WebRTC Workshop

WebRTC Workshop

Björn Helgeson
bhelg@kth.se

Today 9-12

1. Introduction
2. Tasks 1-4
3. Do something new!

This Workshop

What’s this?

```
<!DOCTYPE html>
<html>
<head>
<title>Title of the document</title>
</head>

<body>
<h1> Heading 1! </h1>
<p> This is a paragraph </p>
</body>

</html>
```

The Web & HTML5

- Old web technologies
  - HTML, Javascript & CSS
- New web technologies
  - HTML5, WebRTC

Enabling powerful browser applications without the need to install additional software

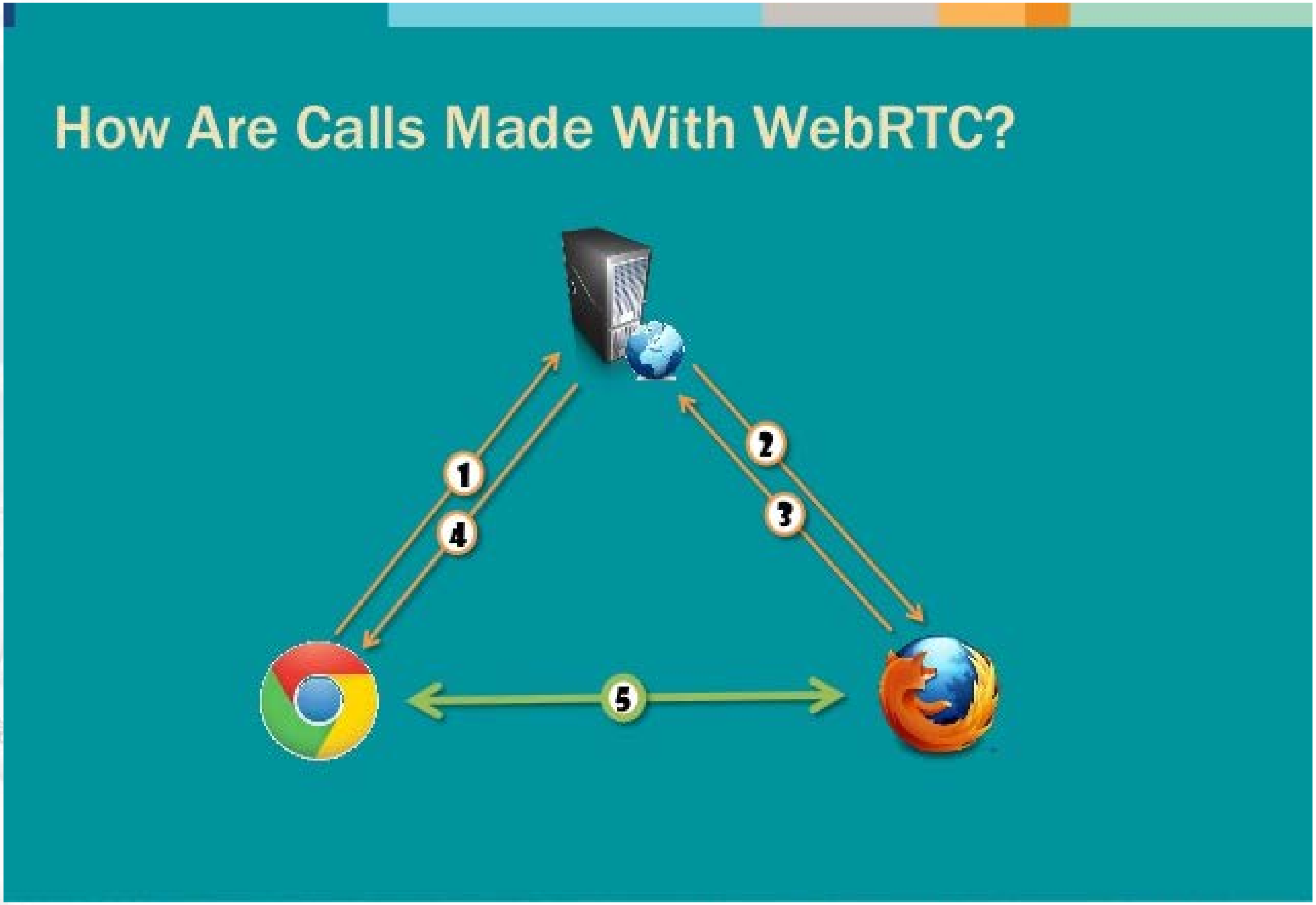
HTML5

- <video>
  - Makes adding video to your page as easy as adding an image...
- <audio>
  - ... and the same goes for sound
- <canvas>
  - A rectangle in your page where you can use JavaScript to draw anything

What is WebRTC?

- Real Time Communication
  - Voice Calling
  - Video Chat
  - P2P file sharing

WebRTC - What’s the thing?



How Are Calls Made With WebRTC?

WebRTC

- It’s not a service, it’s a technology

Is webrtc ready yet?

WebRTC

- An implementation of SRTP with an SDP control mechanism on top
- A media engine with G.711, Opus and VP8 Codecs
- A VoIP implementation using STUN, TURN and ICE for NAT traversal

WebRTC

- MediaStream
  - Audio
  - Video

- http://webcamtoy.com/
- http://seriouslyjs.org/
- Our Demo

WebRTC - Major Components

- `getUserMedia` – grabs your computers camera and/or microphone after being granted permission, and captures the streams.
- `PeerConnection` – sets up audio/video calls between two parts
- `DataChannels -` allow browsers to share files or other data directly via peer-to-peer

WebRTC - getUsermedia

```
navigator.getUserMedia(
constraints, success, error
)
```

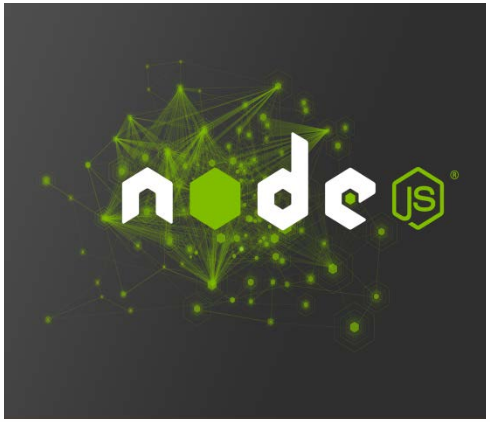
```
1.  var constraints = {video: true, audio:true};
2.  function success(stream){
    document.getElementById("localVideo").src =
    URL.createObjectURL(stream);
    }
3.  function(error) {
        alert(error);
        }
```

WebRTC - Important componens

- **Session control messages:** to initialize or close the communication
- **ICE Candidate:** tells you who you are to the outside world (your computer's IP address and port)
- **Session description:** tells you what codecs and resolutions can be handled by your browser and the browser it wants to communicate with

Other tools

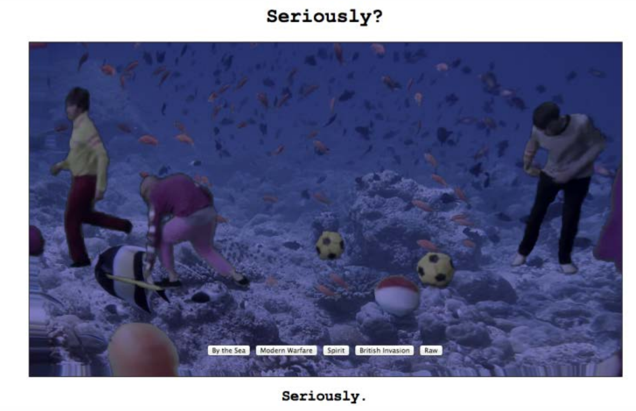
- Node.js



  - Platform to easily build fast network applications
  - Host a server easily on your own computer

Other Tools

- Seriously.js



Seriously?

Seriously.

  - A real-time, node-based video effects compositor for the web built with HTML5 & Javascript

Preparation

1. Go to http://nodejs.org/
2. Download node.js for your system
3. Make sure you have a text editor or download for instance Sublime Text 2
4. Make sure you have the latest version of Chrome
5. Check out some tutorials

   WebRTC at HTML5Rocks

   HTML5-school

First Task

Programmers:

1. Download the material from the course web
2. Run the server. In a terminal:
    a. cd PATH/TO/FOLDER
    b. node channel_server.js
3. Test at http://localhost:8080/task1.html

Everyone else:

Warm-up exercise!

Second Task

Programmers:

1. Try out WebRTC video chat: http://localhost:8080/task2.html
2. See if you can connect two computers. Use ifconfig (Mac) or ipconfig (Windows) to find the IP address of your computer

The rest:

Brainstorm ideas for a WebRTC application that might be useful in your project

Third Task

Programmers:

1. Go to http://localhost:8080/task3.html
2. Play with the filters!
3. Try and replace the image with webcam video
4. Apply CSS filters to your video chat from task 2

Non-programmers:

Pick the best idea you've come up with and begin sketching on a mockup version

Fourth Task

Programmers:

1. More advanced filters: http://seriouslyjs.org/
2. Take a look at http://localhost:8080/Seriously.js/
3. Try and use Seriously.js filters instead of CSS in your video chat
   a. webrtc-workshop/client/Seriously.js/seriously.full.min.js
4. Can you use the Chroma Key filter to replace the background in your video chat?
   a. tip 1: use the 'chroma' effect on the video, then 'blend' with an image
   b. tip 2: http://localhost:8080/task4.html show how to apply effects to video

The rest:

Force the programmers to implement your mockup! :)

Links

W3C

WebRTC Working Group

http://www.w3.org/2011/04/webrtc/

Specifications

http://dev.w3.org/2011/webrtc/editor/getusermedia.htm
http://dev.w3.org/2011/webrtc/editor/webrtc.html

IETF

http://tools.ietf.org/wg/rtcweb/