Introduction to HTML5 & WebRTC

Juan Quemada

WebRTC and Q4S Hackathon
Madrid, 23 October 2012
The Web platform is evolving

And It’s Solving Key Developer Challenges

From Brad Neuberg, Google, http://www.slideshare.net/jobfan/google-html5-tutorial
The Web Platform is Accelerating

From Brad Neuberg, Google, http://www.slideshare.net/jobfan/google-html5-tutorial
JavaScript gets more efficient

More Speed

100x improvement in JavaScript performance

From Brad Neuberg, Google, http://www.slideshare.net/jobfan/google-html5-tutorial
Web as Platform!!

- **Consolidation of HTML5 platform**
  - HTML + CSS + JavaScript
    - [http://www.html5rocks.com/](http://www.html5rocks.com/)

- **The Web browser is getting a net O.S.**
  - Websockets, SSE and other protocols
  - Vector graphics y formula with SVG & MathML
  - Games and 3D with Canvas and WebGL
  - Fantastic interfaces with CSS3
  - Video and Audio
  - WebStorage and Data Bases inside the browser
  - Access to system resources with JavaScript APIs
  - ......
  - Real time Audio y Video with WebRTC
The Cloud

Internet -> is the computer

- Client apps in the **Browser** interacting with
- Server apps in **CDC (the cloud)** using

**Protocols of the HTML5 platform**
- HTTP 1.1 (2.0 starting)
- Websockets
- XMLHttpRequest
- Server side events
- PeerConnection
- ........
HTML5 + WebGL + CSS3

https://webglmeeting.appspot.com
WebRTC history

- Development starts in WHATWG (2009 - 2010)
  - Objective: introduce VoIP to the Web

- W3C, IETF, .. starts activity in 2011
  - W3C: Browser APIs
    - [http://dev.w3.org/2011/webrtc/editor/webrtc.html](http://dev.w3.org/2011/webrtc/editor/webrtc.html)
  - IETF: Communication Protocols

- WebRTC supported now in Chrome
  - Before end of 2012: support for Firefox, Opera, Explorer
    - Mobile device support expected in 2013
PeerConnections

- **Peerconnections**
  - UDP/RTP streams between browsers
    - with low latency
- **Signalling with SDP configurations**
  - But using proprietary protocols
APIs JavaScript de WebRTC

- **getUserMedia**
  - Local stream management: camara, microphone, ...

- **StreamAPI**
  - Management of multimedia streams

- **PeerConnection**
  - UDP/RTP streams between browsers

```javascript
navigator.webkitGetUserMedia("video,audio", gotStream, gotStreamFailed);

function gotStream(stream) {
    var url = webkitURL.createObjectURL(stream);
    document.getElementById("localView").src = url;
    trace("User has granted access to local media.");
    localStream = stream;
}

function gotStreamFailed(error) {
    alert("Failed to get access to local media.");
}

function doCall() {
    createPeerConnection();
}
```
Proyecto Software Libre (C++):
Apoyado por: Google, Mozilla, Opera

WebRTC is a free, open project that enables web browsers with Real-Time Communications (RTC) capabilities via simple Javascript APIs. The WebRTC components have been optimized to best serve this purpose. This website is owned and driven by Chrome's WebRTC team.

**Our mission:** To enable rich, high quality, RTC applications to be developed in the browser via simple Javascript APIs HTML5.

**Our current milestone:** To iterate on our first implementation and use web developer feedback to improve the WebRTC API.

The WebRTC initiative is a project supported by Google, Mozilla and Opera.

---

**Web developers**

getUserMedia, which lets you grant web apps access to your camera and microphone without a plug-in is now available to all Chrome users.

PeerConnection, the other half of WebRTC, is now available in Chrome 23 Beta.

To get you familiarized with the technology and coding, watch this great introduction.

We encourage web developers planning to integrate audio and video chat capabilities to try it out and give us feedback.

---

**Browser developers**

There are various ways to participate in this initial phase:

- Help us improve the W3C WebRTC API draft
- Participate in the IETF Rtcweb workgroup
- Join the W3C WebRTC workgroup
- Join our discussion list
- Engage us on Google+
- Follow us on Twitter: @webrtc
Arquitectura Web RTC

The web

Your browser

WebRTC

WebRTC C++ API (PeerConnection)

Session management / Abstract signaling (Session)

Voice Engine

iSAC / iLBC Codec
NetEQ for voice
Echo Canceler / Noise Reduction

Video Engine

VP8 Codec
Video jitter buffer
Image enhancements

Transport

SRTP
Multiplexing
P2P STUN + TURN + ICE

Audio Capture/Render

Video Capture

Network I/O

API for web developers
API for browser makers
Overrideable by browser makers
AudioEngine

- **iSAC**
  - Wideband audio codec for VoIP and streaming audio.
  - 16 kHz or 32 kHz sampling, adaptive bit rate: 12 to 52 kbps

- **iLBC**
  - Narrowband codec for VoIP and streaming (RFCs 3951, 3952)
  - 8 kHz sampling, 5.2 kbps (20ms frames), 13.33 kbps (30ms fr.)

- **NetEQ** for Voice
  - Dynamic jitter buffer and error concealment algorithm used for concealing the negative effects of network jitter and packet loss.
  - Keeps latency low, while maintaining the highest voice quality.

- **Acoustic Echo Canceler (AEC)**
  - Software signal processing component, for real time removal of acoustic echo of voice feeding into the active microphone

- **Noise Reduction (NR)**
  - Software signal processing component removing background noise usually associated with VoIP, such as Hiss, fan noise, etc.
VideoEngine

VideoEngine is a framework video media chain for video, from camera to the network, and from network to the screen.

◆ VP8
  ▪ Video codec from WebM Project.
    ◆ Well suited for RTC as it is designed for low latency

◆ Video Jitter Buffer
  ▪ Dynamic Jitter Buffer for video.
    ◆ Conceals effects of jitter and packet loss on overall video quality

◆ Image enhancements
  ▪ For example, removes video noise from the image capture by the webcam
El ejemplo intercambia información a través de Servidor Web (chat)
  - Que luego es utilizada en ICE/TURN/STUN
    - Para establecer la conexión a través de NAT

ICE
  - Interactive Connectivity Establishment. RFC 5245
    - Establishing connectivity over NATs with STUN/TURN servers

TURN
  - Traversal Using Relays around NAT (TURN), RFC 5766
    - Para NATs simétricos, con servidor TURN (relay)

STUN
  - Session Traversal Utilities for NAT (STUN), IETF 5389
    - No funciona con nat simétricos

La conexión se realiza a través de UDP/RTP
Proyecto Software Libre: LyncKia: WebRTC MCU
Thanks