# On Embedded Real Time Media Communications

Jean-Charles Grégoire
INRS-EMT, Montréal, Canada
gregoire@emt.inrs.ca

## Abstract

We present WebRTC, describe its strengths and weaknesses and present its different forms of integrations, in the form of mediated, asymmetric or bridged communications. We go on to discuss the issues of providing real peer to peer communications on the basis of WebRTC and propose several building blocks to support such communications.

***Categories and Subject Descriptors*** H.4.3 [*Communications Applications*]: Computer conferencing, teleconferencing, and videoconferencing

***General Terms*** Service design, system architecture.

***Keywords*** WebRTC, Web-based applications, peer-to-peer communications, infrastructure building blocks.

## 1. Introduction

Voice-based communications have historically been delivered through a dedicated, mission–specific infrastructure, on the premise of a simple terminal and network-based functionality. This has certainly been true of traditional telephony and its "dumb" terminal [1], but has remained largely true in an IP-based universe, for example in infrastructures such as the IP multimedia subsystem (IMS) [3].

Certainly, while we can appreciate all the complexity that goes into a cellular telephone, such a device has nevertheless delivered little more than basic voice services, to the point that it appears that smart phones are used mostly for non-call purposes.

In parallel, the industry has always been focused on value-added services, creation and deployment. The terminal has always been part of the issues raised in deploying new services, through limitations in signalling ability, and, more recently, ease of software updates and addition of new features.


*AWeS'15*, April 21, 2015, Bordeaux, France

http://dx.doi.org/10.1145/2749215.2749224

The emergence of media communication features in application-neutral, dynamically accessible as well as programmable environments, as embodied by WebRTC, has allowed the emergence of a new paradigm for the creation and deployment of media communications.

We explore here some features of RTCs in general and the elements of an ecosystem to support different forms of WebRTC-enabled communications.

## 2. WebRTC

WebRTC is the product of the joint work of the RTCWEB (IETF)[7] and WEBRTC (W3C)[8] groups to specify a browser-based client to support real time communication (RTC) services[10]. WebRTC essentially offers browser-embedded audio, video and data communication primitives and an API to access and activate them. This effectively makes A/V communications available to browser-based applications, which can be dynamically downloaded from any web site [4]. Note that it does not define a signalling protocol, but provides primitives compatible with SIP signalling [11], that is, it just defines that SDP and offer/answer exchanges must be used, such that the endpoints can agree on the actual media flows to be exchanged.

In WebRTC we see the effect of the steady evolution of the browser as the new desktop, where all sorts of applications can be deployed. This mode of service delivery has several advantages:

- OS and platform independence–as long as it is supported in the browser;
- programmability;
- dynamic configuration without need for plugins;
- support for multiple concurrent communications;
- uniformity of features available (i.e. codecs) and hence predictability.

Overall, a service can be designed and configured in code that is loaded dynamically, per use.

On the down side, we should note the following:

- platform restrictions–the browser is an environment with limits in communications and storage access (i.e. sandboxing);

- no possibility of extension (additional codecs, other security mechanisms);
- no standard way of integration into a signalling environment.

The approach taken in the design of WebRTC is a "one size fits all" approach for media (Audio, Video, Data), while, on the signalling side, it is essentially non-committal, reduced to SDP–bias for the presentation of connection data. So, we have in effect a building block exploited through a JavaScript API. This leads to the questions, how good a building block is it, and how can it be extended?

## 3. Non-telephony applications

WebRTC easily supports two forms of communications, which we characterize as *mediated* and *asymmetric* and has been studied in connection with platforms such as IMS [5], in a *bridged* model.

### 3.1 Mediated communications

The Internet has–arguably–significantly enlarged the spectrum of communications. Most importantly, we have seen communications being embedded into larger scale applications, such as mail or social media, meaning that, while people are exchanging information, they could always choose to have a person-to-person communication, in the form of text, voice, or even video message recording.

This form of embedded communications benefits from features provided by the application, including:

- identity–an identifier which allows us to recognize the parties involved;
- presence–the notion of who is available;
- notification–means to notify some other party of a communication.

In this case, the context provides all the information required to establish a communication between two parties. The application is thus a *mediator* between two individuals who desire to exchange at a specific time.

Overall, such applications are browser-based, and have a signalling plane which can be extended to support communications and can take advantage of an embedded communication module.

### 3.2 Asymmetric communications

Another large class of applications which benefit from the availability of browser-based RTC is customer to business communications, for call-centres, eCommerce and the like. In the "click to call" model, it is trivial to add a simple button to a web site which, when clicked upon, will put a customer directly in touch with a company representative, increasing the customer-business bond. There is no need to pick up a different device, waiting becomes easier and can be combined with info-tainment, etc.

This only supports one form of communication, offered by the business but used entirely at the discretion of the customer.

### 3.3 The always-on Issue

Mediated communications and asymmetric communications can be readily deployed, but challenges remain in the peer to peer communication model. Telephony communications have evolved over time on the premise that a call shall always terminate–in the worst case in voice mail–and thus generate revenues. Calls can otherwise be forwarded from one device–or one person–to another, often targeting cellular phones which have an extended reach. This leads to the "Always-On" hypothesis: a party must always be reachable. This property rests on an infrastructure which provides some of the features we have mentioned above, that is, a global identity and a signalling mechanism to route and establish calls.

This brings us back to an infrastructure to support calls. The back-to-back user agent (B2BUA) model of IP telephony [11], supported by many soft switches (e.g. Asterisk) maps directly into a mediated communication model as described above. Similarly, bridges have been defined to integrate a WebRTC-based access into a standard telephony infrastructure, e.g. 3GPP's IMS [3].

## 4. Peer-to-peer communications

We have seen so far how WebRTC can be integrated into existing environments, to add new forms of user terminals, or new means of communications. One challenge that remains unanswered to this day is the re-creation of a true peer-to-peer form of communications. The study of a signalling plane for WebRTC is ongoing, and probably futile in light of the diversity of uses and the possibilities of integration with already existing infrastructures. We propose that an analysis of possible building blocks would be more relevant and we present below some key elements.

### 4.1 Quality of Service

RTC requires minimal levels of quality for the comfort of the users. In the realm of the telco, quality of service (QoS) has been a paramount concern, but recent history has shown that RTC services–characterized as Over the Top (OTT)–can offer satisfactory quality of experience without guaranteed quality of service levels.

While such services perform quite satisfactorily over wireline access, different forms of wireless access present a challenge for such services. It has however already been shown how standard QoS mechanisms in cellular networks can be exploited in connection with WebRTC [9]. We must not that it is, to this date, not possible to exploit them independently of a provider-operated environment.

### 4.2 Trusted Identification

It is important to know who is calling, and to be sure you are calling the right person. There is therefore a need for a service of strong identification, supported by an independent party [6].

In traditional telephony, the phone number serves this purpose, with the understanding that only one operator is responsible for the delivery of a call to a specific number. In environments where communications are embedded, authentication is done at a higher level.

Strongly related to identification is discovery and the whole issue of online identity, and whether identities can remain anonymous or not. This topic has caused much ink to flow and it is certainly clear that a unique model will not be sufficient. There are however two building blocks of interest: an independent white-page service, and a user-controlled identity generator, which will generate identifiers which can be communicated to other parties, including the white page service.

### 4.3 Reachability

Identity does not however guarantee reachability. Unlike telephone numbers which act both as identifiers and as addresses, we must be able to procure an address from the identity. This requires a location service.

That being established, the next issue is the connection of the terminal. Browser-based communications imply that the browser be open and a page remain active to keep the location service updated on the reachability of the user. This, in and per itself, remains an inconvenience, especially for mobility.

### 4.4 Real Presence in a Virtual World

The next dimension of service support, and a partial answer to the problem of reachability, is to establish a virtual presence to decouple access, which can be intermittent, from global reachability which is long term. This, again, is not a new concept and in the recent past companies have emerged offering this kind of service.

### 4.5 Features

Most users expect to have, associated with their communication service, a number of features[2]. Call display, selective call blocking and voice mail are typical examples. It must also be noted that the set of features in real use tends to be small, much to the chagrin of operators.

The mechanisms we have discussed so far allow to build such services: a call can be filtered based on caller identity (or lack thereof) and virtual presence would allow the support of voice mail. Further extensions to our building blocks would include storage for things such as filter lists, for example.

### 4.6 Extended functionalities

There will soon be a time when we will be running within the same browser a number of different communication features, as we are actually already doing through a number of applications–a rather uncomfortable proposition. At that stage, we will want to be able to integrate them into a communication environment and not be restricted by the page/tab nature of the browser, or its sandbox.

There are a number of issues in this proposition, including the design of new browser-based features (and respective APIs) to manage other dimensions of the communications, such as identity, address books and related support features. Beyond that, it could even require a new presentation model for browsers or a better way to relate tabs to ongoing background activities, as well as I/O device management. We can certainly expect progress in that area.

### 4.7 Discussion

Overall, we see that a peer-to-peer model requires cloud-based facilitation to overcome a number of limitations, some browser-based such as the stateless nature of the browser and its intermittent connectivity model, other networking-based such as the presence of middleboxes of various kind including the increasing need for IPv4-IPv6 interworking and finally more global issues of reachability.

None of these issues is in itself challenging, but we propose instead that the door is open for a large variety of integration models, which could be supported by a few building blocks, as we have described.

In fact, we appear to see a major transition in the industry with a proliferation of small companies expanding their products with communication features, offering communication features, often WebRTC related, for embedding in various forms of communications or also some support block for either. We can expect this trend to continue.

## 5. Conclusion

We have looked at the established uses of WebRTC through a characterization of embedded, asymmetric and bridged communications. Beyond these scenarios, we have looked at a number of building blocks required to support a true-er peer-to-peer use of browser-based RTC.

In this quick overview, we have exposed that there remains large opportunities for research on this topic, especially in the area of trusted identity management, on which we will concentrate our future efforts.

We must acknowledge that have not discussed any media issues in this paper. This is not to imply that WebRTC is beyond improvement, but rather that it was not our purpose. Many of the issues related to media transport have already been the focus of major research efforts and are beyond the scope of this paper.

*Acknowledgments* The author wishes to thank F. Oueslati and A. Chetoui and their work on WebRTC and the insight it has provided.

## References

[1] Isenberg, D. (1997). "Rise of the stupid network." Computer Telephony, 5(8), 16-26.

[2] Keck, D. O., & Kuehn, P. J. (1998). "The feature and service interaction problem in telecommunications systems: A survey." Software Engineering, IEEE Transactions on, 24(10), 779-796.

[3] Camarillo, G., & Garcia-Martin, M. A. (2007). The 3G IP multimedia subsystem (IMS): merging the Internet and the cellular worlds. John Wiley & Sons.

[4] Jennings, C., Uberti, J., & Rescorla, E. "Javascript Session Establishment Protocol." IETF draft, draft-ietf-rtcweb-jsep-08, Oct. 27, 2014.

[5] Bertin, E., Cubaud, S., Tuffin, S., Cazeaux, S., Crespi, N., & Beltran, V. (2013). "WebRTC, the day after." Service Architecture Lab research report, Apr. 2014.

[6] Muranyi, J., & Kotuliak, I. (2013, October). Identity management in WebRTC domains. In Emerging eLearning Technologies and Applications (ICETA), 2013 IEEE 11th International Conference on (pp. 289-293). IEEE.

[7] IETF Real-Time Communication in WEB-browsers (rtcweb) Working Group, http://tools.ietf.org/wg/rtcweb/charters.

[8] W3C Web Real-Time Communications Working Group, http://www.w3.org/2011/04/webrtc/.

[9] Lanczukowicz, E., Tuffin, S., Braud, A., Bouabdallah, A., Fromentoux, G., & Bonnin, J. M. (2014). Approaches for Offering QoS and Specialized Traffic Treatment for WebRTC. In Advances in Communication Networking (pp. 59-69). Springer International Publishing.

[10] Holmberg, C., Hakansson, S., and G. Eriksson, "Web Real-Time Communication Use-cases and Requirements", draft-ietf-rtcweb-use-cases-and-requirements-16 (work in progress), Jan. 2015.

[11] Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, "SIP: Session Initiation Protocol", RFC 3261, June 2002.