```
[https://www.w3.org/2011/04/webrtc/wiki/January_14_2016]
[08:06] == Erik_ [~Erik@public.cloak] has joined #webrtc
[08:06] <dom> ScribeNick: DrAlex
[08:06] <dom> i/Agenda/ScribeNick: dom
[08:06] <dom> ... Meeting is being recorded
[08:07] <DrAlex> scribe mode
[08:07] == Shijun [~Shijun@public.cloak] has joined #webrtc
[08:07] <DrAlex> waiting for recording and start of the hangout session
[08:07] <dom> Present+ MathieuHofman
[08:07] <DrAlex> erik: welcome everybody, if you can t join the hangout, please
join irc and let us know
[08:08] <DrAlex> ... a slide for you adam
[08:08] <dom> Topic: Contribution guidelines
[08:08] == mreavy1mac [~chatzilla@public.cloak] has joined #webrtc
[08:08] <vivien> -> https://github.com/w3c/webrtc-pc/pull/447  Add contribution
guidelines #447
[08:09] <DrAlex> adamB: pr447 include guidelines for contributions to the deaft
[08:09] <DrAlex> s/deaft/draft
[08:09] <DrAlex> bernard: is there anything committer should be aware of?
[08:09] <DrAlex> admaB: yes, there is a section about line wrapping for
example, for the automatic checking.
[08:10] <DrAlex> fluffy: can we put automatic checking in a separate PR?
[08:10] <DrAlex> adamB: yes, you can see a corresponding comment.
[08:10] == jesup [~chatzilla@public.cloak] has joined #webrtc
[08:10] <DrAlex> dom: it is not enforced yet in the repo. It would be a good
thing for people to start doing it though.
[08:10] <DrAlex> ... while we are on this topic, a heads up, hopefully in a few
weeks there should be a new way to put IDL code in the specs.
[08:11] <DrAlex> ... obviously I will let people know when it happens.
[08:11] <DrAlex> bernard: particular, when there are new things we should start
doing, a post on the mailing list would be welcome.
[08:12] <DrAlex> adamB: I was not aware that we were not enforcing this on the
specs. I think my personal view is that doing automatically line wrapping will
make things more difficult. But ... we can try to use it and see.
[08:12] <DrAlex> erik: so are we ok to move on, culled?
[08:12] <DrAlex> fluffy: I was just requesting that one PR maps to one feature,
not several features. Easier reviewing.
[08:13] <DrAlex> eirk: ok, next.
[08:13] <DrAlex> s/eirk/erik/
[08:13] <dom> Topic: Pull requests
[08:13] <DrAlex> erik: I think there is actually an extra PR we forgot to put
in this list.
```

```
[08:13] <DrAlex> ... if you see anything missing, please let us know, maybe on
IRC.
[08:14] <DrAlex> ... having said that, did anybody already see something
missing?
[08:14] <DrAlex> bernard: I had a list, and put it in the slides, so I believe
everything is there.
[08:14] <DrAlex> harald: 454 is included.
[08:14] <DrAlex> erik: cullen, your turn.
[08:14] <vivien> -> https://github.com/w3c/webrtc-pc/pull/434 Change
setParameters call to be Async #434
[08:15] <DrAlex> cullen:  last time we spoke on setParams Async, it seemed ok,
but harald raised a few timing problems
[08:15] <DrAlex> ... what happen if you have "overlapping" setParams call ? do
you queue them? In which order should you handle it.
[08:15] <dom> Present+ AlanJohnston
[08:15] <DrAlex> ... I think this is the open issue we need to resolve, before
we write anything about it in the draft.
[08:16] <DrAlex> ... harald since you raised the issue, would you mind to
comment.
[08:16] <DrAlex> harald: yes, i realized after the discussion that you won;t be
able to resolve things on the thread to be able to know what to set and what
happen eventually.
[08:17] == jib [~jib@public.cloak] has joined #webrtc
[08:17] <DrAlex> harald: in some cases, depending on orders of calls, and
timing of promise resolving, you have not consistent behavior.
[08:17] <DrAlex> ... is it so complicated that we should stay with Synchronous,
or can we actually write it down and fully specify it?
[08:17] == adambe_ [~adambe@public.cloak] has joined #webrtc
[08:18] <DrAlex> there were some discussions about codec change for example,
changing software codec to hardware codec, discovering that this codec is
already in use, it might not be done synchronuously.
[08:19] <dom> Present+ Justin
[08:19] <DrAlex> ????: the read is also not read only
[08:19] <vivien> s/????/Peter/
[08:19] <dom> s/the read is also not/the rid should be/
[08:19] <DrAlex> justin: <something>
[08:19] <DrAlex> peter: can we live with it being sync ?
[08:20] <DrAlex> cullen: there is no way, we are going to need async.
[08:20] <dom> s/<something>/I'm pretty sure we will encounter cases where we
need time before vetting off parameters
[08:20] <DrAlex> peter: i m more concerned about justin other use case.
[08:21] <DrAlex> harald: when you do getParam, you return a conceptually
```

```
generated value .... and if you set param, it should still be the same params,
otherwise, setParam should fail
[08:21] <DrAlex> ... that would enforce consistency.
[08:21] <DrAlex> cullen: that would be the most miserable API from user
exeprience
[08:21] * vivien RRSAgent, make logs public
[08:21] * RRSAgent I have made the request, vivien
[08:21] <DrAlex> harald: for anything that does not have a race condition, it
would just work.
[08:21] * vivien RRSAgent, draft minutes
[08:21] <RRSAgent> I have made the request to generate
http://www.w3.org/2016/01/14-webrtc-minutes.html vivien
[08:22] <DrAlex> justin; is it clear how you could arrive to a discrepancy in
the params ?
[08:22] <DrAlex> cullen: what you propose harald would work.
[08:23] <DrAlex> adamB: can we have some failure happen at anytime, anyhow, to
have a process to report problems. Say we keep it async, and if it fails, we
report it.
[08:23] <DrAlex> harald: that would be an event, not a promise.
[08:24] <DrAlex> justin: we should still entertain it.
[08:24] <DrAlex> roman: it feels like we are arriving to a transactional model.
It would be cleaner also.
[08:25] <DrAlex> justin: right, but it is almost like a batch set, batch set.
Any error would be surface through the failed promise.
[08:25] <DrAlex> roman: yes, but you would have some sort of object way to
percolate the info up through the promise.
[08:25] <DrAlex> justin: again, why async is problematic again ?
[08:26] <DrAlex> harald: if we have a transaction mechanism like roman
proposed, then it is ok. Otherwise it s problematic. we have to be sure that
the behavior is predictable.
[08:26] <DrAlex> peter: otherwise we have the same problem we are having when
set..... is called twice, before the first one resolves.
[08:27] <DrAlex> cullen: I feel like calling things twice before the first call
resolves would happen frequently enough not to be a corner case.
[08:27] <DrAlex> harald: the place where we could copy this mechanism from is
Dynamic DNS.
[08:28] <DrAlex> cullen, so justin, i m trying to design your thinking, when
you call setParams, if there is already a setParams promise queue or being
processed, it instantaneously fails.
[08:28] <DrAlex> peter: then what happen in this (more convoluted case)
[08:28] <DrAlex> justin: params should not be set until setParam() resolved.
[08:29] <DrAlex> bernard: just to be clear, get is always sync, right?
```

```
[08:29] <DrAlex> harald: yes
[08:29] <DrAlex> erik: just checking on time, and the notes.
[08:29] <hta> Harald promises to create a pull request against Cullen's pull request to make this modification.
[08:29] <DrAlex> peter: if setParam is called while a previous call is unresolved, then fail right away.
[08:30] <DrAlex> justin: if we think there is a case for multiple calls, we can then queue, but as of now, there is no such case.
[08:30] <DrAlex> erik: are we going ahead with justin proposal (vs herald's)?
[08:30] <vivien> Present+ Alan_Johnston, Ted_Hardie
[08:30] <DrAlex> justin: yes, but harald said he would write the PR again it :)
[08:30] <DrAlex> erik: great, can t get bette than that
[08:30] <DrAlex> bernard: next slide will be fast.
[08:30] <vivien> -> https://github.com/w3c/webrtc-pc/pull/454  Add contributing source voice activity flag #454
[08:31] <DrAlex> ... proposal to add support for v flag (RFC6464)
[08:31] <jesup> +1 for adding this
[08:32] <jesup> don't care about the name ;-)
[08:32] <DrAlex> .. instead of using audio level, you could rely on "v" to know who is speaking
[08:32] <DrAlex> harald: voice activated flag sounds like a good name.
[08:32] <jesup> AudioActivityIndicator
[08:33] <jesup> Or voice.
[08:33] <DrAlex> justin: we talk about this at TPAC. we decided not to do it, because it does not get sent down from mixer. In this case (p2p), I can see how it could make sense.
[08:34] <DrAlex> bernard: the use case for this is a very simple UI, for mesh conference.
[08:34] <dom> I'm neutral on adding it; but please let's not use "v" as a name :)
[08:35] <DrAlex> justin: this is gonna lead to a future where someone will ask for the mixer to send the info down.
[08:35] <DrAlex> cullen: the mixer equivalent is the crsc list
[08:36] <DrAlex> justin: in all fairness, some people use the list to determine background noise (??)
[08:36] <DrAlex> jesup: i can see a use for that, as people try to code SFU in the browser.
[08:36] <DrAlex> bernard: let's try to keep it in the p2p model.
[08:37] <DrAlex> cullen: if the signaling is such, that you do not use that bit, what happen?
[08:37] <DrAlex> bernard: it s just unset
[08:37] <DrAlex> justin: you mean undefined
```

```
[08:37] <DrAlex> bernard: yes
[08:37] <DrAlex> cullen: i like that
[08:38] <DrAlex> all: can we just use another name for it?
[08:38] <vivien> -> https://github.com/w3c/webrtc-pc/pull/462  Add
PeerConnection.activeSender() and update early media example. #462
[08:39] <DrAlex> harald: seems good to me
[08:39] <DrAlex> erik: doe sanybody oppose to this?
[08:39] <DrAlex> erik: ok, consensus it is.
[08:40] <vivien> -> https://github.com/w3c/webrtc-pc/pull/463  Add more
explicit language about transceivers, MIDs and rollbacks. #463
[08:40] <DrAlex> erik: rollback
[08:40] <DrAlex> peter: so i went ahead and put some text about the rollback.
[08:40] <DrAlex> .. somebody had already put the nullable part
[08:40] <DrAlex> .. i just made it a little bit more explicit what would happen
during the rollback, and also pointers to JSEP.
[08:40] <DrAlex> ... it s just slightly more explicit
[08:40] <DrAlex> erik: anybody has any comment on this?
[08:41] <DrAlex> cullen: looks like a notarial change, so ...
[08:41] <DrAlex> harald: I like it ....
[08:41] == TedHa [~hardie@public.cloak] has joined #webrtc
[08:41] <DrAlex> erik: ok, next, ice transport policy
[08:41] <vivien> -> https://github.com/w3c/webrtc-pc/issues/441
RTCIceTransports needs to be DOMString not enum? #441
[08:41] <DrAlex> bernard: there was a discrepancy between the draft and JSEP.
[08:41] <dom> Present+ Jan_Ivar
[08:42] <DrAlex> ... our first attenpt at fixing it raised backward
compatibility and extensibility issues
[08:42] <DrAlex> ... harald proposed another solution.
[08:43] <DrAlex> ... we just want to ask the browser guys if that will be a
problem?
[08:43] <DrAlex> justin: i m not sure i would deb on board as the migration,
but the move to a string would be good in the future.
[08:44] <DrAlex> peter: what's the point of having a string? I don't get it.
[08:44] <DrAlex> cullen: so we could return an error. also ENUM are broken in
IDL.
[08:44] <DrAlex> peter: ah ok, enum are broken.
[08:45] <DrAlex> bernard: so what should we do?
[08:45] <DrAlex> jib: comment on enum.
[08:45] <DrAlex> jib: you have no choice if you don't want the browser to choke
on the type.
[08:45] <DrAlex> jib: that s why we used strings for optional constraints.
[08:46] <DrAlex> justin: we want the browser to NICELY fail when it bumps into
```

```
something it does not know. With string you can ignore it, not with ENUM.
[08:46] <DrAlex> justin: i would prefer to keep an enum, but well define the
values in the enum.
[08:46] <DrAlex> cullen: use case with red and green relays.
[08:47] <DrAlex> ... the question is then, how do we extend it? if we do not
extend it, enum is fine.
[08:48] <DrAlex> bernard: PR 432 should be rolled back, and ENUM could be OK.
[08:48] <DrAlex> justin: yeah, i guess, do we want to speak about which values
should be in there?
[08:48] <DrAlex> dom: technically, the enum is missing what is in JSEP?
[08:48] <DrAlex> bernard: has anybody implemented none already?
[08:48] <DrAlex> jib: firefox has.
[08:49] <DrAlex> cullen: what are the privacy implications?
[08:49] <dom> s/enum is missing what is in JSEP?/enum is reflecting what is in
JSEP
[08:49] <DrAlex> justin: my thinking about this has changed across time.
[08:49] <DrAlex> ... originally it was because gathering was starting right
away
[08:49] <DrAlex> ... right now, you need to wait for a call.
[08:50] <DrAlex> ... now is maybe not the right time to speak about that. if
someone wants to keep none, then let s hear the rationale
[08:50] <DrAlex> bernard: ok, let s write down that we need to speak about
"none"
[08:50] <DrAlex> bernard: i think we can move on
[08:50] <DrAlex> erik: can we speak about that on discuss-public as well?
[08:50] <vivien> -> https://github.com/w3c/webrtc-pc/issues/389  Should have a
"closed" RTCPeerConnectionState #389
[08:50] <DrAlex> erik: issue 389, peter
[08:51] <DrAlex> peter: reading the slide.
[08:51] <DrAlex> erik: anybody has any comment on that?
[08:51] <DrAlex> jib: it just reminds me that there is another occasion where
the peer connection can close itself.
[08:52] <DrAlex> jib: in that case, having a state is important for the user.
[08:52] <DrAlex> ... but if the user must close it, then he/she does not need a
state.
[08:52] <DrAlex> harald: so when you call closed on a peer connection, how do
yo know it is actually closed.
[08:52] <DrAlex> ?
[08:53] <DrAlex> pete: so if all the ice connection state are closed, then the
peer connection is closed?
[08:53] <DrAlex> harald: i think that you NEED to call close() for the peer
connection is closed.
```

```
[08:54] <vivien> s/pete:/peter:/
[08:55] <dom> +1 to adambe to update "signalling state is closed" to "connection state is closed" (if we adopt this)
[08:55] <TedHa> I can scribe
[08:55] <vivien> scribe: TedHa
[08:55] <DrAlex> thanks ted, sorry guys
[08:55] == mathieucitrix [~uid123277@public.cloak] has joined #webrtc
[08:56] * vivien thanks DrAlex for scribing so far
[08:56] <TedHa> Conclusion for Issue 389: add the "closed" state.
[08:57] <TedHa> Harald: set all three states to closed.
[08:57] <TedHa> Peter will make the relevant PR.
[08:57] <TedHa> Issue 412:  Framerate knob for Simulcast
[08:57] * trackbot doesn't understand that ISSUE command.
[08:57] <vivien> -> https://github.com/w3c/webrtc-pc/issues/412 Framerate knob for simulcast #412
[08:59] <TedHa> Bernard:  reviews two use case.  Screen sharing conference with devices of different framerates.  scaleResolutionDownBy generates streams with dfifferent resolutions.  maxFramerate generates streams within device max framerate.
[09:00] <TedHa> Video conference is an example where scaleResolutionDownBy would be useful (thumbnails get scaleResolutionDownBy)
[09:01] <TedHa> Cullen:don't care about scale so much, but care deeply about the maxFramerate, because of the interaction with a bridge.
[09:01] <TedHa> Telling a high bandwidth browser what to do in that context gives you a direct impact on user experience
[09:02] <TedHa> Cullen notes that screen share in simulcast situation is another example (low framerate for unchanging presentation).
[09:03] <TedHa> Justin:  If you really want to make changes based on changes to framerate not on under user control, this gives you a way to handle that.
[09:04] <TedHa> Cullen: But we are primarily interested in a different use case, where the capture rate and the maxframerate are not the same.
[09:04] <TedHa> Justin: probably simplest to just go with maxFrameRate.
[09:04] <TedHa> (No objections)
[09:04] <TedHa> Decision: maxFramerate, already in the spec, will be maintained.
[09:05] <vivien> -> https://github.com/w3c/webrtc-pc/issues/370 Add drop option for RTCDegradationPreference #370
[09:05] <TedHa> Now on Issue 370: add drop option
[09:05] <TedHa> How does a simulcast sender stop sending simulcast layers? Note that the firing of a circuit breaker could cause this to be required.
[09:06] <TedHa> (Aside:  how does the application know the circuit breaker has fired)?
```

```
[09:06] <TedHa> Bernards reminds that degredationPreference is codec
independent, where priority is within a codec-specfic context.
[09:07] <TedHa> Bernard asks whether these are enough to do what we want?
[09:08] <TedHa> Note that priority determines the QoS marking; it may also
guide encoder behavior (preference of encoding to drop if all cannot be sent
simultaneously)
[09:08] <TedHa> Bernard notes that there may be a conflict here with circuit
breakers.
[09:08] <TedHa> Harald asks to confirm that circuit breakers talks about
transport, basically?
[09:09] <TedHa> Varoun notes that circuit breakers is not all-or-nothing, if
you detect it before the "all or nothing" level, you can shut off individual
ssrcs.
[09:10] <TedHa> Does priority apply across audio and video streams?  Bernard
answers that it is not just within a sender, but applies across all senders.
[09:11] <TedHa> That might be confusing, but Harald notes that this is a matter
for the IETF transport draft, which is designed not to starve any stream
completely.
[09:12] <TedHa> Cullen says that circuit breakers tells you about number of
bits in a congestion context, transport tells you about allocating them within
the context, this tells the encoder about relative priority at the level of
layers.
[09:12] <TedHa> Bernard:  Is this enough information for app writers?
[09:12] <TedHa> Cullen:  Maybe not.  Harald, does this match your
understanding?
[09:13] <TedHa> Harald:  It would be logical to see that it is application
dependent within a specific context.  It might be logical to give priority to
the smallest stream, so something always get through.
[09:14] <Erik_> 15 mins remaining
[09:14] <TedHa> Justin:  Minimum qp is the only way for the application to
express this; "don't send this unless you can send at least this good"?
[09:14] <TedHa> Cullen;  WebRTC has no answer to the question to "what's the
point at which you give up and stop sending video"?
[09:15] <TedHa> Justin: it's browser dependent right now.
[09:15] <TedHa> Justin:  We need to consider two different scenarios.  The
second of those is the minimum qp example.  I also have concerns that this is
doing both DSCP marking and bit apporitionment.
[09:16] <vivien> s/Cullen;/Cullen:/
[09:16] <TedHa> The group will discuss how to solve this use case, possibly by
looking the minimum qp stuff from ortc.
[09:16] <TedHa> Harald notes that the 1.0 aim is also to stop adding new
features.
```

```
[09:17] <TedHa> Conclusion:  Drop is not sufficient, and the group needs to
figure out what to do beyond that, but it is post 1.0 work.  At the moment,
signaling or stats review is available.
[09:18] <TedHa> Issue 442 up next.
[09:18] * trackbot doesn't understand that ISSUE command.
[09:18] <vivien> -> https://github.com/w3c/webrtc-pc/issues/442  Impossible to
know if ICE agent is "finished checking", for "failed" and "completed" states.
#442
[09:18] <TedHa> Taylor notes that finished checking is ambiguous, since late
candidates may cause you to re-start checking.
[09:19] <TedHa> The trickle ICE spec gives you end-of-candidates indication,
but there is nothing available to signal that in WebRTC.
[09:20] <TedHa> First question:  should 'failed' only occur after local and
remote gathering is done?
[09:21] <TedHa> This would make 'failed' more definitive, and confirms that
only an ICE restart would enable recovery.  That would also match trickle ICE
definition.  This also allows the ICE agent to conclude ICE more quickly in
aggressive nomination
[09:21] <TedHa> If the app never sends end-of-candidates, though, they would
never see failed.  That may be uncommon, since gathering is finished by the
time failed is reached.
[09:22] <TedHa> Apps can also determine this themselves.
[09:22] <TedHa> Taylor's recommendation is that "failed" means you are done
with local gathering.  It can flip back if you get a new candidate.
[09:23] <TedHa> Cullen asks a question about user experience:  I want to
differentiate between 'trying to get you a connection' and 'sorry, can't
connect'.  How do I do that?
[09:23] * vivien RRSAgent, draft minutes
[09:23] <RRSAgent> I have made the request to generate
http://www.w3.org/2016/01/14-webrtc-minutes.html vivien
[09:23] <TedHa> Taylor:  the application can send then if the peer connection
ICE state has failed and the local gathering has concluded.
[09:24] <TedHa> Bernard:  is this ever a final thing?  We may have a new
interface come up in the case of an initial 'failed'
[09:25] <TedHa> Justin:  If you are doing ICE, that's the time to be bringing
up interfaces.  Failed should be terminal.
[09:25] <TedHa> Bernard:  so you wouldn't signal failed until there were no
candidates that might be added.
[09:25] <TedHa> Justin: yes
[09:25] <vivien> Present+ Maire_Reavy
[09:26] <TedHa> Jan-Ivar:  I don't think that it is necessary to be terminal,
if a new candidate comes in.
```

```
[09:26] <Erik_> 5 mins
[09:26] <TedHa> Justin: failed in ICE is just a time-out.  All the mandatory
time out periods have elapsed.
[09:27] <vivien> Present+ Harald, Erik_Lagerway
[09:27] <TedHa> Bernard:  If you have an ICE restart to get out of that, you
get a new timer state.
[09:27] <TedHa> Peter:  what is the technical value for 'failed' to be
terminal?
[09:27] <TedHa> [Several--don't send end of candidates if you plan to bring up
new interfaces]
[09:28] <TedHa> Application may not have end of candidates.
[09:28] <TedHa> Conclusion:  Peter, Justin, Taylor will noodle on this and take
it to the list.
[09:29] <TedHa> Taylor:  if you have been disconnected, does that mean you will
never get out of disconnected, and does that mean "failed" as well?
[09:30] <TedHa> This gives applications a way to know when ICE restart is
absolutely needed.
[09:30] <TedHa> Bernard:  remember that when consent is gone, it's gone.
[09:30] <TedHa> Conclusion:  the question of whether disconnected should be
read as 'failed' will also be taken to the list.
[09:31] <TedHa> Bernard suggests that this topic might require a design meeting
or interim.
[09:31] <TedHa> Harald: might be a joint IETF or even solely IETF issue, since
this is largely trickle-ice behavior question.
[09:31] <varun> +1 to do an interim for the failed scenario case.
[09:32] <TedHa> Justin: let's try this on the list first, before getting a
design meeting together.
[09:32] <TedHa> Conclusion: agreed to try on the list first, since many of the
same people would be in both groups.
[09:32] <TedHa> Chairs: thanks for joining.
[09:33] == TedHa [~hardie@public.cloak] has left #webrtc []
[09:33] <dom> RRSAgent, draft minutes
[09:33] <RRSAgent> I have made the request to generate
http://www.w3.org/2016/01/14-webrtc-minutes.html dom
[09:33] * vivien I'll take care of cleaning up the minutes and sending them to
the list hopefully tomorrow
[09:33] == Shijun [~Shijun@public.cloak] has quit ["Page closed"]
[09:33] == fluffy [~fluffy@public.cloak] has quit [fluffy]
[09:33] == burn [~burn@public.cloak] has quit ["Page closed"]
[09:35] == dom [dom@public.cloak] has quit [""]
[09:37] == varun [~varun@public.cloak] has quit ["Page closed"]
[09:40] == adambe_ [~adambe@public.cloak] has quit [Ping timeout: 180 seconds]
```

```
[09:50] == DrAlex [~DrAlex@public.cloak] has quit ["Page closed"]
[09:52] == jesup [~chatzilla@public.cloak] has left #webrtc []
[09:54] == mreavy|mac [~chatzilla@public.cloak] has quit [Ping timeout: 180
seconds]
[09:56] == adambe_ [~adambe@public.cloak] has joined #webrtc
[10:13] == vivien [vivien@public.cloak] has quit ["Leaving"]
[10:21] == Bernarda [~Bernarda@public.cloak] has quit [Ping timeout: 180
seconds]
[10:41] <Erik_> Thanks to Alex and Ted for scribing
```