Dialogic

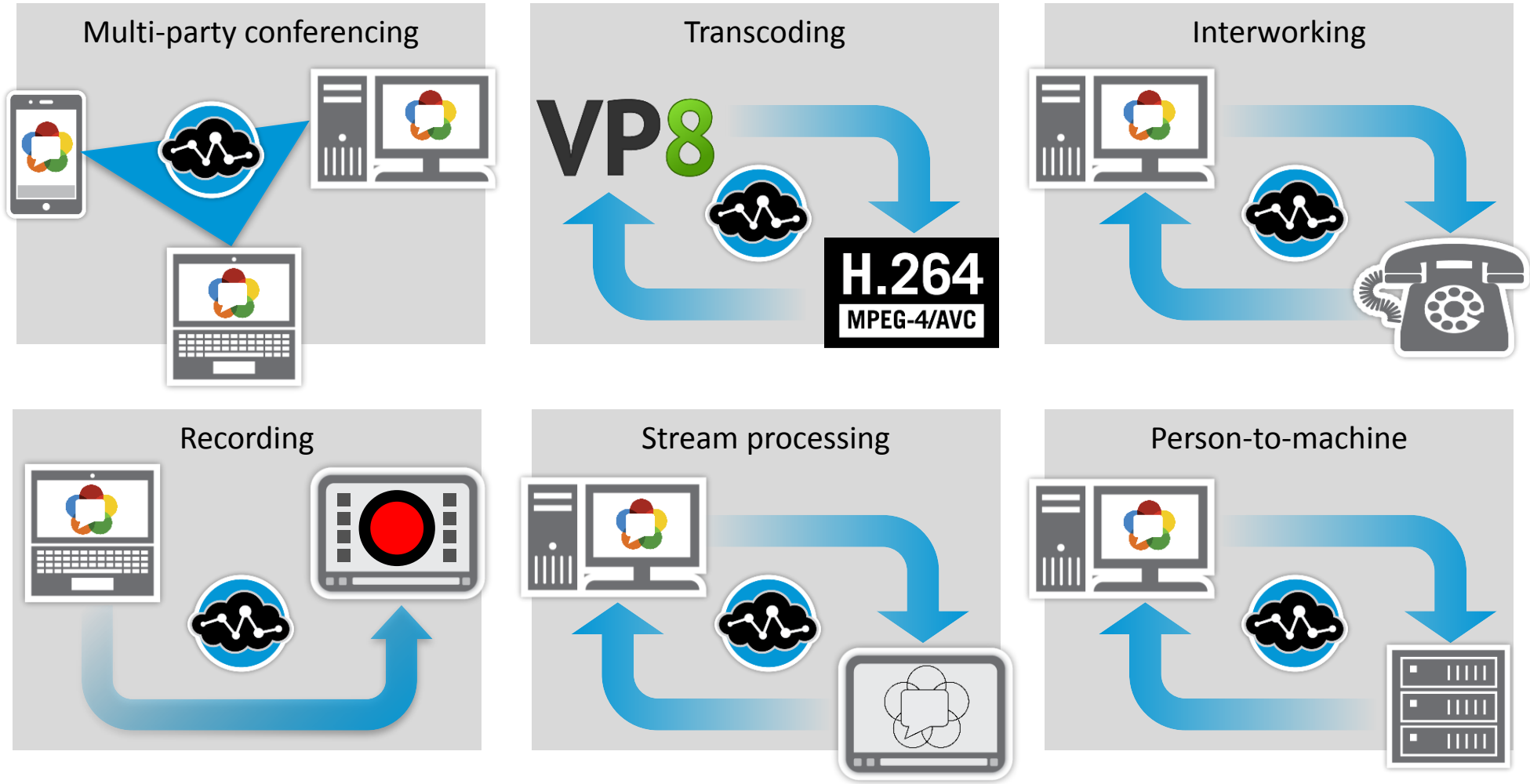
WebRTC Media in the Cloud

Chad Hart

IIT RTC Conference 2014

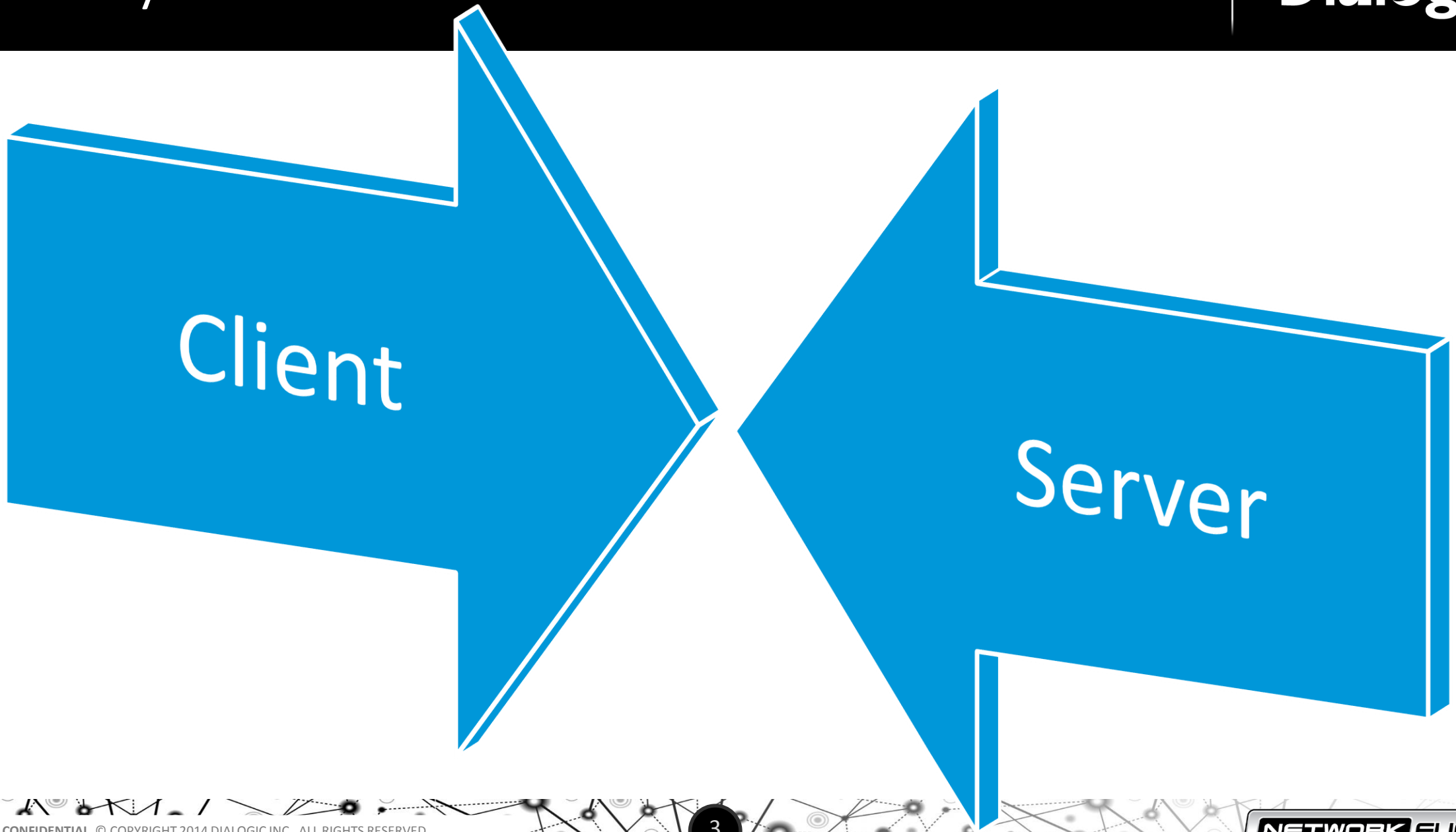
Why do you need a media server?

Dialogic



NETWORK FUEL®

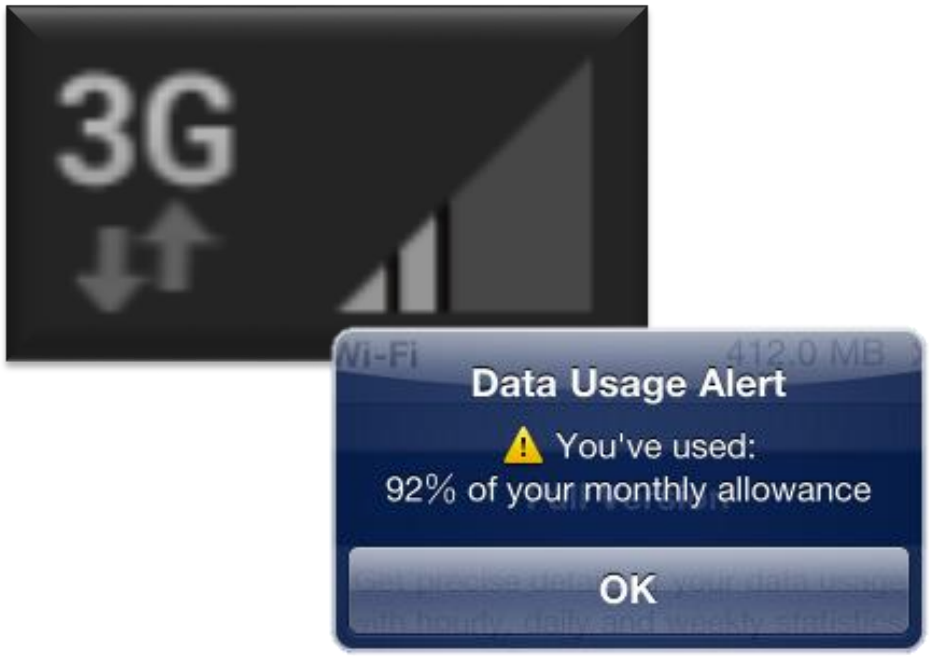
What do you care more about?



Bandwidth is not always ubiquitous or free

Dialogic

Client-side



Server-side

Network

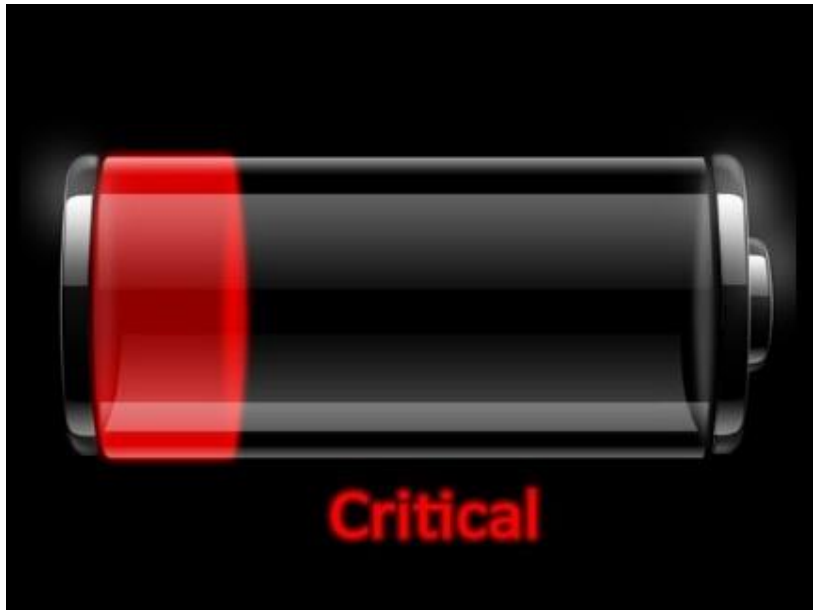
- 5TB/month Outbound Public Bandwidth (with monthly billing)
- Unlimited Inbound Public Bandwidth
- Unlimited Private Network (Server-to-Server) Bandwidth
- 100Mbps Public and Private Network Ports (upgradable to 1Gb
- Private VLAN via Public & Private Network
- Geographically Redundant DNS
- Dual-Stack IPv4 and IPv6 Capable

NETWORK FUEL®

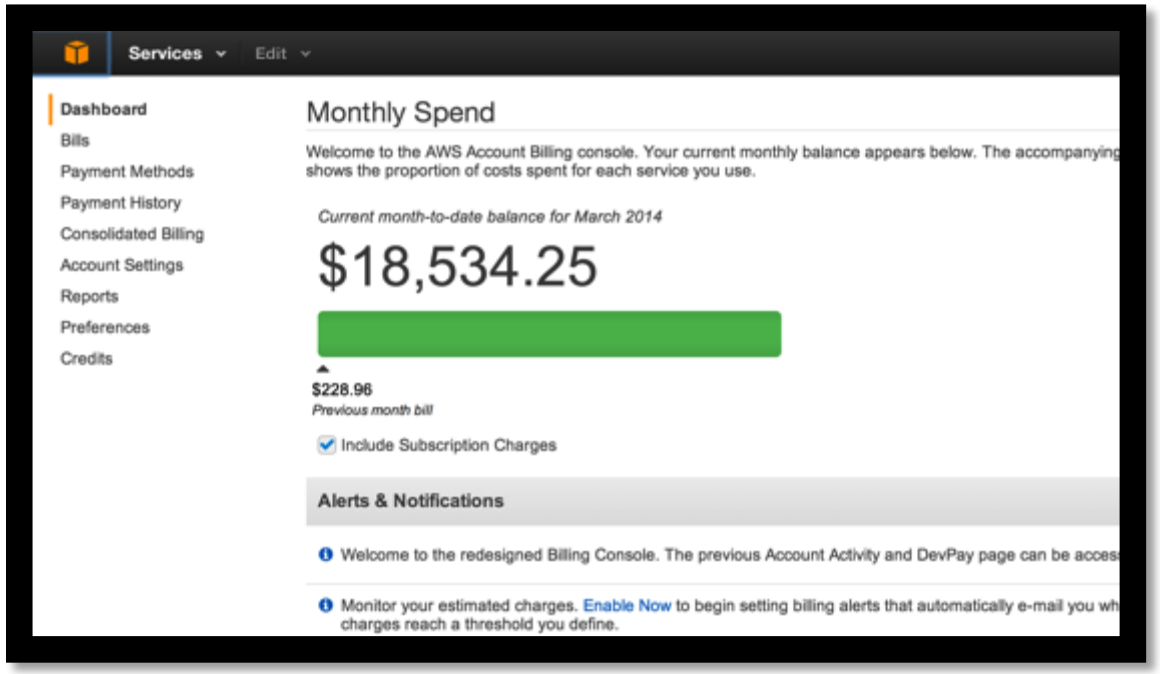
CPU & processing power is expensive

Dialogic

Client view



Provider View



NETWORK FUEL®

Image source: http://thewrestlinglifeupdate.tumblr.com/

Multi-PARTY! Video Conferencing

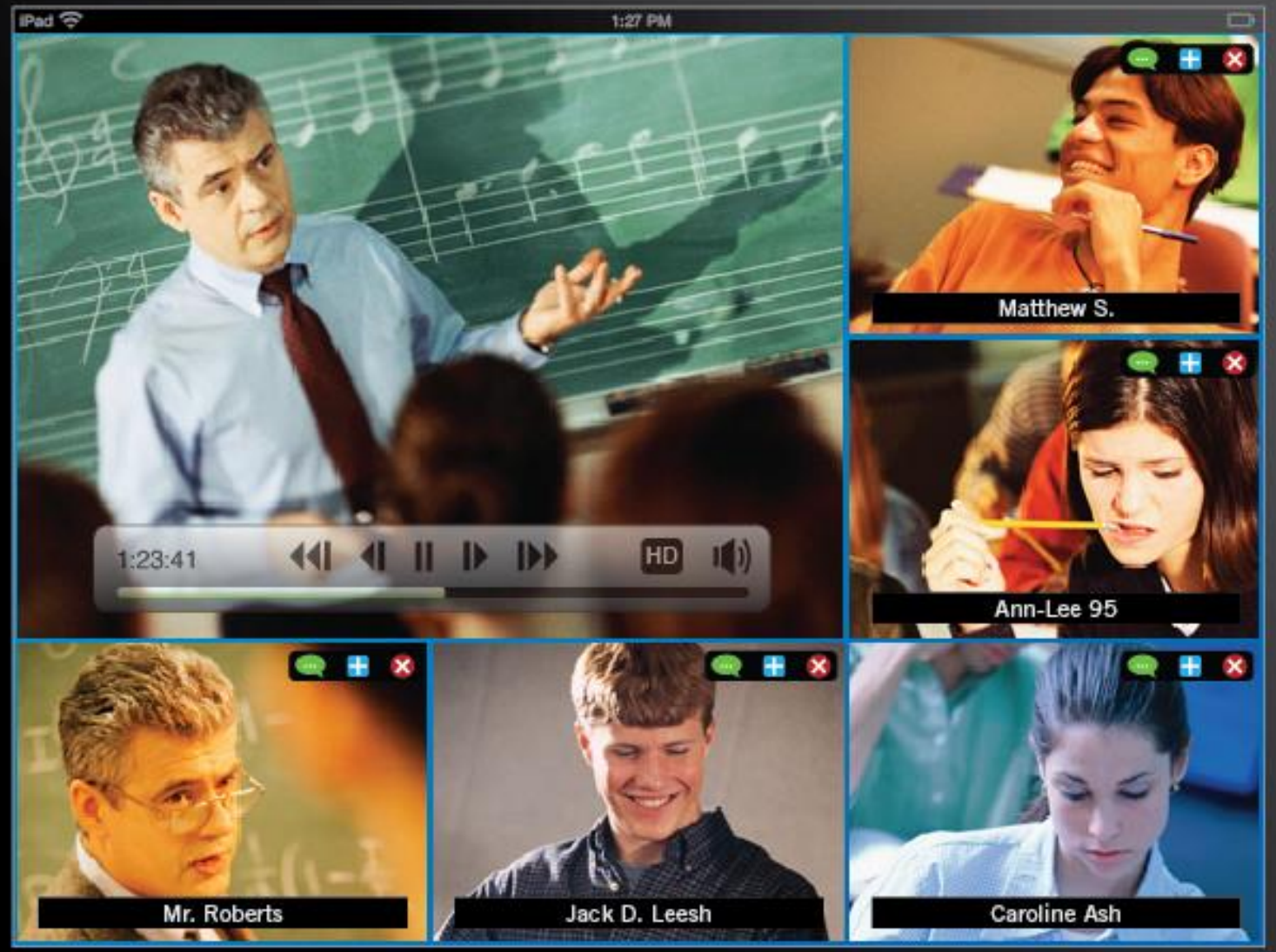


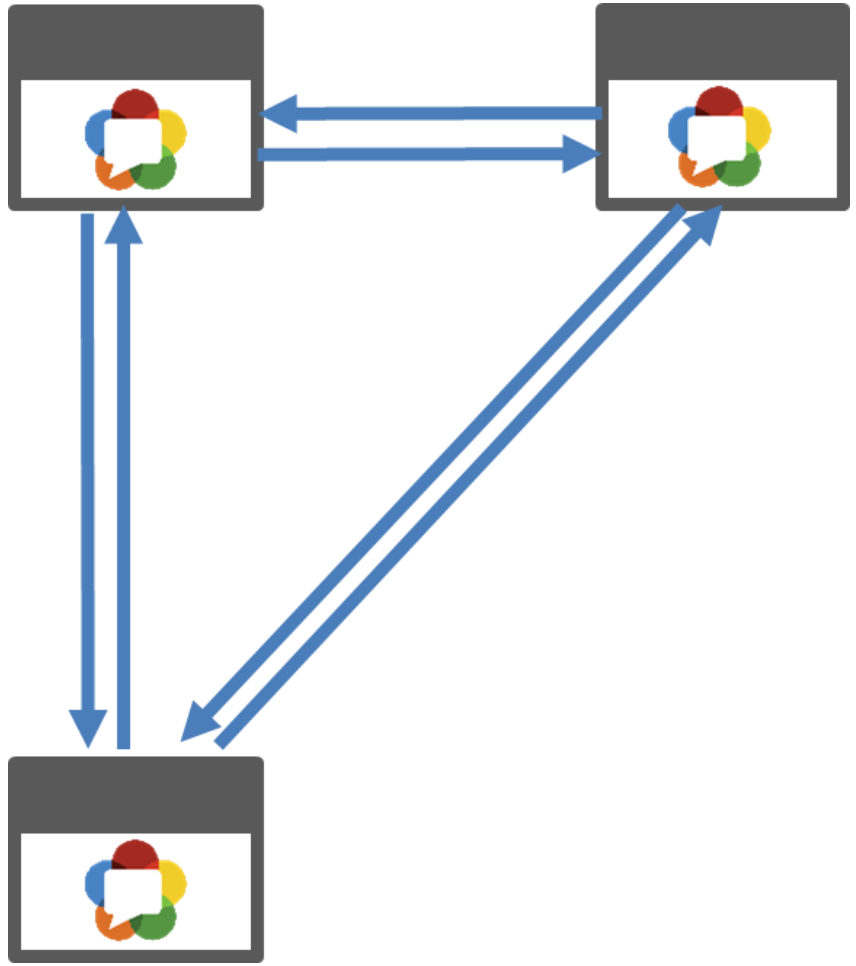
Image source: http://thewrestlinglifeupdate.tumblr.com/

Easy & cheap approach to multi-party: Mesh

Dialogic

Full Mesh

- Works for a few parties
- No server cost
- Lowest latency



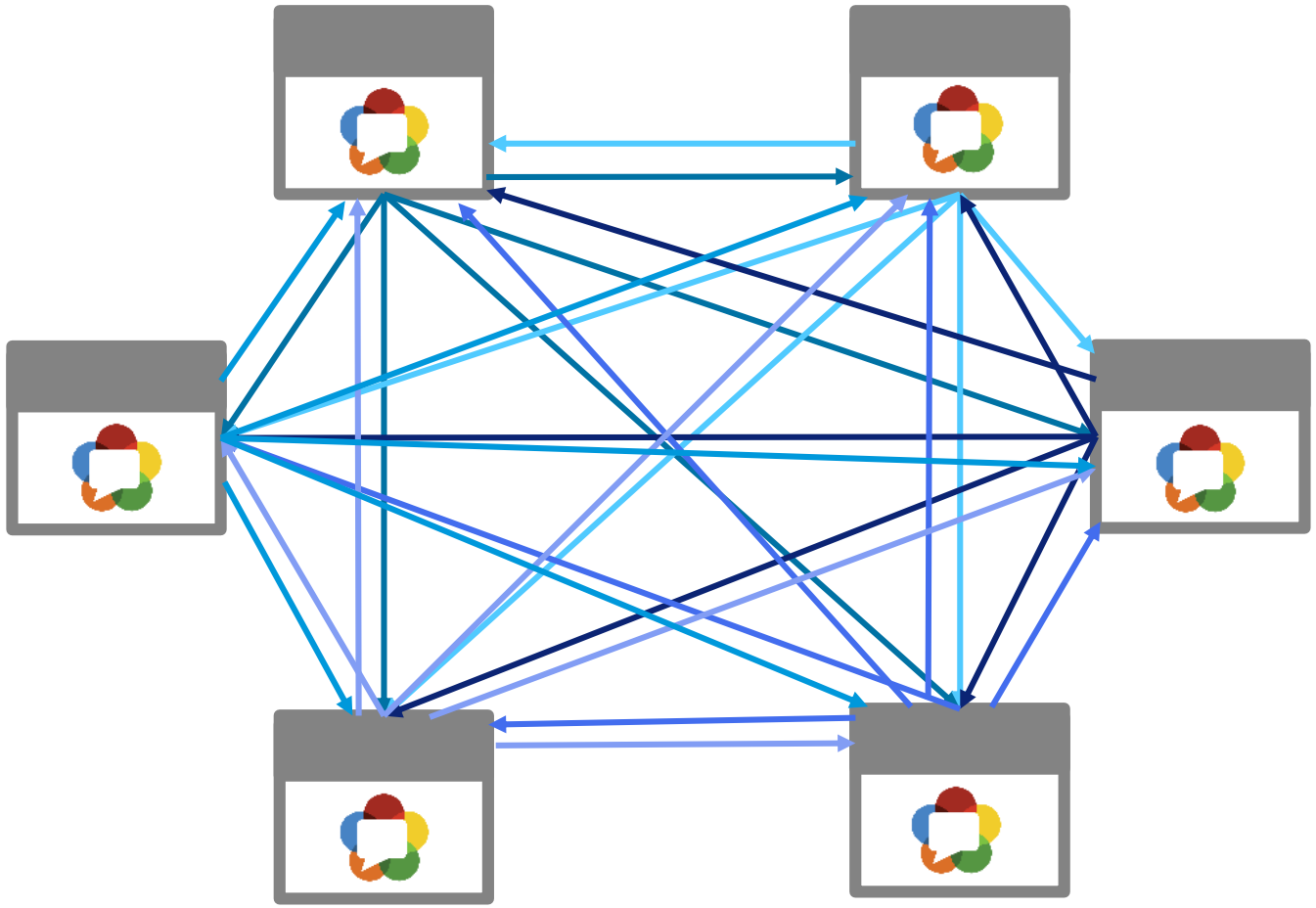
8

Mesh does not scale for many video conferees

Dialogic

Full Mesh

- Clients get overloaded
  - Encode costs more than decode
  - Limited uplink bandwidth
- Inconsistent performance across participants



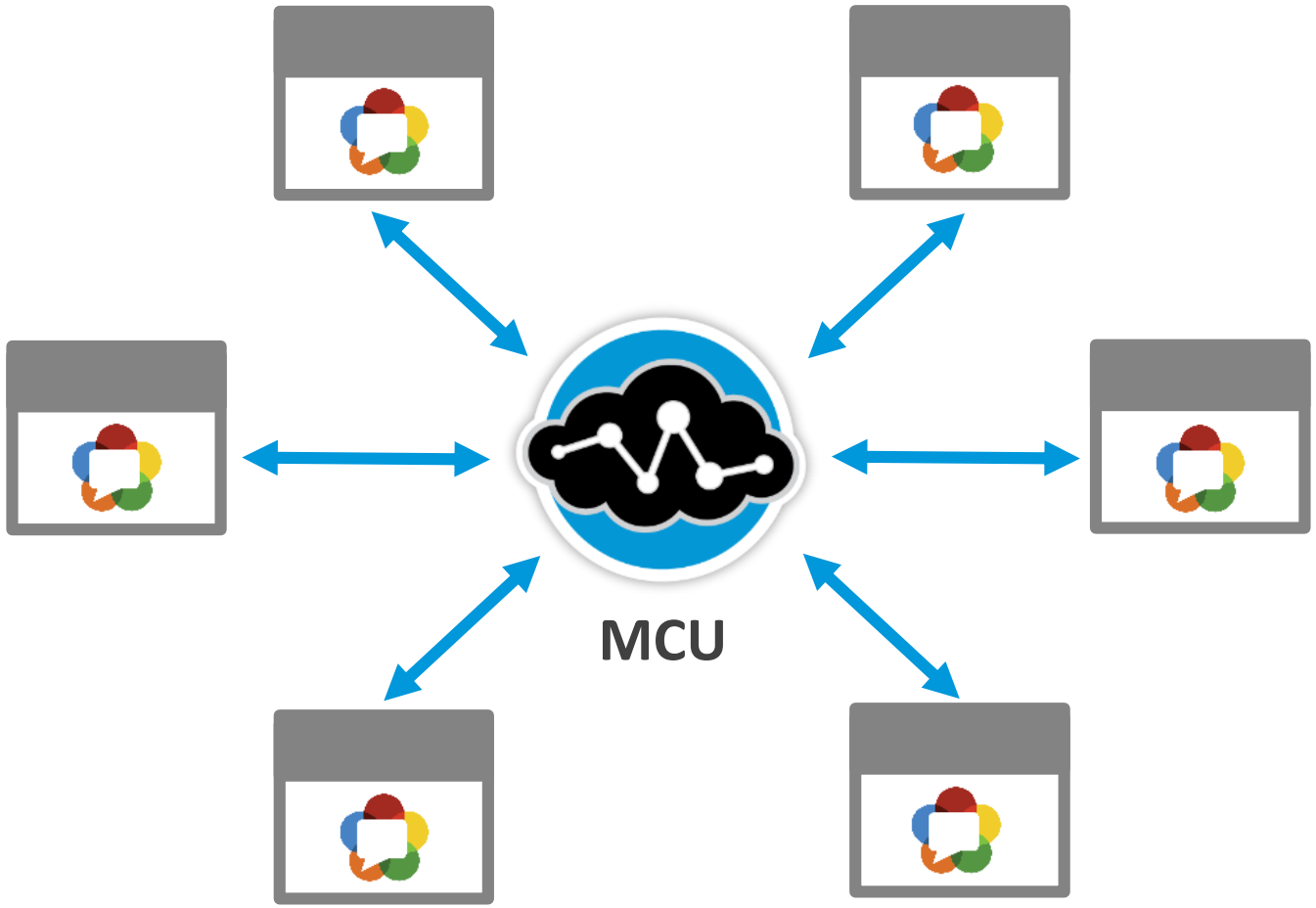
9

Traditional Telephony Approach: MCU

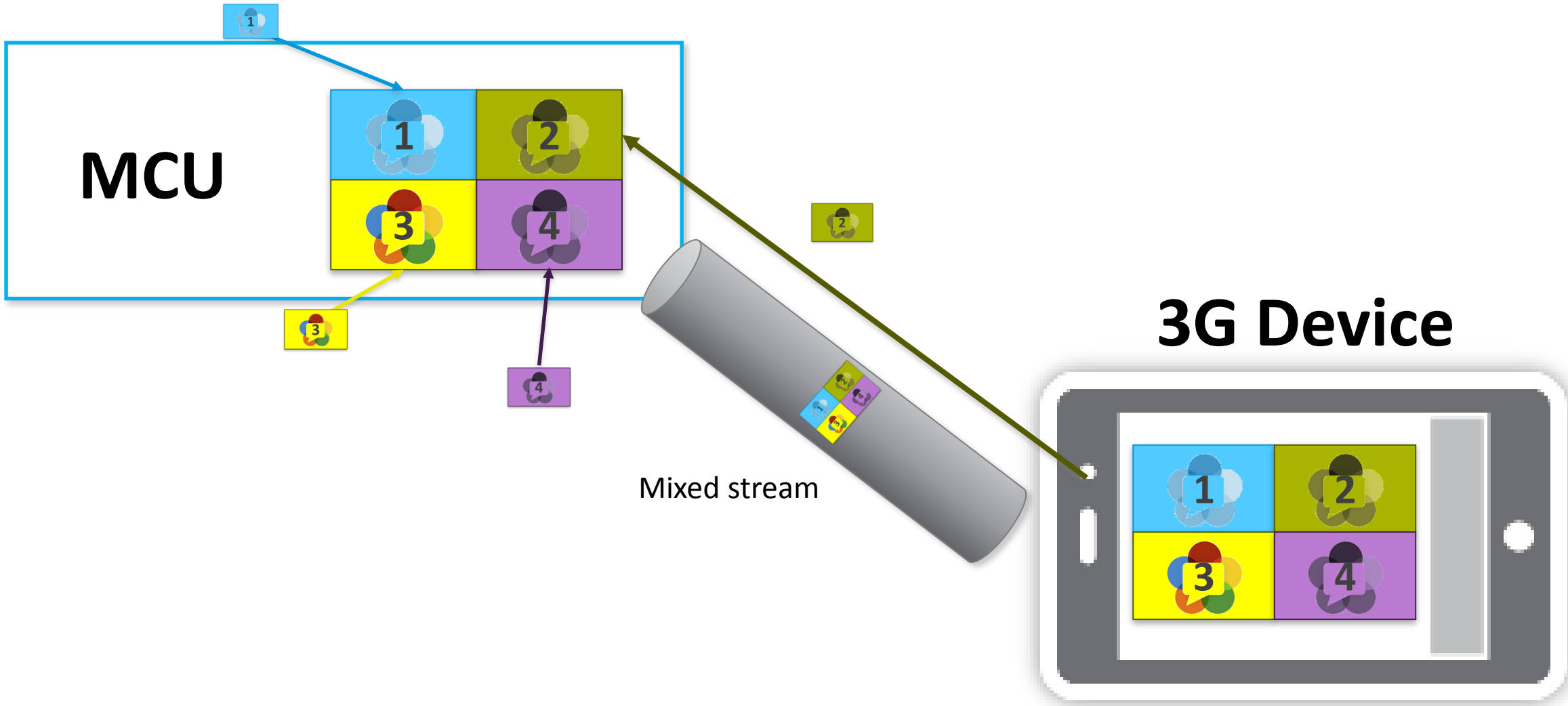
Dialogic

Multipoint Control Unit (MCU)

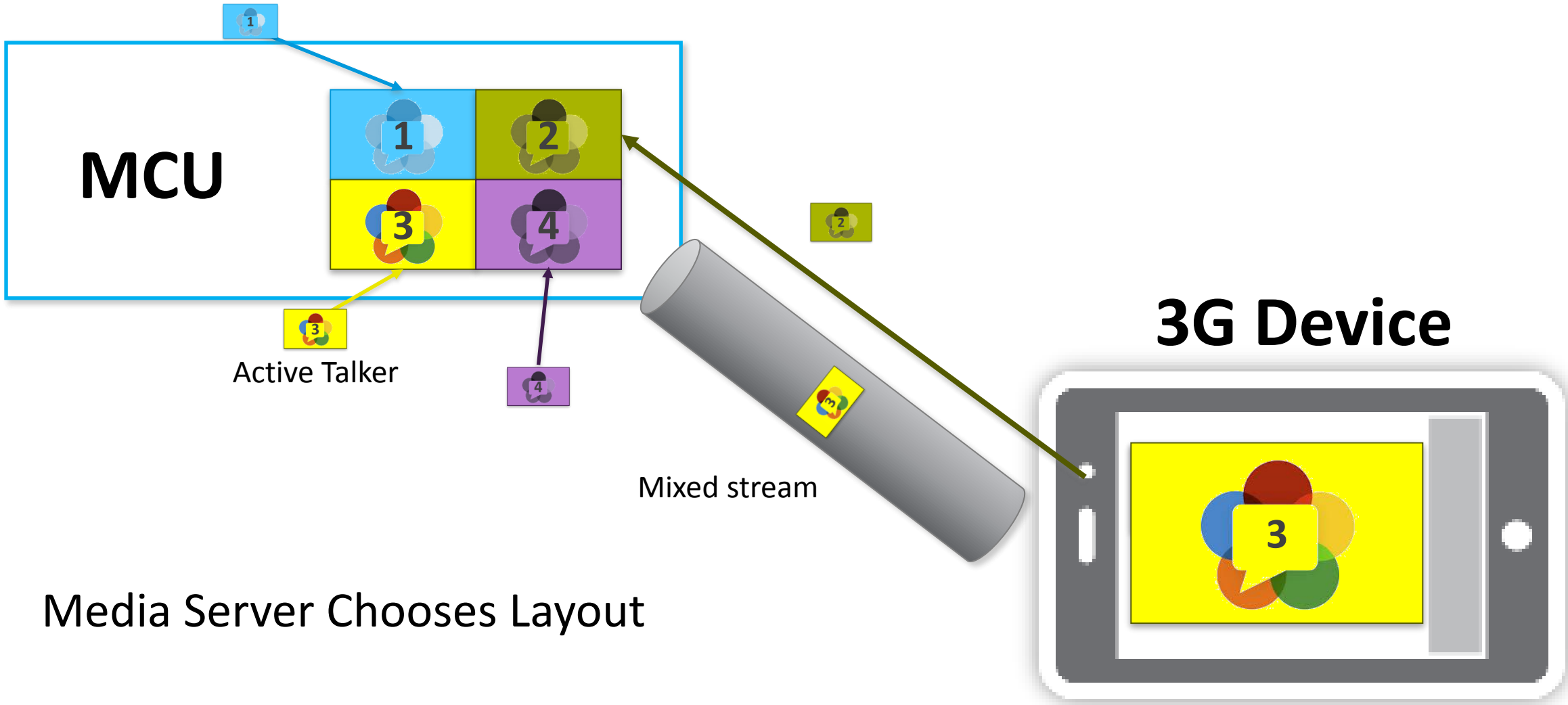
- Good at manipulating all media
  - Transcoding
  - Transizing
  - Transrating
  - Interworking
- Server-side CPU intensive
- Client can request different conference mixes
- Usually client friendly
  - Downsizing & mixing can reduce bandwidth
  - Fewer client streams to process



Video UX: Traditional method – mixed screen



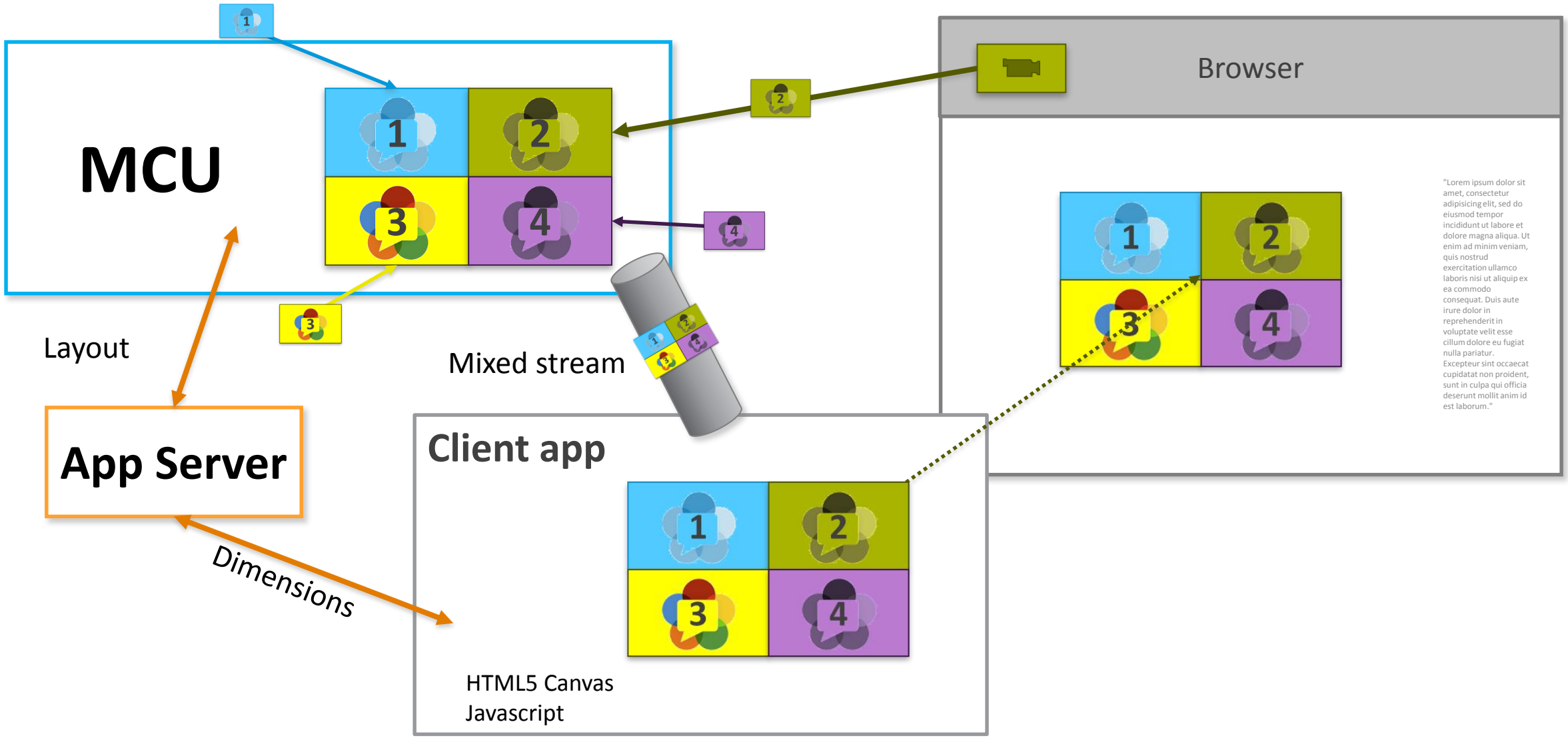
Video UX: Traditional method – active talker



Media Server Chooses Layout

Modern UX with MCU mixing, HTML5 & JavaScript

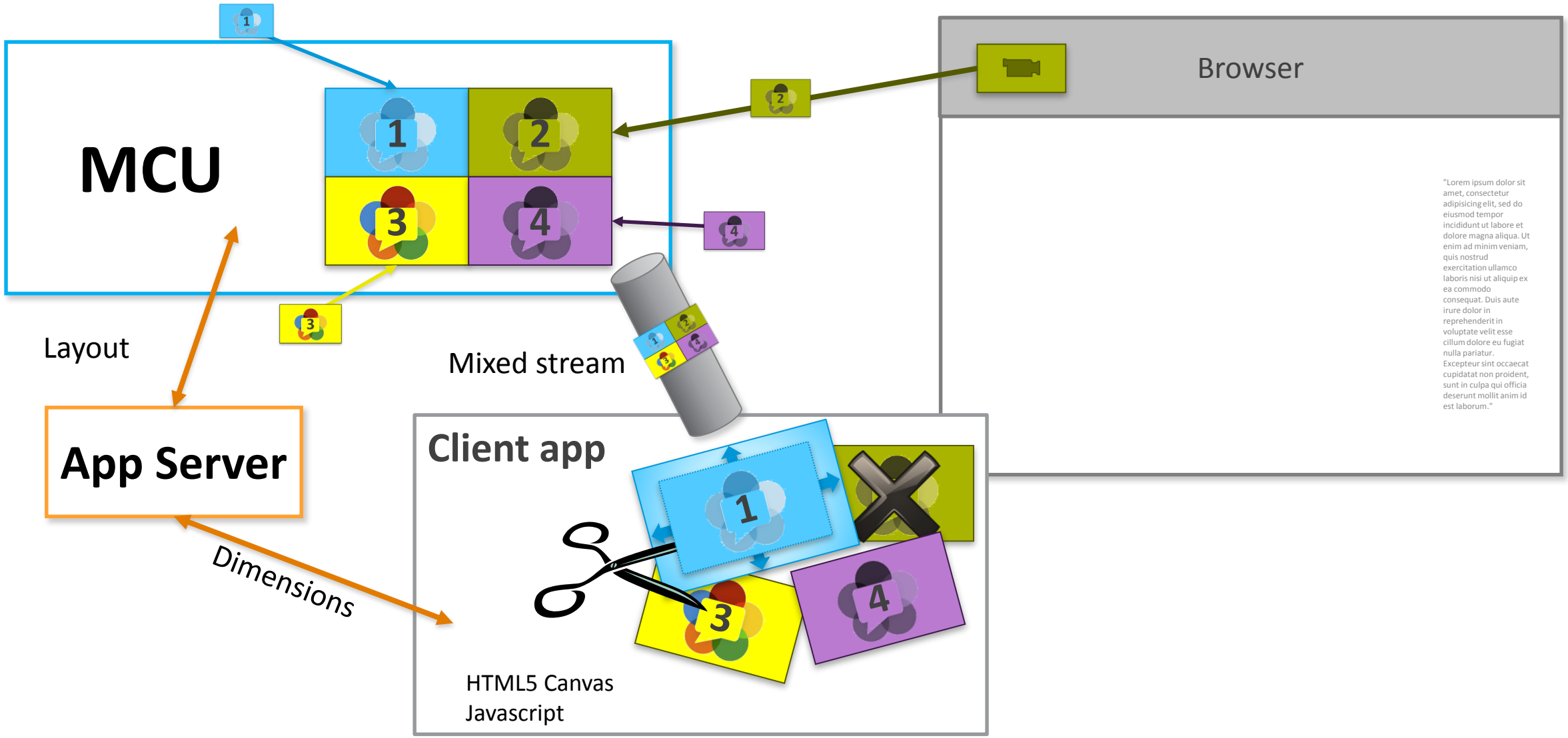
Dialogic



NETWORK FUEL®

Modern UX with MCU mixing, HTML5 & JavaScript

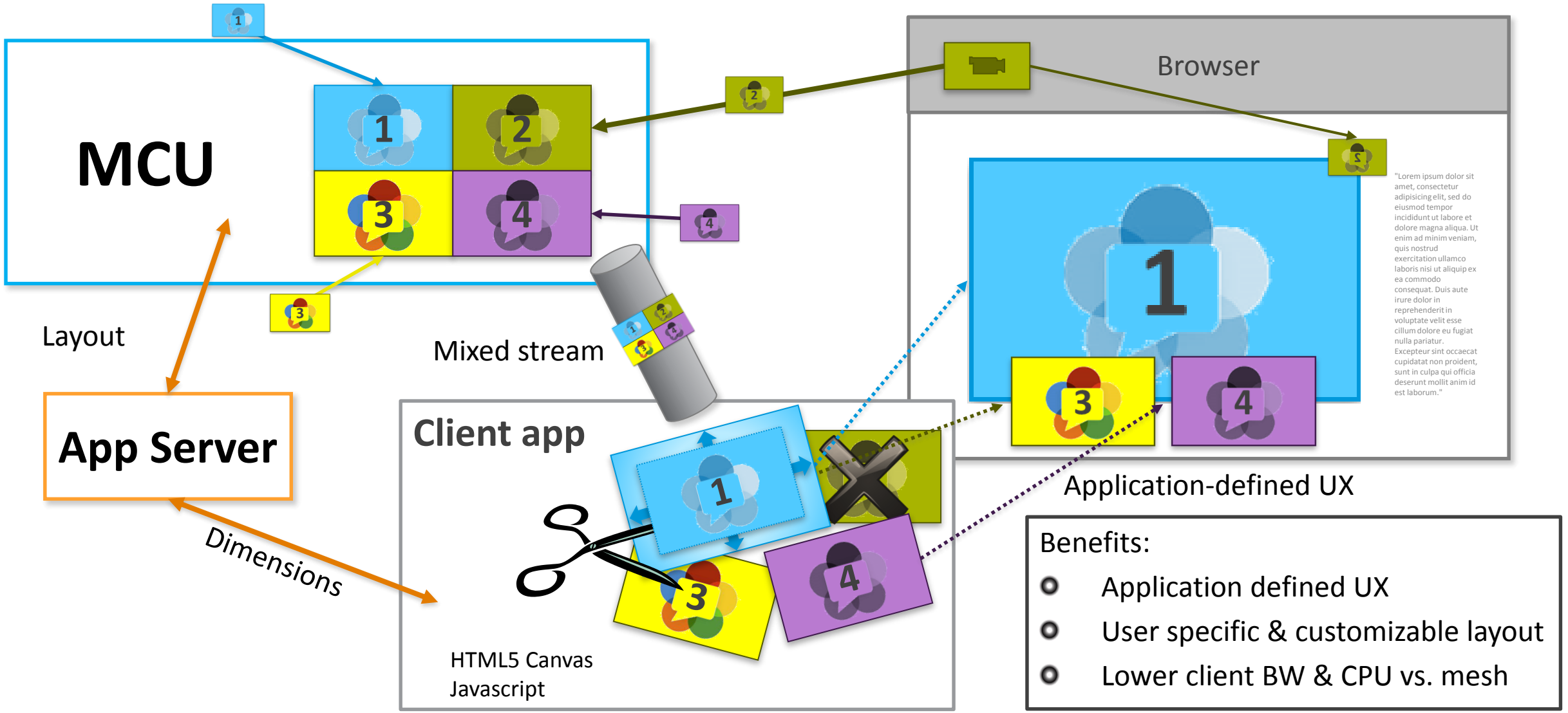
Dialogic



NETWORK FUEL®

Modern UX with MCU mixing, HTML5 & JavaScript

Dialogic



Benefits:

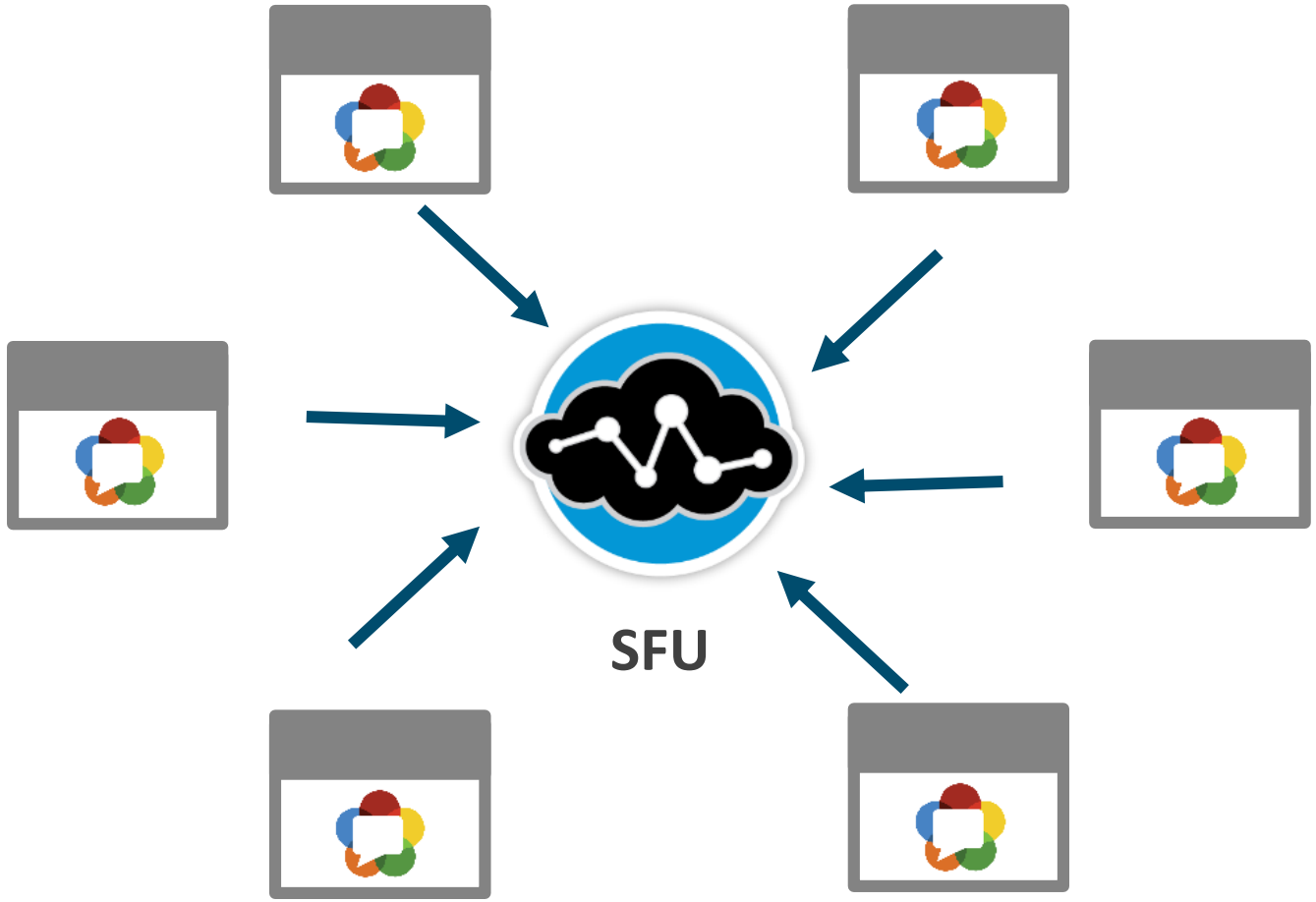
- Application defined UX
- User specific & customizable layout
- Lower client BW & CPU vs. mesh

Newer approach: SFU

Dialogic

Selective Forwarding Unit (SFU) routing

- Clients send one & receive many
- Client can instruct SFU which streams to send
- High throughput
- Can be lots of downlink bandwidth
- Low latency



16

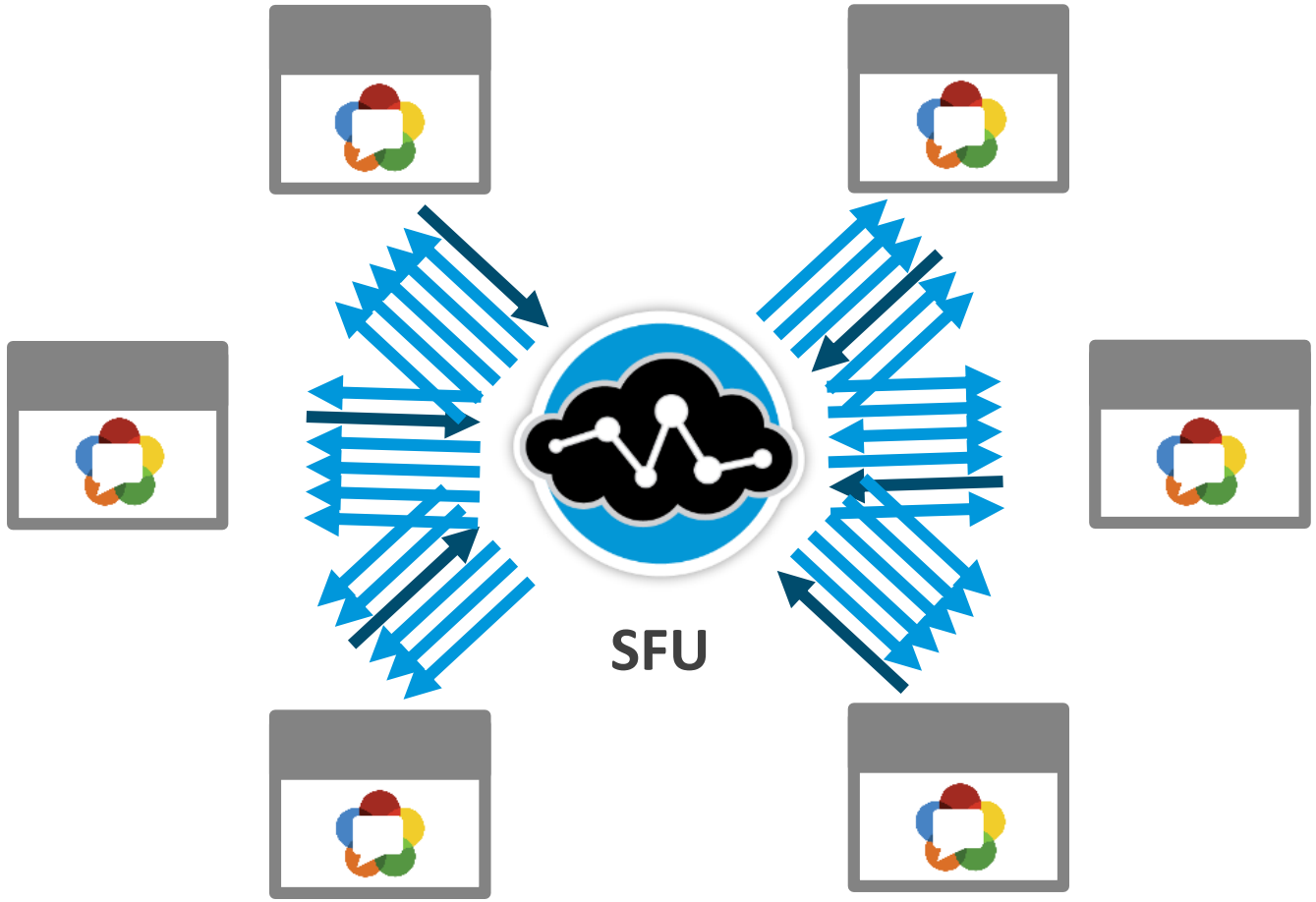
NETWORK FUEL®

Newer approach: SFU

Dialogic

Selective Forwarding Unit (SFU) routing

- Clients send one & receive many
- Client can instruct SFU which streams to send
- High throughput
- Can be lots of downlink bandwidth
- Low latency



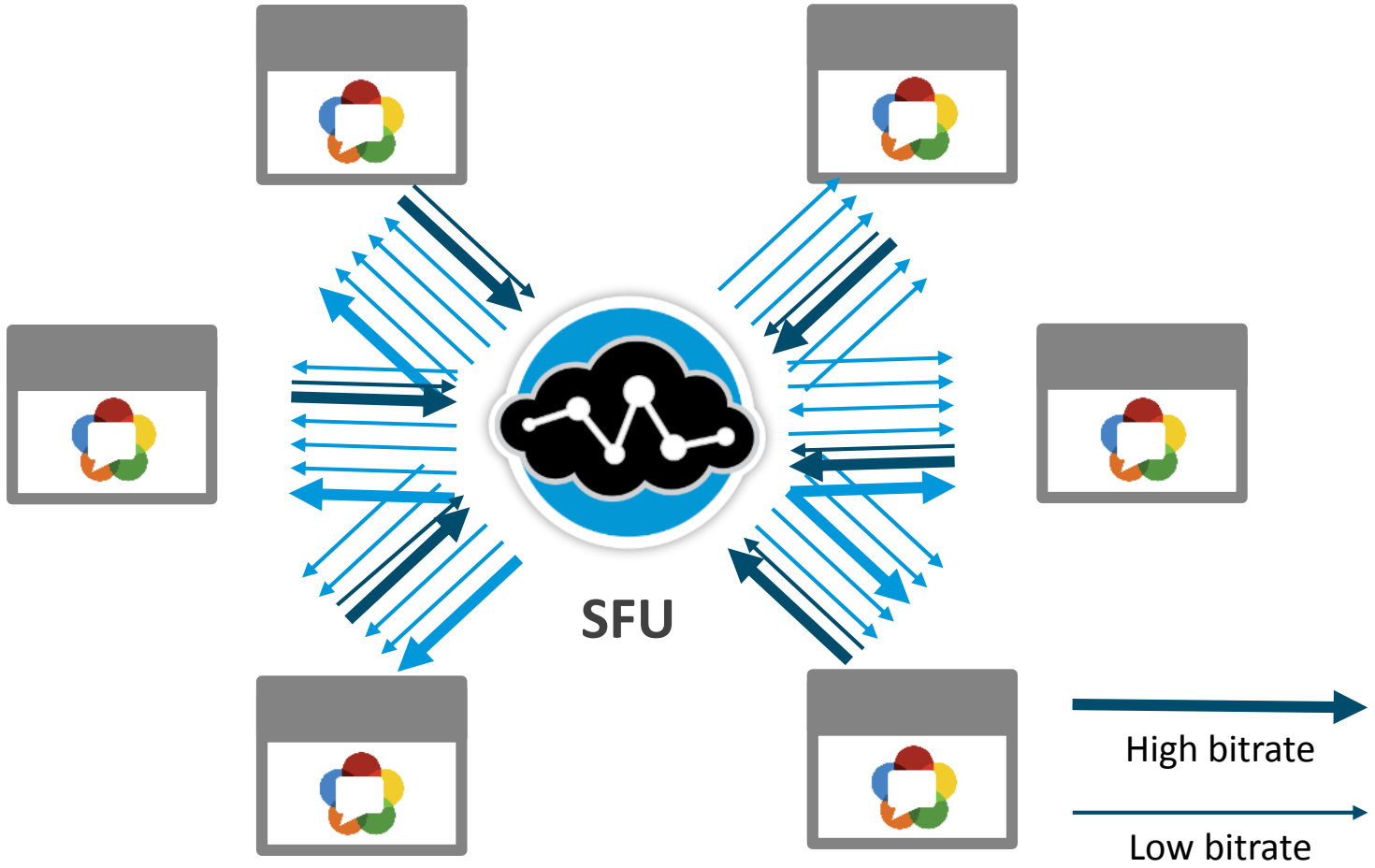
17

New WebRTC approach: Simulcast

Dialogic

Selective Forwarding Unit (SFU) with Simulcast

- Clients send multiple streams to SFU
  - one high-bit rate
  - one or more lower-bit
- Client directs SFU which streams to receive
- Reduces bandwidth vs. SFU
- Simulcast in WebRTC coming



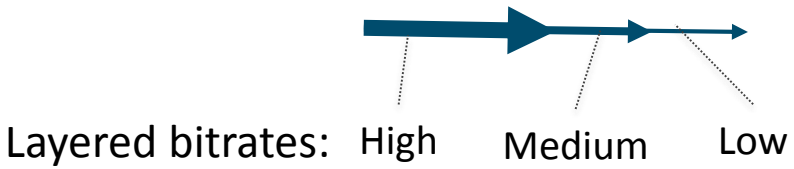
18

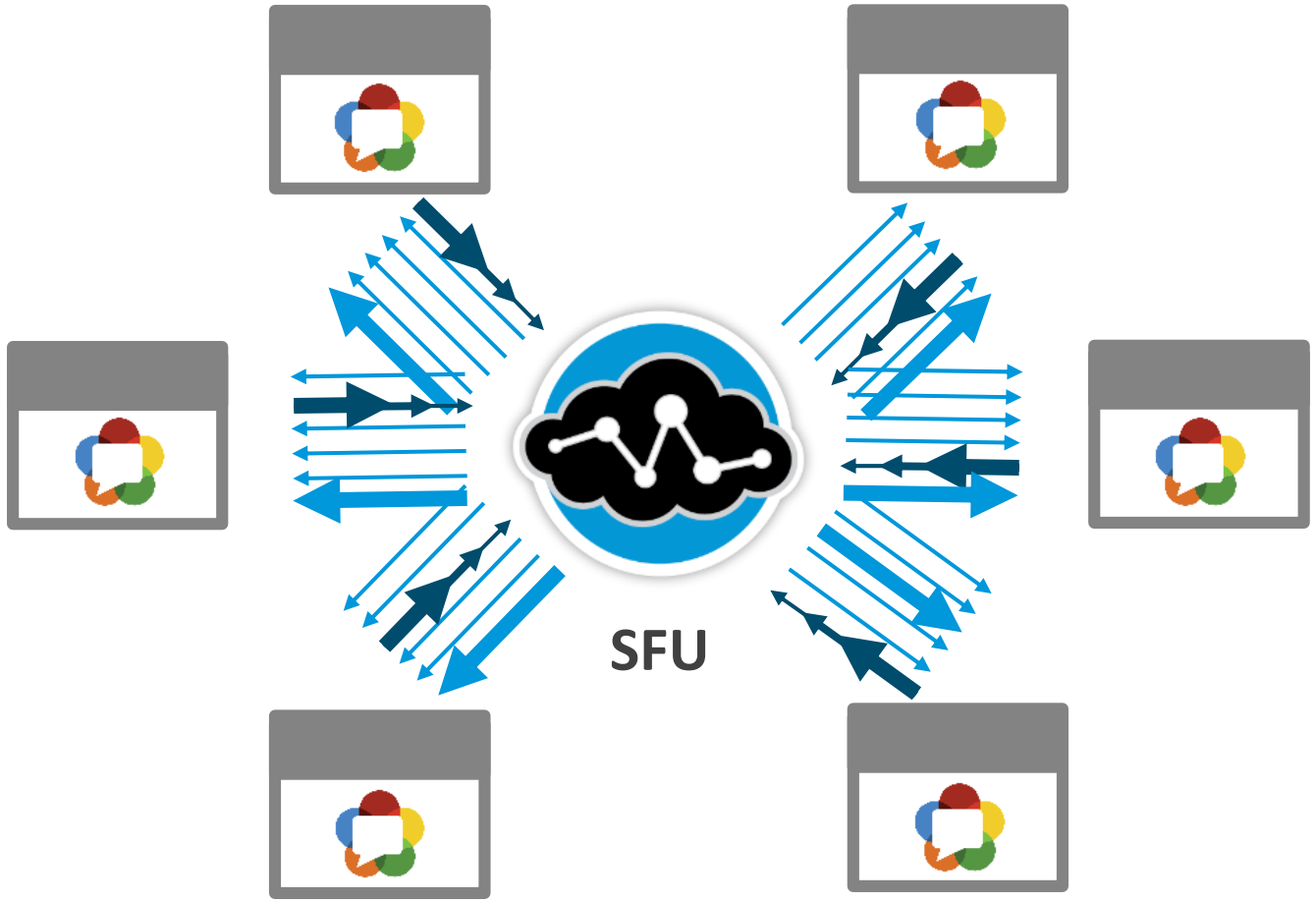
NETWORK FUEL®

Dialogic

Selective Forwarding Unit (SFU) with SVC

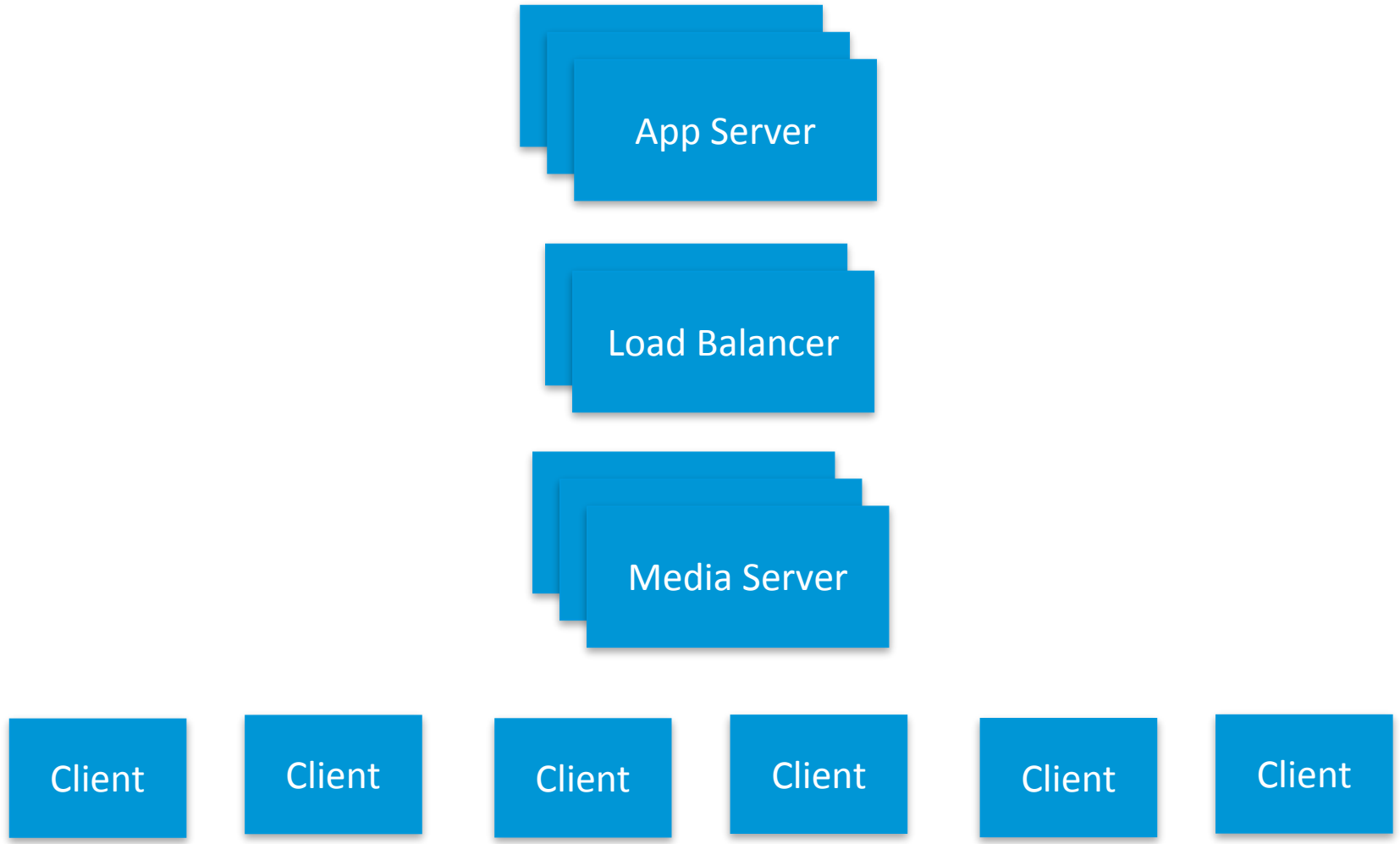
- Clients send layered stream to SFU
  - Varying bitrates – time, size, quality
- SFU directs who gets what
- Coming to WebRTC eventually



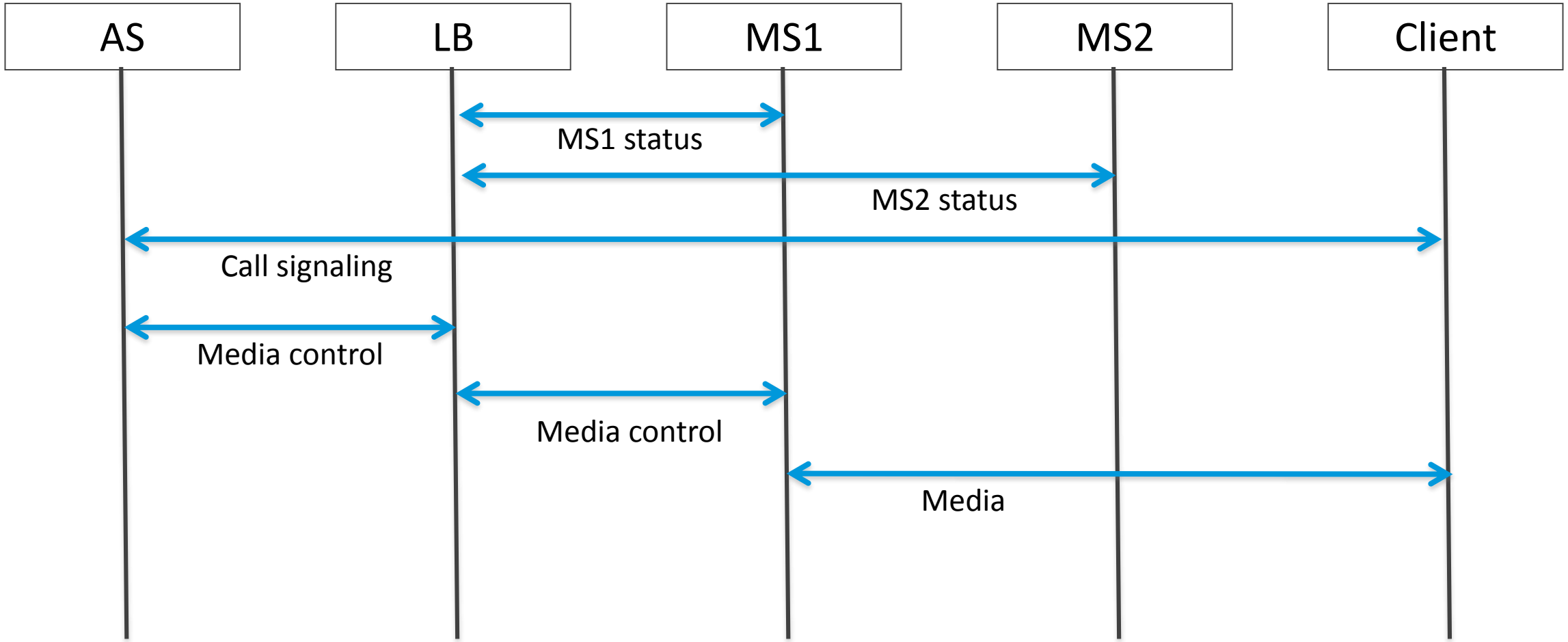


19

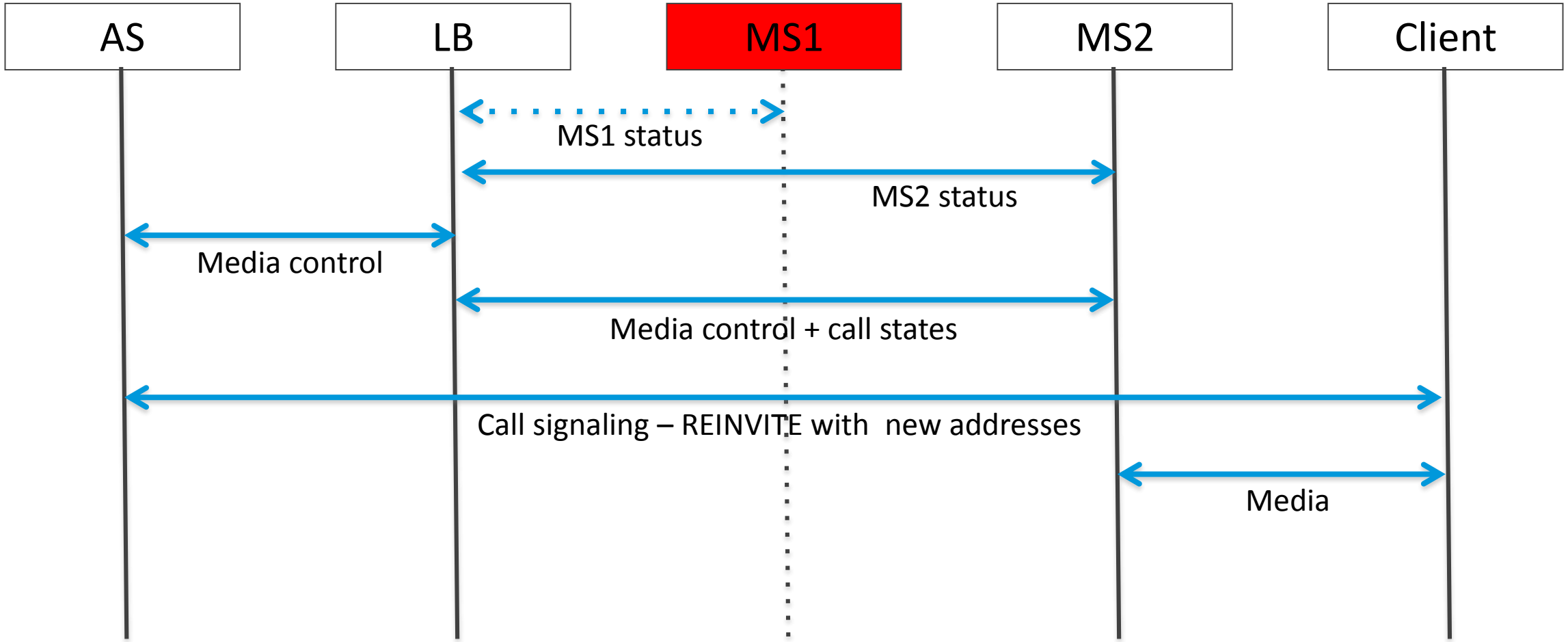
Typical media server network model



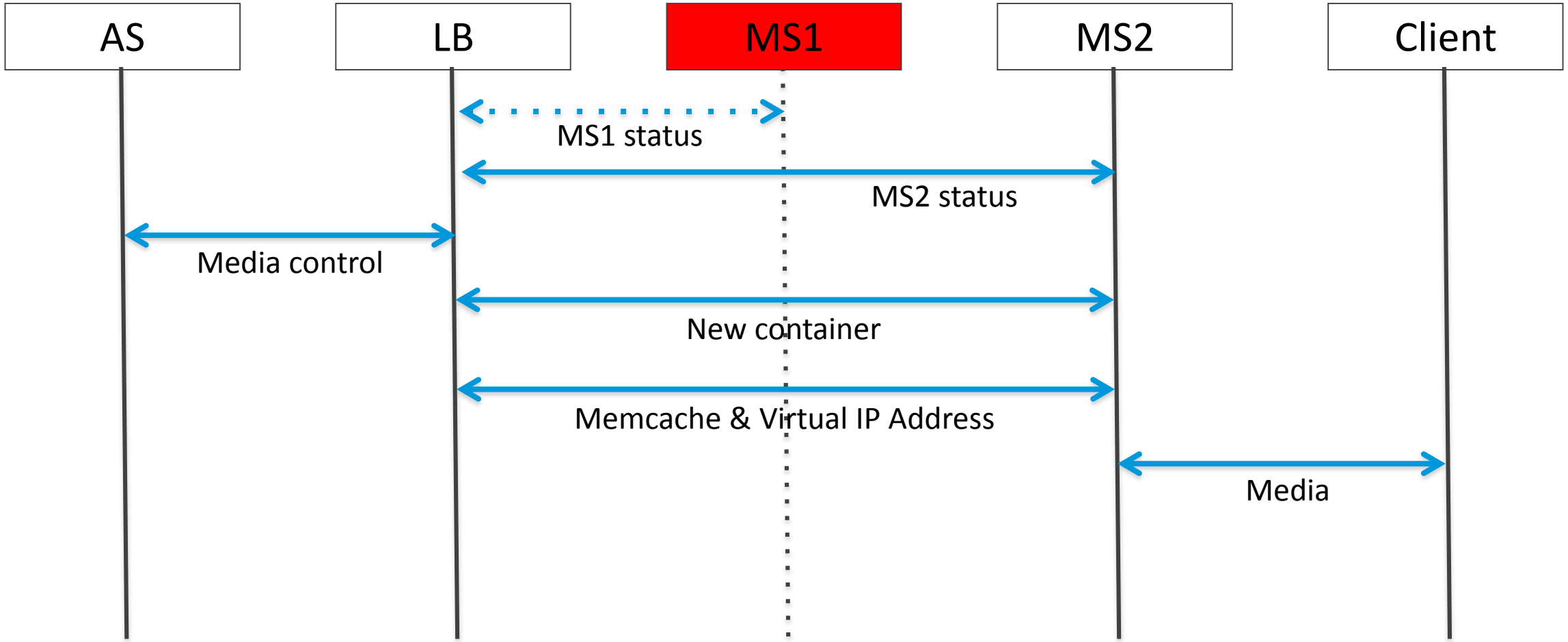
Pseudo Call-flow with a media server



Pseudo Call-flow – traditional fail-over with RE-INVITE



Pseudo Call-flow – fail-over the cloud way



When you are serious about scale

Dialogic

- Declarative
- Concurrency
- Soft real-time
- Robustness
- Distribution
- Hot code loading
- External interfaces
- Portability

ICE, STUN, and TURN save the day

Image source: webrtcHacks- http://webrtchacks.com/an-intro-to-webrtcs-natfirewall-problem/

- Use of Public IP's mitigates need for TURN servers
- Open source TURN servers do reasonably well
- Running the TURN server co-resident with the media server works well when you need it