# WebRTC: The Web Way to Communicate

**Alan Johnston**
Distinguished Engineer
**Avaya**
Adjunct Instructor
**Washington University in St. Louis**

**Dan Burnett**
Director of Standards
**Voxeo Labs**

# WebRTC Tutorial Topics

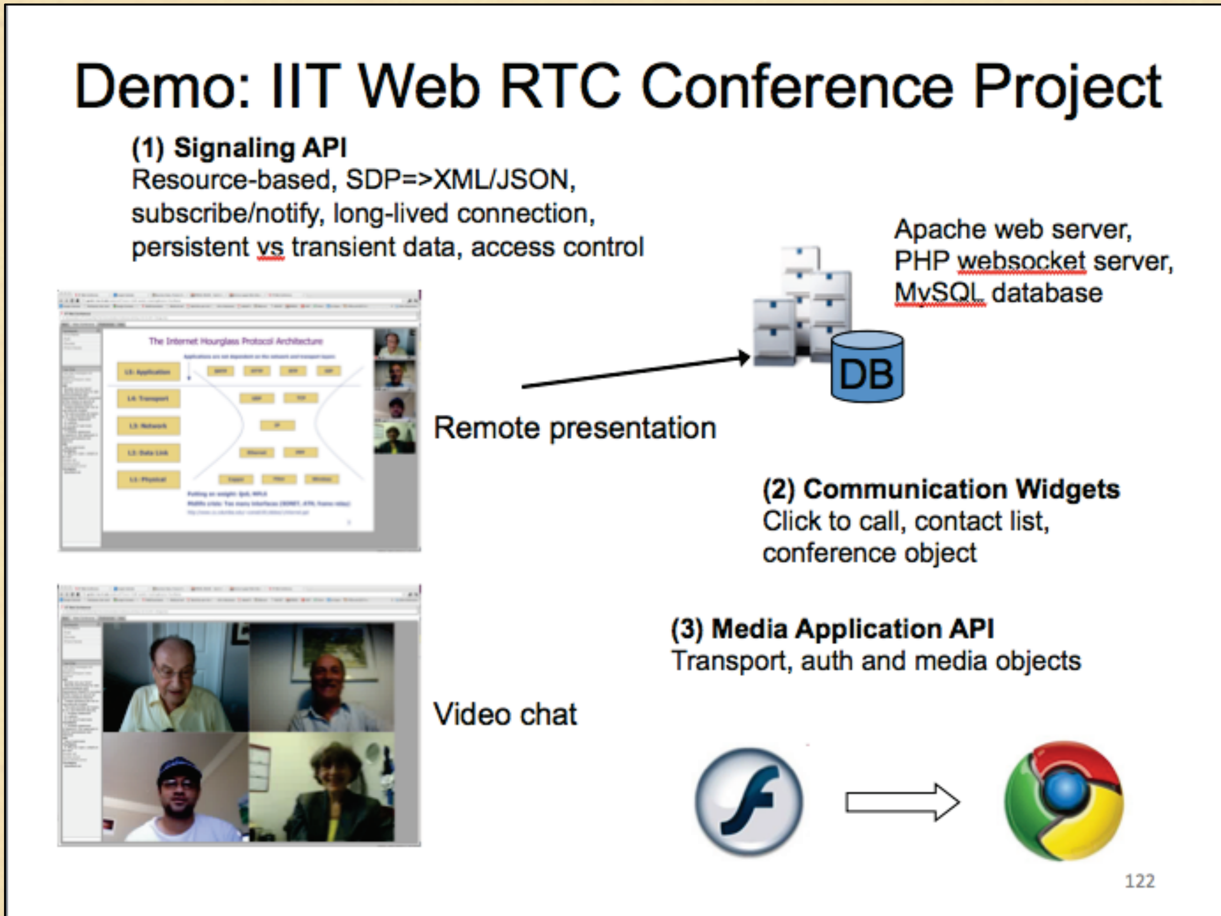
- What is WebRTC?
- How to Use WebRTC – Peer Connection
- WebRTC Peer-to-Peer Media
- WebRTC Protocols and IETF Standards
- WebRTC and the Enterprise
- What's Next?

# What is WebRTC?

# Real-Time Communication on the Internet

- NVP (Network Voice Protocol)
  - 1977
- RTP (Real-time Tranport Protocol)
  - First used in 1992
  - First published as IETF RFC in 1996
  - Still used today for VoIP and with SIP
- ITU H.323 video telephony standard
  - 1996
  - Voice and video conferencing
- IETF SIP – Session Initiation Protocol
  - 1999
  - Unleashed VoIP revolution on telephony
  - Video and room conferencing
  - Protocol widely used by service providers and in enterprises
- Real-Time Communication on the Web
  - Voice and video on the Internet using browser plugins
  - 2006 with GoogleTalk inside Gmail
  - WebRTC standardizes and eliminates need for plugin or download

# IIT Voice and Video over Web Project




http://sites.google.com/site/vvowproject

# Why not just use Flash?

**Pros**

- Most browsers already have Flash plugin
- Streaming audio and video uses Flash today
- Flash supports real-time audio and video
- Web developers familiar with Flash

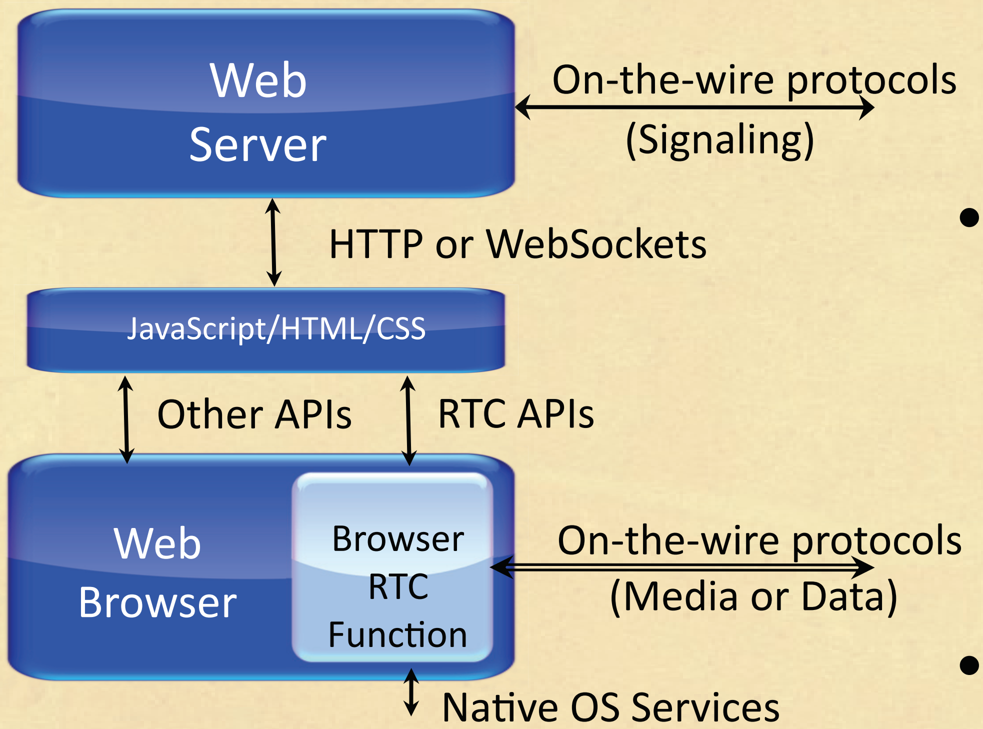
**Cons**

- Flash is single-vendor proprietary and closed
- Losing market share and not available on iOS
- Limited codec and echo cancellation options
- Proprietary development tools

# WebRTC is "Skype in the browser"

- Access to camera and microphone without a plugin
  - No more Flash!
- Audio/video direct from browser to browser
- Why does it matter?
  - Media can stay local
  - Mobile devices eventually dropping voice channel anyway
  - Games

# The Browser RTC Function



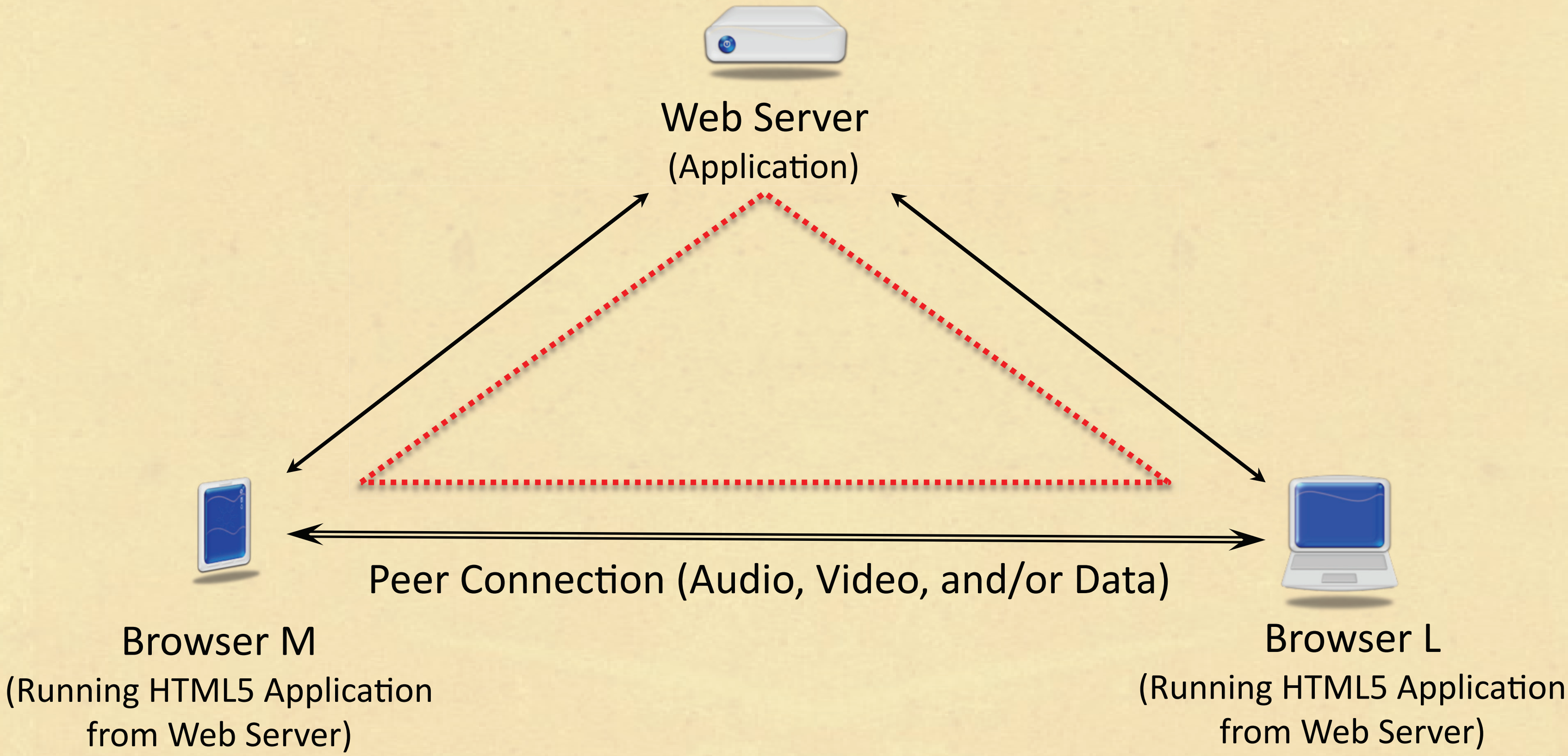

- New Browser Real-Time Communication (RTC) Function built-in to browsers
- Contains
  - Audio and video codecs
  - Ability to negotiate peer-to-peer connections
  - Echo cancellation, packet loss concealement
- In Chrome and Mozilla today, Internet Explorer and Safari eventually

# So What's the Big Deal?

- The web is now a platform for real-time communications development
- Communication will be secure (encrypted) by default
- Latest audio and video codecs means superior quality to anything else
- WebRTC provides peer-to-peer media, even through NATs
- Standard that can interoperate with existing VoIP (Voice over IP), video conferencing, and even PSTN

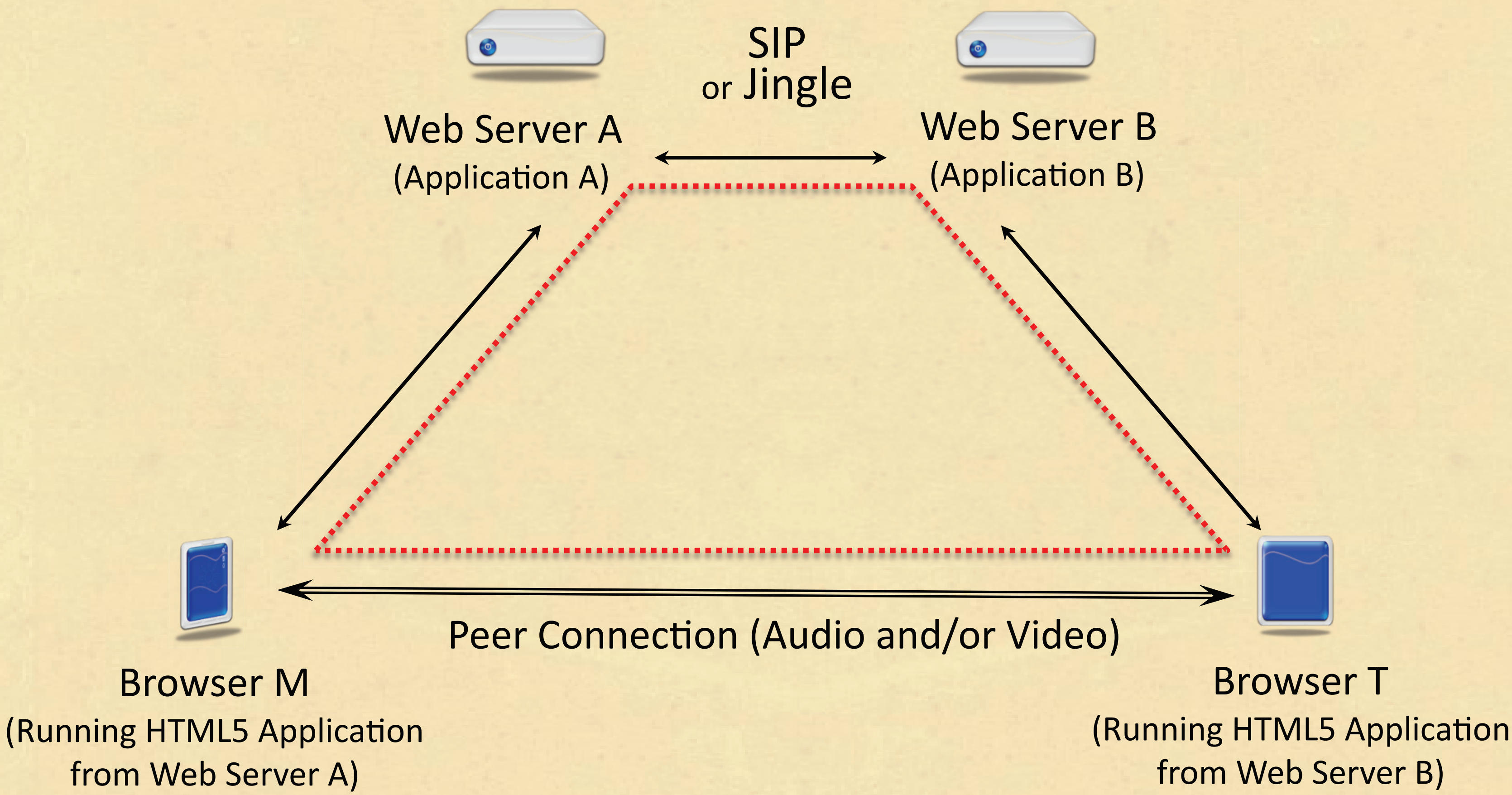
All of these are major innovations for the communications industry

# WebRTC Triangle



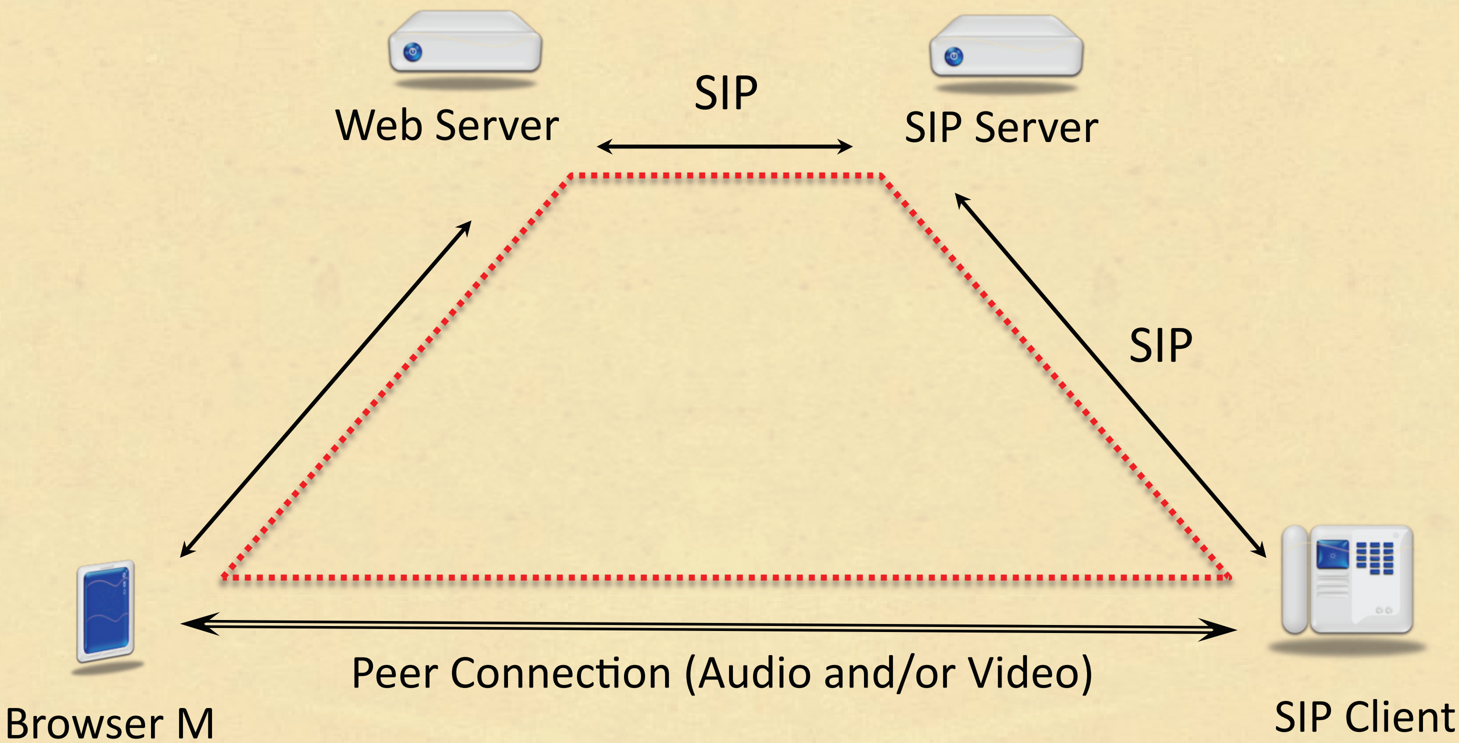

- Both browsers running the same web application from web server
- Peer Connection media session is established between them
- Signaling is not standardized, could be SIP, Jingle, proprietary. Uses HTTP or WebSockets for transport

# WebRTC Trapezoid



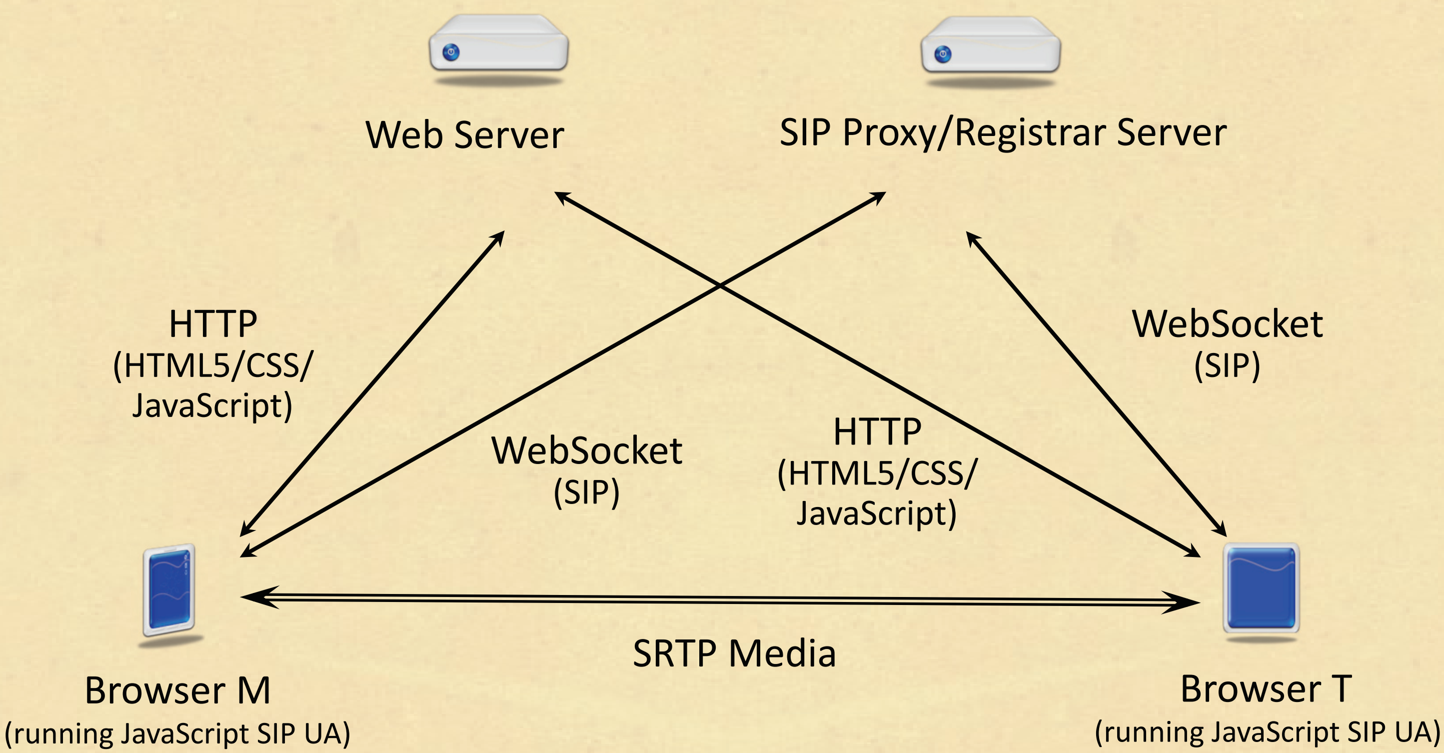

- Similar to SIP Trapezoid
- Web Servers communicate using SIP or Jingle
- Useful for building conventional telephony apps, but unclear how this works in the web world

# WebRTC and SIP



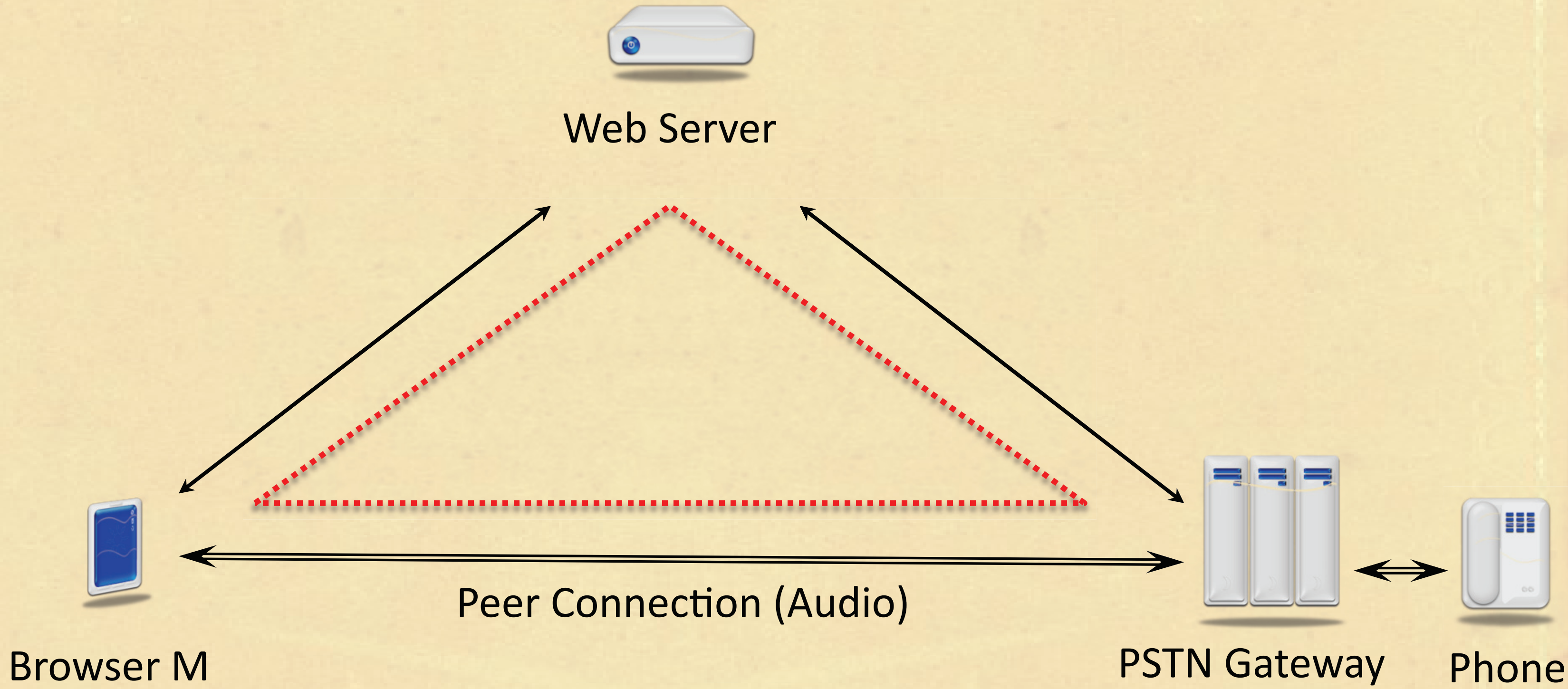

- Peer Connection appears as a standard RTP media session, described by SDP
- Web Server implements a JSEP (JavaScript Session Establishment Protocol) to SIP signaling gateway
- SIP Endpoint must support RTCWEB Media extensions (ICE NAT Traversal, Secure RTP, etc.)

# WebRTC *with* SIP



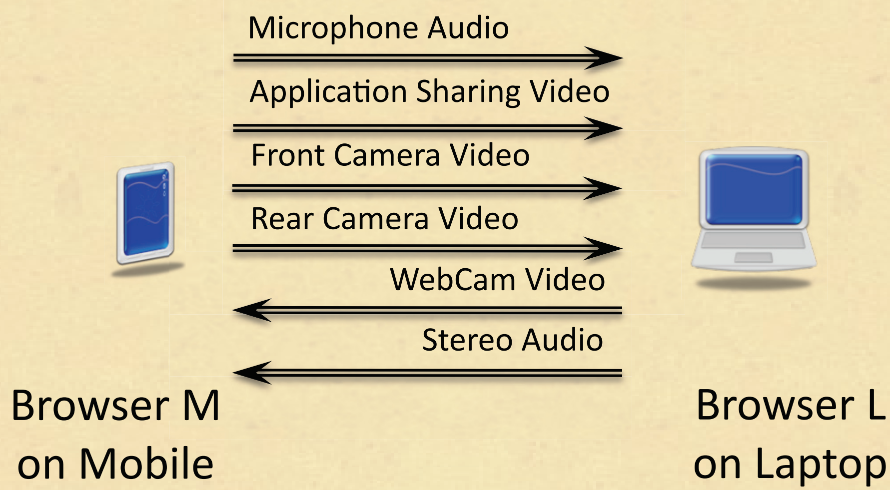

- Browser runs a SIP User Agent by running JavaScript from Web Server
- Secure RTP media connection uses WebRTC APIs
- Details in [draft-ietf-sipcore-websocket] that defines SIP transport over WebSockets

# WebRTC and PSTN




- Peer Connection terminates on a PSTN Gateway
- Audio Only
- Could also use SIP trunking such as SIPconnect 1.1 recommendation

# WebRTC Support of Multiple Media

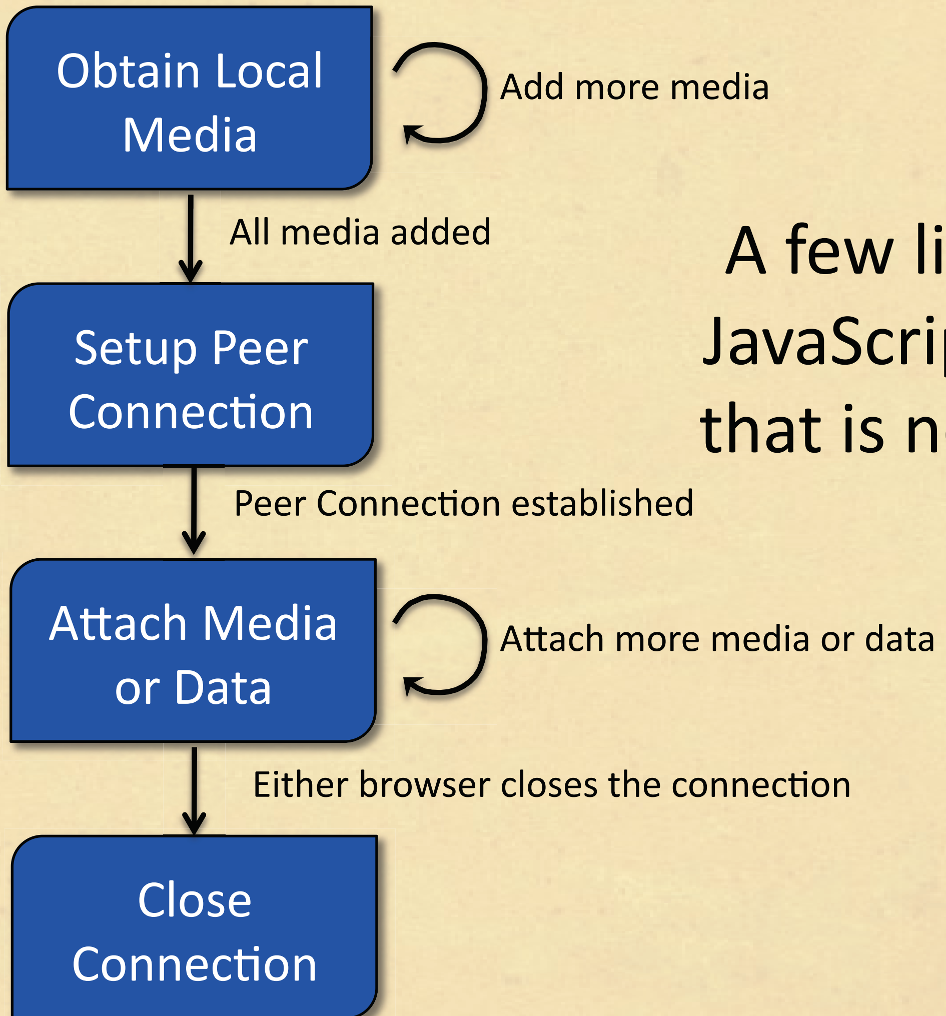



- Multiple sources of audio and video are assumed and supported
- All RTP media, voice and video, and RTCP feedback messages are multiplexed over the same UDP port and address

# How to Use WebRTC

# How to use WebRTC

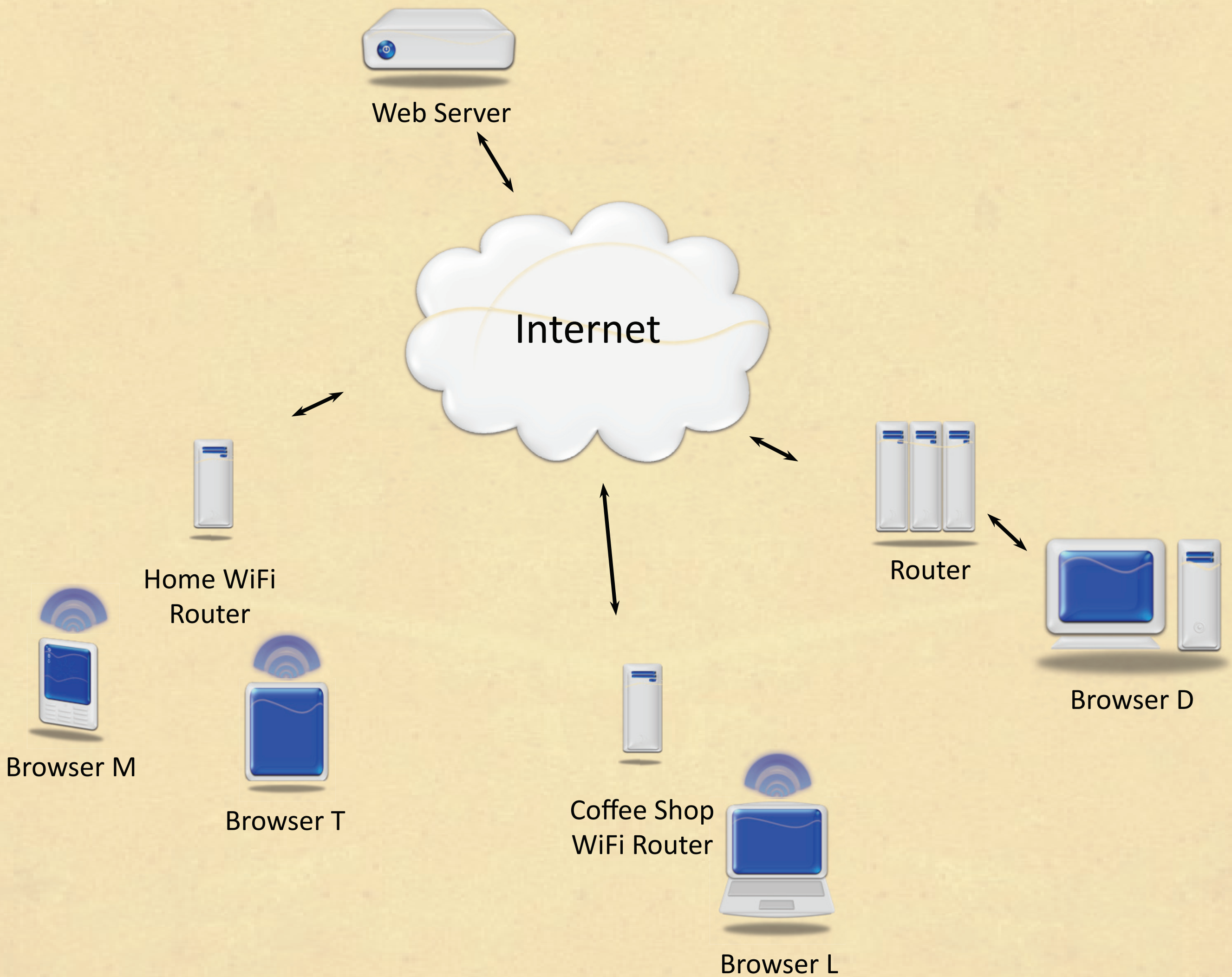
```
getUserMedia()

RTCPeerConnection()

addStream()
createOffer()
createAnswer()
```



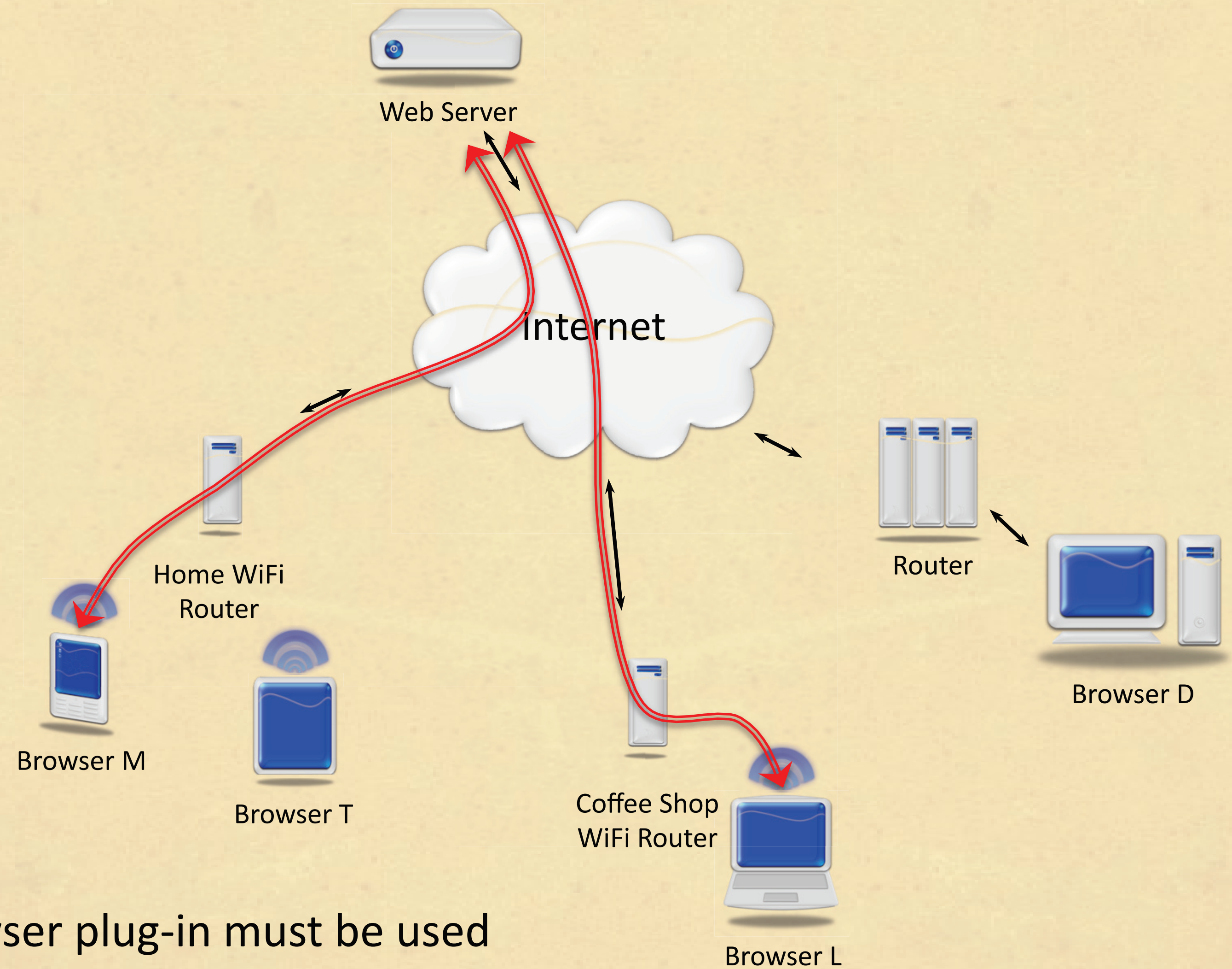

A few lines of JavaScript is all that is needed!

# WebRTC Peer-to-Peer Media

# Media Flows in WebRTC

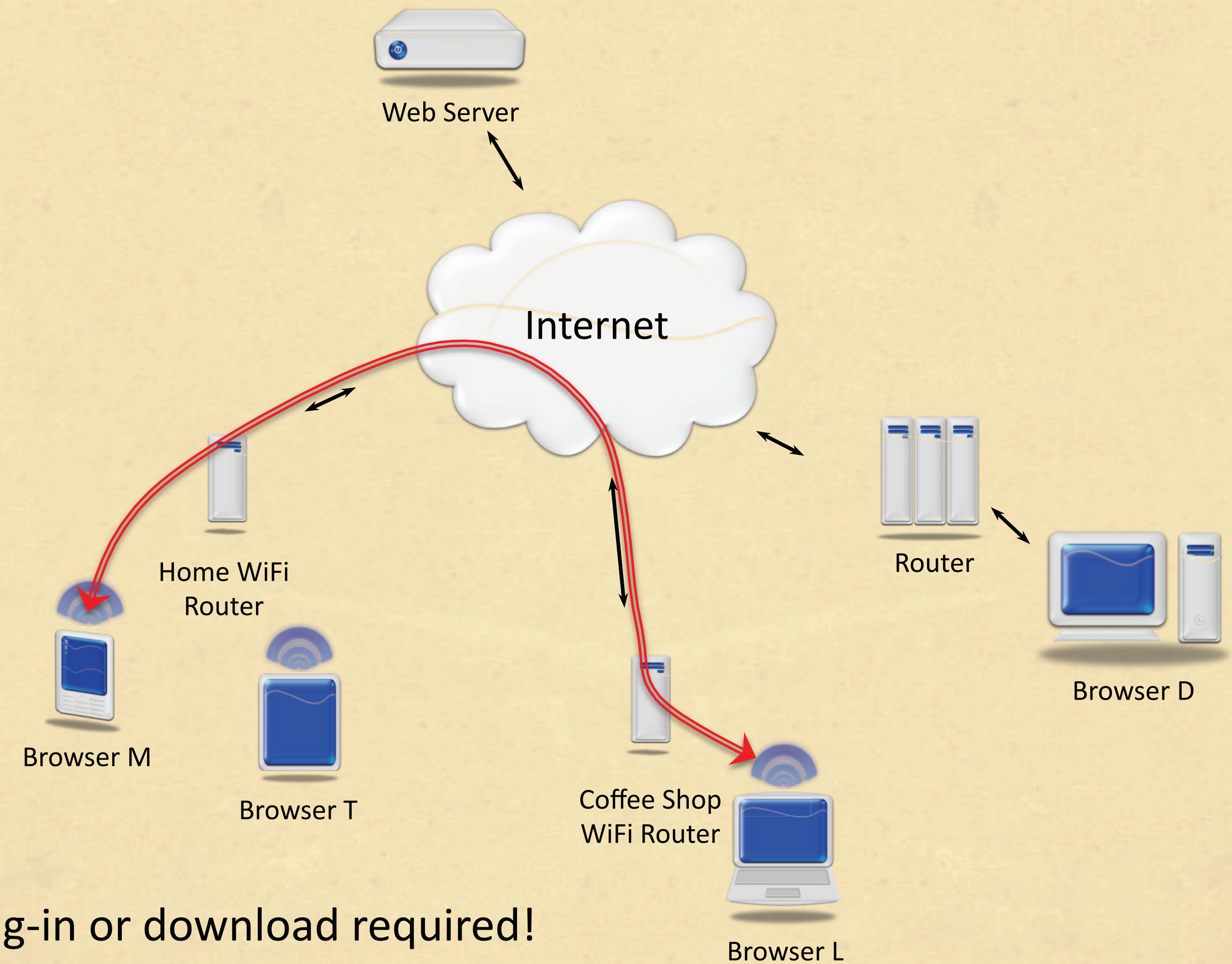


# Media without WebRTC



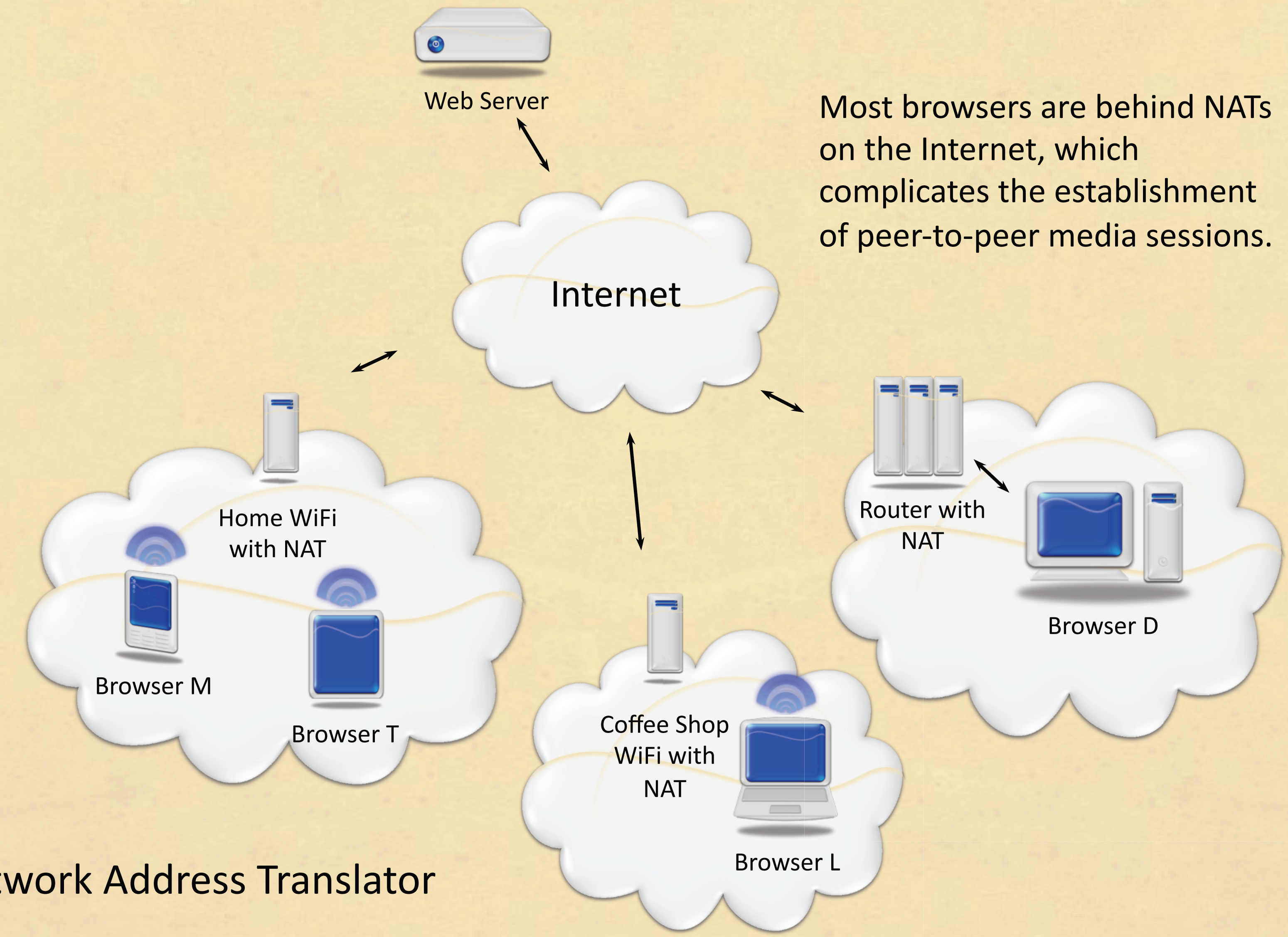

A browser plug-in must be used

# Peer-to-Peer Media with WebRTC




No plug-in or download required!

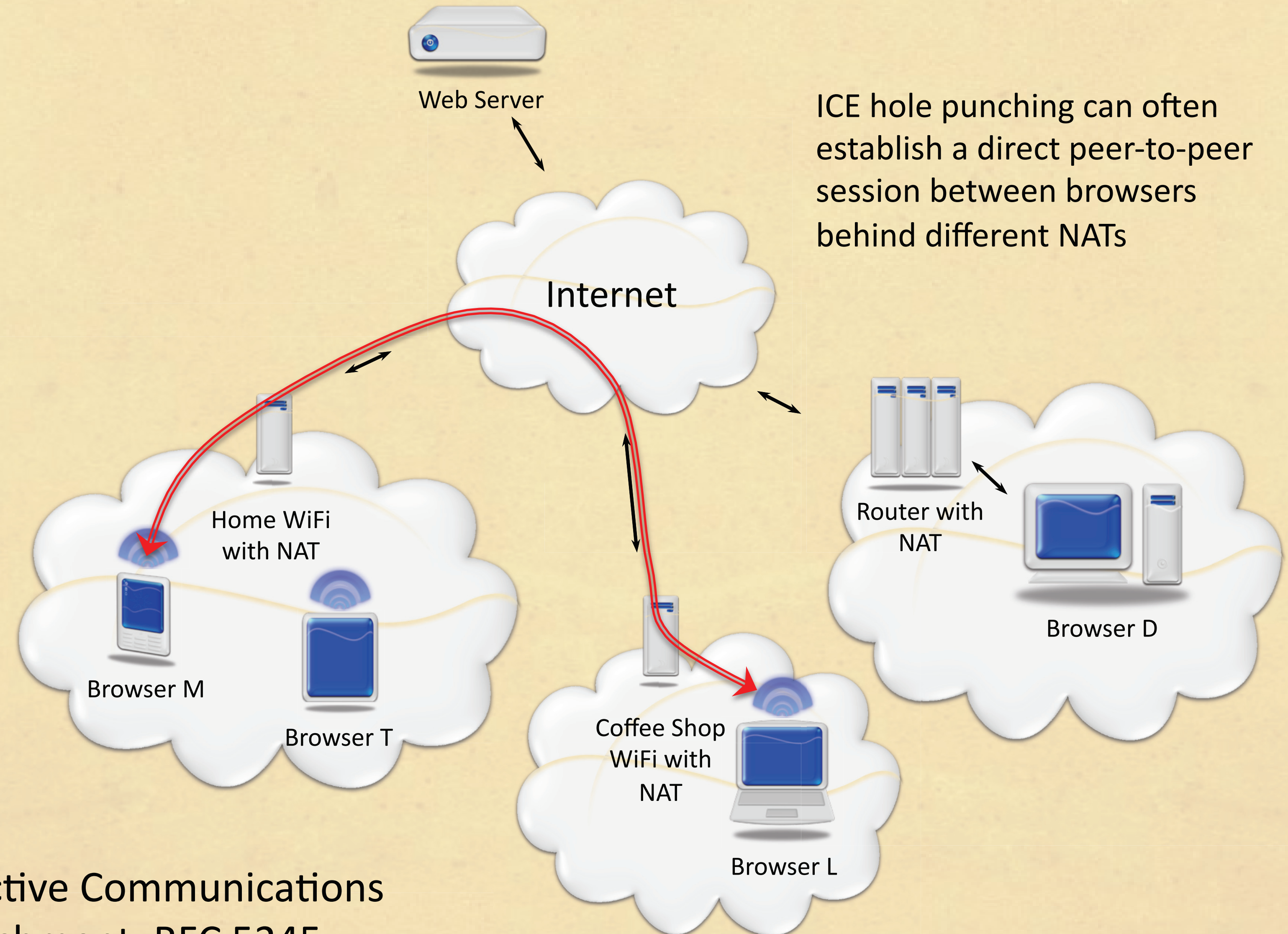
# NAT Complicates Peer-to-Peer Media




Most browsers are behind NATs on the Internet, which complicates the establishment of peer-to-peer media sessions.

Network Address Translator

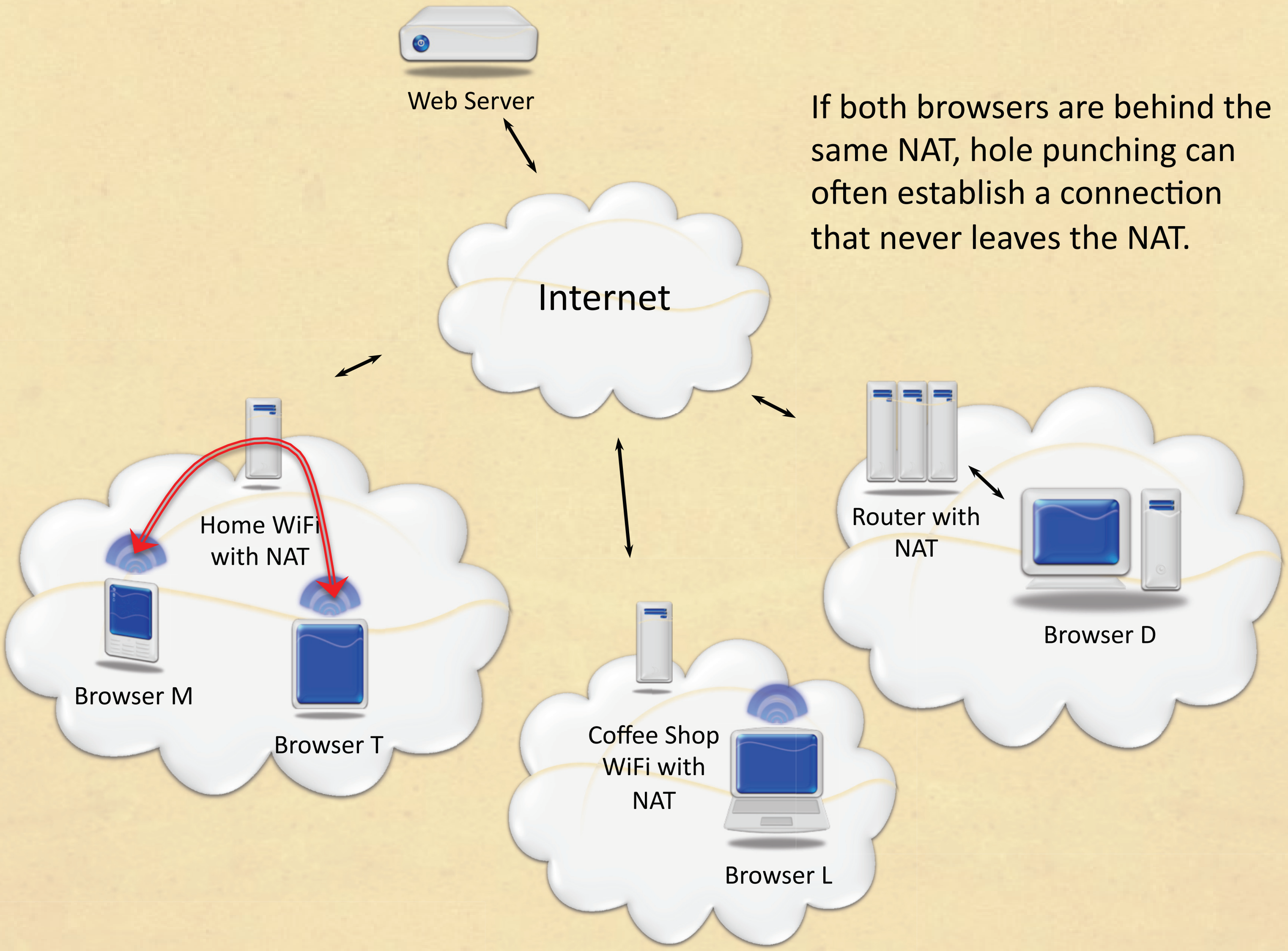
# Peer-to-Peer Media Through NAT




ICE hole punching can often establish a direct peer-to-peer session between browsers behind different NATs

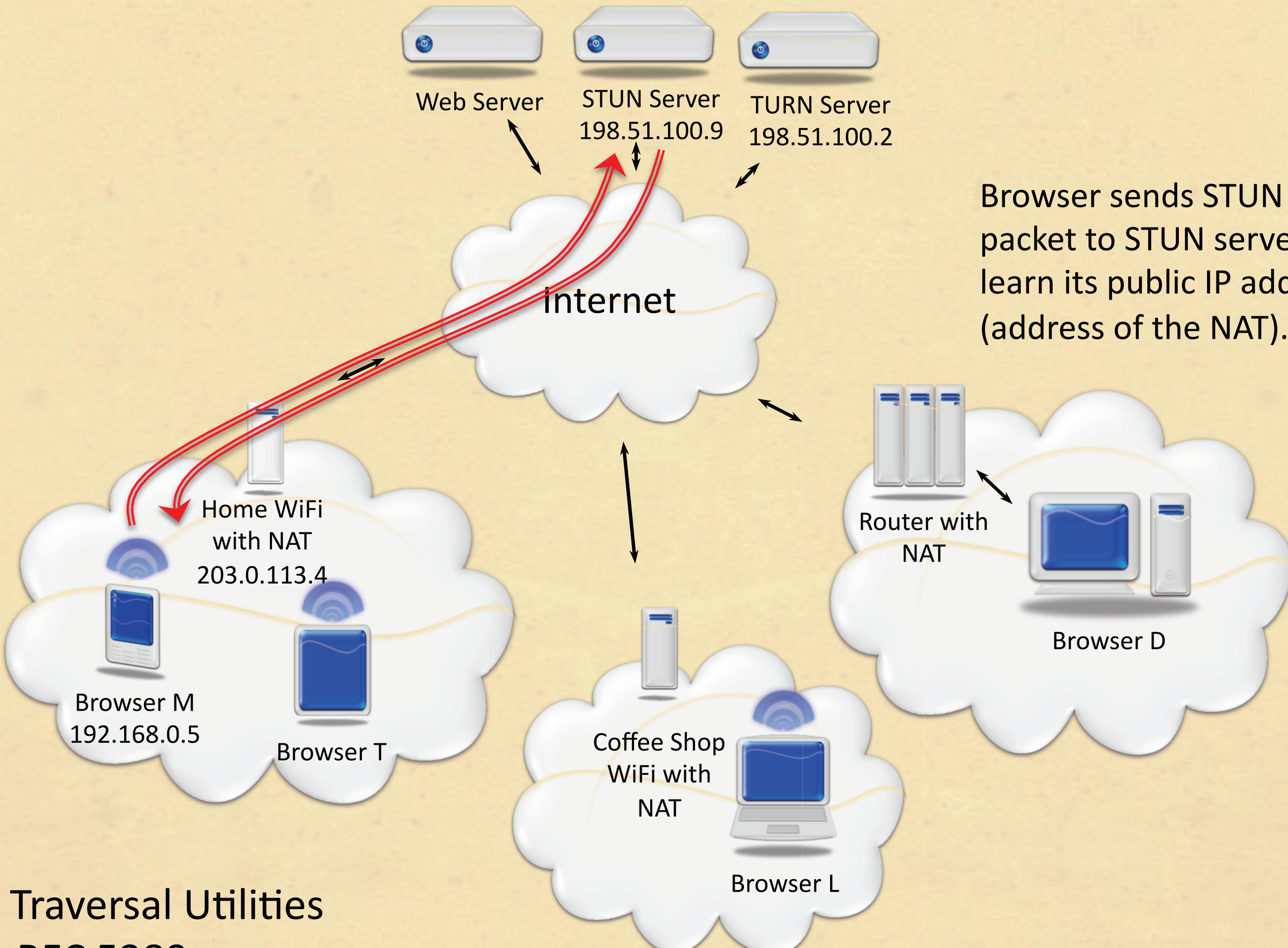
Interactive Communications Establishment, RFC 5245

# P2P Media Can Stay Local to NAT




If both browsers are behind the same NAT, hole punching can often establish a connection that never leaves the NAT.

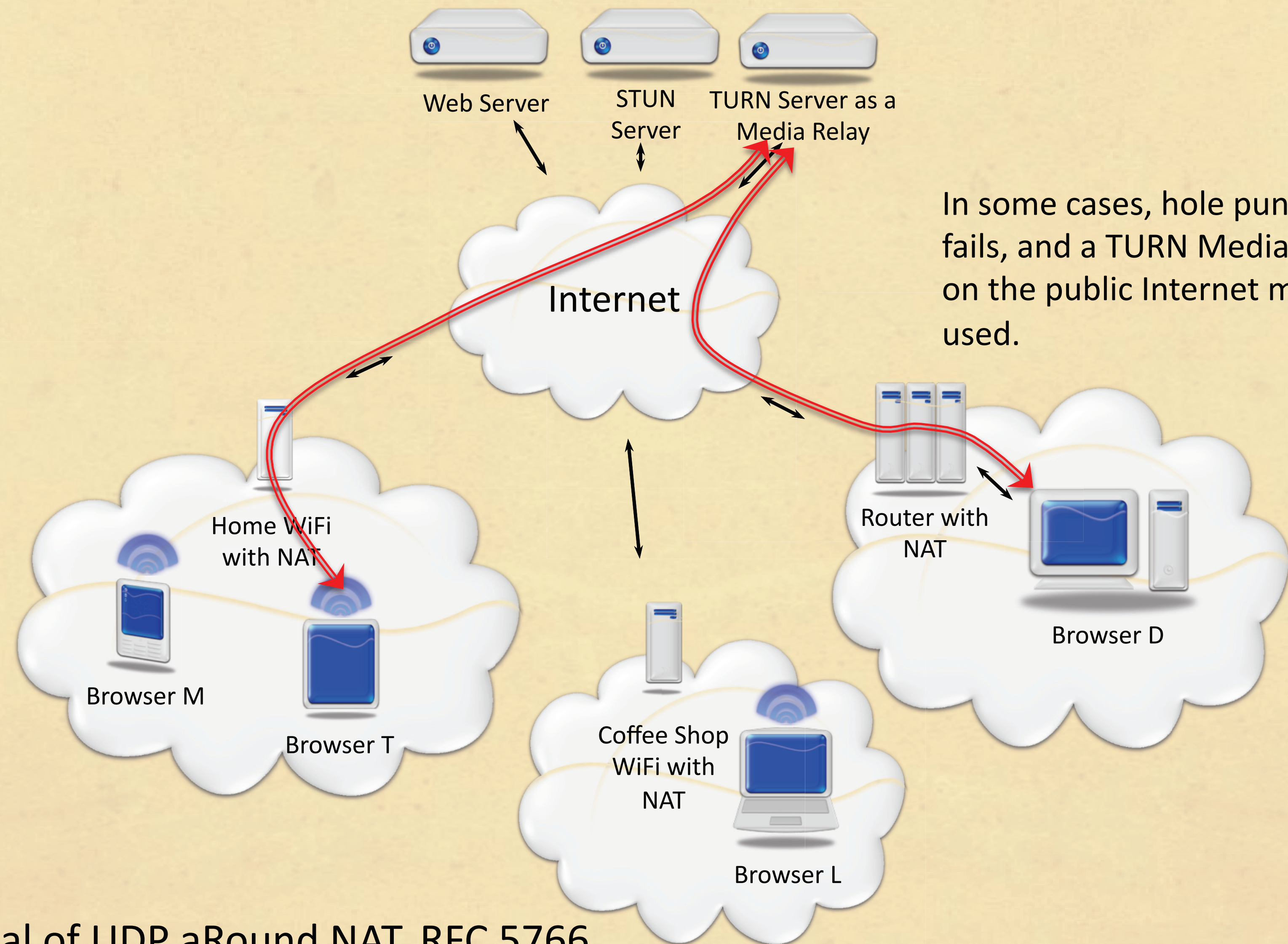
# Browser Queries STUN Server




Browser sends STUN test packet to STUN server to learn its public IP address (address of the NAT).

Session Traversal Utilities for NAT, RFC 5389

# TURN Server Can Relay Media

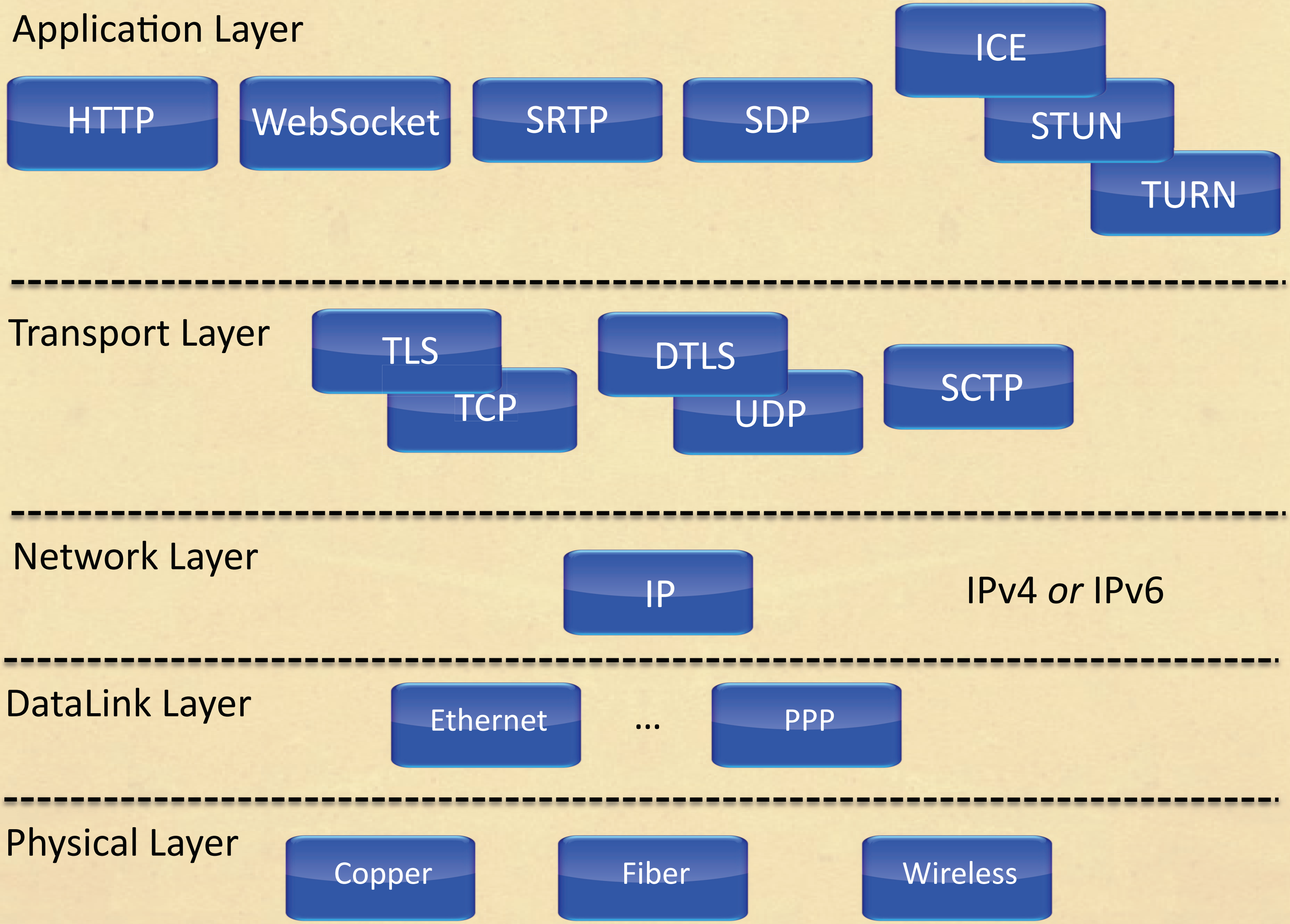



In some cases, hole punching fails, and a TURN Media Relay on the public Internet must be used.

Traversal of UDP aRound NAT, RFC 5766

# WebRTC Protocols and IETF Standards

# WebRTC Protocols



# A Joint Standards Effort

- World Wide Web Consortium (W3C)
  - Standardizing APIs (Application Programming Interfaces)
  - Most work in WEBRTC Working Group
  - Used by JavaScript to access RTC function
- Internet Engineering Task Force (IETF)
  - Standardizing protocols (bits on the wire)
  - Codecs (more on this next)
  - Peer Connection will use RTP, SDP, and extensions
  - Some work in RTCWEB Working Group
  - Lots of related work in MMUSIC, AVTCORE, etc.

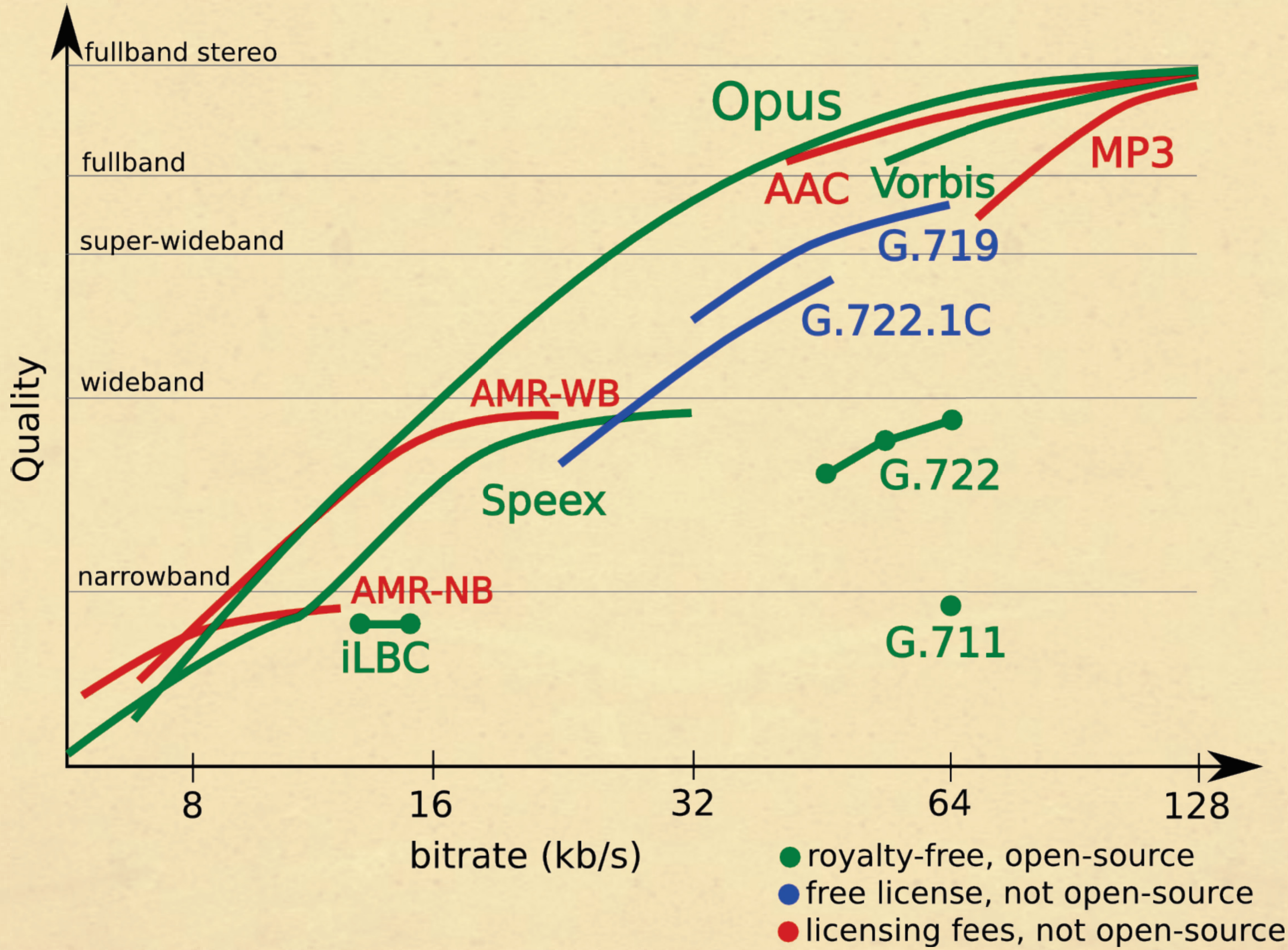
# Opus Audio Codec

### Codec Feature Comparison

| Codec | Sample rate (kHz) | Bitrate (kbps) | Frame size (ms) | Total delay (ms) | Robustness | License |
|---|---|---|---|---|---|---|
| Opus | 8-48 | 6 - 255 (mono)<br>12 - 510 (stereo) | 2.5 - 20 | 5 - 22.5 | packet loss, limited bit errors | Open-source (BSD) |
| Speex | 8<br>16<br>32 | 2.2 - 24.6<br>4 - 42.2<br>4.2 - 44 | 20 | 30 - 35 | packet loss | Open-source (BSD) |
| G.722.1C (Siren14) | 32 | 24, 32, 48 | 20 | 40 | packet loss, bit errors | no charge, but not open-source |
| iLBC | 8 | 15<br>13.3 | 20<br>30 | 25<br>40 | packet loss | Open-source (BSD) |
| AAC-LD | 16 - 48 | 16 - 128 | 10 - 11.6 | 20 - 50+ | packet loss | proprietary, MPEG |

From http://opus-codec.org/comparison/

# Audio Codec Comparison




http://opus-codec.org/comparison/quality.png

# Standard Codecs in WebRTC

| Codec | Use | Specification |
|---|---|---|
| Opus | Narrowband to wideband Internet audio codec for speech and music | RFC 6716 . |
| G.711 | PCM audio encoding for PSTN interworking and backwards compatibility with VoIP systems | RFC 3551 |
| Telephone Events | Transport of Dual Tone Multi Frequency (DTMF) tones | RFC 4733 |
| H.264 | Video codec requiring licensing | RFC 6184 |
| VP8 | Open source video codec | RFC 6386 |

- Mandatory to Implement (MTI) audio codecs are settled on Opus and G.711 (finally!)
- Video is not yet settled
  - H.264 vs VP8 fight is ugly

# WebRTC and the Enterprise

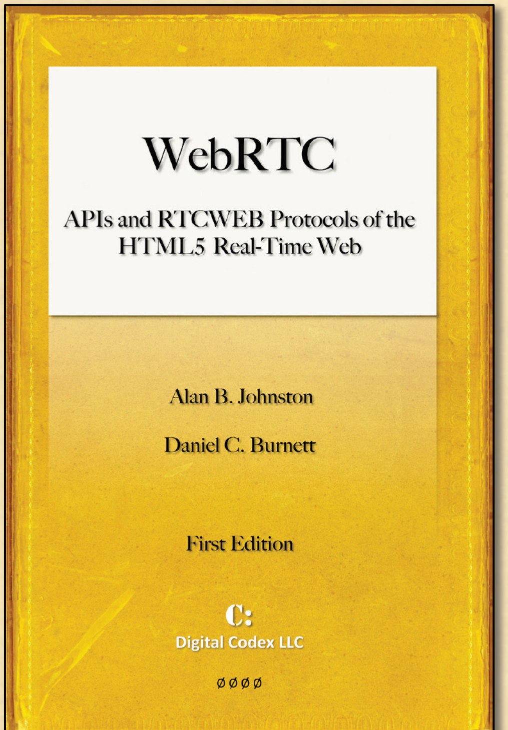
- Enterprise has unique requirements on WebRTC
- Security
  - Firewall traversal
  - Access control
  - Peer-to-peer data flows
- Compliance
  - Recording & logging
  - Policy compliance
- Integration with existing infrastructure
- New element proposed:
  - “Secure Edge” located in enteprise DMZ

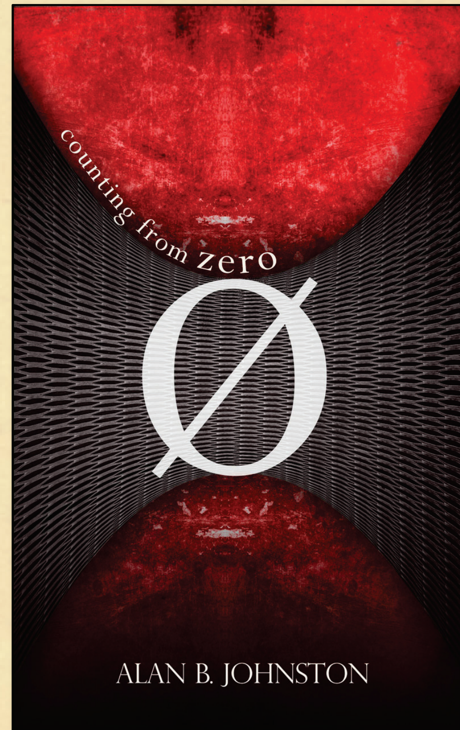
See our article “Taking WebRTC to the Enterprise” in April IEEE Communications Magazine

# What's Next?

- W3C and IETF standards still need to be finalized (early 2014)
- Browsers need to add support
  - Chrome browser has much of this functionality now!
  - Firefox will have shortly (in nightly builds)
  - Mobile browsers need to support
    - In Android Beta now
- Mandatory to Implement video codec needs to be decided
- Enterprise use of WebRTC need to be worked out

# Questions?




https://webrtcbook.com

https://countingfromzero.net

# References

1. C. Davids, A. Johnston, K. Singh, H. Sinnreich, W. Wimmreuter, "SIP APIs for Voice and Video Communications on the Web", Principles, Systems and Applications of IP Telecommunications (IPTcomm), Chicago, IL, Aug 2011
2. A. Johnston, D. Burnett, “WebRTC: APIs and RTCWEB Protocols of the HTML5 Real-Time Web”, Digital Codex, St. Louis, MO, 2012
3. A. Johnston, J. Yoakum, K. Singh, “Taking on WebRTC in an Enteprise, (to appear in) IEEE Communications Magazine, Vol 51, No. 4, April 2013