

[print](#) | [close](#)

## WebRTC And V.VoIP: Friends Or Enemies?

[Electronic Design](#)

[Saraj Mudigonda](#)

Saraj Mudigona, Contributing Technical Expert

Wed, 2014-02-26 10:25

There is some confusion in the industry today, as WebRTC has emerged as a viable IP-based (Internet Protocol) communications solution alongside traditional voice and video over Internet Protocol (V.VoIP). With strong industry backing, WebRTC is now heavily promoted. Are these technologies complementary or competitive? Can they coexist, and how can such coexistence benefit the end user? What are the use cases for each, and where is there overlap?

### Related Articles

- [V.VoIP Adds Value To Next-Generation Applications](#)
- [Choose The Right Video Technology For Your Real-Time V.VoIP](#)
- [HEVC Improves Mobile Video Performance](#)

WebRTC and V.VoIP both aim to enhance the user experience and enable any consumer device to seamlessly connect from anywhere and on any network. But while V.VoIP over the past decade has been deployed in different variants such as VoIP over DSL/cable modem, voice over Wi-Fi/3G (VoWiFi/3G), voice over LTE (VoLTE), and Rich Communication Suite (RCS), WebRTC primarily is focused on browser-based communications.

### V.VoIP Building Blocks

Some of the essential elements of V.VoIP include signaling, media engine, Session Description Protocol (SDP), Real-Time Protocol/Real-Time Control Protocol (RTP/RTCP), Network Address Translation (NAT), security protocols, quality of service (QoS), and other telephony components. V.VoIP essentially wraps all of this with a user interface that includes a dialer, address book/contacts list, and call history for missed/received/dialed calls to provide a complete V.VoIP client.

Signaling primarily is used to establish, maintain, and terminate calls between two or more users. Some of the popular signaling protocols include Session Initiation Protocol (SIP), H.323, and Extensible Messaging and Presence Protocol (XMPP). SIP is the most widely deployed signaling protocol. It handles all call management and supplementary features such as call forwarding, call waiting, call transfer, and other Class 5 call features. SIP also supports multi-way conferencing. In conjunction with a media server, it can support mixing of multiple video and voice channels as well. SIP servers, on the other hand, help locate and register user locations and provide the ability to send/receive messages.

A media engine comprises two key components: a voice engine and a video engine. A voice engine includes voice pre-processing, voice codecs, voice activity detection (VAD), and comfort noise generation (CNG). A video engine consists of a video codec, audio/video lip sync, video jitter buffer, and other video components.

As part of the media engine, voice pre-processing includes acoustic echo cancellation (AEC), which removes the acoustic echo from the audio channel; noise cancellation (NC), which removes ambient noise; and automatic gain control (AGC), which maintains a consistent audio level.

There also is a wide range of narrow-band (sampled at 8 kHz) voice codecs that support as low as 4.75 kbits/s (AMR-NB codec) to 64 kbits/s (G.711 codec). V.VoIP has now transitioned to HD (sampled at 16 kHz) voice such as AMR-WB for superior voice quality. The most widely used video codec is H.264 AVC. Some legacy systems still use H.263, and some enterprises use H.264 SVC. They all eventually will migrate to H.265, which will reduce the bitrate by 50% while maintaining the same quality as H.264.

V.VoIP supports Transport Layer Security (TLS) for signaling and the Secured Real-Time Protocol (SRTP) for media. TLS provides communications security over the Internet. It supports privacy and data integrity between two communicating applications. TLS also is used to provide authentication and encryption of SIP signaling. SRTP provides confidentiality, message authentication, and replay protection to the RTP traffic and to the control traffic for RTP and the RTCP.

Lastly, V.VoIP supports the Interactive Connectivity Establishment (ICE) protocol in combination with Session Traversal Utilities for NAT (STUN) and Traversal Using Relay NAT (TURN), which enables media flows through corporate firewalls and NAT boxes to establish connections across various types of networks.

### **WebRTC Building Blocks**

WebRTC is a real-time voice and video communication engine that primarily works in the context of the browser. So how does it differ from V.VoIP?

The WebRTC media engine is quite similar to the traditional V.VoIP media engine integrating the elements discussed above, such as ICE/STUN/TURN, security protocols, RTP/RTCP, SDP, and audio/camera/display interfaces needed for secure peer-to-peer video communication. It also integrates the Opus full-band voice codec and VP8 video codec.

WebRTC is a shrink-wrapped software package with well-defined application programming interfaces (APIs) that make it easy for Web developers to enable V.VoIP in their Web-based applications. It does not include any signaling protocol, though, leaving this choice and development/procurement/integration to the developer. By integrating such a signaling protocol into WebRTC, one can create a full V.VoIP soft client on a browser.

In addition to a media engine, WebRTC provides a data channel. Data transfer is usually considered to be non-real-time with buffering and multiple re-transmissions over the TCP connection. However, the WebRTC data channel enables a low-latency peer-to-peer User Datagram Protocol (UDP) connection between the browsers, which is ideal for applications such as interactive multi-party gaming, file sharing, and screen sharing. WebRTC also enables concurrent data transfer and video conferencing operations. It supports Datagram Transport Layer Security (DTLS) for secure UDP data connection.

To set up a WebRTC peer-to-peer session, WebRTC provides simple APIs such as `getUserMedia`, `RTCPeerConnection`, `RTCDataChannel`, and `RTCSessionDescription` to respectively obtain local audio and video interfaces, establish a connection between peers, attach audio/video streams and arbitrary data, and exchange session descriptions. But what does it take for legacy V.VoIP providers to support WebRTC?

WebRTC can be supported with minimal changes on the server side. It's browser-based, though, so it needs a Web server in addition to traditional SIP servers. Most of the traditional V.VoIP ecosystem today supports G.7xx/AMR voice codecs

and H.264 video codecs while WebRTC supports the G.711/Opus voice codec and VP8 and/or H.264 codec (not finalized yet, still under discussions). Some transcoding then may be required, which would increase latency and costs. Efforts are well underway to address the interoperability concerns.

### **Conclusion**

WebRTC is an extension of V.VoIP to the browser world. It can reuse the existing V.VoIP infrastructure with incremental upgrades. This is good news for V.VoIP, as adoption of WebRTC only serves to increase overall V.VoIP proliferation.

Also, WebRTC is ideal for low-cost browser-based contact center applications. V.VoIP can serve embedded operator-driven VoLTE applications. Consequently, between WebRTC and V.VoIP, they can support wide range of consumer and enterprise applications.

As with any technology in its early adoption, there is room for improvement. An obvious solution seems to be combining the complementary WebRTC and V.VoIP technologies, leveraging the optimizations for battery consumption, audio and video interfaces, and the infrastructure already in place for V.VoIP deployments. There is continual effort to integrate WebRTC in all forms of V.VoIP, and we will see more of this happening very soon.

*Saraj Mudigonda is the business development manager for Imagination Technologies USA. He can be reached at [saraj.mudigonda@imgtec.com](mailto:saraj.mudigonda@imgtec.com).*

**Source URL:** <http://electronicdesign.com/communications/webrtc-and-vvoip-friends-or-enemies>