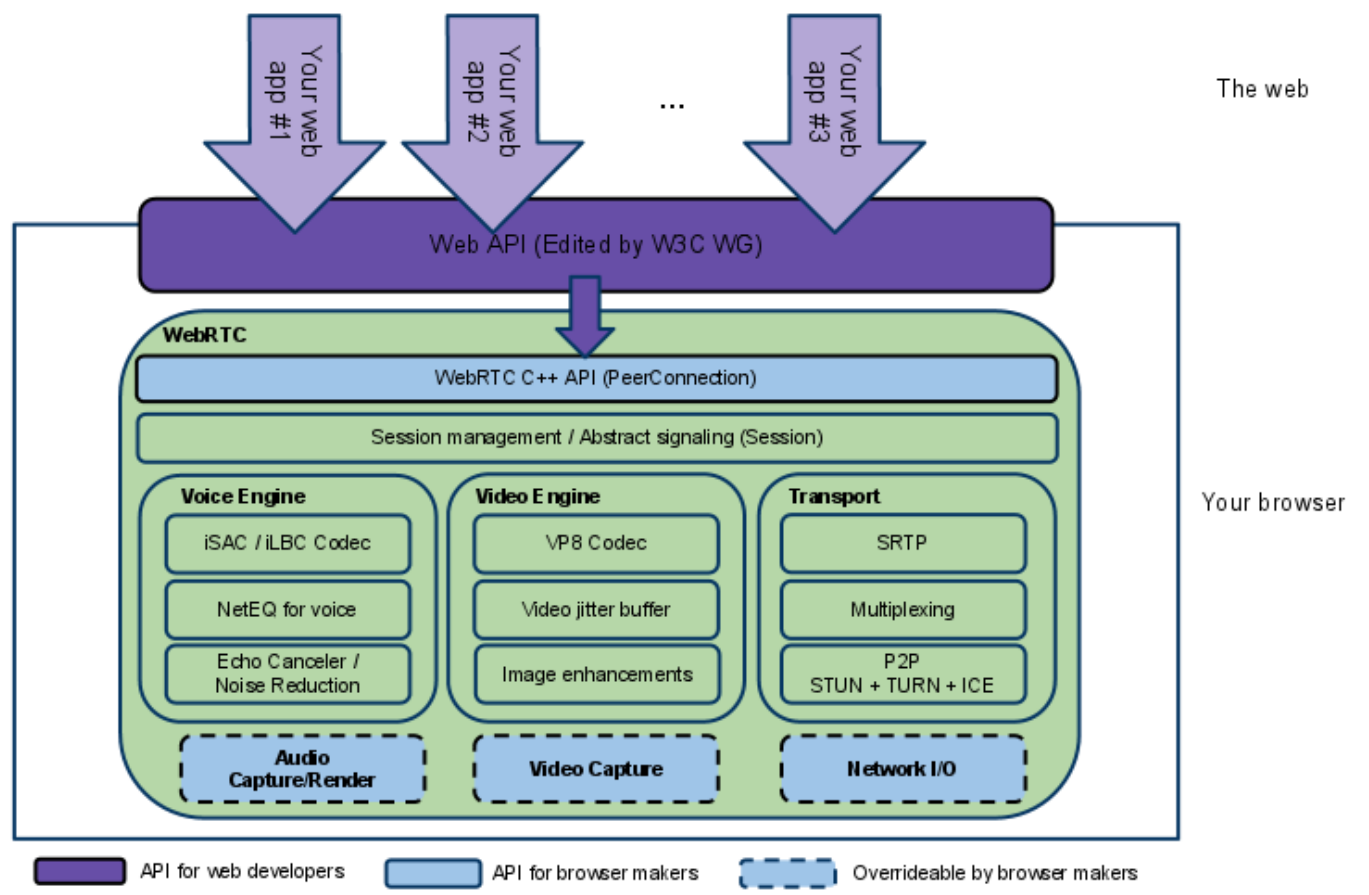


# Architecture | WebRTC

WebRTC offers web application developers the ability to write rich, realtime multimedia applications (think video chat) on the web, without requiring plugins, downloads or installs. It's purpose is to help build a strong RTC platform that works across multiple web browsers, across multiple platforms.

The overall architecture looks something like this:



You will notice two distinct layers.

1. Browser developers will be interested in the WebRTC C++ API and the capture/render hooks at their disposal.
2. Web App developers will be interested in the Web API.

## Your Web App

A third-party developer web based application with video and audio chat capabilities powered by the web API for real time communications.

## Web API

An API to be used by third party developers for developing web based videochat-like applications. Latest proposal can be found [here](https://webrtc.org/architecture/#).

## WebRTC Native C++ API

An API layer that enables browser makers to easily implement the Web API proposal.

### Transport / Session

The session components are built by re-using components from libjingle, without using or requiring the xmpp/jingle protocol.

### RTP Stack

A network stack for RTP, the Real Time Protocol.

### STUN/ICE

A component allowing calls to use the STUN and ICE mechanisms to establish connections across various types of networks.

### Session Management

An abstracted session layer, allowing for call setup and management layer. This leaves the protocol implementation decision to the application developer.

### VoiceEngine

VoiceEngine is a framework for the audio media chain, from sound card to the network.

### iSAC / iLBC / Opus

iSAC: A wideband and super wideband audio codec for VoIP and streaming audio. iSAC uses 16 kHz or 32 kHz sampling frequency with an adaptive and variable bit rate of 12 to 52 kbps.

iLBC: A narrowband speech codec for VoIP and streaming audio. Uses 8 kHz sampling frequency with a bitrate of 15.2 kbps for 20ms frames and 13.33 kbps for 30ms frames. Defined by IETF RFCs 3951 and 3952.

Opus: Supports constant and variable bitrate encoding from 6 kbit/s to 510 kbit/s, frame sizes from 2.5 ms to 60 ms, and various sampling rates from 8 kHz (with 4 kHz bandwidth) to 48 kHz (with 20 kHz bandwidth, where the entire hearing range of the human auditory system can be reproduced). Defined by IETF RFC 6176. NetEQ for Voice

A dynamic jitter buffer and error concealment algorithm used for concealing the negative effects of network jitter and packet loss. Keeps latency as low as possible while maintaining the highest voice quality.

### Acoustic Echo Canceler (AEC)

The Acoustic Echo Canceler is a software based signal processing component that removes, in real time, the acoustic echo resulting from the voice being played out coming into the active microphone.

## Noise Reduction (NR)

The Noise Reduction component is a software based signal processing component that removes certain types of background noise usually associated with VoIP. (Hiss, fan noise, etc...)

## VideoEngine

VideoEngine is a framework video media chain for video, from camera to the network, and from network to the screen.

## VP8

Video codec from the [WebM Project](#). Well suited for RTC as it is designed for low latency.

## Video Jitter Buffer

Dynamic Jitter Buffer for video. Helps conceal the effects of jitter and packet loss on overall video quality.

## Image enhancements

For example, removes video noise from the image capture by the webcam.