Push Notifications for WebRTC

WebRTC WG Face to Face meeting
Mountain View, CA Jan 2012
Dan DRUTA
AT&T
Problem Statement

• Provide unsolicited notifications for incoming connection requests in WebRTC clients
• Make notifications available to the user irrespective if the app is running, in focus or not.
• Provide notifications irrespective of the browser or network environment
Use cases

• Bob uses Ringo’s STAR webrtc service on his mobile device. Bob calls Alice at work and leaves a voice mail for her to call him back. Alice comes back from her meeting and calls Bob. Bob receives an incoming call. He can see it’s Alice and answers.

• Alice is using her browser on her laptop to chat with her mom overseas. She goes online and marks her status available. Then she closes her window to work on a presentation due next week. Her mom wants to tell her the great news about her cousin's new baby and calls her. Alice receives a notification that her mom wants to talk and answers
Factors to be considered

• Maintaining multiple connections is expensive on the battery for most mobile devices
• Web Development effort has to be minimized and the complexity of the API use should be abstracted
• Web Apps Working Group has an API for Server-Sent Events
• Existing frameworks and technologies for connectionless push exist and will continue to be used
WebRTC Client Scenarios

• User online, available and with the app running in focus
• User online, available and app in the background
• User available and the app closed
Options for the WebRTC WG

• Leave notifications out of scope – Service provider specific
  – Con: Silo effect and interoperability issues
  – Pro: No work for the group

• Leverage existing signaling protocols
  – Pro: Leverage Service ready architectures
  – Con: It still requires implementers to choose which protocols to support and provide wrapper libraries for these protocols

• Leave notification protocol out of scope and focus on the client. Lightweight abstract mechanism and API definition for clients
  – Pro: Consistent with W3C focus on data formats and APIs
  – Con: Protocol aware service
Focus on the Client (option 3):

- Register a URI scheme or data format (MIME) and associate the browser as the handler
- Define a specific structure for the notification that would be passed
- Define an API to parse the structure
- API to attach to the notification service