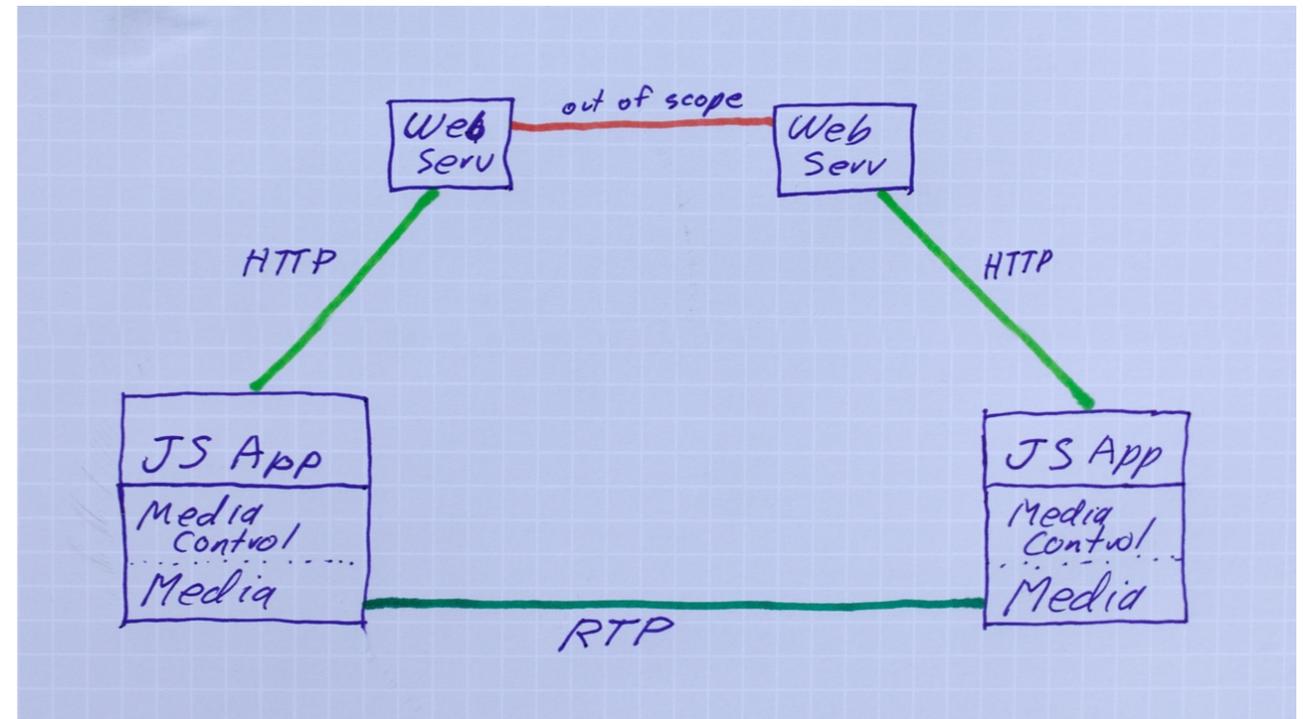


# Media Negotiation

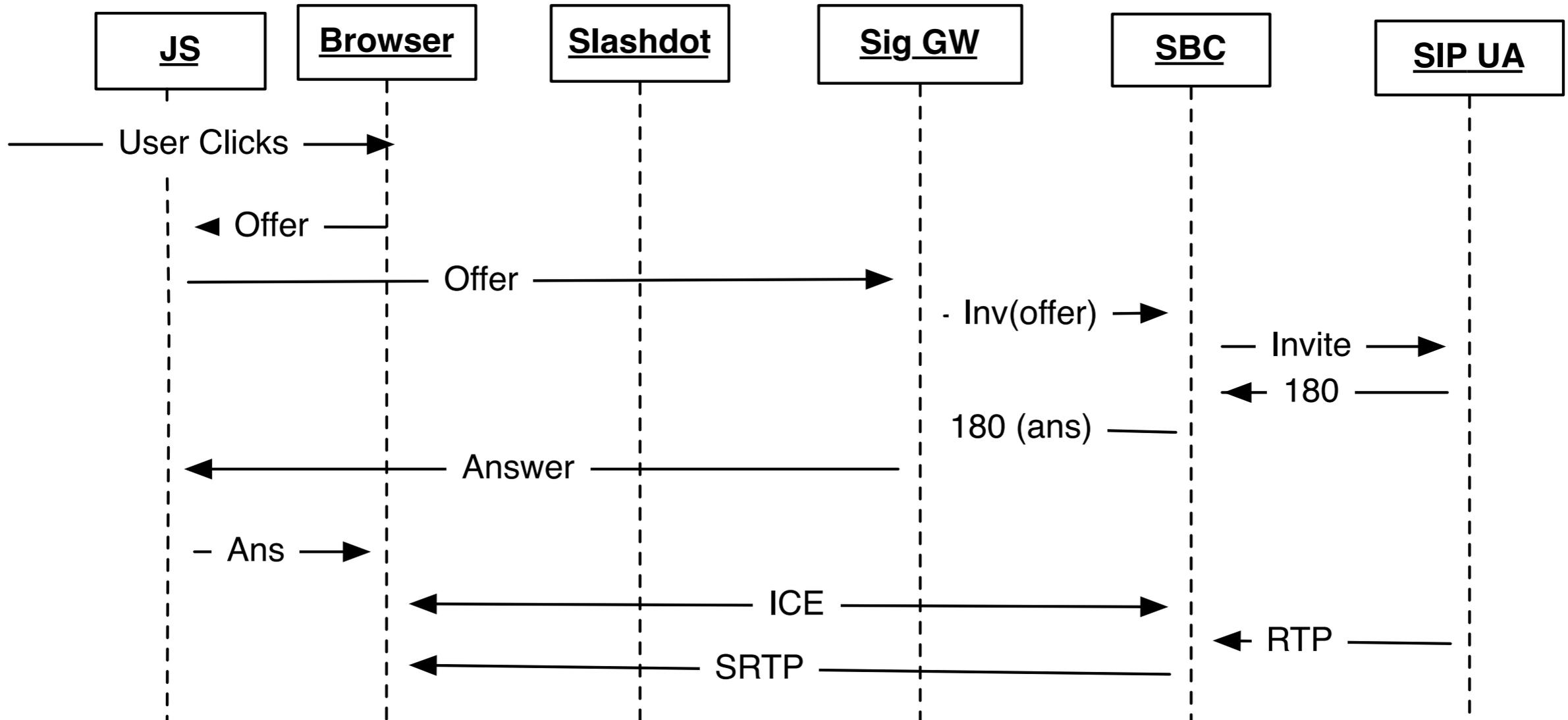
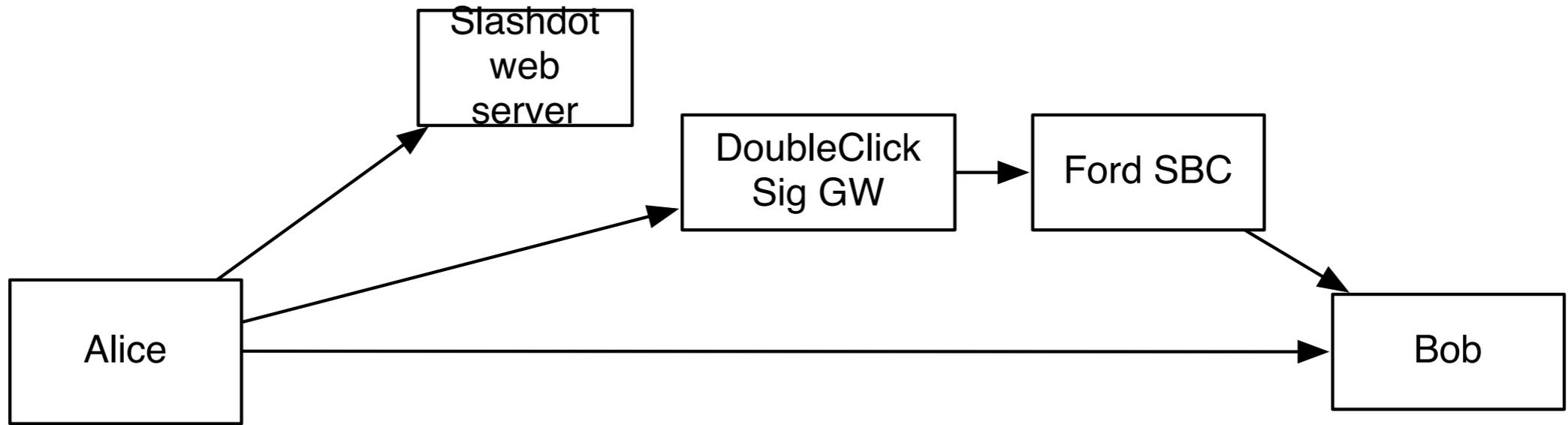
Cullen Jennings, July 2011

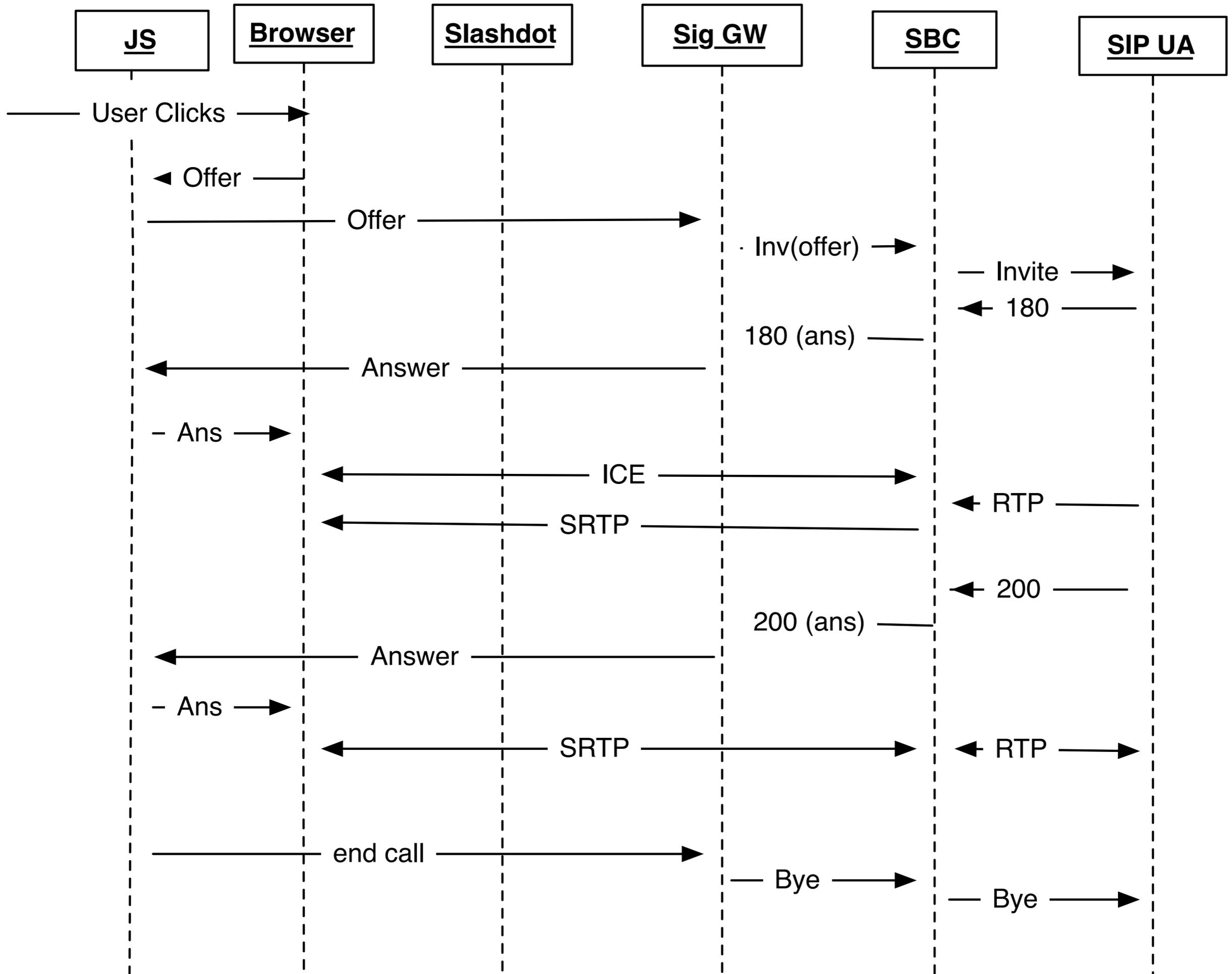
# What we seem to agree on



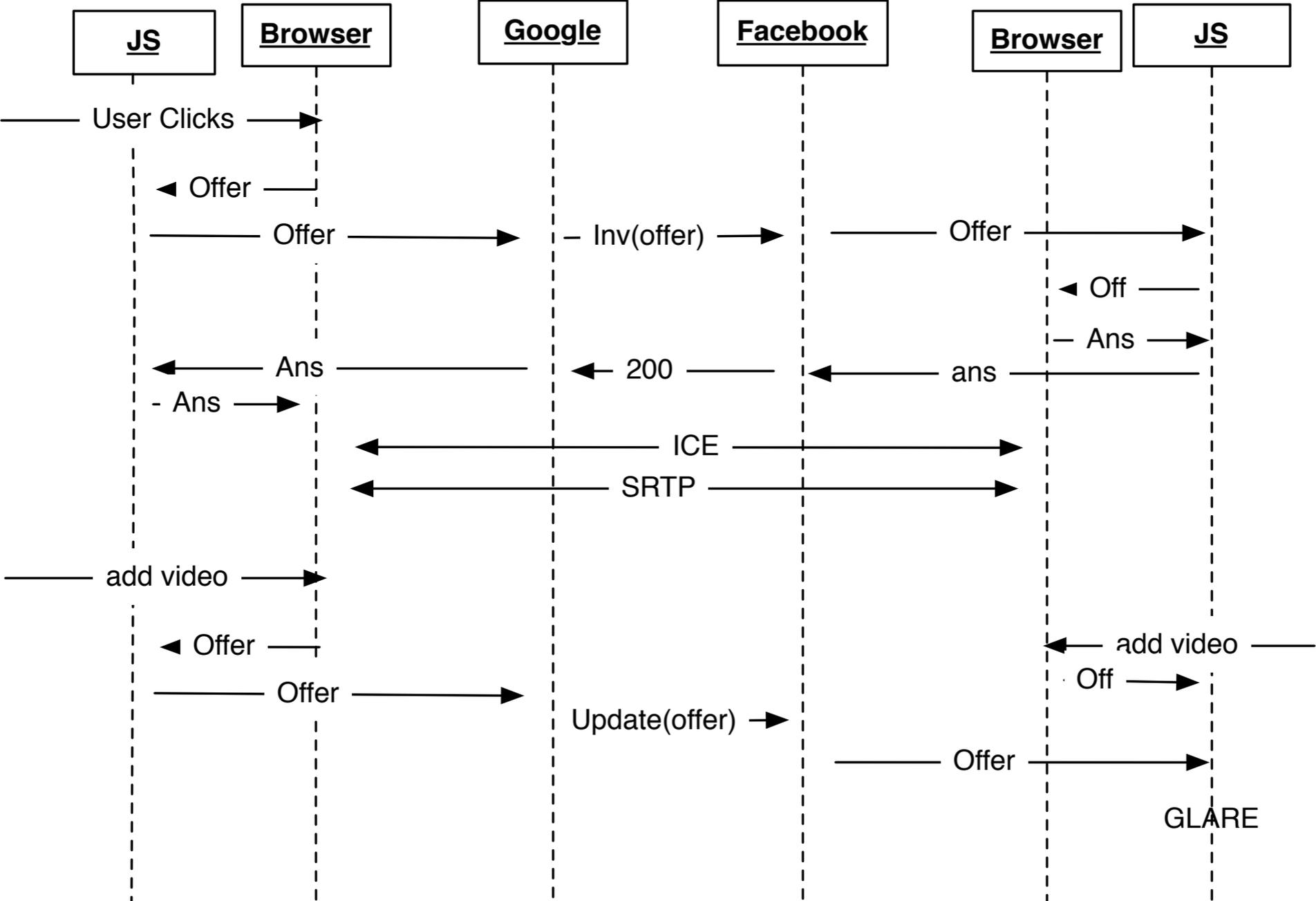
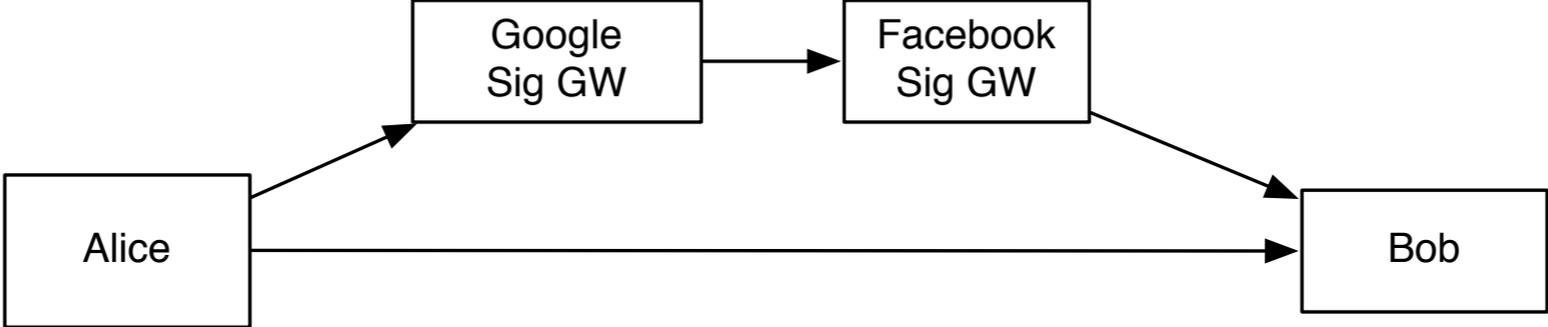
- We agree there is some RTP and/or SRTP from one browser to the other ( green line on bottom)
- We agree there is some HTTP and if there is more than one web server, they can choose to use SIP to talk to each other
- We agree there is media codecs with parameters that need coordination between the two sides.
- Question is what path does the media negotiation information flow along and what does it look like. Two broad classes of solutions..
  - Device provide API to report capabilities and another to be told what media to send and receive (Advertisement / Proposal). Not discussed today
  - Reuse SIP Offer/Answer negotiation

# Call from Slashdot to Ford





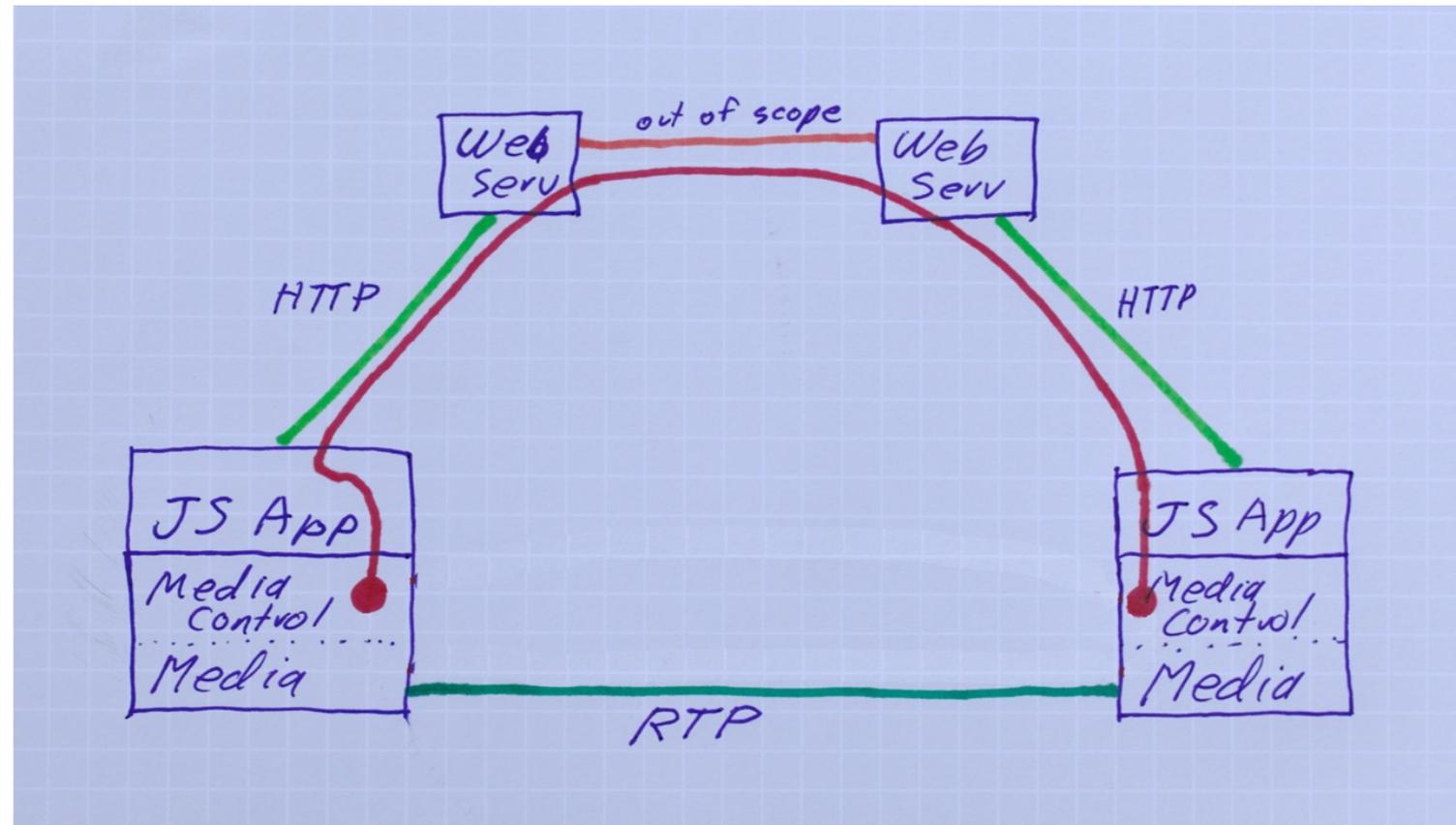
# Call from Google to Facebook



# Requirements

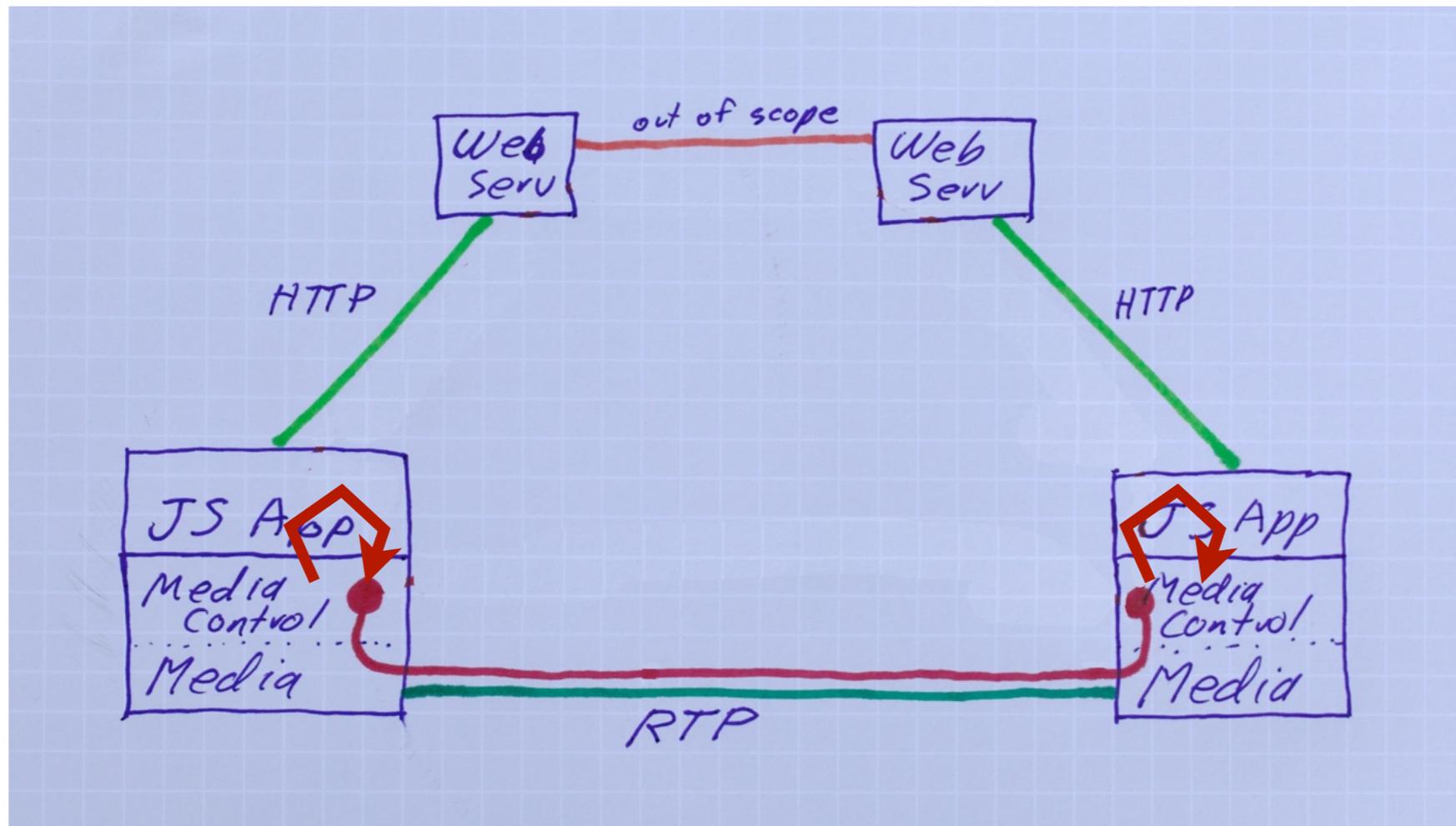
- Be able to pass SDP offer / answers
- Be able to indicate context of interpreting SDP as offer or answer and what other pair it maps to
- Be able to tie the SRTP crypto to the identity of who you are talking to
- Be able to deal with SIP two phase media commit 180 / 200
- Be able to signal errors in SDP interpretation and glare
- .... and a bunch more along these lines .....

# Option: High Path



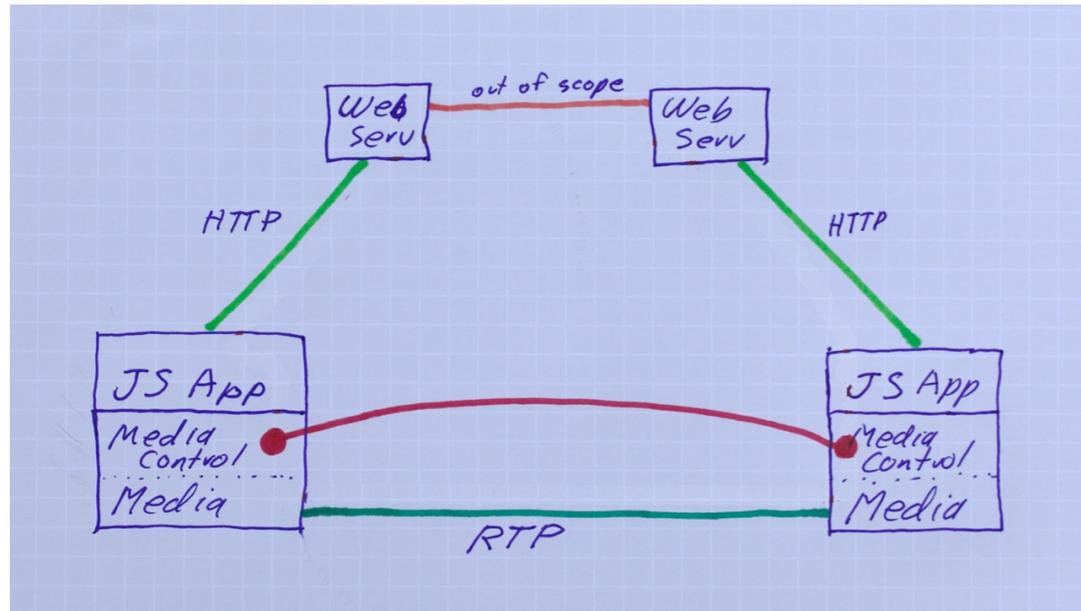
- In this model, what is standardized is an interface from the core part of the browser to the JavaScript application running in the browser
- The JS App takes care of transporting this in whatever formats it wants to the far side
- In the case where there are two federated web servers as shown in the diagram, it gets complicated to map to SIP on orange line depending on what happened on light green line.

# Option: Low Path

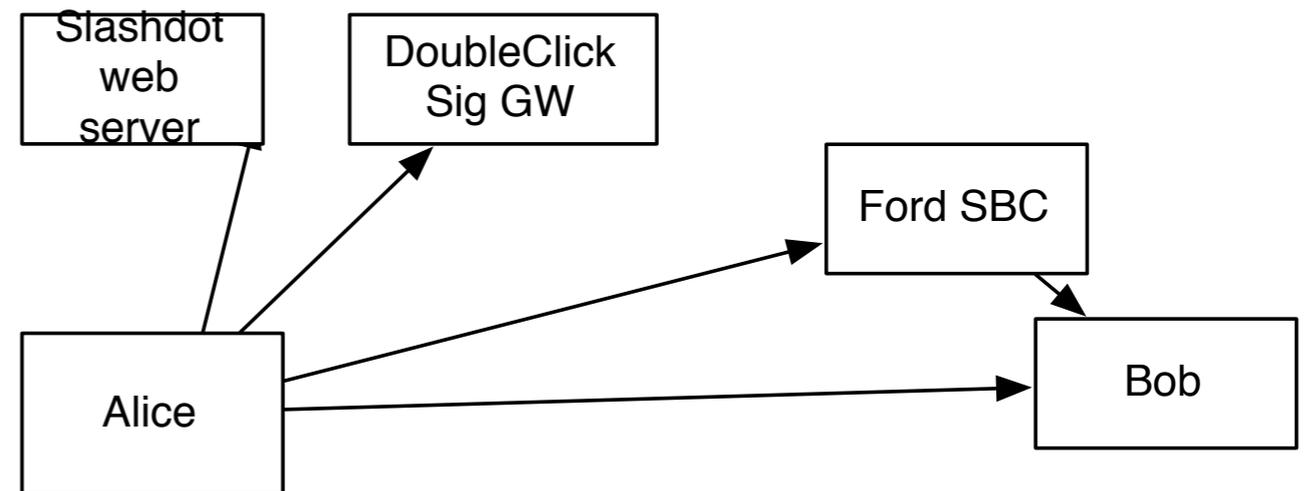


- In this model, the HTTP / orange path communicates enough information that the two browsers can perform ICE. and set up a channel between the browser
- This channel is then used to exchange the media signaling information
- Can be done with current API proposal by using high path with SDP to set just a data channel, then using data channel loop the signaling between peerConnection objects

# Option: Mid Path



Call from Slashdot to Ford



- In this model, the browser speaks a first class signaling protocol such as Jingle or SIP and it can use that to negotiate media with the far end
- The HTTP green/orange path is used only to find the address of who you want to communicate with
- This makes some things like identity for the purposes of authorizing access to the microphone easier

# Choices we need to make soon

- SDP Offer/Answer or not
  - Path A is to use SDP Offer/answer
  - Path B is to create something new to replace Offer/Answer and then mapping from it to Offer/Answer
- Assuming path A, then questions is how to carry the SDP in a protocol to make it be able to be used as offer/answer (the things in the previous requirements slide)
  - Path A1: Specify a new protocol in the form of a bunch of data that needs to be passed along with the SDP to make it be able to do offer/answer
  - Path A2: Steal the parts of SIP (or Jingle) used to do offer / answer and use theses as a container for the SDP. This would take the offer/answer part of SIP and toss out the rendezvous and subscribe / notify parts of SIP.