

PeerConnection API

A few slides to get the
discussion going

A simple example

Example Overview

```
// set up the call, get access to local media, and establish connectivity
function start(iscaller) {
  pc = new PeerConnection(null);

  // send any ice candidates to the other peer
  pc.onicecandidate = function (evt) {
    signalingChannel.send(JSON.stringify({ "type": "candidate", "sdp": evt.candidate }));
  };

  // once remote stream arrives, show it in the remote video element
  pc.onaddstream = function (evt) {
    remoteView.src = URL.createObjectURL(evt.stream);
  };

  // get the local stream and show it in the local video element
  navigator.getUserMedia({ "audio": true, "video": true }, function (stream) {
    selfView.src = URL.createObjectURL(stream);
    pc.addStream(stream);

    var type;
    if (iscaller) {
      pc.createOffer(gotDescription);
      type = "offer";
    } else {
      pc.createAnswer(pc.remoteDescription, gotDescription);
      type = "answer";
    }

    function gotDescription(desc) {
      pc.setLocalDescription(type, desc);
      signalingChannel.send(JSON.stringify({ "type": type, "sdp": desc }));
    }
  });
}

signalingChannel.onmessage = function (evt) {
  var msg = JSON.parse(evt.data);
  switch (msg.type) {
    case "offer":
      // create the PeerConnection
      start(false);
      // feed the received offer into the PeerConnection
      pc.setRemoteDescription(msg.type, new SessionDescription(msg.sdp));
      break;
    case "answer":
      pc.setRemoteDescription(msg.type, new SessionDescription(msg.sdp));
      break;
    case "candidate":
      pc.addIceCandidate(new IceCandidate(msg.sdp));
      break;
  }
};
```

Create PeerConnection and handle generated candidates

Handle incoming streams

Request media, add media to be sent

Create offer/answer

Process incoming offers, answers, and candidates

Example Walkthrough

```
signalingChannel.onmessage = function (evt) {  
  var msg = JSON.parse(evt.data);  
  switch (msg.type) {  
    case "offer":  
      // create the PeerConnection  
      start(false);  
      // feed the received offer into the PeerConnection  
      pc.setRemoteDescription(msg.type, new SessionDescription(msg.sdp));  
      break;  
    case "answer":  
      pc.setRemoteDescription(msg.type, new SessionDescription(msg.sdp));  
      break;  
    case "candidate":  
      pc.addIceCandidate(new IceCandidate(msg.sdp));  
      break;  
  }  
};
```

```
// set up the call, get access to local media, and establish connectivity  
function start(callId) {  
  pc = new RTCPeerConnection({});  
  // send any ice candidates to the other peer  
  onIceCandidate = function (evt) {  
    if (evt.candidate) signalingChannel.send(JSON.stringify({ "type": "candidate", "id": evt.candidate.sdp }));  
  };  
  // once remote stream arrives, show it in the remote video element  
  onRemoteStream = function (evt) {  
    remoteVideo.src = URL.createObjectURL(evt.stream);  
  };  
  // get the local stream and show it in the local video element  
  navigator.getUserMedia({ video: true, audio: false }, function (stream) {  
    localVideo.src = URL.createObjectURL(stream);  
    onLocalStreamReady();  
  }, error);  
  // create offer  
  var offer = {  
    type: "offer",  
    sdp: pc.createOffer({ video: true, audio: false }, function (offerSdp) {  
      pc.setLocalDescription(offerSdp, offerSdp);  
      signalingChannel.send(JSON.stringify({ "type": "offer", "sdp": offerSdp }));  
    }));  
  // create answer  
  function onAnswerReceived(offerSdp, desc) {  
    signalingChannel.send(JSON.stringify({ "type": "answer", "sdp": desc }));  
  };  
  // create answer  
  pc.createAnswer(offerSdp, desc, onAnswerReceived);  
  pc.setLocalDescription(desc, desc);  
  // create candidate  
  signalingChannel.onmessage = function (evt) {  
    var msg = JSON.parse(evt.data);  
    switch (msg.type) {  
      case "offer":  
        // create the PeerConnection  
        start(false);  
        // feed the received offer into the PeerConnection  
        pc.setRemoteDescription(msg.type, new SessionDescription(msg.sdp));  
        break;  
      case "answer":  
        pc.setRemoteDescription(msg.type, new SessionDescription(msg.sdp));  
        break;  
      case "candidate":  
        pc.addIceCandidate(new IceCandidate(msg.sdp));  
        break;  
    }  
  };  
}
```

Question to people reviewing these slides:
Should we include another example of this kind?

A bit more advanced example

A few topics for discussion

Got More Candidates?

- How do we tell that we got enough candidates?
 - null candidate? (as suggested by Justin on the list)
 - New (simple) event?

```
pc.onicecandidate = function (evt) {  
    var candidate = evt.candidate;  
    //...  
};
```

```
[Constructor(DOMString type, PeerConnectionIceEventInit eventInitDict)]  
interface PeerConnectionIceEvent : Event {  
    readonly attribute PeerConnection peer;  
    readonly attribute IceCandidate candidate;  
};
```

Need to negotiate?

- Event to indicate that negotiation is needed
 - Track added/removed
 - Track ended
- Drive call setup(?)

```
[Constructor (IceServers configuration, optional MediaConstraints constraints)]  
interface PeerConnection {  
    ...  
    attribute Function? onrenegotiationneeded;  
    ...  
};
```

A lot of Tweaking

- We currently got five places for tweaking when sending a stream

```
navigator.getUserMedia(constraints, function (stream) { ... }); // 1
```

```
pc.addStream(stream, constraints); // 2
```

```
pc.createOffer(function (offer) {  
    offer.tweakOffer(tweaks); // 4  
    pc.setLocalDescription("offer", offer);  
    // ...  
}, constraints); // 3
```

```
tweakSdp(offer.sdp); // 5
```

Configuring PeerConnection

- Does this need to be an object or would `DOMString[]` do?
 - Future extensions

```
interface IceServers {  
    attribute DOMString servers[];  
};  
  
{ servers:[  
    "stun:stun.example.net",  
    "turn:user@turn.example.org myPassword" ]  
}
```

More topics

- What's being offered?
- How long is my newly created offer valid?
- ICE states (current use case?)

Question to people reviewing these slides:
More topics?

```

[Constructor (IceServers configuration, optional MediaConstraints constraints)]
interface PeerConnection {
    void        createOffer (SessionDescriptionCallback successCallback,
                            optional PeerConnectionErrorCallback failureCallback,
                            optional MediaConstraints constraints);
    void        createAnswer (SessionDescription offer,
                              SessionDescriptionCallback successCallback,
                              optional PeerConnectionErrorCallback failureCallback,
                              optional MediaConstraints constraints,
                              optional Boolean createProvisionalAnswer=false);
    void        setLocalDescription (SdpType action, SessionDescription description);
    readonly attribute SessionDescription localDescription;
    void        setRemoteDescription (SdpType action, SessionDescription description);
    readonly attribute SessionDescription remoteDescription;
    readonly attribute PeerState          readyState;
    void        updateIce (optional IceServers configuration,
                           optional MediaConstraints constraints,
                           optional Boolean restart=false);
    void        addIceCandidate (IceCandidate candidate);
    readonly attribute IceState          iceState;
    readonly attribute MediaStream[]    localStreams;
    readonly attribute MediaStream[]    remoteStreams;
    DataChannel createDataChannel ([TreatNullAs=EmptyString] DOMString? label,
                                   optional DataChannelInit? dataChannelDict);
    attribute Function?                 ondatachannel;
    void        addStream (MediaStream stream, optional MediaConstraints constraints);
    void        removeStream (MediaStream stream);
    void        close ();
    attribute Function?                 onrenegotiationneeded;
    attribute Function?                 onicecandidate;
    attribute Function?                 onconnecting;
    attribute Function?                 onopen;
    attribute Function?                 onstatechange;
    attribute Function?                 onaddstream;
    attribute Function?                 onremovestream;
    attribute Function?                 onicechange;
};

```

Always good to have
the idl at hand