A Tale Of Two Worlds: Bridging SIP And WebRTC With Janus

Lorenzo Miniero
@elminiero

Kamailio World
19th May 2016,

Outline

What's Meetecho?

- A company born in 2009 as an academic spin-off
  - University research efforts brought to the market
  - Proudly brewed in sunny Napoli, Italy ☺
- Focus on real-time multimedia applications
  - Web conferencing only, at first
  - Then widened the scope to multimedia in general
  - Strong perspective on standardization and open source
    - *WebRTC rulez!*
- Several activities
  - Consulting services
  - Commercial support & licenses
  - Streaming of live events (e.g., IETF, ACM SIGCOMM, ...)
  - Products (conferencing, webinar, ...)

What's Meetecho?

- A company born in 2009 as an academic spin-off
  - University research efforts brought to the market
  - Proudly brewed in sunny Napoli, Italy ☺
- Focus on real-time multimedia applications
  - Web conferencing only, at first
  - Then widened the scope to multimedia in general
  - Strong perspective on standardization and open source
    - *WebRTC rulez!*
- Several activities
  - Consulting services
  - Commercial support & licenses
  - Streaming of live events (e.g., IETF, ACM SIGCOMM, ...)
  - Products (conferencing, webinar, ...)

What's Meetecho?

- A company born in 2009 as an academic spin-off
  - University research efforts brought to the market
  - Proudly brewed in sunny Napoli, Italy ☺
- Focus on real-time multimedia applications
  - Web conferencing only, at first
  - Then widened the scope to multimedia in general
  - Strong perspective on standardization and open source
    - *WebRTC rulez!*
- Several activities
  - Consulting services
  - Commercial support & licenses
  - Streaming of live events (e.g., IETF, ACM SIGCOMM, ...)
  - Products (conferencing, webinar, ...)

(*Napoli looks a bit like this...)

Ok, ok, enough about you... what's WebRTC about?

- Real-time media in a browser
- Up to some time ago, no standard solution!
  - No interoperability
  - Plugins needed to be installed anyway

WebRTC = Joint standardization efforts

- Internet Engineering Task Force (IETF)
- World Wide Web Consortium (W3C)

- RTCWEB (IETF)
  - Real-Time Communication in WEB browsers WG
  - Defines protocols and formats to use
- WEBRTC (W3C)
  - Web Real-Time Communications WG
  - Defines UI and API to access devices

Ok, ok, enough about you... what's WebRTC about?

- Real-time media in a browser
- Up to some time ago, no standard solution!
  - No interoperability
  - Plugins needed to be installed anyway

WebRTC = Joint standardization efforts

- Internet Engineering Task Force (IETF)
- World Wide Web Consortium (W3C)

- RTCWEB (IETF)
  - Real-Time Communication in WEB browsers WG
  - Defines protocols and formats to use
- WEBRTC (W3C)
  - Web Real-Time Communications WG
  - Defines UI and API to access devices

Ok, ok, enough about you... what's WebRTC about?

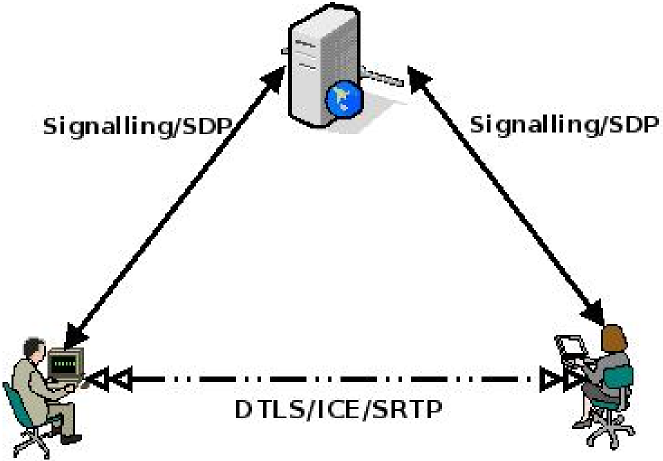
- Real-time media in a browser
- Up to some time ago, no standard solution!
  - No interoperability
  - Plugins needed to be installed anyway

WebRTC = Joint standardization efforts

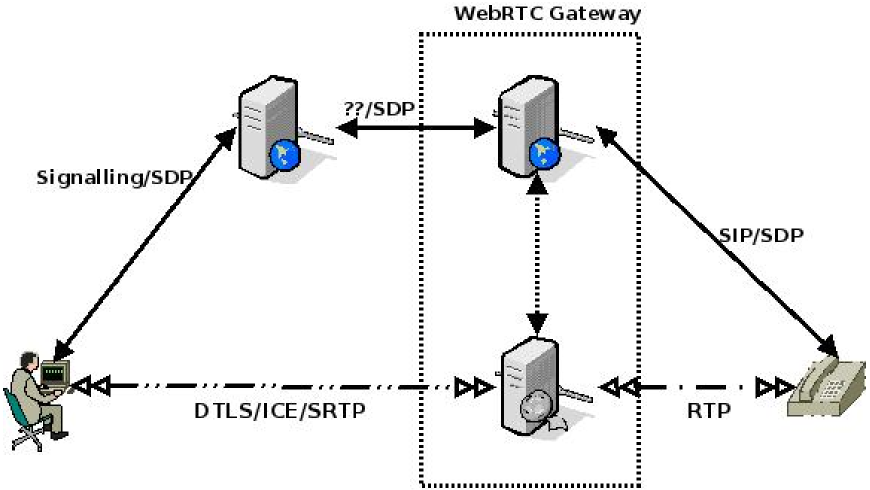
- Internet Engineering Task Force (IETF)
- World Wide Web Consortium (W3C)

- RTCWEB (IETF)
  - Real-Time Communication in WEB browsers WG
  - Defines protocols and formats to use
- WEBRTC (W3C)
  - Web Real-Time Communications WG
  - Defines UI and API to access devices

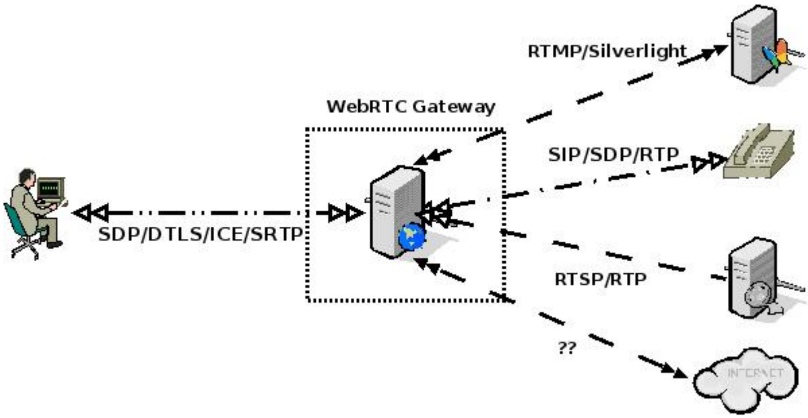
WebRTC reference architecture



Involving a gateway (and applications)



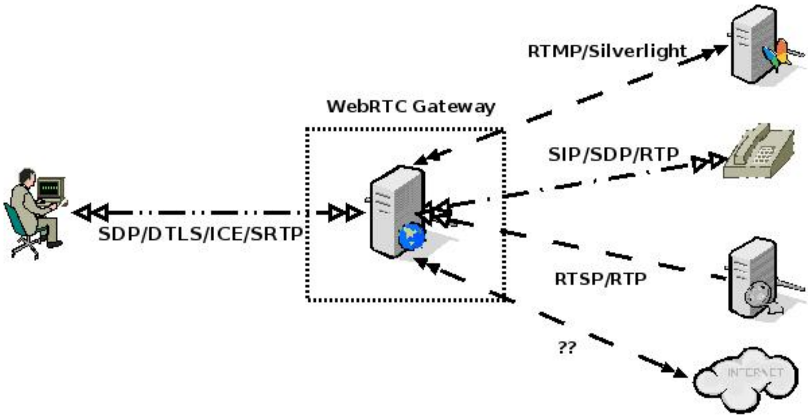
Involving different technologies as well



"What is a WebRTC Gateway anyway?"

- https://webrtchacks.com/webrtc-gw/

Involving different technologies as well



"What is a WebRTC Gateway anyway?"

- https://webrtchacks.com/webrtc-gw/

Janus: a general purpose WebRTC gateway

"In ancient Roman religion and myth, Janus [..] is the god of beginnings and transitions, and thereby of gates, doors, passages, endings and time. He is usually depicted as having two faces, since he looks to the future and to the past."

— http://en.wikipedia.org/wiki/Janus

Janus: a general purpose WebRTC gateway

- A door between the communications past and future
  - Legacy technologies (the “past”)
  - WebRTC (the “future”)

Janus

General purpose, open source WebRTC gateway

- https://github.com/meetecho/janus-gateway
- Demos and documentation: https://janus.conf.meetecho.com
- Community: https://groups.google.com/forum/#!forum/meetecho-janus

Modular architecture

- The core only implements the WebRTC stack
  - JSEP/SDP, ICE, DTLS-SRTP, Data Channels, ...
- Plugins expose Janus API over different transports
  - Currently HTTP / WebSockets / RabbitMQ / Unix Sockets
- "Application" logic implemented in plugins too
  - Users attach to plugins via the gateway core
  - The gateway handles the WebRTC stuff
  - Plugins route/manipulate the media/data
- Some proof of concept plugins implemented
  - Echo Test
  - Streaming ($\rightarrow$ Plain RTP to WebRTC!)
  - Video Room ($\rightarrow$ Selective Forwarding Unit!)
  - SIP Gateway ($\rightarrow$ "Legacy" SIP!)
  - ...

Modular architecture

- The core only implements the WebRTC stack
  - JSEP/SDP, ICE, DTLS-SRTP, Data Channels, ...
- Plugins expose Janus API over different transports
  - Currently HTTP / WebSockets / RabbitMQ / Unix Sockets
- “Application” logic implemented in plugins too
  - Users attach to plugins via the gateway core
  - The gateway handles the WebRTC stuff
  - Plugins route/manipulate the media/data
- Some proof of concept plugins implemented
  - Echo Test
  - Streaming ($\rightarrow$ Plain RTP to WebRTC!)
  - Video Room ($\rightarrow$ Selective Forwarding Unit!)
  - SIP Gateway ($\rightarrow$ “Legacy” SIP!)
  - ...

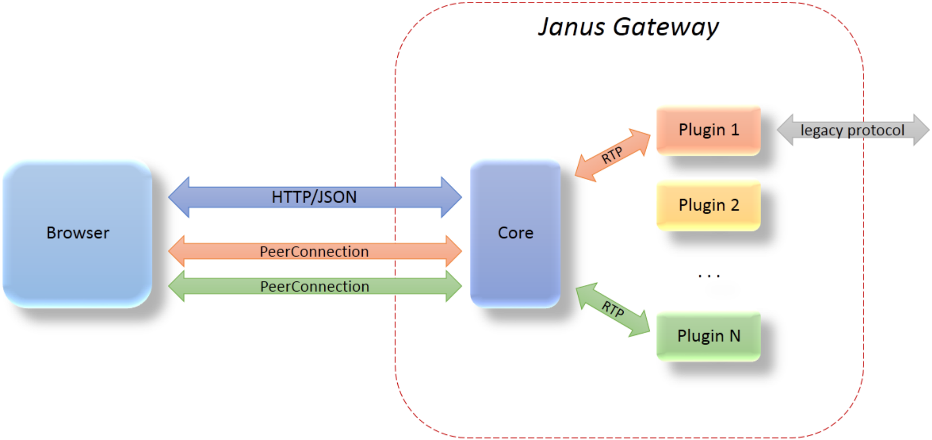
Modular architecture

- The core only implements the WebRTC stack
  - JSEP/SDP, ICE, DTLS-SRTP, Data Channels, ...
- Plugins expose Janus API over different transports
  - Currently HTTP / WebSockets / RabbitMQ / Unix Sockets
- “Application” logic implemented in plugins too
  - Users attach to plugins via the gateway core
  - The gateway handles the WebRTC stuff
  - Plugins route/manipulate the media/data
- Some proof of concept plugins implemented
  - Echo Test
  - Streaming ($\rightarrow$ Plain RTP to WebRTC!)
  - Video Room ($\rightarrow$ Selective Forwarding Unit!)
  - SIP Gateway ($\rightarrow$ “Legacy” SIP!)
  - ...

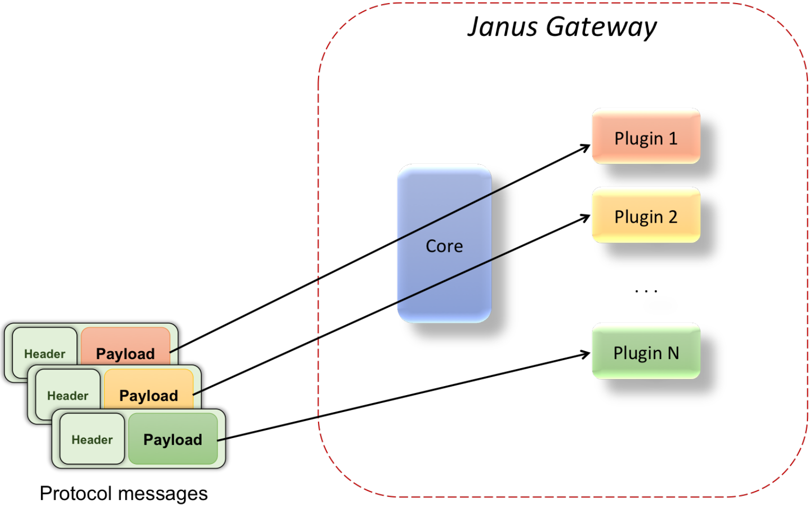
Modular architecture

- The core only implements the WebRTC stack
  - JSEP/SDP, ICE, DTLS-SRTP, Data Channels, ...
- Plugins expose Janus API over different transports
  - Currently HTTP / WebSockets / RabbitMQ / Unix Sockets
- “Application” logic implemented in plugins too
  - Users attach to plugins via the gateway core
  - The gateway handles the WebRTC stuff
  - Plugins route/manipulate the media/data
- Some proof of concept plugins implemented
  - Echo Test
  - Streaming ($\rightarrow$ Plain RTP to WebRTC!)
  - Video Room ($\rightarrow$ Selective Forwarding Unit!)
  - SIP Gateway ($\rightarrow$ “Legacy” SIP!)
  - ...

Extensible Architecture and API



Extensible Architecture and API



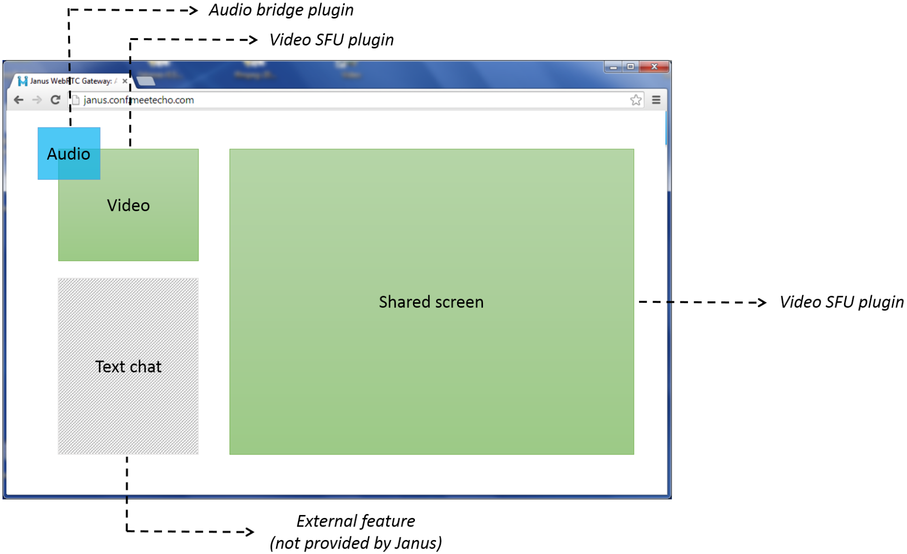
Plugins as “bricks”

- Each plugin is a feature, not an application
- Application can be composed out of different features
  - Features as “bricks” for a complex scenario
- A few examples...
  - Multimedia conferencing with PSTN support
    - Video Room (participants video & screen) + SIP (participants audio)
  - Webinar with Q&A
    - Video Room (screen) + Video Room (speakers) + Audio Bridge (questions)
  - Social TV
    - Streaming (TV channel) + Video Room (interaction)
  - Contact center / Communication in social networks
    - SIP plugin (calls) + Echo Test (diagnostics) + Record & Play (messaging)

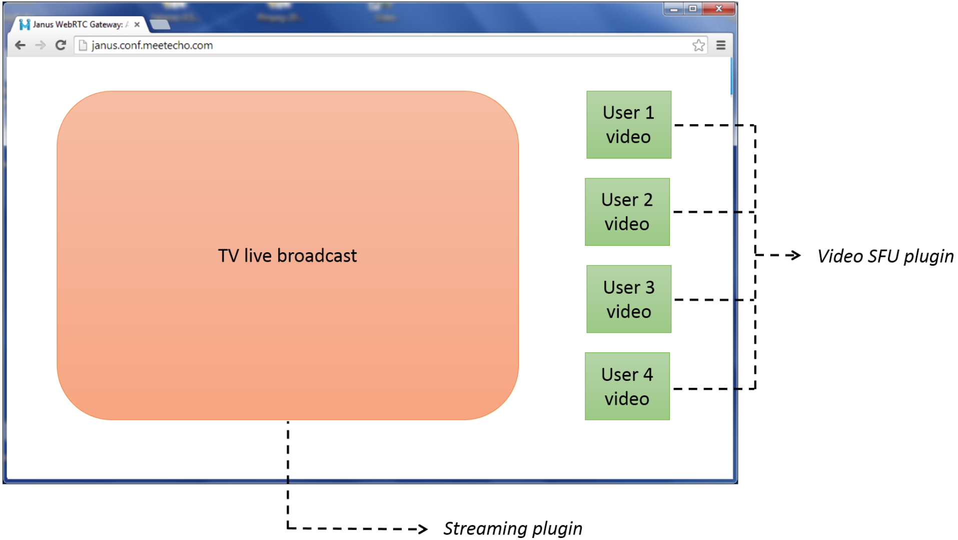
Plugins as “bricks”

- Each plugin is a feature, not an application
- Application can be composed out of different features
  - Features as “bricks” for a complex scenario
- A few examples...
  - Multimedia conferencing with PSTN support
    - Video Room (participants video & screen) + SIP (participants audio)
  - Webinar with Q&A
    - Video Room (screen) + Video Room (speakers) + Audio Bridge (questions)
  - Social TV
    - Streaming (TV channel) + Video Room (interaction)
  - Contact center / Communication in social networks
    - SIP plugin (calls) + Echo Test (diagnostics) + Record & Play (messaging)

Webinar with Q/A



Social TV


Janus WebRTC Gateway:
janus.conf.meetecho.com
TV live broadcast
User 1 video
User 2 video
User 3 video
User 4 video
Video SFU plugin
Streaming plugin

Anything wrong? Check the Admin API!

- Requests/response API to interrogate Janus
  - Query server capabilities
  - Control some aspects (e.g., enable/disable debugging)
  - Inspect handles and WebRTC "internals"
- What about asynchronous events? More on that later!

http://www.meetecho.com/blog/understanding-the-janus-admin-api/

Yeah, yeah, but what about SIP?

- As anticipated, SIP already available as a Janus plugin
  - Demo: https://janus.conf.meetecho.com/siptest
- Basically a WebRTC-to-SIP gateway
  - WebRTC on one side, SIP(S)/(S)RTP on the other end
- Janus SIP plugin acts as a SIP endpoint
  - SIP stack implemented with Sofia-SIP
  - WebRTC users only see the Janus API (JSON)
  - No transcoding, media is only relayed
- Simplifies life for web developers
  - No need to worry about a SIP stack (only SIP URIs)
  - Basic methods/events to handle call (call, answer, hangup)
  - Allows headers injection, for ad-hoc cases

Yeah, yeah, but what about SIP?

- As anticipated, SIP already available as a Janus plugin
  - Demo: https://janus.conf.meetecho.com/siptest
- Basically a WebRTC-to-SIP gateway
  - WebRTC on one side, SIP(S)/(S)RTP on the other end
- Janus SIP plugin acts as a SIP endpoint
  - SIP stack implemented with Sofia-SIP
  - WebRTC users only see the Janus API (JSON)
  - No transcoding, media is only relayed
- Simplifies life for web developers
  - No need to worry about a SIP stack (only SIP URIs)
  - Basic methods/events to handle call (call, answer, hangup)
  - Allows headers injection, for ad-hoc cases

Yeah, yeah, but what about SIP?

- As anticipated, SIP already available as a Janus plugin
  - Demo: https://janus.conf.meetecho.com/siptest
- Basically a WebRTC-to-SIP gateway
  - WebRTC on one side, SIP(S)/(S)RTP on the other end
- Janus SIP plugin acts as a SIP endpoint
  - SIP stack implemented with Sofia-SIP
  - WebRTC users only see the Janus API (JSON)
  - No transcoding, media is only relayed
- Simplifies life for web developers
  - No need to worry about a SIP stack (only SIP URIs)
  - Basic methods/events to handle call (call, answer, hangup)
  - Allows headers injection, for ad-hoc cases

Yeah, yeah, but what about SIP?

- As anticipated, SIP already available as a Janus plugin
  - Demo: https://janus.conf.meetecho.com/siptest
- Basically a WebRTC-to-SIP gateway
  - WebRTC on one side, SIP(S)/(S)RTP on the other end
- Janus SIP plugin acts as a SIP endpoint
  - SIP stack implemented with Sofia-SIP
  - WebRTC users only see the Janus API (JSON)
  - No transcoding, media is only relayed
- Simplifies life for web developers
  - No need to worry about a SIP stack (only SIP URIs)
  - Basic methods/events to handle call (call, answer, hangup)
  - Allows headers injection, for ad-hoc cases

But what if you DON'T want it simple?

- You may want to have more control on SIP messaging
  - e.g., to re-use stacks like JsSIP or SIP.js, or other reasons
- The existing SIP plugin doesn't allow for that
  - Complexity hidden from users, on purpose
  - Only partial control (e.g., custom headers, INFO DTMF, negotiating security)
- **BUT!** Janus is extensible, so why not a new plugin?
- @saghul's idea: "BoringSDP"!
  - A new plugin to only handle media gatewaying
    - WebRTC and SIP SDPs both available to web user
  - You handle SIP transactions yourself, and leave media to Janus
    - You still need to communicate with Janus as well, of course

But what if you DON'T want it simple?

- You may want to have more control on SIP messaging
  - e.g., to re-use stacks like JsSIP or SIP.js, or other reasons
- The existing SIP plugin doesn't allow for that
  - Complexity hidden from users, on purpose
  - Only partial control (e.g., custom headers, INFO DTMF, negotiating security)
- **BUT!** Janus is extensible, so why not a new plugin?
- @saghul's idea: "BoringSDP"!
  - A new plugin to only handle media gatewaying
    - WebRTC and SIP SDPs both available to web user
  - You handle SIP transactions yourself, and leave media to Janus
    - You still need to communicate with Janus as well, of course

But what if you DON'T want it simple?

- You may want to have more control on SIP messaging
  - e.g., to re-use stacks like JsSIP or SIP.js, or other reasons
- The existing SIP plugin doesn't allow for that
  - Complexity hidden from users, on purpose
  - Only partial control (e.g., custom headers, INFO DTMF, negotiating security)
- **BUT!** Janus is extensible, so why not a new plugin?
- @saghul's idea: "BoringSDP"!
  - A new plugin to only handle media gatewaying
    - WebRTC and SIP SDPs both available to web user
  - You handle SIP transactions yourself, and leave media to Janus
    - You still need to communicate with Janus as well, of course

But what if you DON'T want it simple?

- You may want to have more control on SIP messaging
  - e.g., to re-use stacks like JsSIP or SIP.js, or other reasons
- The existing SIP plugin doesn't allow for that
  - Complexity hidden from users, on purpose
  - Only partial control (e.g., custom headers, INFO DTMF, negotiating security)
- **BUT!** Janus is extensible, so why not a new plugin?
- @saghul's idea: "BoringSDP"!
  - A new plugin to only handle media gatewaying
    - WebRTC and SIP SDPs both available to web user
  - You handle SIP transactions yourself, and leave media to Janus
    - You still need to communicate with Janus as well, of course

What is Janus used for today, and by whom?

- We use it ourselves for many things (obviously)
  - Web conferencing and Webinars
  - WebRTC-to-SIP gateway
  - Streaming of live events (e.g., IETF meetings)
- Many folks/companies also using it in creative ways!
  - E-learning
  - Coworking
  - Contact centers
  - TV broadcasting and Social TV
  - Surveillance systems
  - E-health
  - Home automation & Internet of Things
  - Mobile devices, Raspberry Pis, drones, etc.
- New third-party tools are starting to come out
  - https://janus.conf.meetecho.com/docs/resources

What is Janus used for today, and by whom?

- We use it ourselves for many things (obviously)
  - Web conferencing and Webinars
  - WebRTC-to-SIP gateway
  - Streaming of live events (e.g., IETF meetings)
- Many folks/companies also using it in creative ways!
  - E-learning
  - Coworking
  - Contact centers
  - TV broadcasting and Social TV
  - Surveillance systems
  - E-health
  - Home automation & Internet of Things
  - Mobile devices, Raspberry Pis, drones, etc.
- New third-party tools are starting to come out
  - https://janus.conf.meetecho.com/docs/resources

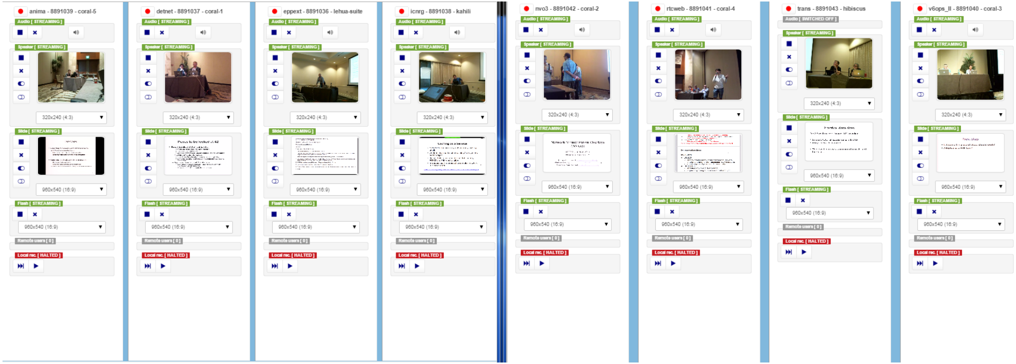
What is Janus used for today, and by whom?

- We use it ourselves for many things (obviously)
  - Web conferencing and Webinars
  - WebRTC-to-SIP gateway
  - Streaming of live events (e.g., IETF meetings)
- Many folks/companies also using it in creative ways!
  - E-learning
  - Coworking
  - Contact centers
  - TV broadcasting and Social TV
  - Surveillance systems
  - E-health
  - Home automation & Internet of Things
  - Mobile devices, Raspberry Pis, drones, etc.
- New third-party tools are starting to come out
  - https://janus.conf.meetecho.com/docs/resources

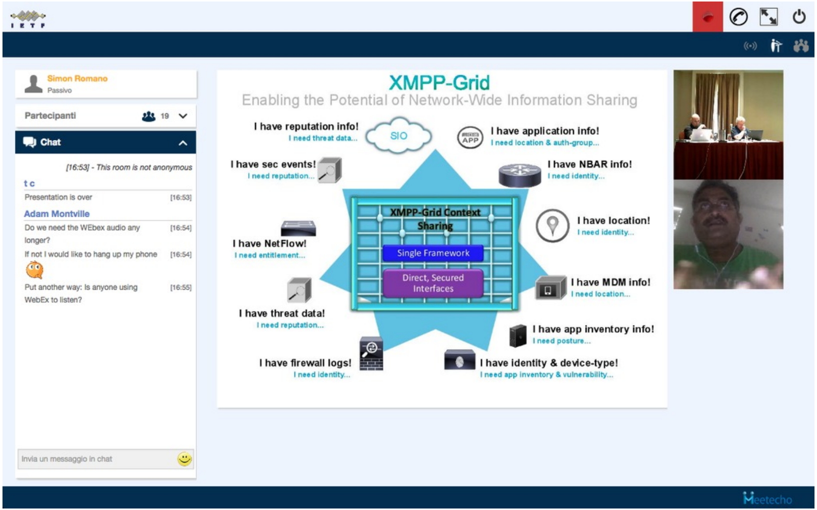
"Director" room @ IETF meetings

Completely WebRTC-based media streams

- Slides as a video feed from the beamer
- Static video feed from the room
- Dynamic video feeds for remote speakers

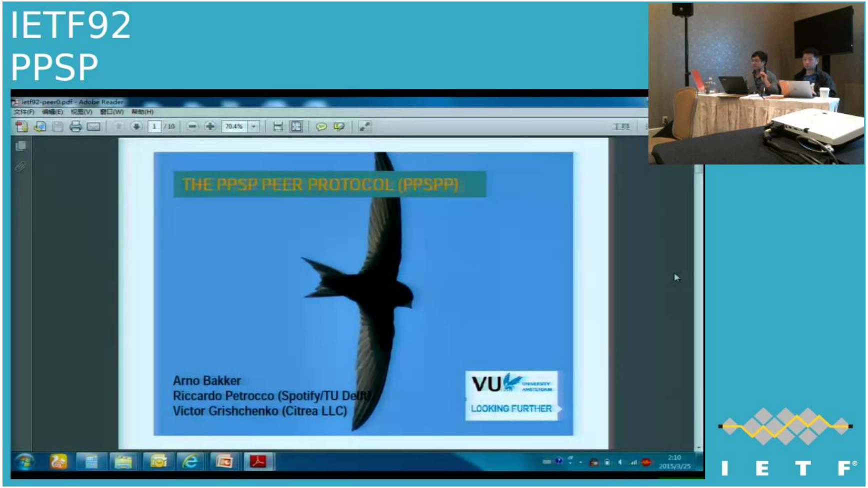


Meetecho: IETF meeting example



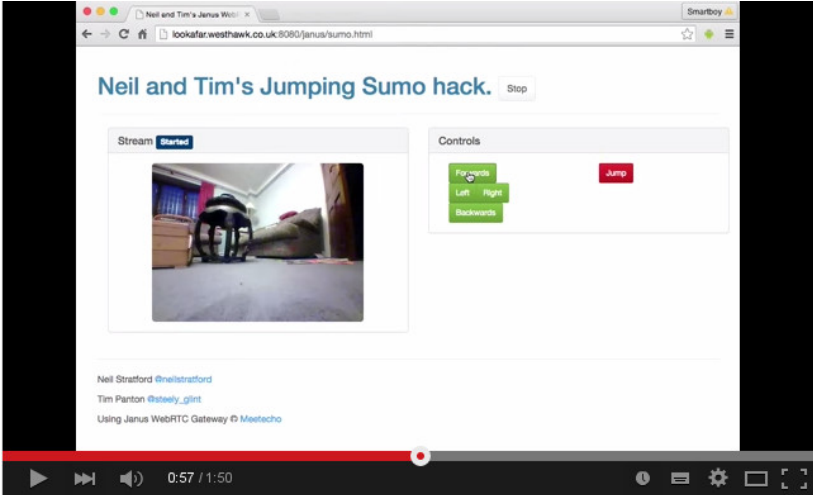
https://ietf.org/meeting/remote-participation.html

Meetecho: IETF recordings



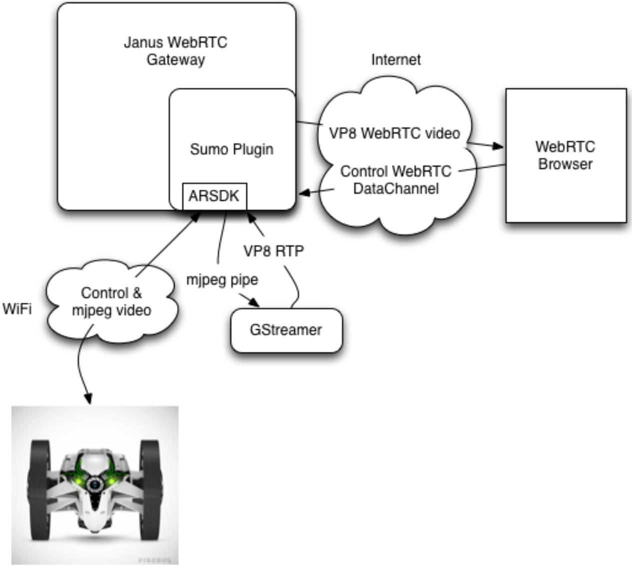
https://www.youtube.com/user/ietf

A “silly” use case: The Jumping Sumo!



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

A “silly” use case: The Jumping Sumo!



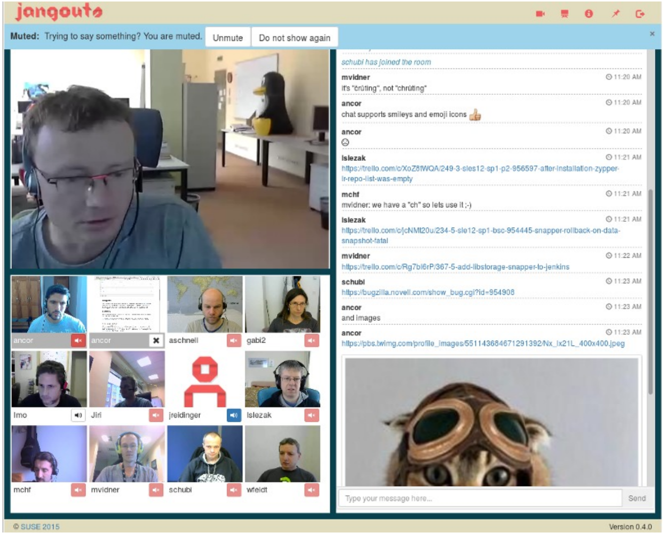
“Matrix wins Best of Show at WebRTC World!”

https://www.youtube.com/watch?v=OMzDklvDS3c

"Matrix wins Best of Show at WebRTC World!"

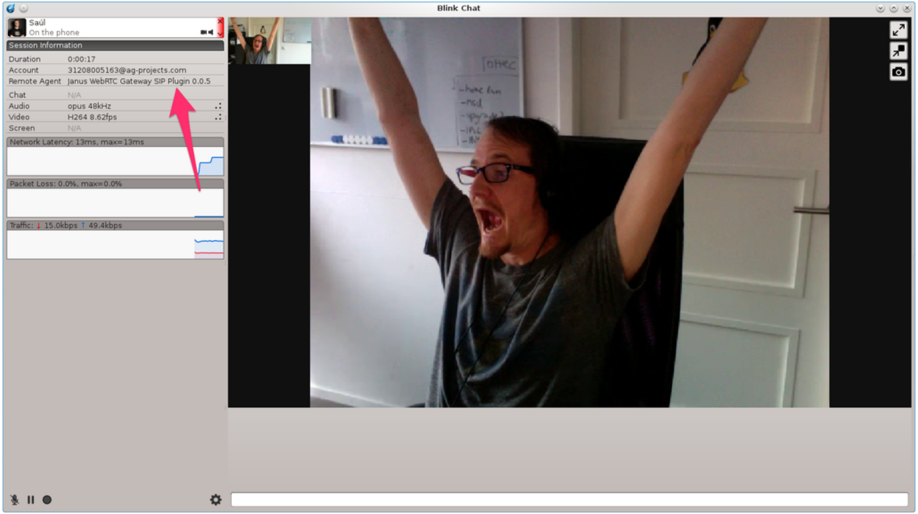
https://www.youtube.com/watch?v=NpBStIlq6fM

Jangouts (for "Janus Hangouts" ☺)



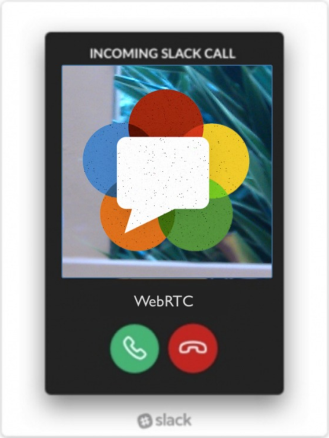
https://github.com/jangouts/jangouts

SylkServer (SIP/XMPP Application Server)



http://sylkserver.com/

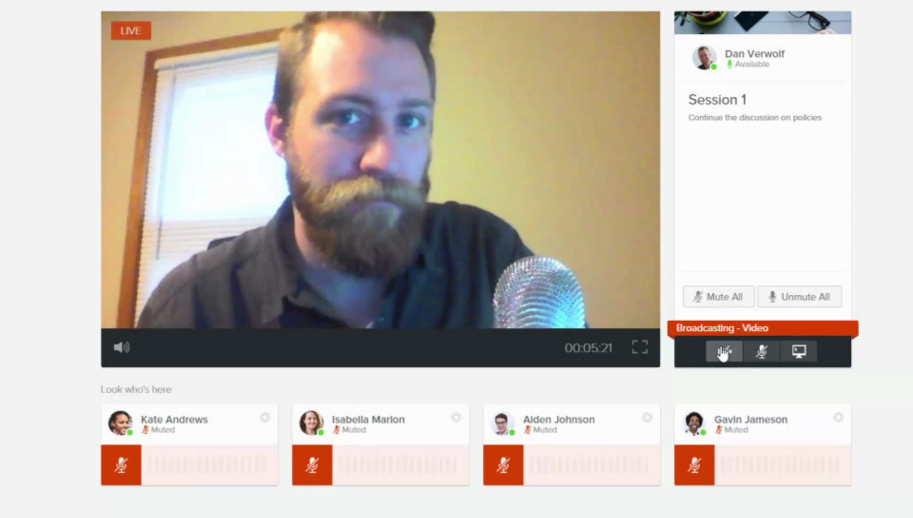
Slack? (team co-working)



https://webrtchacks.com/dear-slack/

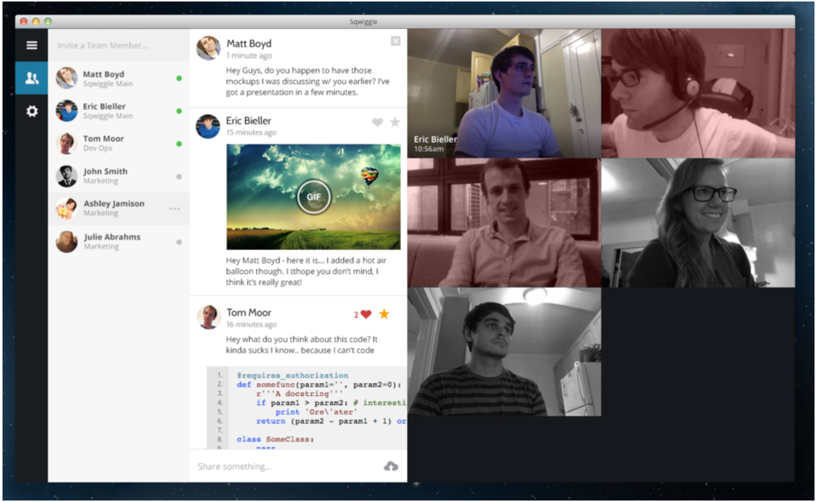
https://webrtchacks.com/slack-webrtc-slacking/

Lenovo's AirClass (e-learning)



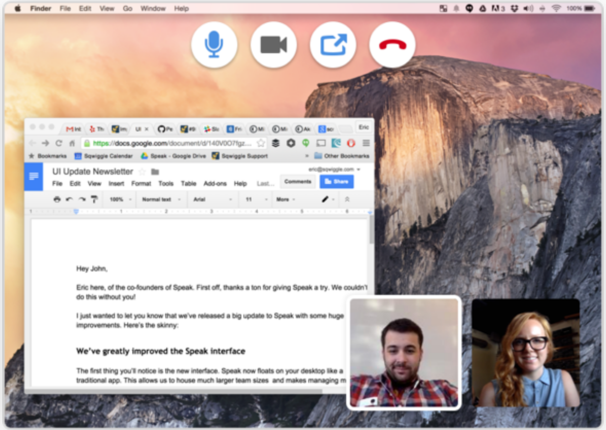
https://www.airclass.com

Sqwiggle / Speak.io (team co-working)



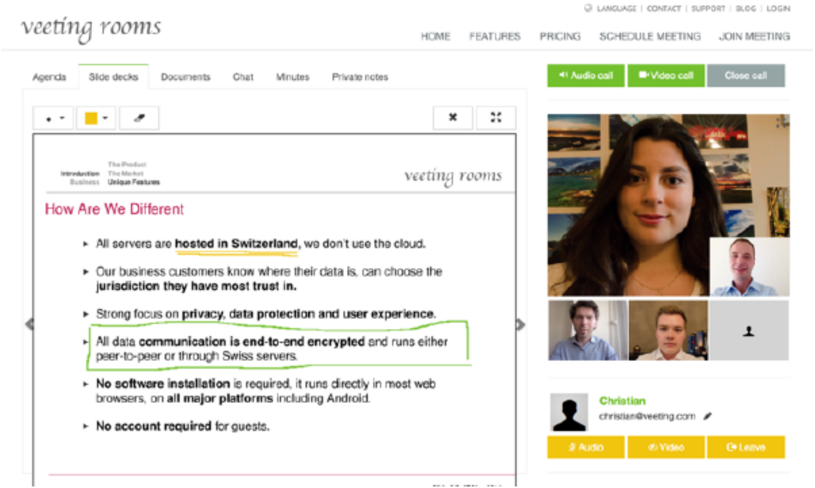
https://www.sqwiggle.com
https://speak.io

Sqwiggle / Speak.io (team co-working)



https://www.sqwiggle.com
https://speak.io

Veeting rooms (web conferencing)



https://www.veeting.com

What to do next?

- Finalize the WebRTC implementation
  - Renegotiation, multistream, ...
- Keep on improving and fixing things
  - Code cleanup, reference counters, new modules, ...
- A recent idea: modular Events API!
  - Core and plugins generate events
  - Custom modules can subscribe to and handle them
    - e.g., save to DB, send to external service, etc.
  - A couple of potentially interesting integrations
    - Homer/HEP, for SIP calls and their relation to WebRTC
    - Live stats collection via callstats.io

Help us improve this!

- Play with it, more testing is important
- Write your own applications/wrappers/plugins!

What to do next?

- Finalize the WebRTC implementation
  - Renegotiation, multistream, ...
- Keep on improving and fixing things
  - Code cleanup, reference counters, new modules, ...
- A recent idea: modular Events API!
  - Core and plugins generate events
  - Custom modules can subscribe to and handle them
    - e.g., save to DB, send to external service, etc.
  - A couple of potentially interesting integrations
    - Homer/HEP, for SIP calls and their relation to WebRTC
    - Live stats collection via callstats.io

Help us improve this!

- Play with it, more testing is important
- Write your own applications/wrappers/plugins!

What to do next?

- Finalize the WebRTC implementation
  - Renegotiation, multistream, ...
- Keep on improving and fixing things
  - Code cleanup, reference counters, new modules, ...
- A recent idea: modular Events API!
  - Core and plugins generate events
  - Custom modules can subscribe to and handle them
    - e.g., save to DB, send to external service, etc.
  - A couple of potentially interesting integrations
    - Homer/HEP, for SIP calls and their relation to WebRTC
    - Live stats collection via callstats.io

Help us improve this!

- Play with it, more testing is important
- Write your own applications/wrappers/plugins!

What to do next?

- Finalize the WebRTC implementation
  - Renegotiation, multistream, ...
- Keep on improving and fixing things
  - Code cleanup, reference counters, new modules, ...
- A recent idea: modular Events API!
  - Core and plugins generate events
  - Custom modules can subscribe to and handle them
    - e.g., save to DB, send to external service, etc.
  - A couple of potentially interesting integrations
    - Homer/HEP, for SIP calls and their relation to WebRTC
    - Live stats collection via callstats.io

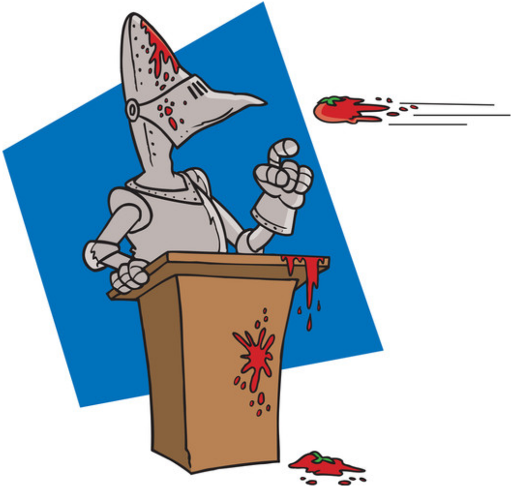
Help us improve this!

- Play with it, more testing is important
- Write your own applications/wrappers/plugins!

Shameless bragging! Souvenirs from San Francisco ☺

Innovation Challenge Panel/Pitchfest @ INFOCOM2016... we won!

Questions? Comments?



@elminiero