WebRTC in WebKit
2015

What is WebRTC?

- A set of APIs and protocols to enable real-time audio-visual communication natively in a browser
  - Standardized in W3C and IETF
- Main Components
  - GetUserMedia()
    - Get access to the user’s devices
  - MediaStream API
    - Control real-time media from JavaScript
  - PeerConnection
    - Establish connectivity through firewalls and NATs
    - Send real-time media (and data) to others

OpenWebRTC

- We got some code; why not share it!
- Permissive License (BSD)
- We are on Github

OpenWebRTC

- Built as framework from ground up
- Second independent implementation
- Cross platform (incl. iOS)

WebRTC (.org)

- Interoperability:
  - Chrome
  - Firefox

gstreamer

- Based on GStreamer
- Hardware codecs (OSX and iOS a.t.m.)

- Internal projects:
  - Remote excavator
  - Back office support with Google Glass
- Next steps: WebKit

- Video: H.264 & VP8
- Audio: OPUS & G.711

WebRTC API Updates

- Promises
  - MediaDevices - unprefixed getUserMedia() and more
  - RTCPeerConnection – overloaded functions
- Functionality move
  - From MediaStream to MediaStreamTrack
- Renaming
  - MediaStreamSource → RealTimeMediaSource
- No more Audio/VideoStreamTrack
- HTMLMediaElement.srcObject
  - Implications on MediaPlayer (platform)
- Constraints

MediaEndpoint

- Powers RTCPeerConnection
- Is not RTCPeerConnection
- Configure how to send and receive
- Can power different JS API Alternatives

Stuff to talk about

- Rendering (OpenWebRTC)
  - GStreamer based MediaPlayer dependency (step 1)
  - Generic rendering (step 2)
- Promises
  - Two signatures, one implementation
  - Automatically generated bindings
- Other ports (not using GStreamer)
  - Rendering