

P2P vs MCU vs SFU at a glance

CRITERION	P2P	MCU	SFU
Server in audio path	None	Yes, heavy	Yes, light
Server decodes audio	n/a	Every stream	Never
Server CPU per room	Zero	Very high	Low
Client downloads	1 per person	1 (the mix)	1 per speaker
Added latency	Lowest	Highest	Low
Scales to big rooms	No	CPU-limited	Yes (default)
Single-file recording	No	Easy	Needs egress
Dumb receivers (PSTN)	No	Yes	No

The SFU trick: forward only the talkers (RFC 6464)

What it is	Each sender stamps every audio packet with its loudness, 0-127 in -dBov.
Where it lives	In an RTP header extension (RFC 6464 / RFC 5285) - readable without decode.
Why it matters	The SFU ranks speakers from the number alone, then forwards only the loud few.
The V bit	One bit flags whether the sender thinks the packet holds real voice.

What real engines do

mediasoup	SFU. ActiveSpeakerObserver uses RFC 6464 levels (Volfin-Cohen). No server decode.
LiveKit	SFU. Forwards Opus untouched; recording = subscribe + decode + mix in egress.
Jitsi	SFU. Dominant-speaker + Last-N forwards only the most relevant streams.
Opus DTX	Drops a quiet mic from ~24-32 kbit/s to ~1 kbit/s; can confuse level readers.

Remember

- Two people, no recording need: use P2P - direct path, lowest latency, zero server cost.
- Above four people, or once you add video: put a server in the middle, almost always an SFU.
- SFU = forward, never decode. MCU = decode, mix into one N-1 stream, re-encode. That sets the cost.
- An MCU sends each person everyone's voice EXCEPT their own (N-1 mix) to stop feedback.
- SFU recording is NOT free: no mix exists, so budget a separate egress service that mixes.
- Use an MCU when a receiver can only play one stream (PSTN dial-in, set-top, gateway).
- Active-speaker UI flickers when people go quiet? Suspect Opus DTX gaps, not the algorithm.
- Hybrids win: start P2P for two, promote to SFU on the 3rd join; add a small MCU for PSTN legs.