Migrating From Twilio Video

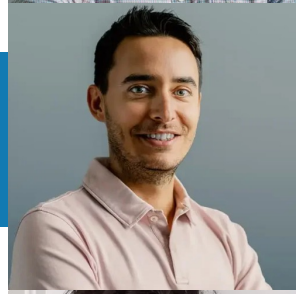
Jan 12, 2024

DISCLAIMER: Content in this presentation has been assembled by WebRTC.ventures from a wide range of sources and vendors, but should not be construed as specific advice for your situation. The whole point of this presentation is to cover a wide range of options and considerations, and that you need to assess and prototype those options to see what works best for you!

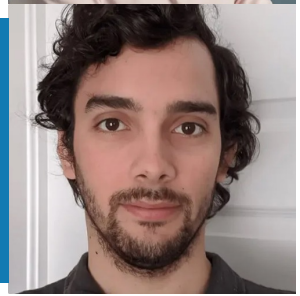
Migrating From Twilio Video

Arin Sime, CEO

Alberto Gonzalez, CTO



Justin Williams, Senior WebRTC Engineer



Agenda (with Q&A throughout)

- Our background
- Twilio Video EOL announcement
- Twilio's Zoom recommendation
- WebRTC Architectures
- Open Source options
- CPaaS options (Amazon Chime SDK, Daily, Vonage)
- Abstracting the media server layer (optional)
- Factors to base your choice on
- Your 2024 Twilio Video Migration Plan
- How WebRTC.ventures can help

Notes from today will be available at:
https://webrtc.ventures/twilio-video/

Our Team of Experts

- Over 40 team members dedicated to Video work
- Offices in Charlottesville, Virginia and Panama City, Panamá
- Working virtually around North and South America for clients globally
- Global team that allows for 24/7 MSP and support
- Founded in Virginia in 2010
- Working in Video since 2015
- Mix of Web/Mobile devs, DevOps, UX/UI design, QA, & project leads

Example clients we have worked with

Our team of developers, designers, testers, DevOps and project leads work with ENTREPRENEURS and funded startups who need development expertise to get their idea to market. We also work with MID-MARKET and ENTERPRISES to integrate, fine-tune, or scale their communications solution.

Twilio Video End-of-Life in December 2024

Almost buried in a December layoff announcement to Twilio employees was this news:

> *"Lastly, we've decided to end-of-life (EOL) Twilio Programmable Video as a standalone product. Given it's such a niche area and a relatively small part of our portfolio, we believe partnering with video industry leaders is the best way to ensure long-term product innovation for our customers. Removing Programmable Video from our portfolio will also allow Communications to more effectively focus on our pillar products – Messaging, Voice, and Email.*
>
> *Those Twilions impacted by our Video EOL are aware of this decision, and we'll continue working closely with them on transition plans."*

Dec 5, 2024 - Official end date for Twilio Programmable Video
Not that far away when you consider development cycles and smooth migration planning are necessary!

Arin Sime crossing sabers with Twilio CEO Jeff Lawson at a Twilio SIGNAL conference

Twilio's recommendation: Zoom

- Zoom is fine for generic meeting use cases, and high quality for desktop use cases, but does not rely on the WebRTC standard for web and so is not a popular choice with WebRTC traditionalists
- Using WebAssembly allows Zoom to utilize their own custom codecs, which they feel are superior.
- Zoom has published this guide to their technical stack which explains more:

https://www.zoom.com/en/blog/how-zooms-video-sdk-stacks-up/

This choice has met with skepticism...

Tsahi Levent-Levi, industry analyst

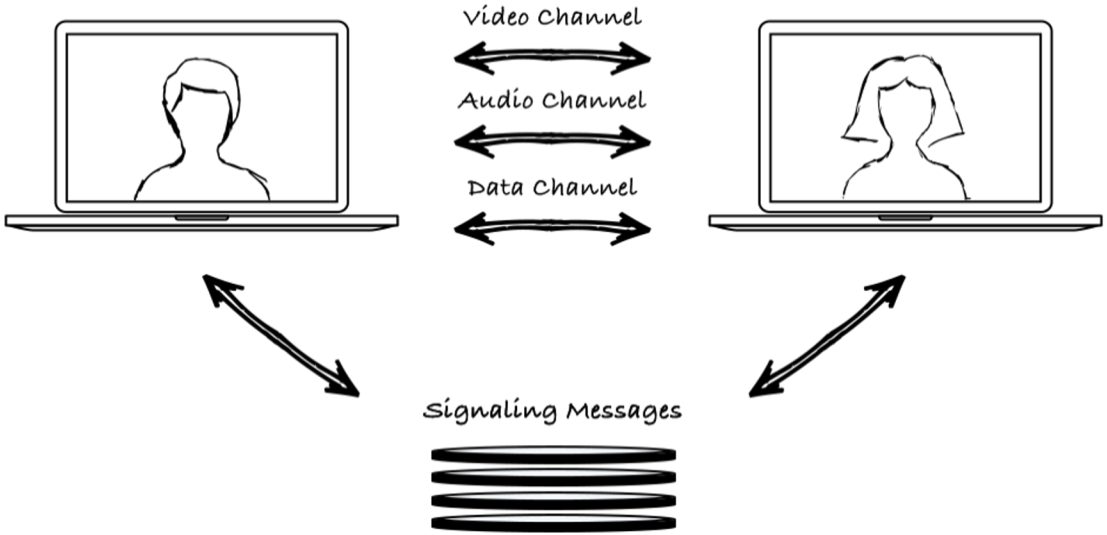
*"Why Zoom? **Zoom isn't a competitor of Twilio in anything** … Their web experience isn't on par with the rest of the pack."*

Kwindla Hultman Kramer, Daily.co CEO

*"More recently, Zoom launched developer SDKs. These developer SDKs are less mature than the Zoom end-user products. In particular, **the Zoom Web SDK has important feature gaps and major performance issues that developers should be aware of before attempting to port web applications to Zoom**."*

https://www.daily.co/blog/zoom-web-sdk-technical-notes/

WebRTC architectures are not quite this simple...



- STUN/TURN servers
- Application Signaling
- Video codecs
- Group chat/scaling
- Browser/Mobile Support
- Recording

... and so switching architectures won't be either!

WebRTC Architectures

WebRTC architecture	WebRTC Standard	Unbundled WebRTC
Up front cost to build	High	High
Ongoing cost transactionally	Low	Low
Technical difficulty	High	Medium-High
Features included	Low	Low

OUR TAKE

Why build from scratch?

Also hard, and not as stable across browsers!

We recommend either of these!

Open Source Media Servers	CPaaS
Medium	Low
Low	High
Medium	Low
Medium	High

Provides low transaction costs but you must invest in infrastructure and DevOps

Best way to leverage the work of others in scaling and integrations and get to market quickly

Open Source Media Servers

Media Servers will handle:

- Video/audio stream details
- Part or all of the signaling
- Possibly STUN, TURN
- Some scaling capabilities
- Could be SFUs or MCUs
- Browser/Mobile support

But you host/manage:

- All infrastructure and updates
- The rest of the scaling
- Likely some optimization for your use case

Tsahi just posted a nice summary of the top 4 open source media servers:
https://bloggeek.me/webrtc-open-source-media-servers-github-2024/

Written in C, very stable project
And well supported by MeetEcho
janus.conf.meetecho.com

Written in Java, 8x8 owns it,
JaaS option
jitsi.org

Built with Node.js, Rust, C++
Purchased by Miro
mediasoup.org

Written in Go, very versatile, strong community, LiveKit is built on top of Pion
Pion.ly

CPaaS options

We will cover these CPaaS providers specifically…

Amazon Chime SDK

daily

VONAGE

But there are others too..

agora

digitalsamba

Dolby.io

dyte

LiveKit

LiveSwitch

… and more!

CPaaS options:

Amazon Chime SDK

- Uses WebRTC's open standards
- Global coverage: 21 AWS Regions, including GovCloud
- Simplification by staying in AWS cloud
  - AWS cross-product discounts* may apply
- Single security boundary across AWS
- AWS Compliance
  - (FedRAMP, HIPAA, PCI, SOC, …)
- Elastic scale with usage-based pricing
- Amazon Connect integrations for contact centers and Click-to-call

Twilio Video Specific Notes:

- Public price is 57.5% lower on Chime SDK vs Twilio ($0.0017 vs $0.0040)
- Twilio Video ran on AWS infrastructure

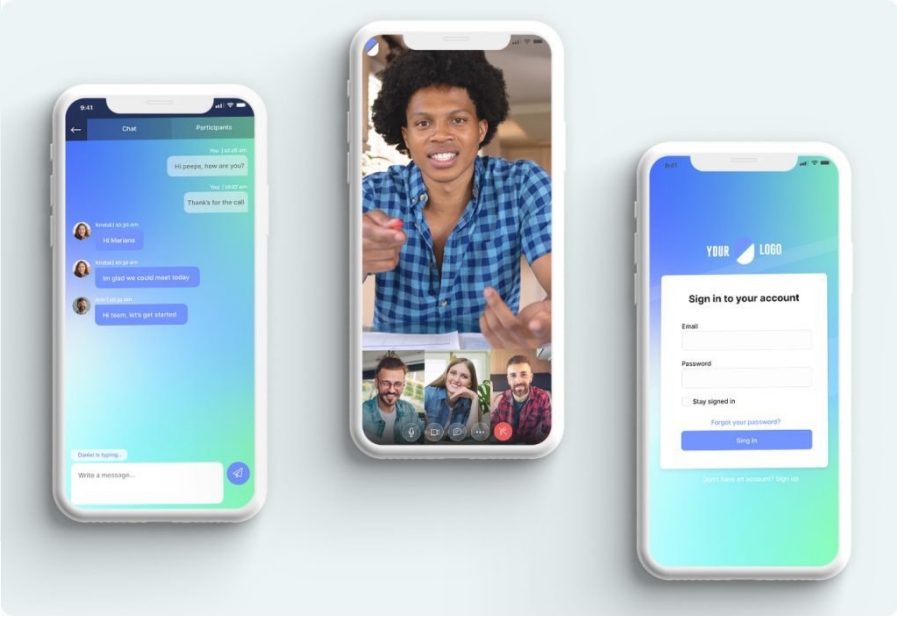
* Formerly known as Enterprise Discount Plan

https://webrtc.ventures/partners/amazon-chime-sdk/

Our Starter Kit using the Amazon Chime SDK

https://webrtc.ventures/partners/amazon-chime-sdk/

Amazon Chime SDK



CPaaS options: daily

https://webrtc.ventures/partners/daily/

- Performance equivalent to Zoom as benchmarked by 3rd parties
- HIPAA compliant - 25% of clients are in telehealth and AI powered SOAP Clinical Notes capabilities built in (uses a HIPAA compliant REST API call)
- End-to-End encryption available
- SIP Video and dial-in / dial-out abilities
- Option to still use Twilio STUN and TURN
- Support for large call use cases with 100k participants
- Daily Prebuilt is a reference app easily installed as a starting point or testing point, full APIs also available to build own interface
- AWS Marketplace availability for enterprises with AWS spend requirements

CPaaS options: daily

Daily has a comprehensive Twilio migration guide:

https://www.daily.co/competitive/twilio-alternative/

$30,000 migration credit!

Head-to-head comparison	daily	twilio	zoom
Maximum call size	100,000	50	1,000
Maximum video resolution	4k	1080p	720p
HIPAA	✓	✓	✓
Camera, microphone, and screen sharing	✓	✓	✓
Virtual backgrounds and background blur	**Fully supported**	**Fully supported**	**Partial** (not supported in Safari)

(table continues with more details on their migration page)

Flexible migration credit

Support your transition to Daily with our credit programs. Early stage teams can join our Startup Program. Enterprise customers can apply a $30,000 credit to Daily usage, or we'll contribute it to support development with one of our expert partners.

CPaaS Options: VONAGE

https://webrtc.ventures/partners/vonage/

Maturity

- Most well established CPaaS (we have worked w/ them since 2015 when it was Tokbox)
- Backed by Ericsson (low chance they go away)
- Enterprise Support packages, higher tiers included dedicated support engineers

Topology / Flexibility

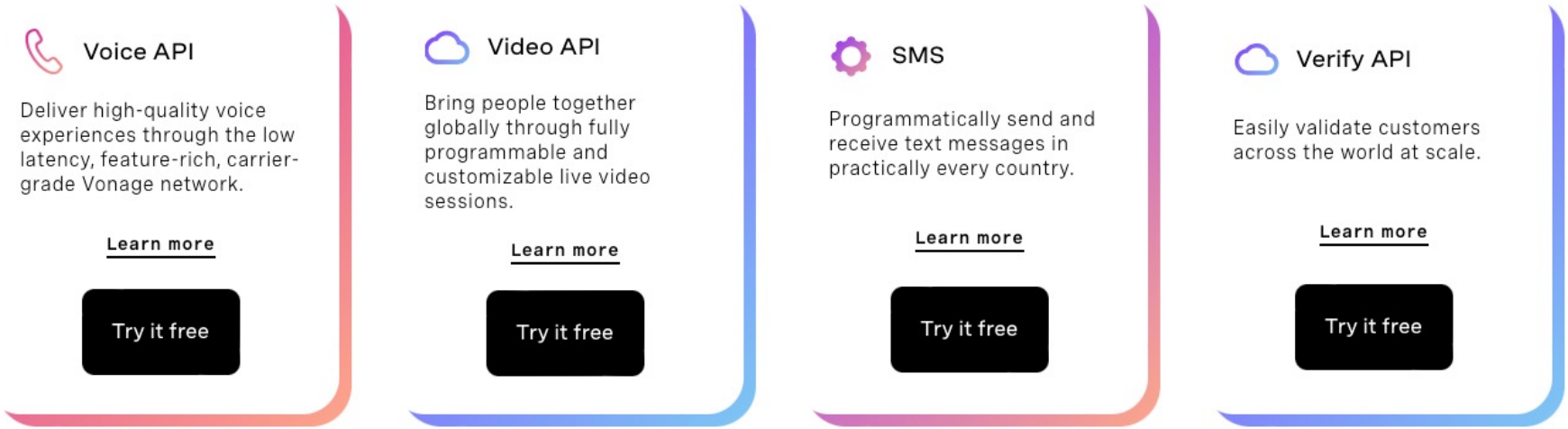
- P2P, 1-1, groups
- 1080p available, especially important for recording & broadcasts
- Broadcasting features equivalent to Twilio Live
- Flexible recording composition engines
- SIP video & audio
- Individual streams for AI integration

Security/Compliance

- Recordings can be encrypted at rest
- Regional media zones to isolate all media
- Deploy own TURN servers optionally
- Experience working with Healthcare/Financial Services to EU and US standards
- E2EE available

CPaaS Options: VONAGE

https://www.vonage.com/communications-apis/video/

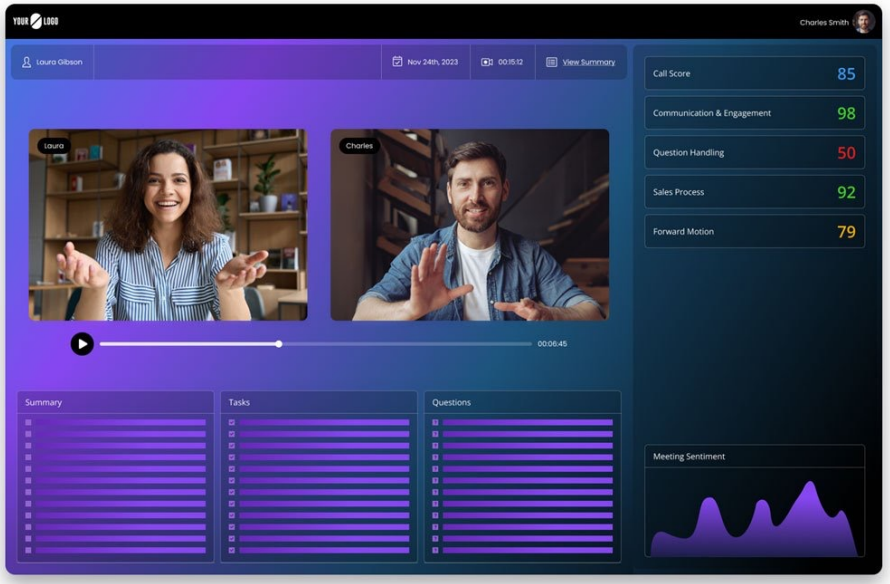


https://developer.vonage.com/en/video/twilio-vonage-video-migration-guide

AI/ML integration with: symbl.ai

https://webrtc.ventures/partners/symbl-ai/

- AI/ML platform for audio (streaming or recorded)
- Can produce Call Scores, Summaries, Questions, and Action Items
- APIs for Transcription, Sentiment Analysis, Entities, Trackers, Redaction
- On-premise LLM option available for sensitive use cases
- We can integrate this with any of the CPaaS's and projects mentioned today
- Great for use cases like contact centers and sales teams to supercharge productivity and save time for agents and customers



Dan Nordale from Symbl.ai will be our guest on WebRTC Live!
Wednesday February 21st, 12:30pm Eastern
Register now at: https://webrtc.ventures/webrtc-live/

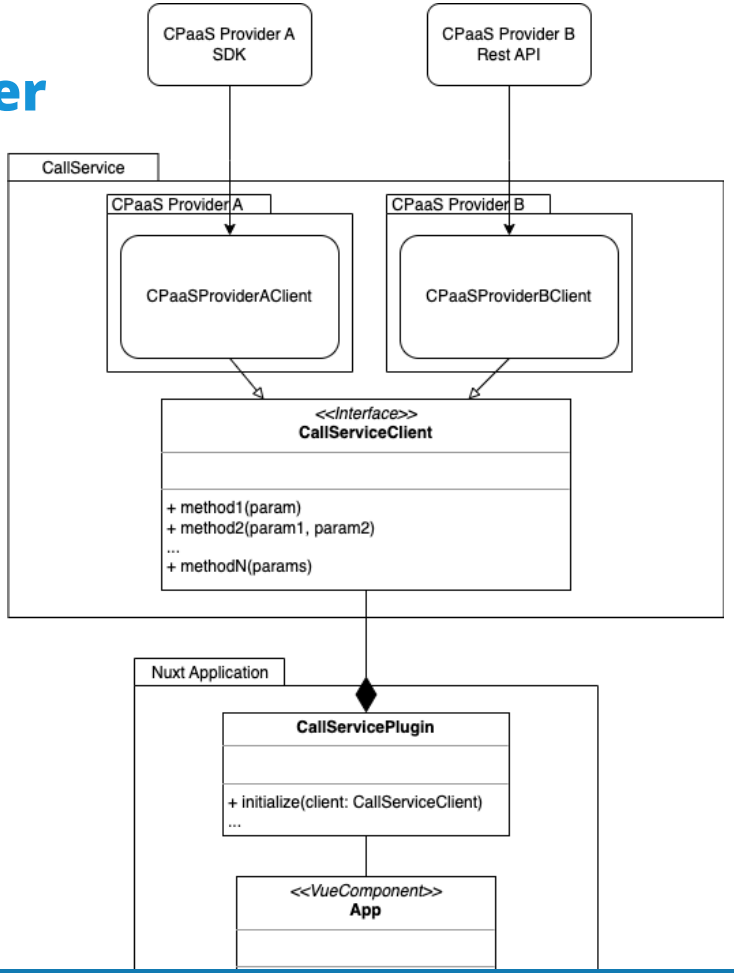
Abstracting the media server layer

https://www.youtube.com/@WebrtcVentures

https://www.youtube.com/watch?v=WGXPuivb7dM

Abstracting the media server layer

```
export interface CallServiceBase {

  initSession(info: CPaaSProviderInfo): Promise<void>

  screenShareImgPreview: string | null

  connectSession(sessionId: string, token: string): Promise<void>

  disconnectSession(sessionId: string): void

  destroySession(sessionId: string): void

  isSessionConnected(sessionId: string): boolean

  getStreamsForSession(sessionId: string): Stream[] | null

  getStreamForSession(sessionId: string, streamId: string): Stream | null

  subscribeToStream(
    sessionId: string,
    stream: Stream,
    targetElement: HTMLElement,
    fitMode?: FitMode,
    preferredResolution?: Resolution
  ): Promise<Subscriber>

  unsubscribeFromStream(
    sessionId: string,
```



https://webrtc.ventures/2023/07/how-to-incorporate-a-cpaas-abstraction-layer-in-webrtc-applications/

Abstracting the media server layer

Benefits

- **Vendor-agnostic Approach** - Decouples the application from any specific CPaaS provider's implementation details. This allows developers to switch between providers or add support for new ones without having to make significant changes to their codebase.
- **Standardized Interface** - Exposes a standardized interface or set of functions that encapsulate common features and functionality offered by CPaaS providers. Makes your app independent of the diverse approaches of specific features across various CPaaS providers.
- **Code Maintainability and Reusability** – Improves code maintainability and reusability. Changes or updates required due to provider-specific updates or API changes can be implemented in a single location within the abstraction layer, rather than scattered throughout the application codebase. This promotes cleaner code separation and easier maintenance.

There are Risks!

- Adds time and cost initially
- May make it more difficult to optimize for specific use cases, or in the way specific vendors have optimized their APIs to work for a use case

https://webrtc.ventures/2023/07/how-to-incorporate-a-cpaas-abstraction-layer-in-webrtc-applications/

5 Factors to base your choice on

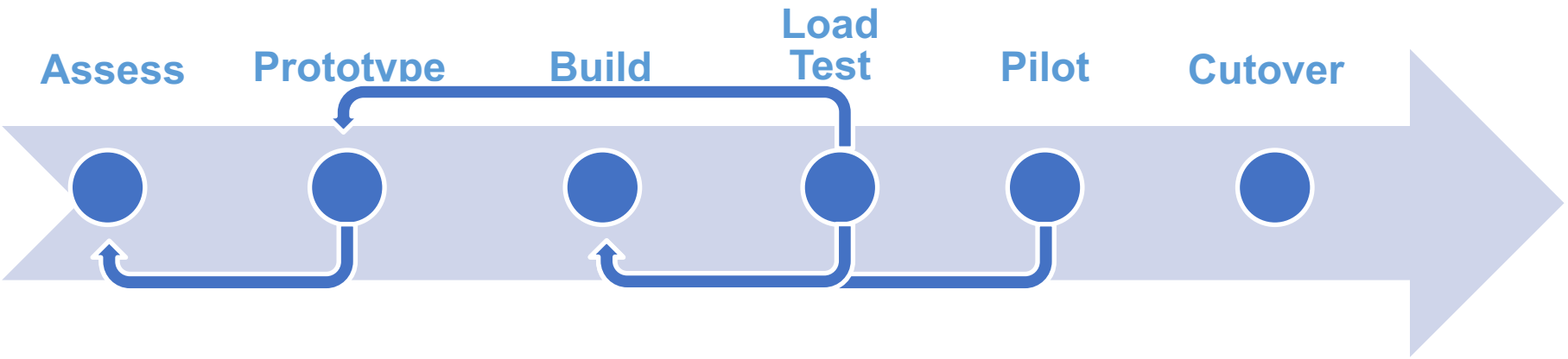
1. Feature Compatibility
   More than just listing supported features, but how are they implemented and how will they scale?
2. Media Server Architecture
   SFU's generally better and provide individual streams to work with, but telephony integrations benefit from MCU architectures
3. Programming Language
   The language the server is built in, as well as the languages they support in SDK's for web/mobile
4. Compliance / Data Protection
   HIPAA, GDPR, FedRAMP, etc
5. Cost / Pricing Structure
   Analyze based on your usage patterns and don't forget extra charges like recording. And possibly Enterprise Support options!

NOTE:

- May need to compromise something, there's lots of tradeoffs between providers
- Even if features are the same, the implementations likely vary a bit, so **prototype key features!**

https://webrtc.ventures/2023/12/5-technical-things-to-consider-when-switching-cpaas-providers/

Your 2024 Twilio Video Migration Plan



- Compare key features
- Consider pricing and APIs
- Select 1-3 options for prototyping

- Confirm pricing
- Negotiate?
- Confirm UI interactions
- Document gaps & changes to users

- Complete development and testing
- Set up user testing

- Test in actual conditions with realistic data

- Limited test with real users

- Need to deploy well before December 5th, 2024!!
- Plan for ongoing support afterwards

We are here to help with your Twilio Video migration!

Notes from today available at: https://webrtc.ventures/twilio-video/

BUILD
Complete web/mobile application development or work side by side with your team

INTEGRATE
Integrate communication APIs and ML/AI services into your existing application

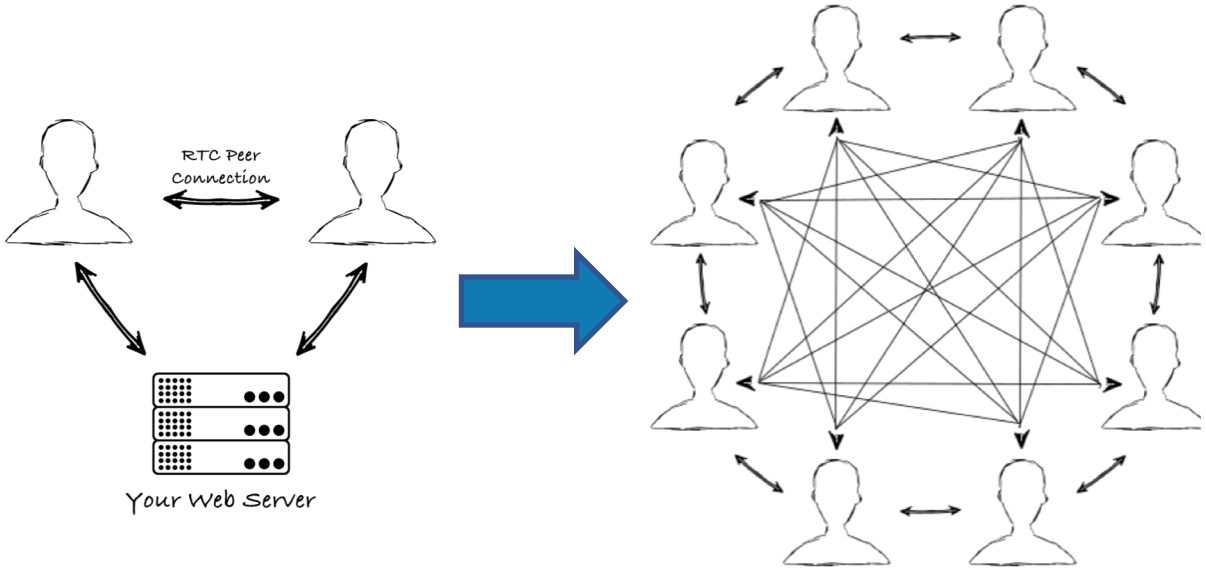
TEST
Web & Mobile
Manual, exploratory, automated, and load testing

ASSESS & OPTIMIZE
Leverage our years of experience to make your project successful!

DEPLOY & MANAGE
Complete DevOps and production deploys, with Managed Services and Support Contracts

Additional Slides for Q&A

WebRTC Scaling Challenges



SFUs or MCUs can help scale WebRTC

MCU – Multipoint Control Unit

- Handles mixing of video/audio streams in a central server so each participant only has one stream to deal with

SFU – Selective Forwarding Unit

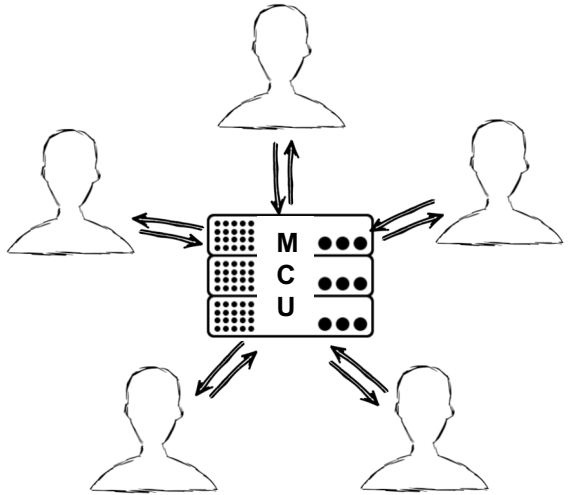
- Each participant only connects to the SFU, but receives unique streams for each participant

Either can add features beyond scaling

- Recording
- Broadcasting
- Interface to other services like transcription or VoIP legacy systems

MCU example

- Multipoint Control Unit
- Central server mixes all audio and video
- Each participant only gets one downloaded stream each for audio and video
- MCU controls a composited layout of that video for everyone, which can be nice but also introduces latency
- Heavy processing required on MCU, but more predictable bandwidth requirements



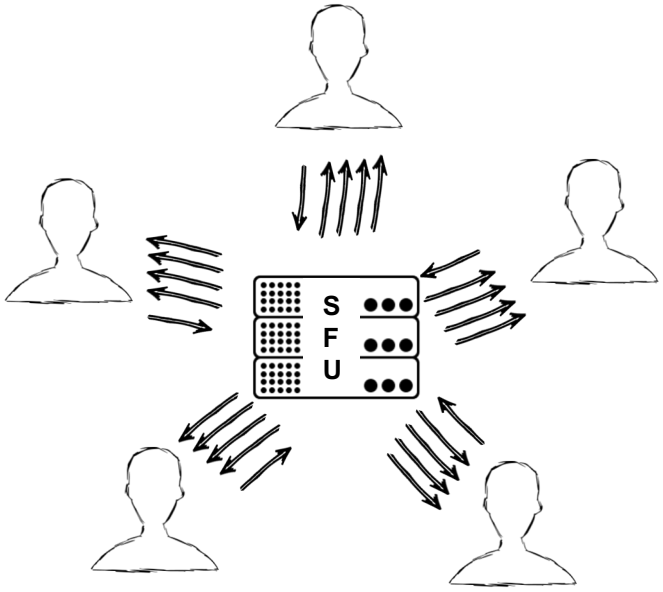
Media Servers offering MCU capability (not a comprehensive list):

FreeSWITCH

KURENTO

SFU example

- Selective Forwarding Unit
- Routes the correct stream to each user
- Still unique streams for each participant (allows for layout changes on user side)
- More powerful and more modern option but more complicated implementation
- Lower CPU requirements but more variable bandwidth required (based on # of users)
- Possible to do end-to-end encryption

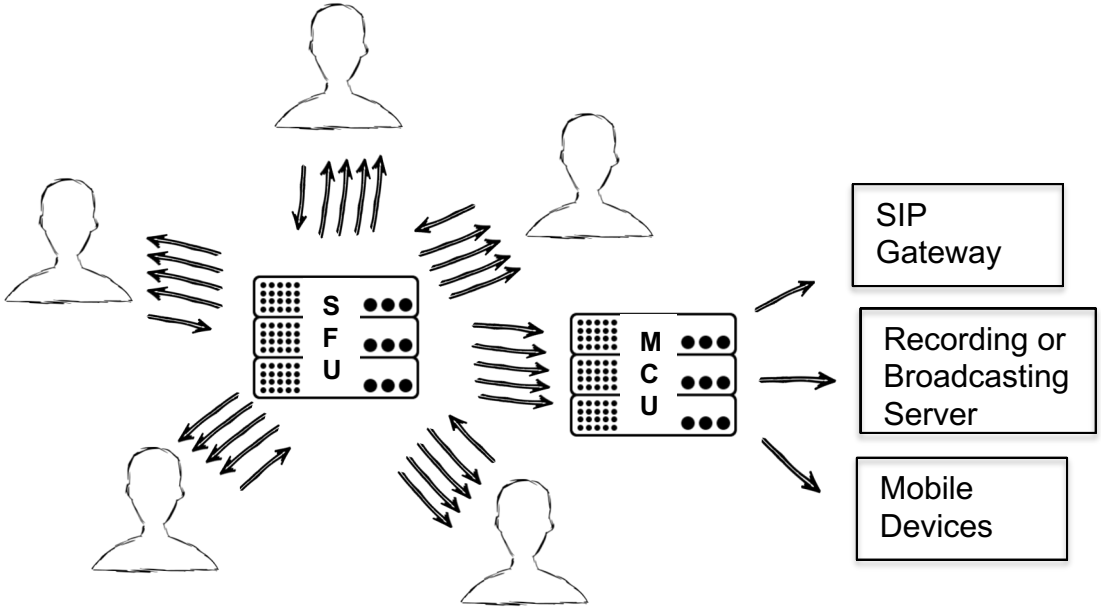


Media Servers offering SFU capability (not a comprehensive list):

jitsi.org

JANUS WebRTC Gateway

Why not both?



Concept from IIT RTC 2020 presentation by Lorenzo Miniero of MeetEcho / Janus, on “Can SFUs and MCUs be friends?”

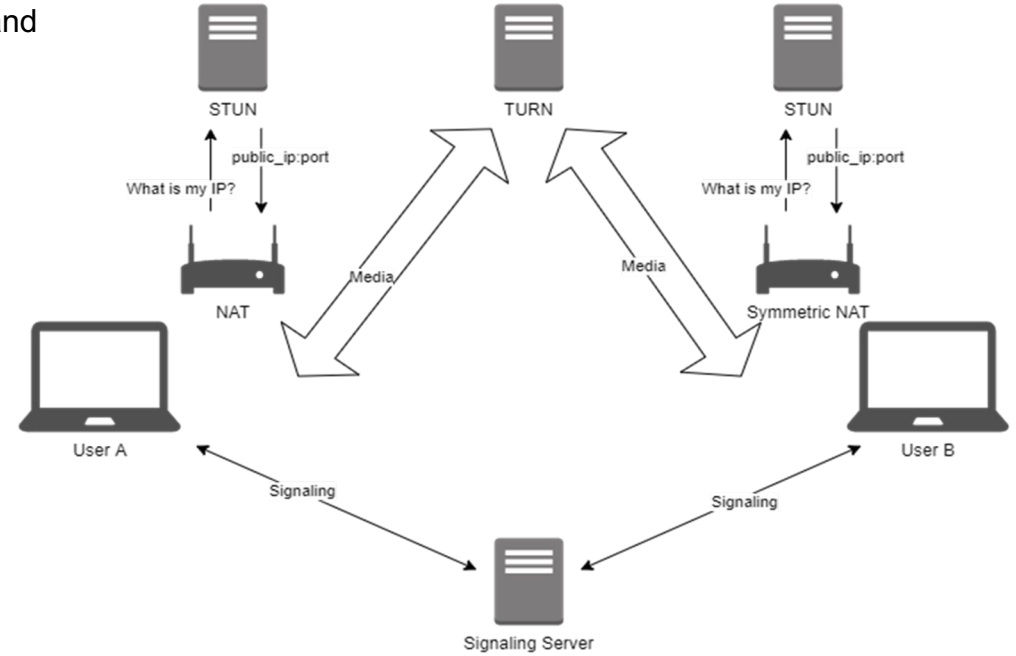
Use Case: A Standard 1-1 Video Chat Application

Overview of this use case

- Common for telehealth, intercom, remote control and basic click to call applications
- Browsers most tested and supported use case

Things to consider

- CPaaS or native application
- Forget about recording or broadcasting to many
- Free is not always better (the free approach might mean worst performance)
- Real E2EE (End to End Encryption) out of the box
- TURN



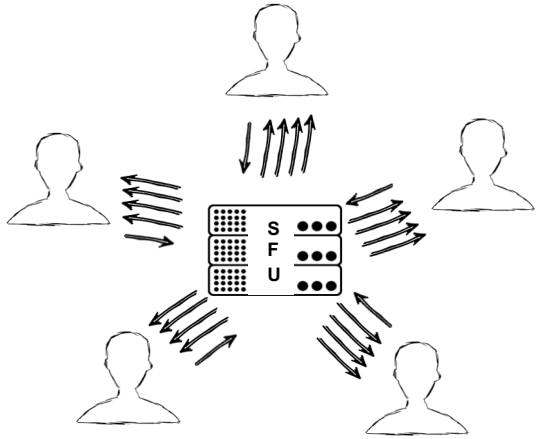
Use Case: A Group Chat Video Application

Overview of this use case

- Common for conferencing solutions like Jitsi, Google Meet, Zoom...
- SFU architecture is the most common

Things to consider

- Again, CPaaS or self hosted
- Scalability
- Interoperability is more complex
- Many videos = Resource intensive
- Simulcast and SVC
- Do you really need video?



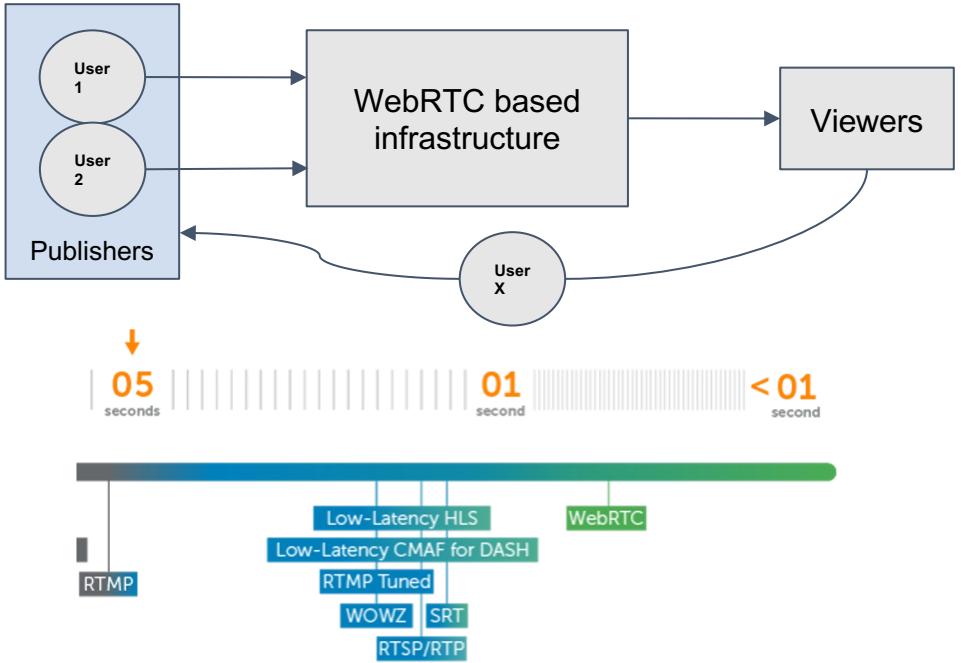
Use Case: Live Interactive Broadcasting

Overview of this use case

- Required for broadcasting media where latency matters: gaming, payments/betting, sports and other live events

Things to consider

- Estimated active publishers/viewer numbers
- More expensive than streaming on demand video
- Scalability
- Simulcast/SVC
- Do you really need low latency?



Use Case: Contact Centers

Overview of this use case

- Very common use case for medium/large companies that offer support/marketing

Things to consider

- Multiparty video conferencing support?
- Integration of multiple channels
- Integration with VoIP legacy systems
- Recording/voicemail and speech to text
- IVRs or voicebots