



WebRTC
DEMYSTIFIED

Anant Narayanan, Tim Terriberry

Warsaw

September 8, 2012

WHAT?

- A set of technologies to enable real time communication in web pages through a ***simple JavaScript API***
- You need three things for this to work:



Access to the webcam & microphone



Peer to peer networking



Media encoding and playback



GETUSERMEDIA

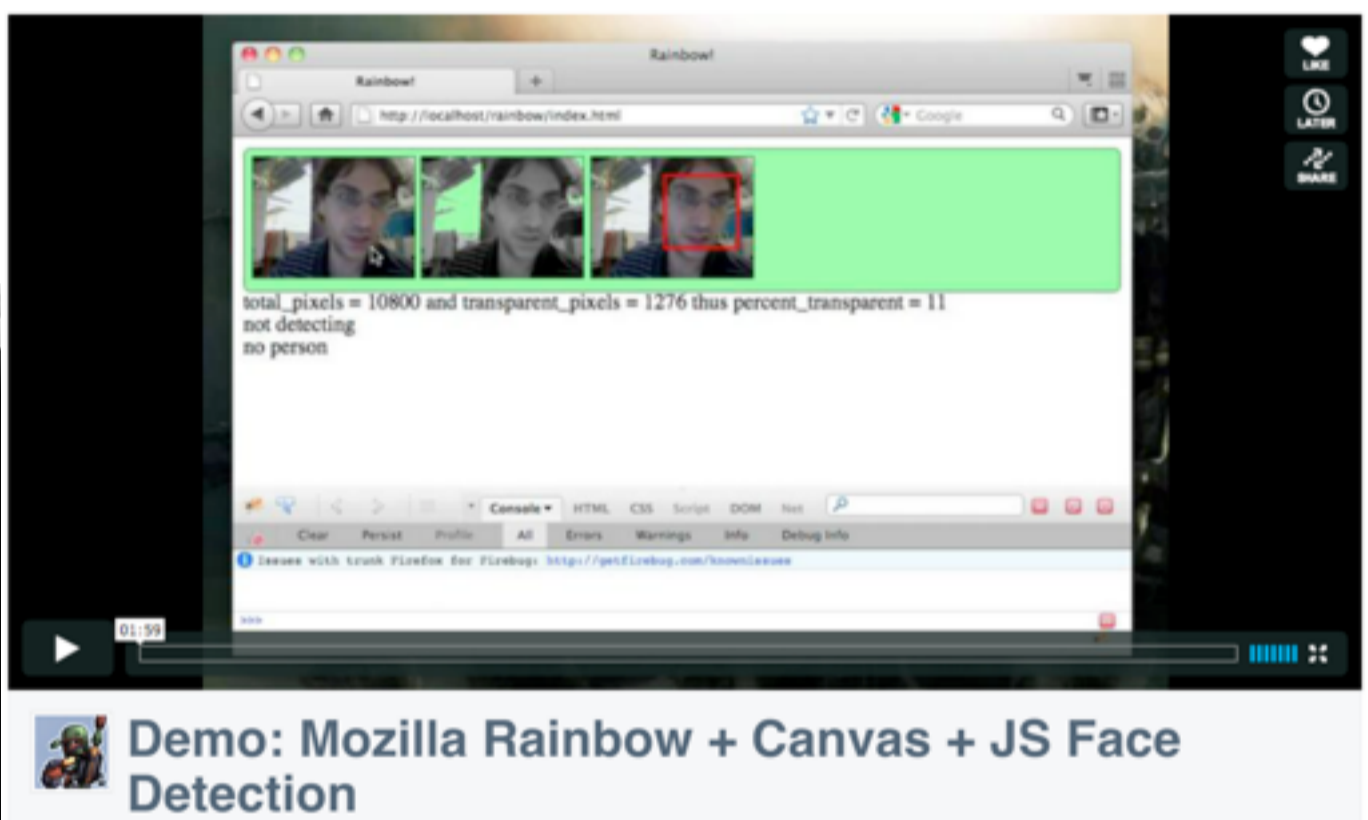
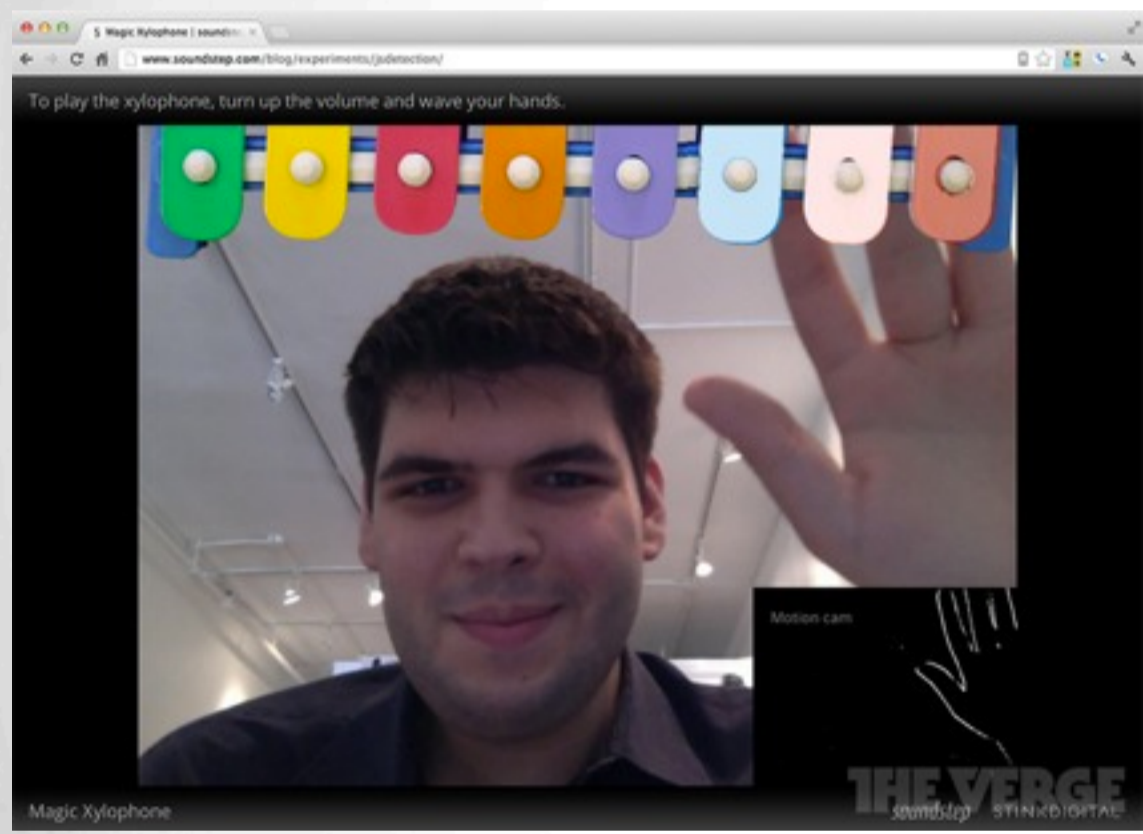
- Web camera and microphone access with a simple API that integrates well with other web elements like `<audio>` and `<video>`

```
1 <!DOCTYPE html>
2 <html>
3   <video id="tehvideo" controls/>
4   <script>
5     var video = document.getElementById("tehvideo");
6     navigator.mozGetUserMedia(
7       {video:true},
8       function(stream) {
9         video.src = stream;
10        video.play();
11      },
12      function(err) {
13        alert("Error! " + err);
14      }
15    );
16  </script>
17 </html>
```



GETUSERMEDIA

- Integration with <video> and <canvas> opens up many possibilities!



PEERCONNECTION

- Allows creating a peer to peer channel for audio, video and data
- High-level API, the grimy details of connectivity checks and NAT traversal is (*mostly!*) hidden

```
1 <script>
2   function error(e) { alert(e); }
3   var pc = new mozPeerConnection();
4
5   pc.addStream(videoStream);
6   pc.onremotestreamadded = function(stream) {
7     document.getElementById("remotevideo").src = stream;
8   };
9
10  pc.createOffer(function(offer) {
11    pc.setLocalDescription(offer, function() {
12      $.post("/offer", offer, function(answer) {
13        pc.setRemoteDescription(answer, function() {
14          alert("Call established!");
15        });
16      });
17    }, error);
18  }, error);
19 </script>
20
```



DATA CHANNELS

- Like WebSockets, but directly from one peer to another!
- Both reliable (like TCP) and unreliable (and UDP) modes available

```
1 <script>
2   function error(e) { alert(e); }
3   var pc = new mozPeerConnection();
4   // ...
5   pc.onDataChannel = function(channel) {
6     channel.onmessage = function(e) {
7       // e.data instanceof Blob, DOMString, ArrayBuffer
8     };
9     channel.onopen = function(e) {
10      // ...
11    };
12  };
13  pc.onConnection = function() {
14    var reliable = pc.createDataChannel("hello world", {});
15    var unreliable = pc.createDataChannel("hello world", {reliable:false});
16    reliable.send(...);
17    // ...
18  }
19 </script>
```



MEDIA

- VP8 for Video and Opus for Audio, just like `<audio>` & `<video>`
- Opus - a high quality audio codec that's great for realtime speech, music **and** local playback

web▶m

opus

- You don't (*necessarily*) have to worry about encoding or decoding, PeerConnection will automatically select the most appropriate codec usable on both ends.
- Integration with the Audio Data API for manipulation coming soon!



ROADMAP

- ***getUserMedia*** available in FF17 (behind a pref), targeting FF18 for turning it on by default
- Targeting ***PeerConnection*** for FF18, currently available on *alder* for experimental purposes
- Current focus is on shipping a usable, "**1.0**" version of both the above

<https://wiki.mozilla.org/Media/WebRTC>



WE NEED YOU!

- **Developers:** Tons of blocker bugs that need to be worked on, for both `getUserMedia`, `PeerConnection` and other WebRTC related efforts
 - hg.mozilla.org/projects/alder - build just like mozilla-central
 - Write demos that use *getUserMedia* and/or *PeerConnection*
- **Designers:** We need creative ideas and mockups for the UI that should be presented to the user
 - On device access, permission grant, during use, and more...
- **Everybody:** Tutorials, Documentation and Evangelism!



Q&A

#MEDIA

IRC.MOZILLA.ORG

DEV-MEDIA@LISTS.MOZILLA.ORG

