WHITEPAPER

Integrating WebRTC with Existing VoIP Networks

Executive Summary

WebRTC has the potential to be one of the most disruptive Web/telecoms innovations for years. According to Google, there will be 1 billion WebRTC enabled devices by the end of 2013, and Disruptive Analysis projects that there will be 3.9 billion WebRTC enabled devices and over 1.5 billion WebRTC users by the end of 2016.

This paper offers a perspective on how WebRTC service networks will evolve to interwork with and extend today's VoIP Service Networks.

Introduction

WebRTC has the capacity to fundamentally change the way the communications are leveraged in applications. For years the application focus of service providers has been communications. Not anymore. With WebRTC in particular, this paradigm is being shifted: The application now becomes the focal point of value creation, and communications is a consequence of these applications. The shift of business value from the communications infrastructure to the application has been evolving for years. WebRTC has the potential to fully empower the application to introduce new business models and drive higher revenues for service providers.

WebRTC, and in particular RTCWeb, represents a sea change in how endpoints, applications and communications infrastructures need to interact to conform to the new ways subscribers make use of real-time communications. The reward for this evolution will be more immersive and natural communications applications, such as eliminating the need for a dial pad and tedious voice directed menus to control subscriber interactions with automated attendants: "Press 1 if you speak English." Applications can be built that already know the context of the communication before it starts, and the service infrastructure must allow the application to fulfill the communications request in the flow of its operation. The new experience will be that the voice, video, messaging and data sharing just happens as part of your interactions with the application you are engaged in.

Dispelling Myths

We're already hearing comments in the industry heralding the death of service providers as we know them. That WebRTC doesn't need that sort of centralized infrastructure to operate. That finally, communications will live in the endpoints. Weren't those the same words echoing in the halls when SIP was being introduced to the world?

Although predictions of service provider extinction are premature, WebRTC will certainly have an impact on their revenue opportunities and competitive positions. It has the capacity to fundamentally change the way communications is blended into applications. The application now can become the primary driver of how people and machines will interact, not the communications network.

Based on history, we can make a bold statement: The communication-enabled application will never live completely in the endpoint, which in WebRTC's case is viewed as the browser. The endpoint will continue to play the role of renderer and user control. There will always be the need for a service network to provide connectivity assistance, media manipulations, persistent subscriber presence and value-added services.

One of the strengths of the WebRTC architecture is that the application can control communications session establishment. This ability is one reason for the perception that WebRTC doesn't require a service provider at all because client applications require only an HTTP network to connect. This may be the case for the simple point-to-point connections shown in Figure 1, where the application server is just a Web server.

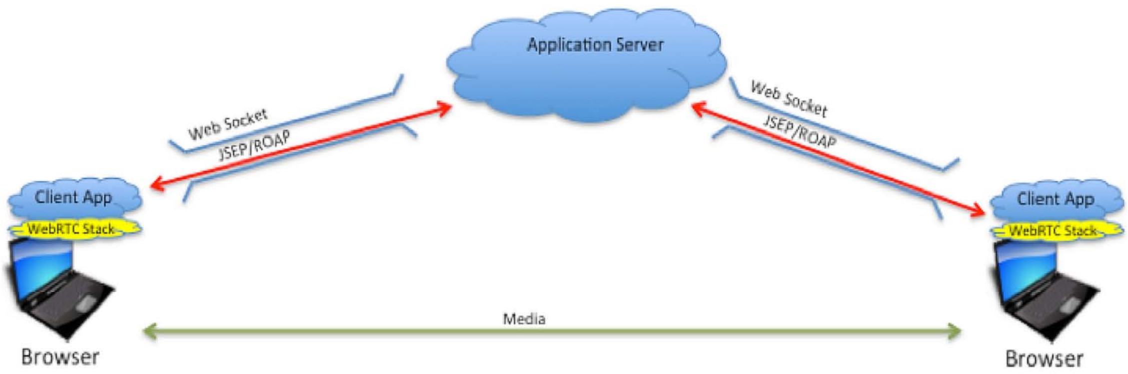

Figure 1

However, delivering enterprise- or carrier-class value-added features and services would require a centralized service network to act as the value engine. This can be as simple as a Web service or as complex as an operator. In order to make use of service networks that exist today, it is likely that any session establishment protocol will need to be adapted to existing VoIP protocols such as SIP.

The question then becomes what form does this take and how will it deliver on the promise of application enablement? As shown in Figure 2, the VoIP/SIP service networks that exist today do not have a consistently available API to enable the application to control it. Another element is going to be required to provide a path for the application to add the necessary functions to make WebRTC appear like any other client the service network is designed to work with.

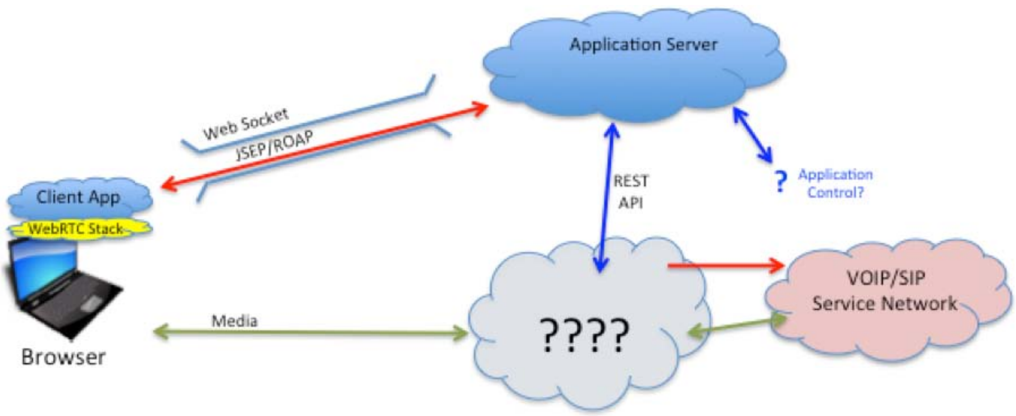

Figure 2

The Gateway Effect: Adapting to "Legacy" VoIP

Multiple connection session protocols are being considered for use in WebRTC clients and service networks. One of the early protocols considered is SIP Over WebSockets (SOW). Others, such as ROAP and JSEP, are more natural WebRTC protocols, but SOW has gained early mindshare because of its similarity to existing service network protocols.

SOW as a WebRTC session establishment protocol is a natural evolution of thought because SIP is the predominant session establishment protocol used in today's service networks. As such, it is believed to be straightforward to adapt to communicate over the existing SIP VoIP infrastructures. The gateway or SBC function that connects the endpoint to the core SIP switching infrastructure is simplified to a basic bridging of the session control.

But at what cost? SOW pushes the communications establishment complexity and intelligence back to the service network and away from the application, the very thing that makes WebRTC so valuable. So, in effect, SOW relegates the WebRTC client to being little more than a browser-based SIP client and forces the client to bridge communications to the application.

SOW also requires that the WebRTC endpoint (browser) now needs to have a plug-in that contains a SIP stack, wrapped in a JavaScript module, adding more complexity while delivering less innovation. The combined effect is that the browser application plug-in becomes complex and is now a point of failure in the communications infrastructure!

The first step in addressing this endpoint complexity and service quality issue, and adapting WebRTC to a legacy VoIP network, is to introduce a gateway to the solution. This removes many of the quality issues from the endpoint and puts service delivery control back into the hands of the service network.

Of course, the WebRTC-SIP gateway, shown in Figure 3, is more than a simple signaling exchange between the WebRTC client and the VoIP service network. In addition to the signaling, the gateway must also adapt the characteristics of the WebRTC client to match the characteristics of a legacy client that the VoIP service network was designed for.

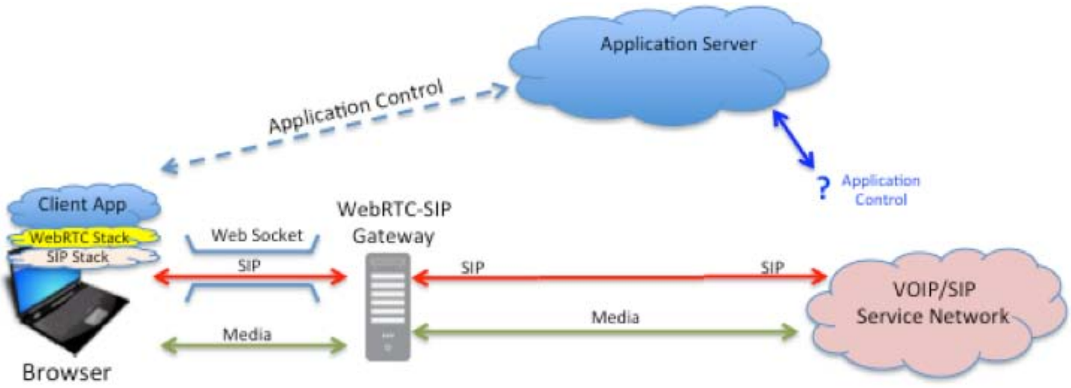

Figure 3

An even bigger issue that must be addressed is handling Web security issues such as CORS (Cross-origin Resource Sharing). A question for the industry now is: Where does this WebRTC-SIP gateway functionality reside? Is it a separate element that front ends the service network? Is it built into an existing element such as an SBC, media server or legacy application platform?

As a separate element, the WebRTC-SIP gateway adds cost and complexity to the service network while adding no value to the application enablement that WebRTC promises to unleash. As you can see, the application has been ignored in this solution; nothing has been added to the service network to satisfy the delivery of a new application enablement capability.

Let's not forget that most service networks will still require the inclusion of an SBC to provide security, session routing, protocol adaption and scalability. Very few SBCs today provide visibility and support for applications and application enablement.

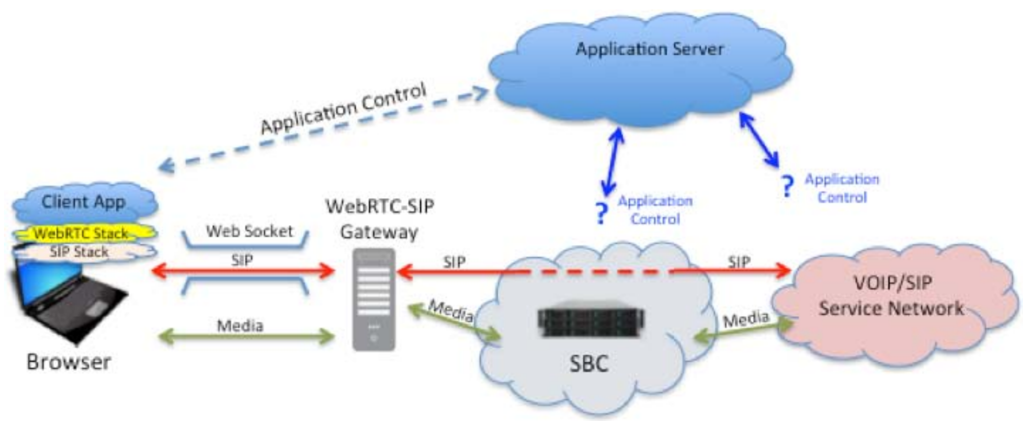

Figure 4

A solution to the issues identified above is to evolve the SBC into a new class of border control element that encompasses the functions being considered in the WebRTC-SIP gateway and creates an interface to provide session visibility and control to the application. Figure 5 illustrates this new class of SBC that forms an RTCWeb Service Network Edge element, which embodies all of the functionality needed to manage bridging the WebRTC client and VoIP/SIP service network behaviors.

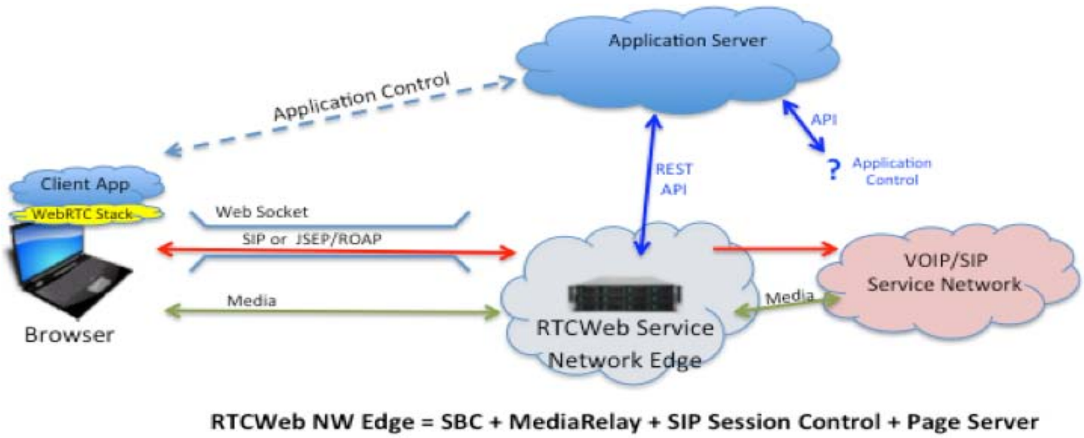

Figure 5

SOW is a reasonable short-term measure to leverage existing VoIP infrastructure and applications while the WebRTC standards and service-enabling technologies are being defined and developed. However, as we can see, this is not a good long-term solution to enable more immersive and natural communications through applications.

The same issues are present in other session establishment protocols (e.g., ROAP, JSEP) under consideration. So it is reasonable to assume that any session establishment protocol adopted will require some form of gateway plus the additional endpoint support functions outlined above.

The Evolution of the Service Network

If the gateway is the first step, what's next? The service network side of the WebRTC infrastructure is being defined through an additional effort: RTCWeb. RTCWeb is represented by a series of IETF drafts that cover the service network side of the WebRTC infrastructure. The early focus of these drafts is on session establishment. However, this is also where new service capabilities will be defined that will cover more of the service value creation aspects necessary to involve the application layer more closely with the session establishment.

New service network capabilities necessary for enhanced application layer support will need to include device and application registration and control, application session routing, subscriber identity mapping and management, subscriber presence and persistence,and notifications. Sansay is among the vendors already trialing platforms that encompass many of these features and more.

Conclusion: Gateway to the Future

WebRTC-SIP gateway functionality is an early means of adapting the emerging WebRTC-enabled endpoints into existing service networks. The most cost-effective and secure means of providing this gateway functionality is through evolving today's SBC into a new class of platform capable of not only protocol adaption but also providing the application enablement capability required to fulfill the promise of the new WebRTC infrastructure.

The Sansay WebSBC™ platform is positioned to provide the WebRTC gateway functionality needed to bridge these WebRTC endpoints into the existing VoIP service network and applications while providing a growth path for its customers to evolve to the new application enablement capabilities promised by the emerging WebRTC infrastructure.

Corporate Headquarters:
Sansay, Inc.
4350 La Jolla Village Dr. Ste. 888
San Diego, CA 92122

Toll Free: +1 888.889.8906
Office: +1 858.754.2200
Fax: +1 858.550.2044

www.sansay.com