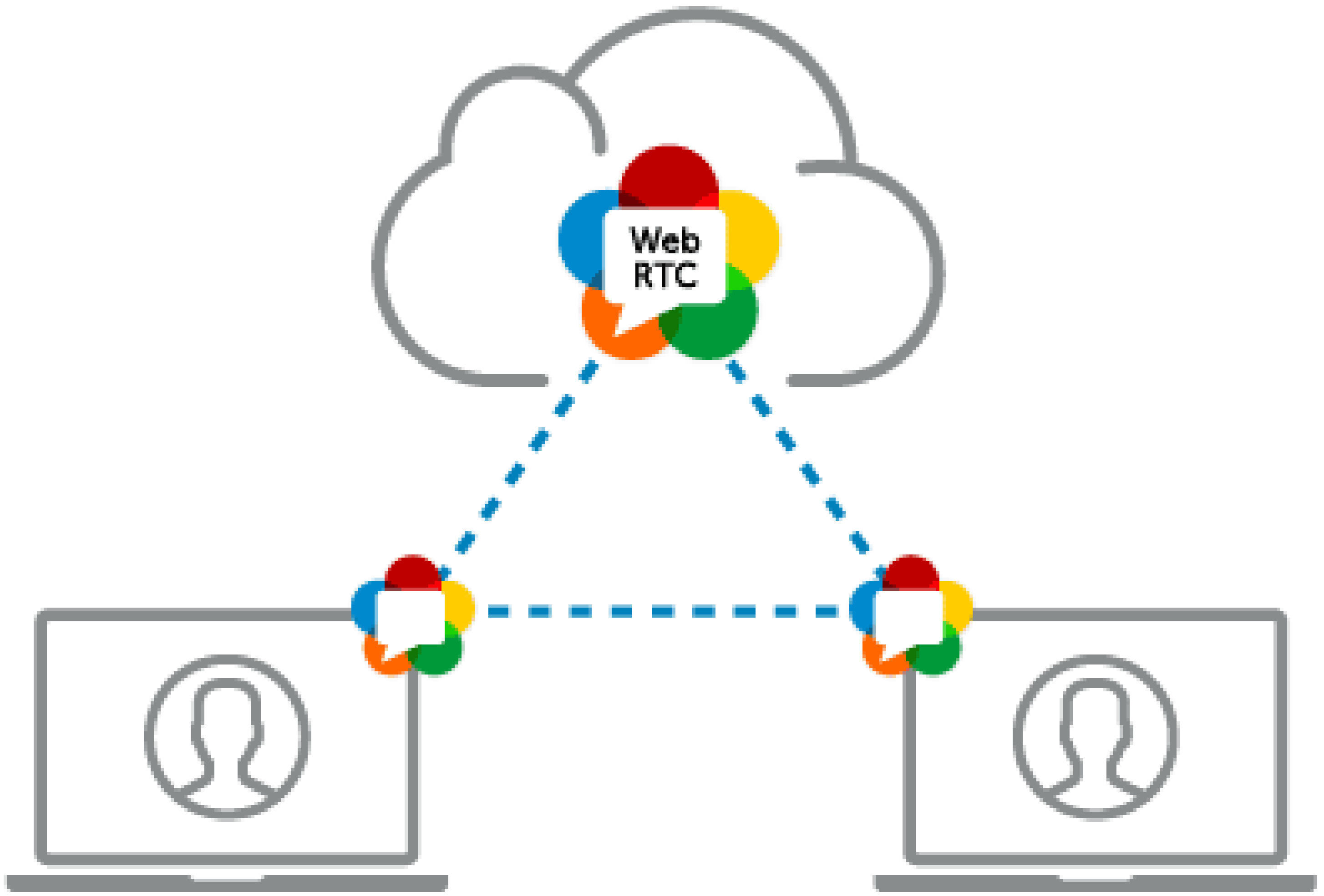
- enables video conferencing, voice calling, file sharing, online gaming, and more.
- built into modern web browsers like Chrome, Firefox, Safari, Edge, Android and iOS.

WebRTC vs Previous Technologies


WebSockets
Real Time Communication through server
WebRTC
Real Time Communication between browsers

ADVANTAGES

- enables real-time communication **directly within web browsers** without requiring users to install additional software or plugins.
- peer-to-peer architecture **minimizes communication delays**, providing a near-instantaneous response in voice and video interactions

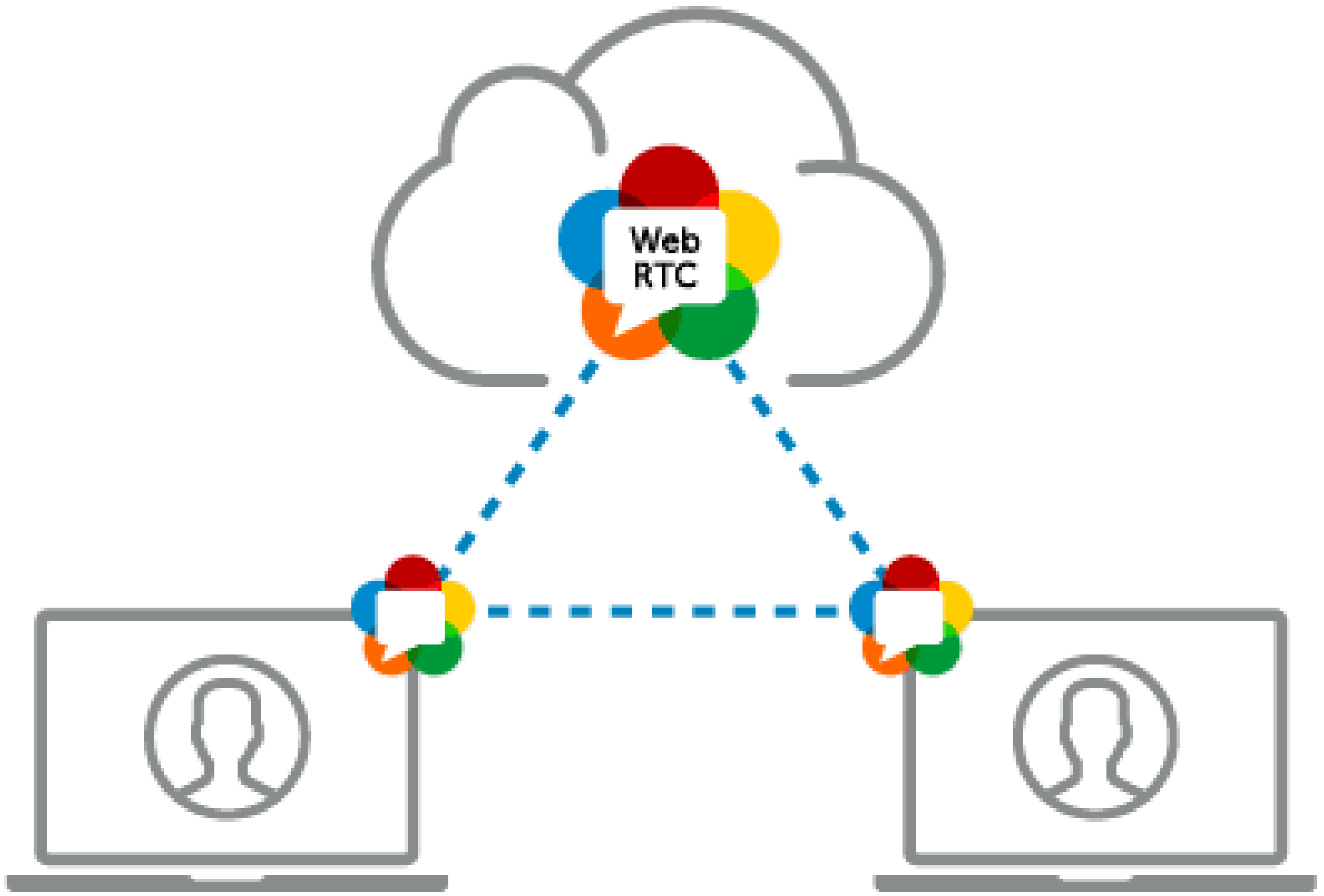


UDP (User Datagram Protocol) is a connectionless, lightweight networking protocol that enables fast data transmission without built-in error checking or reliability.

Overhead refers to the extra data or resources required by a system or protocol, often beyond what is necessary for its core functionality.

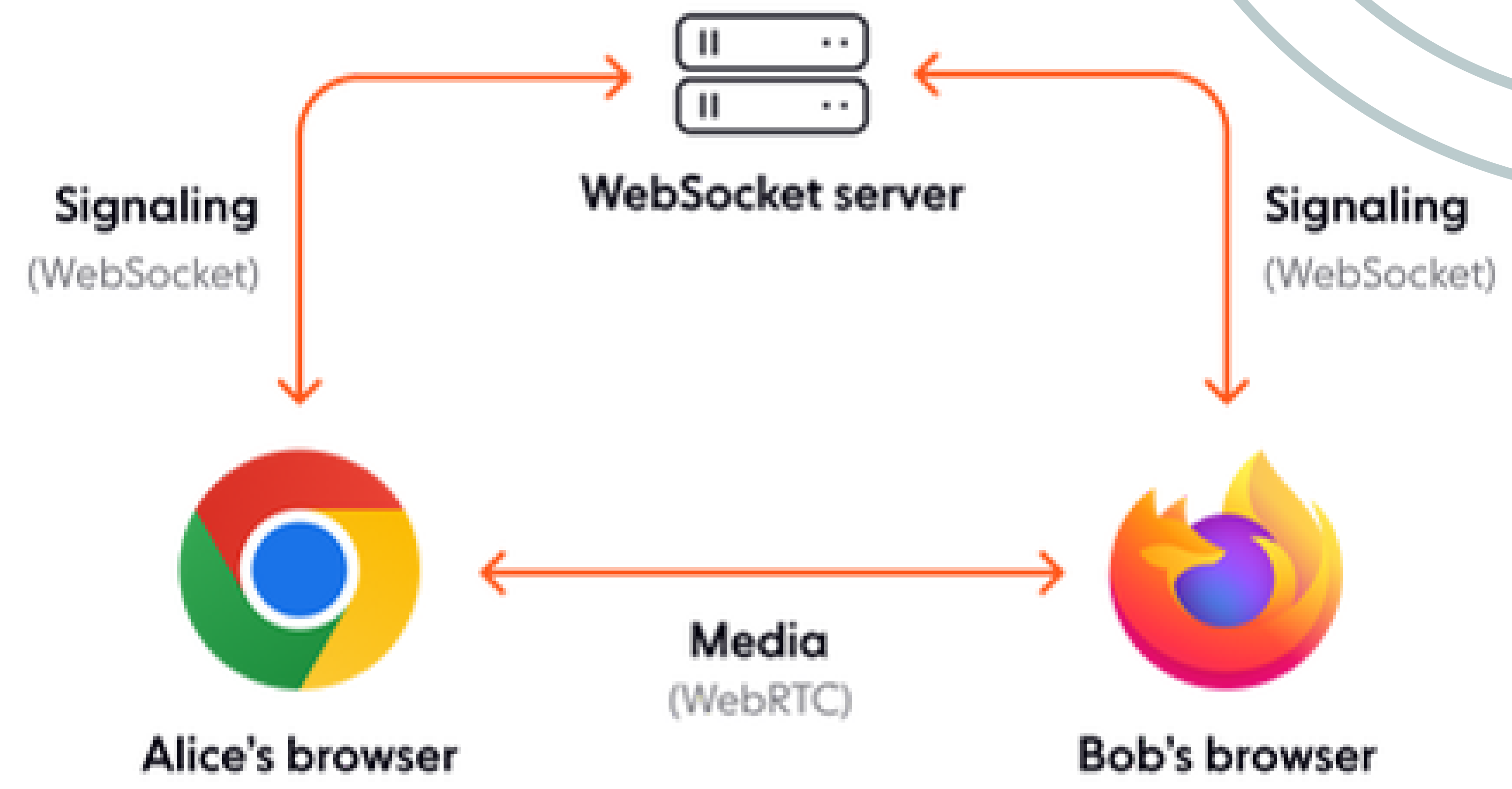
ADVANTAGES

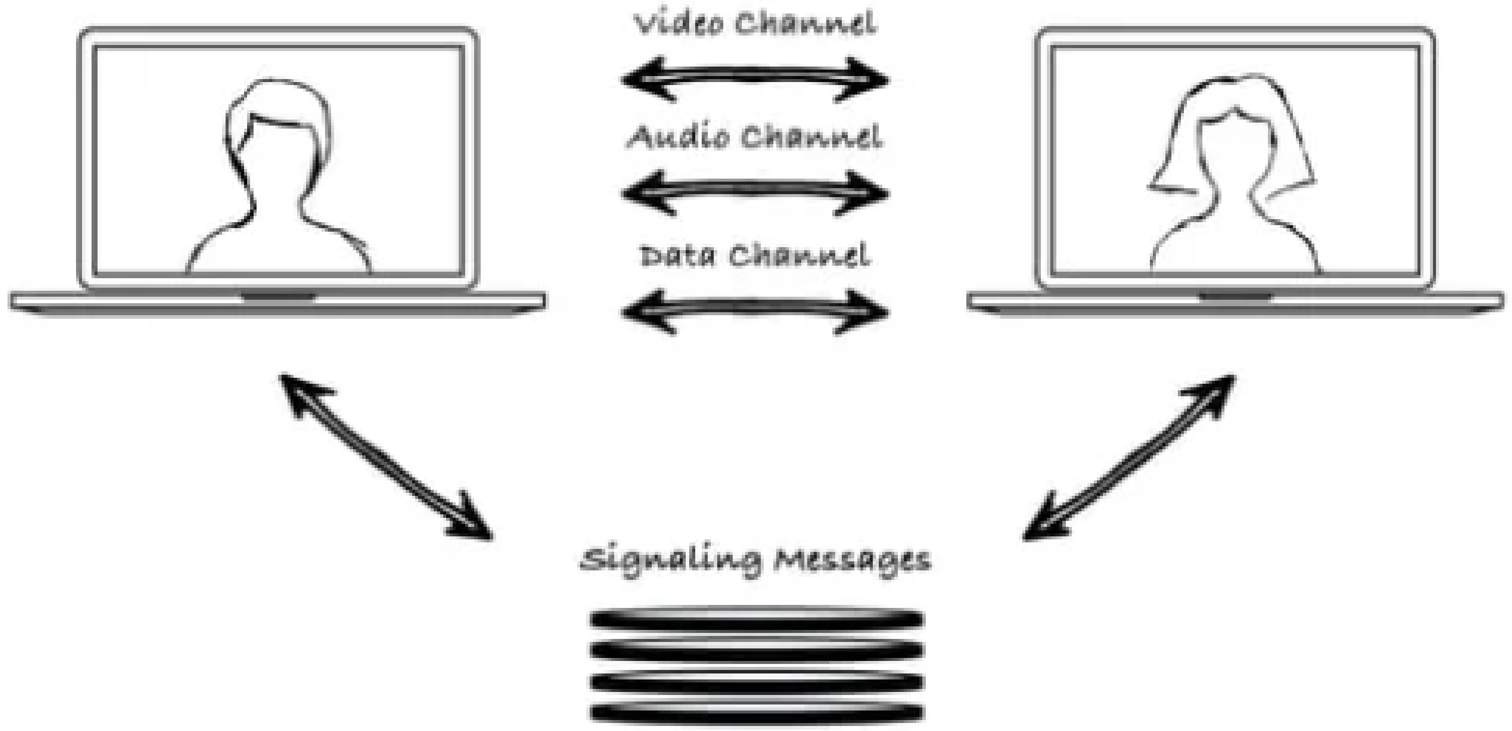
- leverages UDP for media streams because UDP is connectionless and provides lower overhead than TCP, making it suitable for applications where timely delivery is crucial, and a small amount of lost data can be tolerated.
- provides easy-to-use JavaScript APIs that streamline the integration of real-time communication features into web applications.



WORKING

- Signaling
- Offer and Answer Exchange
- ICE (Interactive Connectivity Establishment)
- Establishing Peer-to-Peer Connection
- Media Streaming
- Data Channel





WEBRTC APIS

MediaStream (GetUserMedia)

- access device cameras and microphones using JavaScript.
- controls where multimedia stream data is consumed.
- provides some control over the devices that produce the media.
- exposes information about devices able to capture and render media.

RTCPeerConnection

- enables creation of direct connections with their peers without the need for an intermediary server.
- SDP negotiation.
- Codec implementations.
- NAT Traversal.
- packet loss.
- bandwidth management.
- media transfer.

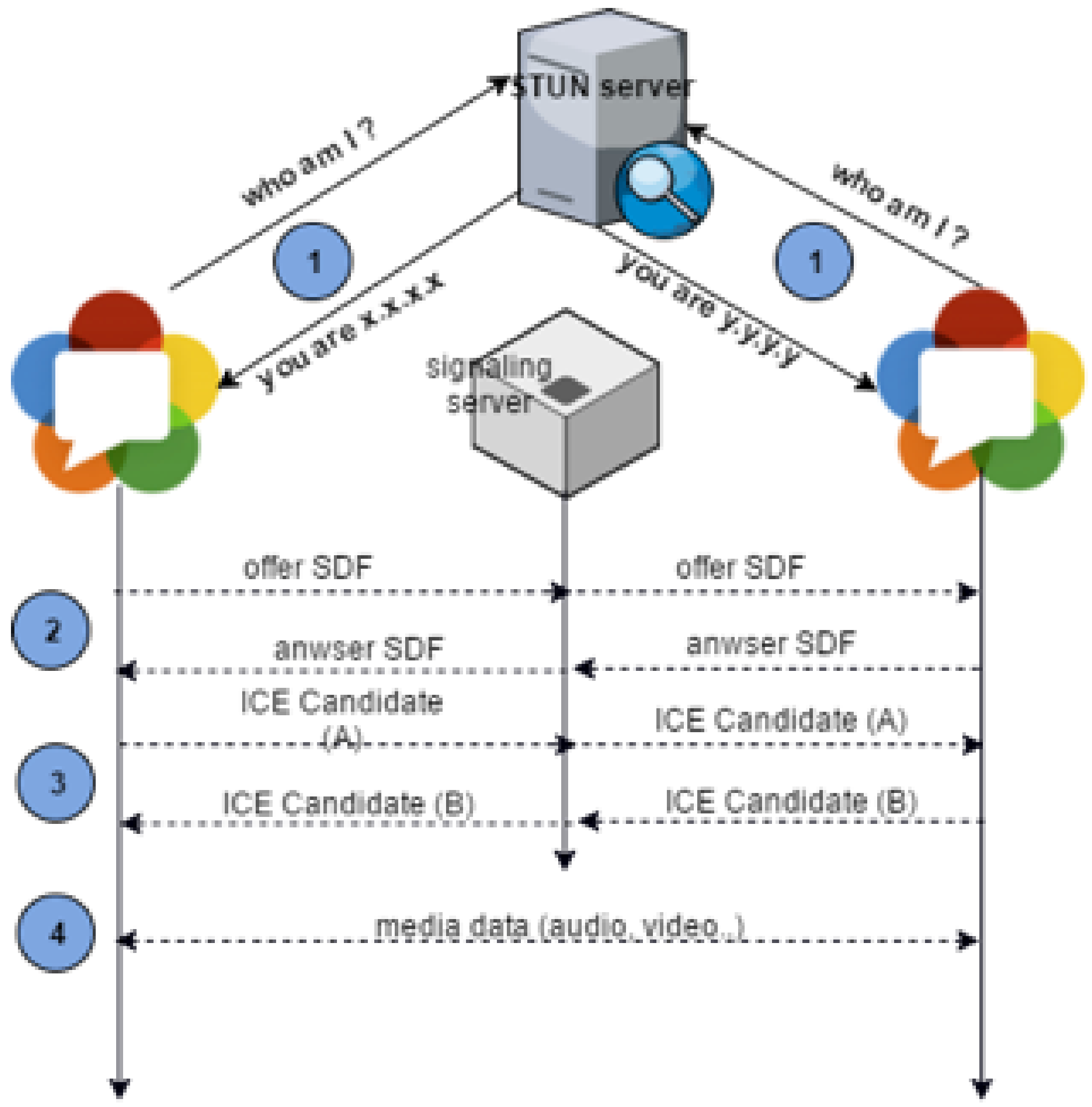
RTCDataChannel

- allow bi-directional data transfer.
- data channels use UDP-based streams with the Stream Control Transmission Protocol (SCTP) protocol

SIGNALLING

- exchange metadata to coordinate communication.
- often formatted using the Session Description Protocol (SDP).
- Done to know each others capabilities and network addresses.
- solved through a regular HTTP-based Web API

SDP (Session Description Protocol) is a text-based format used to describe multimedia sessions for establishing communication over networks.



NAT Traversal

NAT traversal is done to enable direct communication between devices behind different Network Address Translation (NAT) setups, allowing them to overcome the barriers imposed by private IP addresses and communicate over the public internet.

Interactive Connectivity Establishment (ICE)

ICE is a mechanism that allows devices to find the best possible connection path for real-time communication while navigating the complexities of various network configurations.

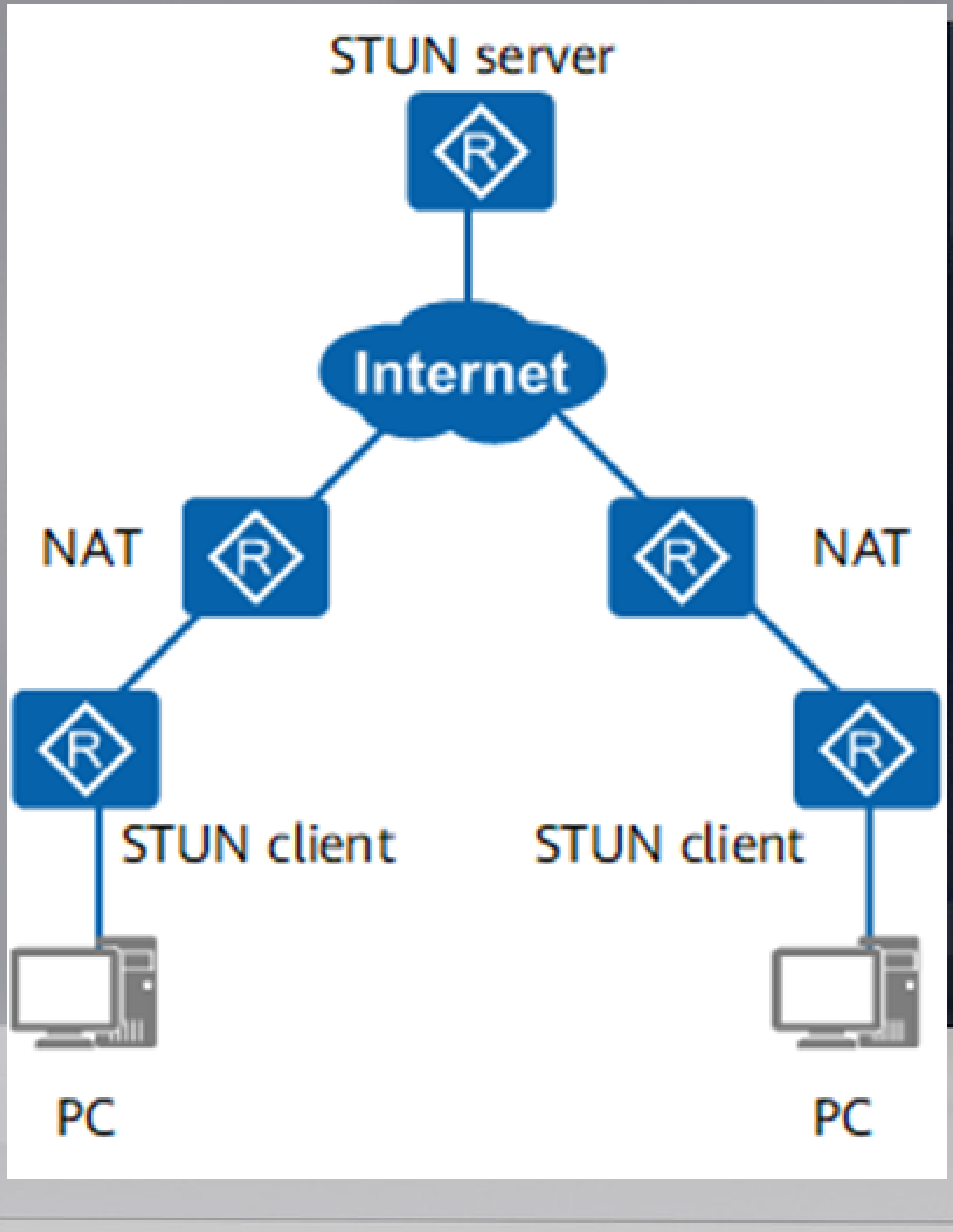
It employs a combination of candidate gathering, connectivity checks, prioritization, and adaptive switching to ensure successful peer-to-peer connections in a wide range of network scenarios.

ICE

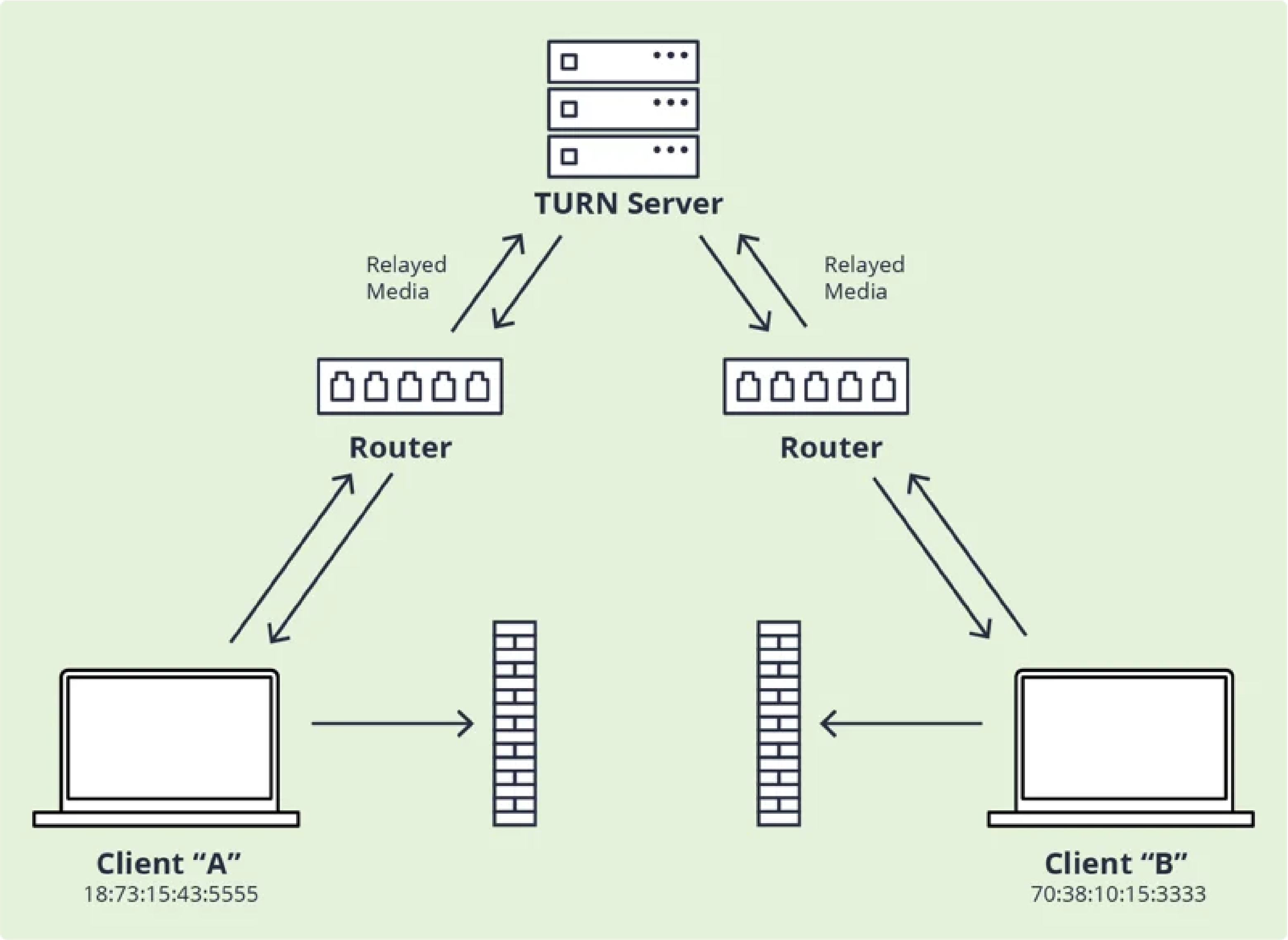
Session Traversal Utilities for NAT (STUN)

STUN allows devices to discover their public IP address and determine the type of NAT they are behind, which is crucial for setting up peer-to-peer connections.

One **disadvantage** of STUN is that it only works in scenarios where the NAT is simple. In complex cases, STUN may not be able to correctly determine the external port mapping and IP address, leading to connection issues



Traversal Using Relays around NAT (TURN)



The TURN server assists in the NAT traversal by helping the endpoints learn about the routers on their local networks, as well as blindly relaying data for one of the endpoints where a direct connection is not possible due to firewall restrictions.

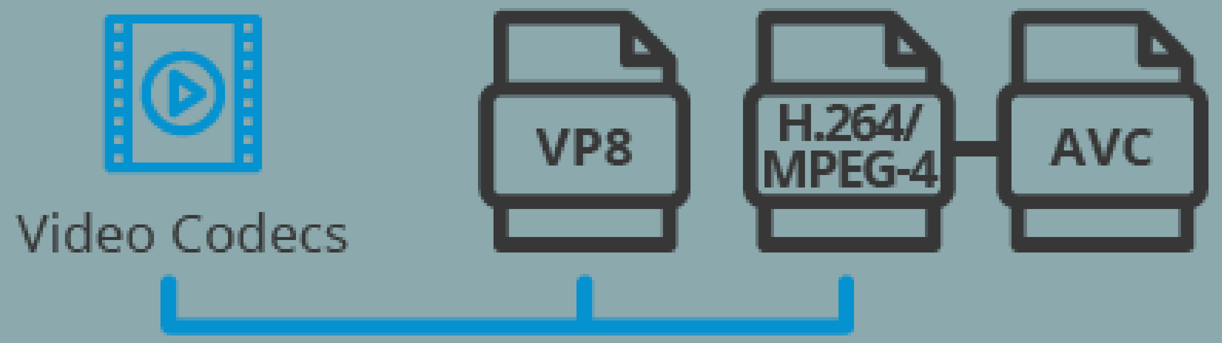
The main drawback of TURN is the potential increase in latency due to an additional hop in the communication path.TURN servers can experience high load, especially in scenarios with a large number of users or intense media streaming, which can affect their performance.

CODECS

Before sending the media over a peer connection, it has to be compressed. Raw audio and video is simply too large to send efficiently in our current Internet infrastructure. Likewise, after receiving media over a peer connection, it has to be decompressed.

WebRTC has mandated three audio codecs and two video codecs:

1. Audio - PCMU (G.711μ) running at 8,000Hz with a single channel (mono).
2. Audio - PCMA (G.711a) running at 8,000Hz with a single channel (mono).
3. Audio - Opus running at 48,000Hz with two channels (stereo).
4. Video - VP8.
5. Video - H.264/AVC using Constrained Baseline Profile Level 1.2.

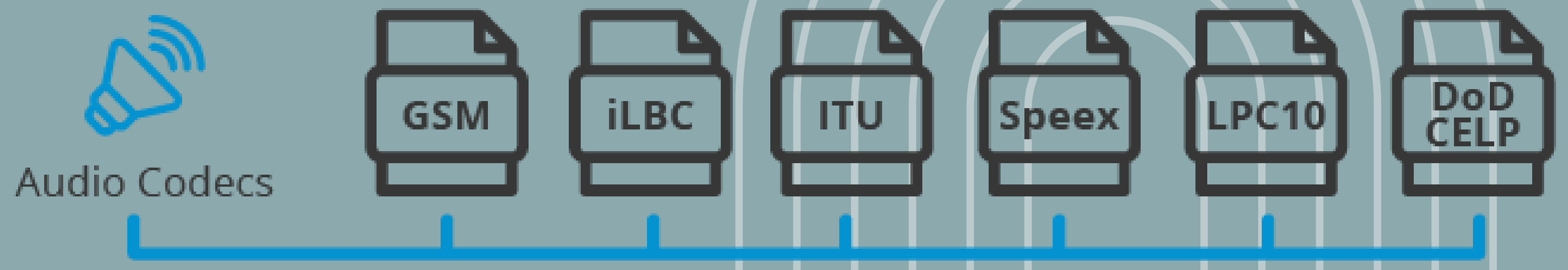


Video Codecs

VP8

H.264/ MPEG-4

AVC



Audio Codecs

GSM

iLBC

ITU

Speex

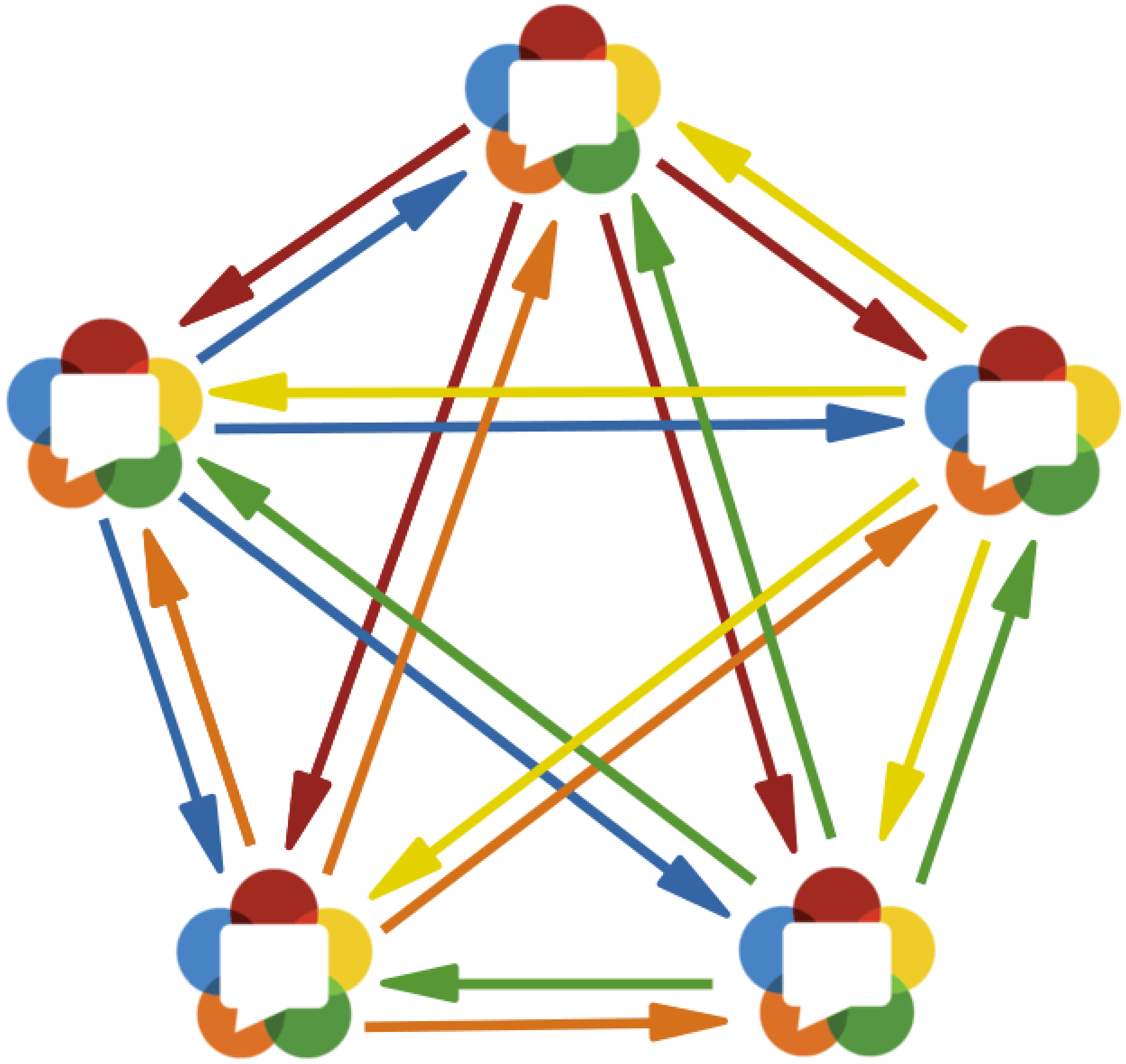
LPC10

DoD CELP

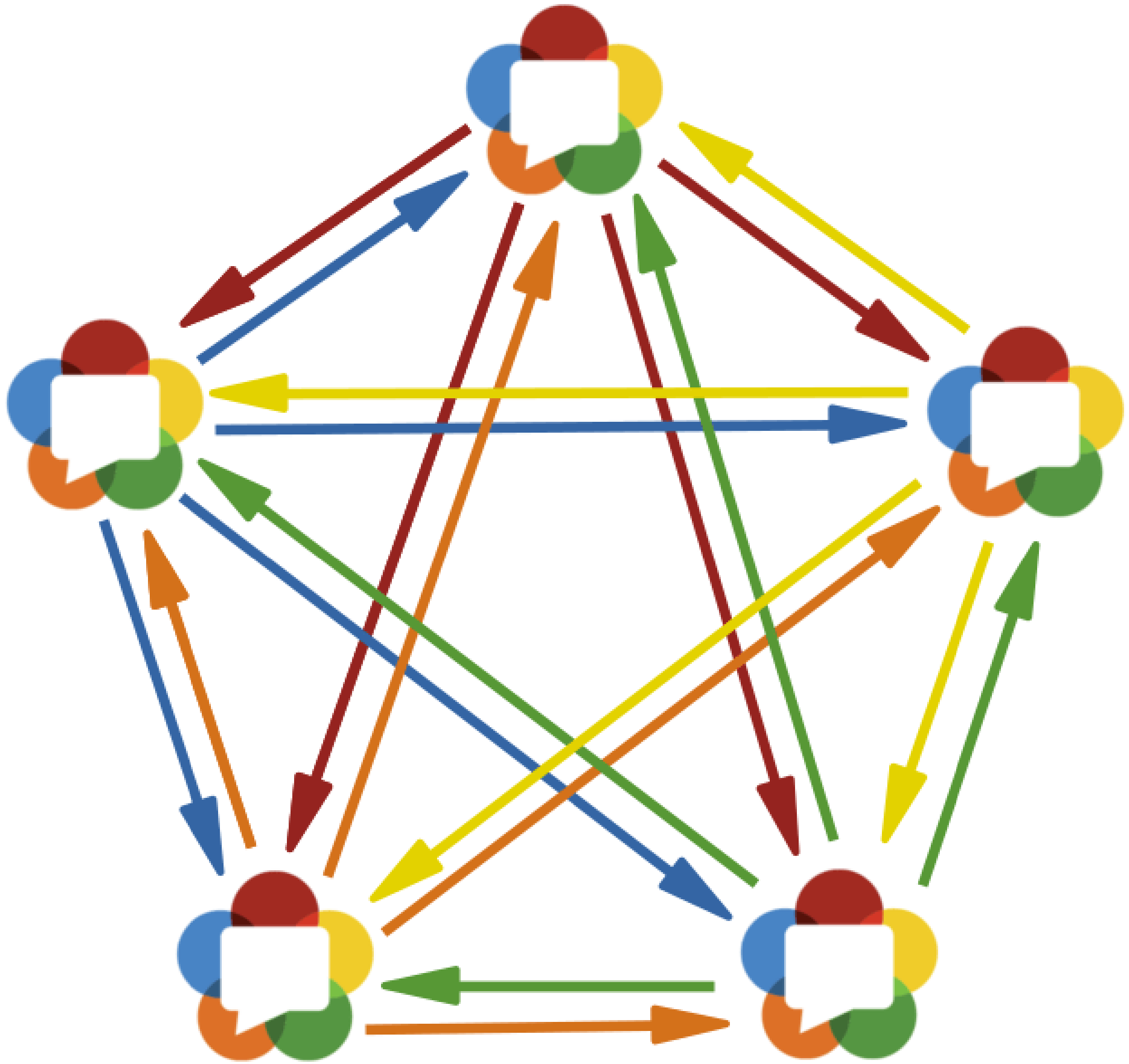
WebRTC Topologies

1. Peer-to-Peer

Definition: Peer-to-peer WebRTC topology refers to a communication architecture where data, such as audio, video, and messages, is exchanged directly between individual users (peers) without the need for a central server to relay the communication.
There are n*(n1) number of connections where n is the number of peers.



1. Peer-to-Peer



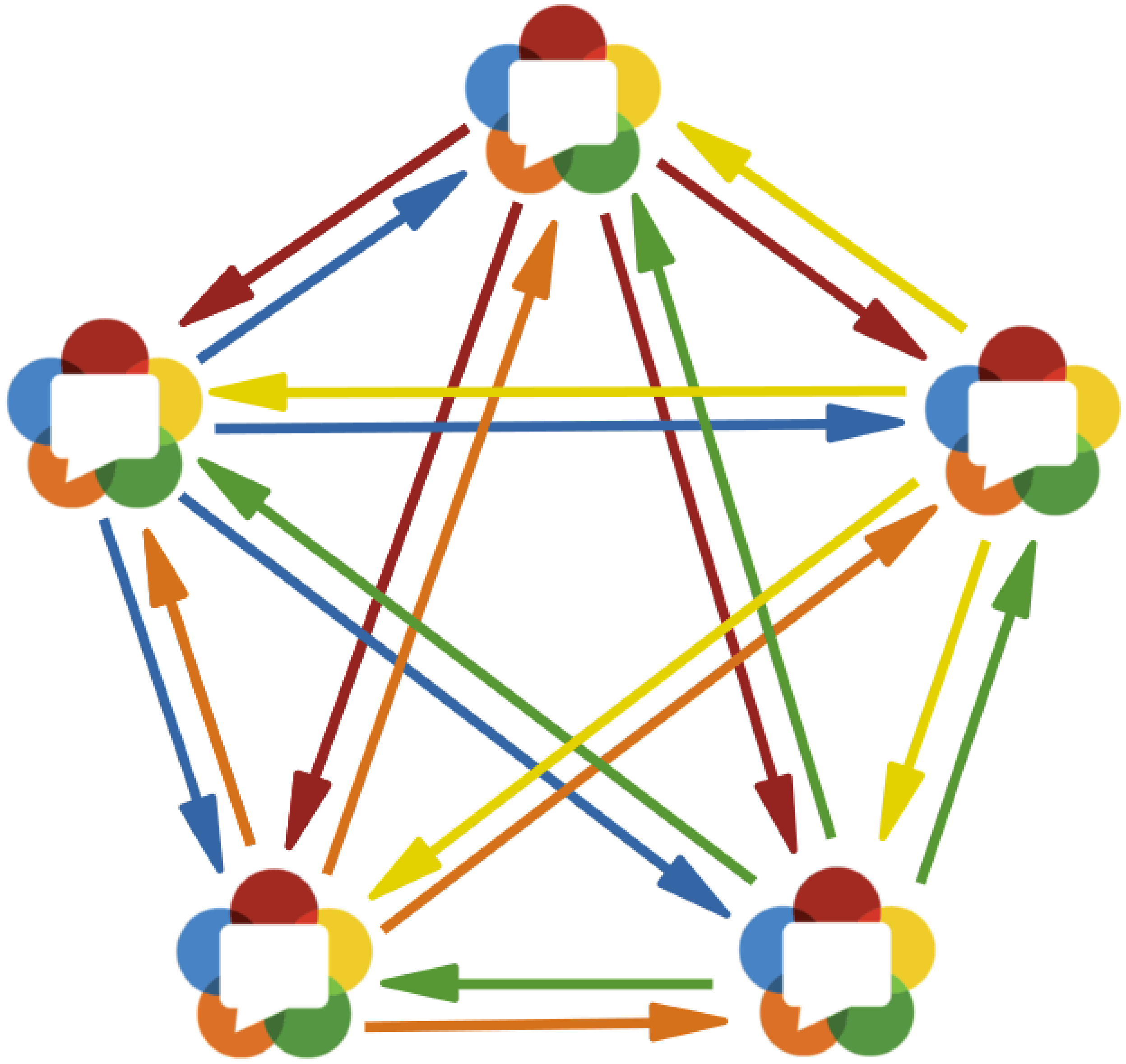
Advantage:

- Low Latency: lack of hops between initial and final destination.
- Reduced Server Load: communication happens directly between pairs thus reducing load on other servers.
- Privacy: no interference of other devices during transmission.
- Scalability: load on central servers is reduced making it suitable for applications with a large user base.
- Real-Time Interaction: suitable for video conferencing, online gaming, and collaborative applications.

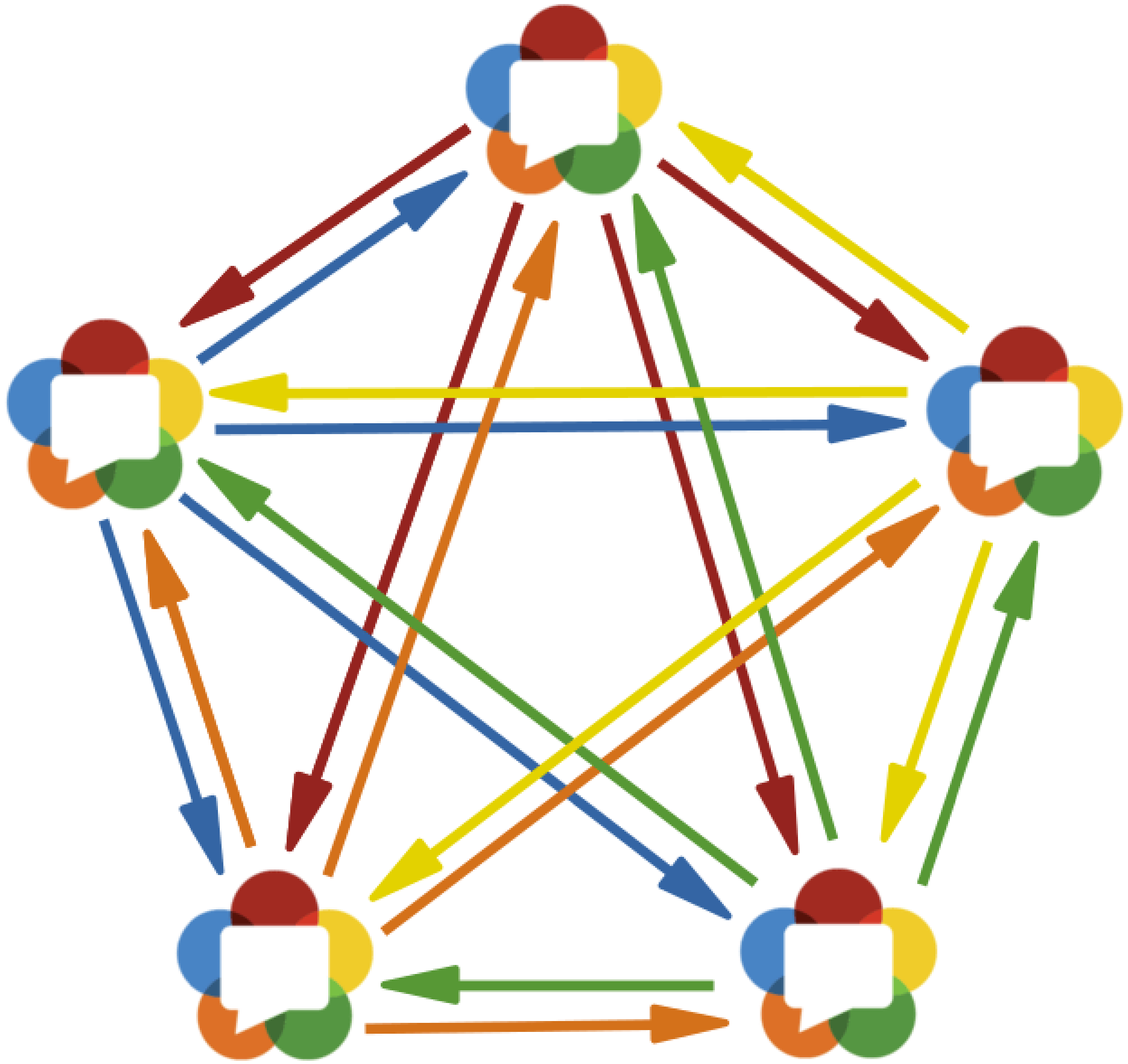
1. Peer-to-Peer

Disadvantages :

- NAT and Firewall Limitations: Some users might be behind restrictive NATs or firewalls, making direct peer-to-peer connections challenging to establish.
  Solution: TURN Servers
- **Reliability**: Since connections rely on the stability of users' network connections, If one peer experiences network disruptions, it can impact the entire connection.
- **Complexity**: NAT traversal and candidate gathering, can be challenging to implement correctly.



1. Peer-to-Peer



- Use Cases:
- 1-on-1 Customer Service
- Language Tutoring
- Legal Consultations.
- Financial Advising.
- Online Therapy and Counseling

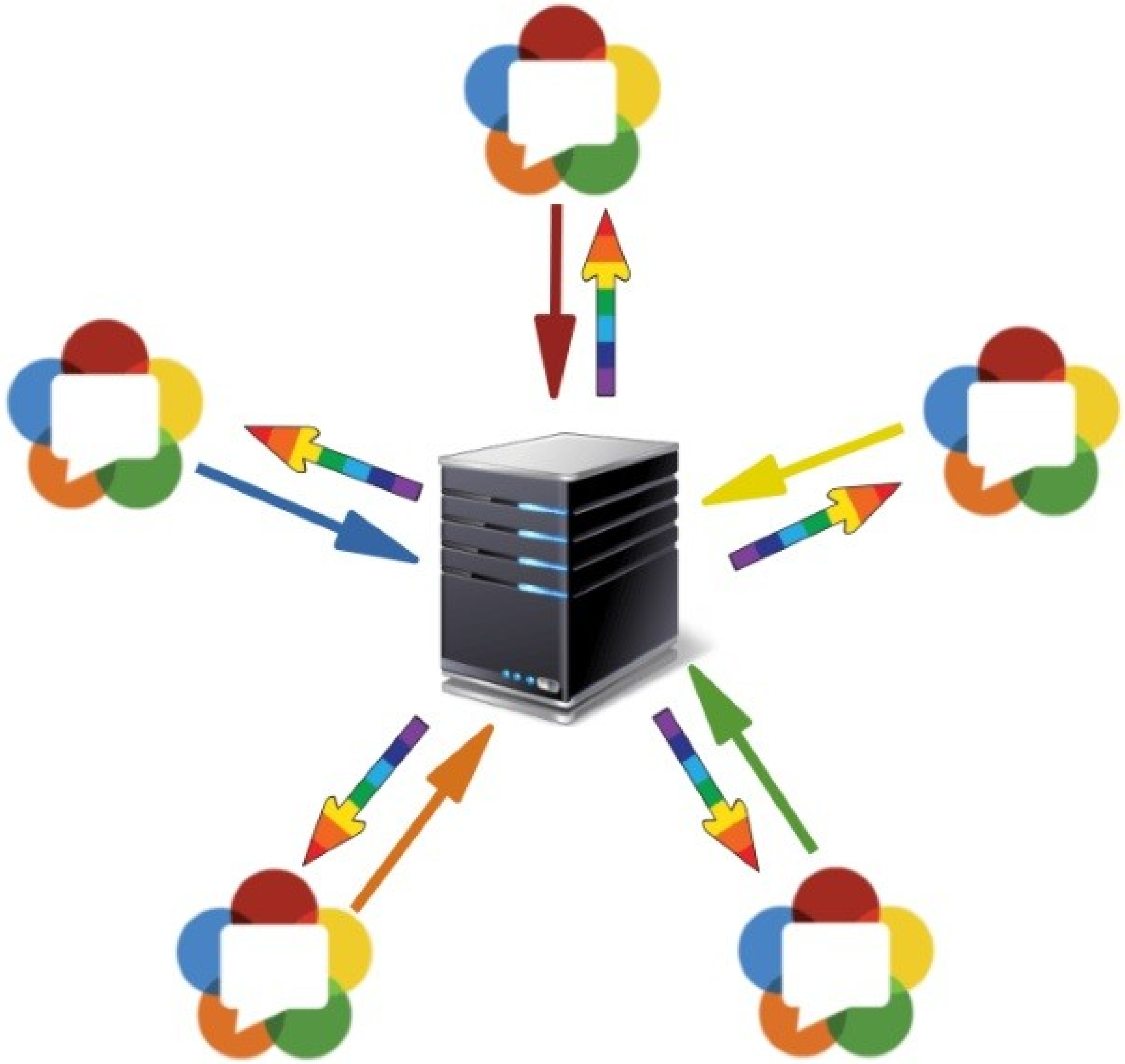
2. Multipoint Control Unit

Definition:

each participant in a session connects to a server that acts as a multipoint control unit (MCU)
The MCU acts as a central hub that receives audio and video streams from multiple participants and then mixes, processes, and redistributes these streams to all other participants.

Advantage:

This requires less bandwidth usage and device CPU but it does require additional server CPU for mixing audio/video into single streams. MCU's are also a great option for dealing with poor network conditions as it provides the lowest possible bandwidth usage for each individual participant.



Disadvantage:

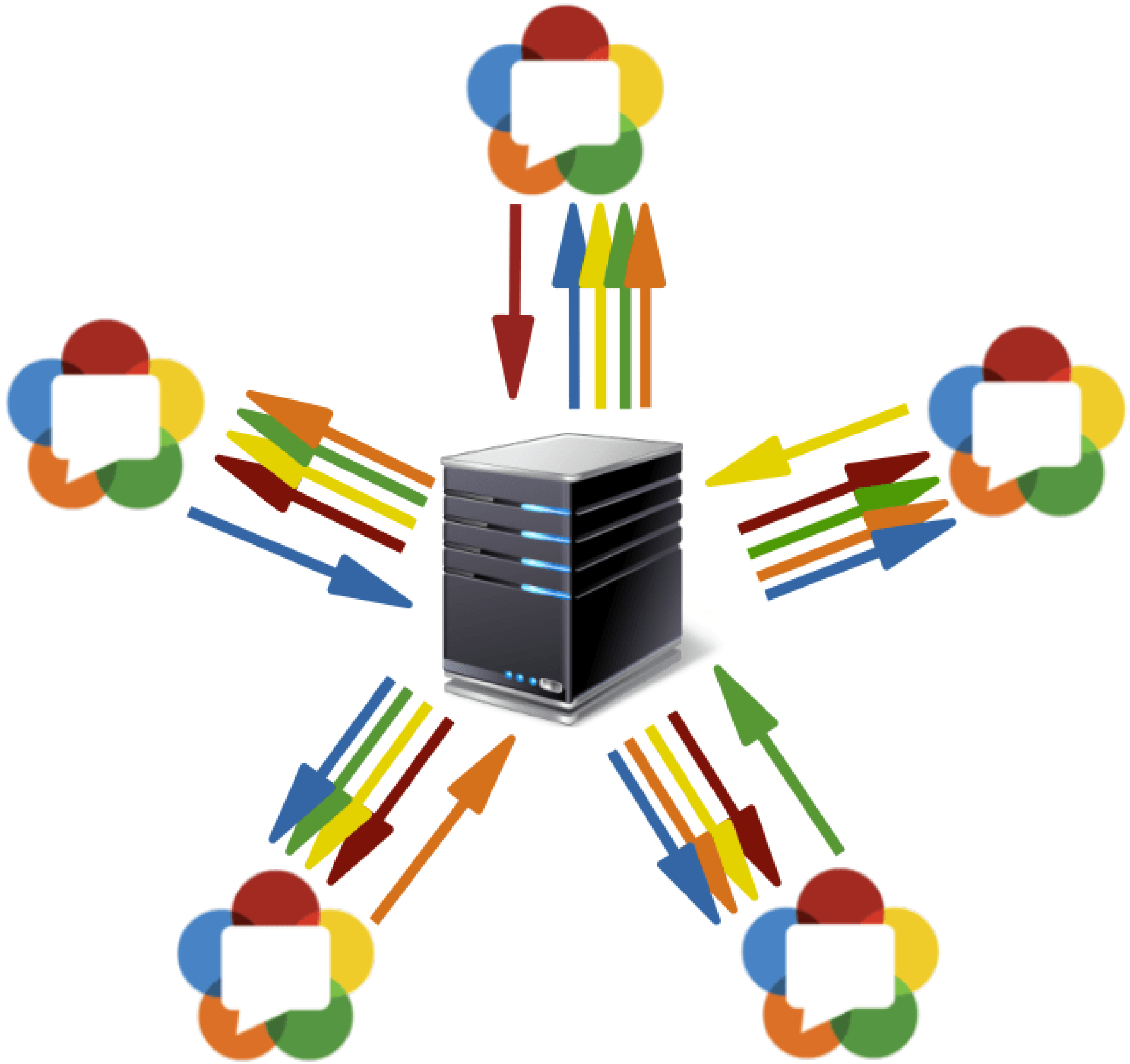
- Single point failure
- Added latency
- Compared to peer-to-peer, higher overall bandwidth consumption
- Need for powerful MCU server

3. Selective Forwarding (SFU)

Definition:
In a selective forwarding topology, each participant in a session connects to a server that acts as a selective forwarding unit (SFU). Each participant uploads their encrypted video stream one time to the server. The server then forwards those streams to each of the other participants using multiple downstream connections.

Advantage:
This reduces latency and also permits things like transcoding, recording, and other server-side integrations such as SIP which would be much more difficult in a peer-to-peer connection.
This topology is generally considered the most balanced.

Disadvantage:
having multiple downstream connections means each client will eventually run out of resources once a certain number of participants is active in the session.



HYBRID TOPOLOGY

maintain a mix of Peer-to-Peer, Selective Forwarding, and Multipoint Control (Mixing) architectures

topologies can change as participant counts increase and decrease

Utilizes MCU for larger conferences while retaining mesh for smaller groups, optimizing resource usage.

Mesh minimizes server load for point-to-point, while MCU streamlines resource-intensive multiparty scenarios.

Suited for platforms needing versatile communication capabilities, from small team collaborations to large conferences.

FILE TRANSFER

WebRTC Datachannels API allows developers to transmit arbitrary data directly between two users.

Before we can send messages via WebRTC, we need to verify that the other peer we want to connect to is online, as WebRTC would not connect with offline peers.

The idea behind file transfers using WebRTC is to convert files into buffers or DataUrl, transmit them in chunks, then compile them back to the file it originally was, on the receiving end.

THANK YOU