



# WebRTC 关键技术分析及运营商引入策略研究

屈振华,李慧云,张 凌,杨新章,龙显军

(中国电信股份有限公司广州研究院 广州 510630)

**摘 要:** WebRTC 是一项在浏览器内部进行实时视频和音频通信的技术,可利用 JavaScript API 和 HTML5 实现丰富且质量高的实时通信。分别介绍了 WebRTC 在信令和媒体处理方面的关键技术,分析了 WebRTC 的业务模式和可能对传统运营商带来的冲击,并就运营商引入 WebRTC 的策略提出了建议。

**关键词:** WebRTC; 多媒体; 运营商; 运营商 OTT

**doi:** 10.3969/j.issn.1000-0801.2013.Z1.029

## Key Technologies and Strategies Analysis of WebRTC for Telecom Operator

Qu Zhenhua, Li Huiyun, Zhang Ling, Yang Xinzhang, Long Xianjun

(Guangzhou Research Institute of China Telecom Co., Ltd., Guangzhou 510630, China)

**Abstract:** WebRTC is a newly emerged trend of real-time communications. It enables the developers to realize rich, high quality, RTC applications in the browser via simple JavaScript API and HTML5. The key technologies within the WebRTC architecture including both the signaling and media plane were addressed, and some strategically analysis for the probable business model or impact were given when introducing WebRTC to the telecom operators.

**Key words:** WebRTC, multimedia, operator, telco-OTT

### 1 引言

Web 技术正在面临前所未有的巨大变革,Google 最新倡导的 WebRTC(web based real-time communication)技术正使得音视频媒体的采集、播放及传输逐渐摆脱浏览器插件(如 Adobe Flash Player、ActiveX、Java Applet 等)的控制。Web 应用开发者仅需要进行简单的 JavaScript API 调用,即可在两个浏览器间轻松实现双向多媒体实时通信。WebRTC 技术的诞生源自基于互联网的多媒体通信业务模式与 Web 应用开发模式的有机结合,P2P 的业务模式使得多媒体通信“碎片化”、“去中心化”的趋势更加明显,低成本、跨平台、组网灵活等天生优势使得其更易于快速规

模推广。可以预见,原本竞争激烈的多媒体实时通信市场即将面临新一轮的洗牌,传统运营商在这场变革中将何去何从?本文将分别介绍 WebRTC 在信令和媒体处理方面的关键技术,分析 WebRTC 的业务模式和可能对传统运营商带来的冲击,并就运营商引入 WebRTC 的策略提出建议。

### 2 WebRTC 关键技术

WebRTC 技术框架最初被设计用于实现浏览器间的 P2P 通信,现有规范<sup>[1,2]</sup>主要定义了浏览器侧(客户端)的接口与交互方式。如图 1 所示,WebRTC 浏览器侧的功能模块至少包括音频引擎、视频引擎、网络传输、WebRTC 的本地 API 以及对音频采集、视频采集和网络 I/O 的接口。浏



浏览器需要实现音频采集、视频采集、网络 I/O 的具体业务逻辑和功能模块,并通过统一、与平台无关的 native API 与 JavaScript API 向本地或 Web 应用开发者提供功能调用。现有功能模块大致可以划分为信令层面与媒体层面两部分。

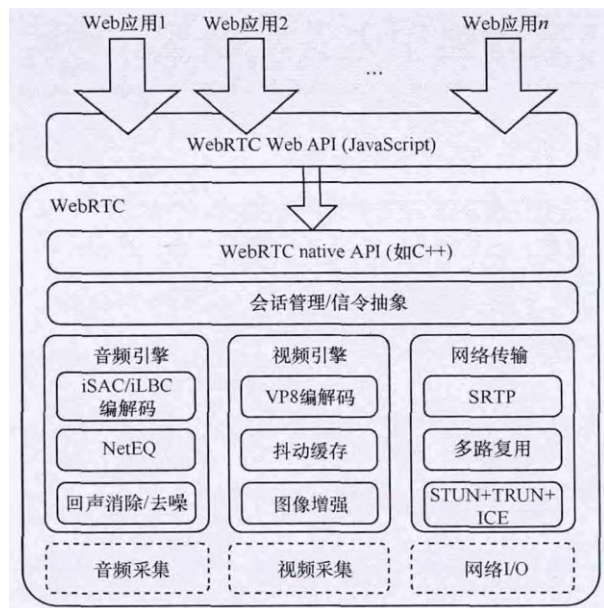


图 1 浏览器侧功能架构

## 2.1 信令层面关键技术

WebRTC 的信令消息由两部分组成:一部分为承载协议,另一部分为信令协议,如图 2 所示。

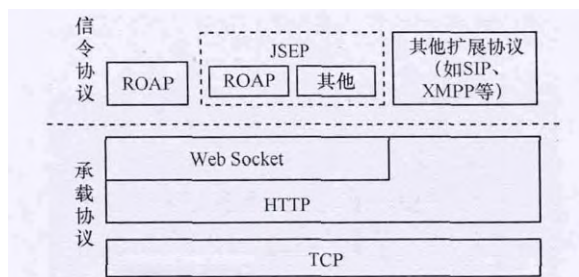


图 2 信令协议关系

### (1) 承载协议

目前 WebRTC 信令的承载协议可以采用 Web Socket 协议或传统的 HTTP,以支持浏览器和服务器或者浏览器和浏览器之间维持双向的连接通道。Web Socket 和 Socket 技术本质上一样,均是基于 TCP 的协议,所不同的是,在建立 Web Socket 连接之前,首先需要通过 HTTP 消息进行协商(HTTP 经过了扩展),协商完成后建立 TCP 双向通道,直到客户端或者服务器端的某一方主动关闭连接;基于

HTTP 的轮询、长轮询或流技术并不是真正的实时技术,只是在用 Ajax 方式模拟实时的效果,在每次客户端和服务端交互时都是一次 HTTP 请求和应答的过程,而每一次的 HTTP 请求和应答都带有完整的 HTTP 头信息。从传输的数据量大小、编程复杂度以及实时通信的效果看,Web Socket 都优于基于 HTTP 的轮询、长轮询和流技术。

### (2) 信令协议

WebRTC 的信令协议并没有硬性的标准,目前常用的信令协议包括 ROAP<sup>[3]</sup>、XMPP (Jingle)、SIP 以及 JSEP<sup>[4]</sup>等。其中,JSEP 从严格的角度来讲,可以说是一种协议编程框架,它定义了一组 JavaScript API 以及调用的顺序,在 API 中封装了浏览器和应用平台/浏览器之间的信令交互过程,这样开发者只需要关注业务逻辑和信令内容的构造,大大降低了开发的难度。在 JSEP 中,信令内容可采用 XMPP (Jingle 协议)、SIP、ROAP 或私有协议,因此可以说,JSEP 仅是对信令协议的更高级封装。

## 2.2 媒体处理层关键技术

WebRTC 为实现多媒体实时通信提供了从音视频采集、编解码到网络传输控制等一系列必要的软件处理模块,并将它们组成一个完整的媒体处理架构。WebRTC 采用了当前较为先进的音视频编解码格式,并根据 VoIP 应用场景进行针对性优化。在软硬件架构设计方面,也充分考虑到平台间的差异以及今后的扩展能力。

### (1) 媒体采集和播放

WebRTC 提供了多平台下的媒体采集和播放功能,如音频设备、视频捕捉、视频渲染等与硬件相关的模块,根据不同的操作系统进行量身定制,并封装为统一的 API。例如,视频渲染模块就是根据 Windows、Mac、Linux、Android 等不同窗口系统,分别调用 DirectX、Carbon、X11、Java 等本地 API 进行视频帧绘制。这部分 API 的维护需要对操作系统有较为透彻的认识,由开发者自行维护的难度很大,受限于 Google、微软、苹果等操作系统开发商。

### (2) 媒体增强

WebRTC 提供了包括高通滤波、噪声抑制、回声消除、自动增益控制、静音检测等语音增强功能。视频处理包括颜色增强、去闪烁、下采样、去噪。这些媒体信号处理操作在媒体编码前或解码后执行,对媒体播放或编码起到辅助作用,可以提升音视频通信中的用户体验。虽然部分媒体采集/播放设备或操作系统也提供类似功能,但考虑到不同设备商或系统平台间的驱动及接口差异,WebRTC 对这些功能提

供了软件实现,以便在无法使用硬件时进行补充。

WebRTC 媒体增强模块的特点是实现代码的高度整合,例如,共用部分数据结构、宏、底层汇编优化指令等,虽然会增加代码间的耦合度,但有利于代码的进一步优化。例如,只需要对部分通用函数,如卷积、相关等,采用 SSE、NEON 等 SIMD 指令进行优化,就可以实现整体性能的提升。PC 平台与嵌入式平台间处理性能的差异,导致 WebRTC 需要对不同平台分别提供不同的算法实现或代码优化,但受嵌入式设备硬件处理能力所限,目前仍然存在一定的性能瓶颈,需要在未来进一步完善。

### (3) 媒体编码

Google 实现的 WebRTC 音频引擎中集成了 G.711a/u、iLBC(RFC 3951)<sup>[5]</sup>、iSAC、G.722<sup>[6]</sup>、Opus(RFC 6716)<sup>[7]</sup>等音频编解码器,其中 G.711a/u、iLBC 属于窄带语音编码器,而 iSAC、G.722、Opus 属于宽带语音/音频编码器。部分编码器(如 iLBC、Opus)针对互联网分组丢失问题采用了分组丢失隐藏技术,配合 GIPS 独有的 NetEQ 技术,有效地提高了分组丢失网络下的通话体验。其中特别值得关注的是 Opus 编码,其由 Mozilla 和 Xiph.org 基金会开发,是一种窄带到全频段的音频编解码器,支持动态可调的码率、帧长和音频带宽,是目前最先进的音频编码器之一。Opus 作为一种全能型的音频编码器,很可能在未来全面取代其他编码器,成为 WebRTC 的首选。Opus 也面临着复杂度较高、缺少硬件实现以及专利纠纷等问题,需要时间进一步检验。

相对音频编解码而言,不同视频编码间的差异性较小,专利成为影响编码标准成功的重要因素。VP8 视频编码(RFC 6386)<sup>[8]</sup>源自被 Google 收购的 On2 Technologies 公司,现已成为 WebM 开源项目的一部分。免费与开源是 VP8 格式的最大优势,得到 Chrome、Firefox、Opera 等厂商的普遍支持,自然成为 WebRTC 的首选。事实上,WebRTC 对视频编码的选用,一直存在 VP8 和 H.264 之争。Google 虽然明确表示其 Chrome 浏览器中的 WebRTC 不会支持 H.264,但 H.264 由于牵涉众多业界巨头的商业利益,软硬件实现都要比 VP8 成熟得多,已经成为视频通信事实上的标准;而 VP8 最大的问题就是和 H.264 实在太像,其使用的许多技术(如宏块划分、帧内预测等),几乎是照搬 H.264 的方法,由此也带来很多专利纠纷。最近 Google 与 MPEG-LA 已经达成专利授权协议,进一步扫除了 VP8 应用的障碍,但诺基亚又提出对 VP8 的专利侵权起诉。Google 也在积极推动新一代编码技术 VP9 的应用,

在将要发布的 Chrome 28 中支持 VP9。

### (4) 网络传输

WebRTC 的媒体流传输采用标准的 RTP/SRTP(secure RTP, RFC 3711),并提供一种基于 ICE(STUN+TURN)<sup>[9-11]</sup>的私网穿越机制。由于 WebRTC 采用了全自动的网络传输控制,用户几乎无需对媒体流的传输过程进行干预。尽管 WebRTC 为点对点的通话场景而设计,但 RTP/SRTP 流既可以用于单播也可以用于实现多播,这意味着通过媒体流重定向或拆分/汇聚,有可能实现视频监控、媒体广播、视频会议等目前不具备的新功能。对于运营商而言,需要特别注意的一点是,WebRTC 默认采用 SRTP 进行媒体流传输,这一方面提高了媒体传输的安全性,但同时也造成今后的监管风险。

## 3 WebRTC 业务模式

WebRTC 目前有两种主要的业务提供模式:P2P 方式和 Web2 Server 方式。

### (1) P2P 方式

如图 3 所示,在 P2P 方式下,Web 浏览器之间进行协议和媒体的点对点通信交互,媒体流可以取道最短路由,特别适合局域网、企业网等带宽有保证的通信场景。这种方式受到以 Google 为首的新兴互联网商、虚拟运营商、应用提供者以及个人开发者的青睐,应用的场景包括在线客服、远程教学、培训、外勤汇报、垂直网站等。

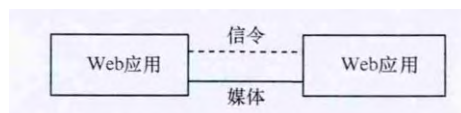


图 3 P2P 方式组网

### (2) Web2 Server 方式

如图 4 所示,在 Web2 Server 方式下,服务提供商需要部署 WebRTC 服务器实现 WebRTC 之间的信令互通,并可通过网关与其他业务系统(如传统 PSTN/GSM/CDMA 或 IMS 等)互通。这种方案需要考虑服务器架设、网关部署、数据承载等一系列问题,主要阵营是传统运营商、通信设备制造商、企业解决方案提供商以及部分 OTT 企业。应用场景包括实时会议系统、呼叫中心、融合信息服务、社交应用、视频监控等。

## 4 WebRTC 对运营商的影响

WebRTC 为运营商带来的影响是多方面的,既带来新



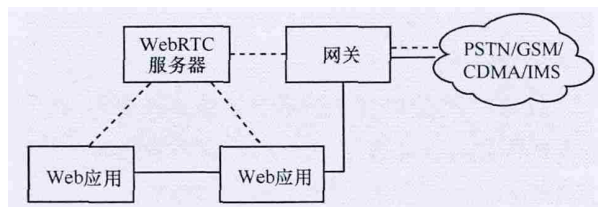


图4 Web2 Server方式组网

的挑战,也提供了新的机遇。

#### 4.1 WebRTC带来的新挑战

WebRTC大幅降低了多媒体实时通信应用的开发门槛,为市场带来更多新的竞争对手,直接冲击运营商的语音业务。

(1)加剧通信碎片化趋势,分流运营商的传统语音业务

OTT业务的出现,已经促使通信的碎片化,并分流了运营商的语音业务;而WebRTC技术允许开发者在网页中建立实时通信,使语音业务成为一种可以不依赖运营商平台的原子能力。目前成熟的互联网应用集成WebRTC后可能产生更丰富的应用形态,对现有运营商的融合通信产品产生的冲击更大。可以预见,当WebRTC能够简单、低成本地集成到这些应用中时,通信的碎片化趋势将加剧。

(2)“去中心化”的趋势更加明显

在WebRTC的体系架构中,平台的作用被弱化,媒体协商可以不通过平台完成。WebRTC需要平台协助的功能集成到已有的业务平台中,而不需要额外部署网元。这种业务模式大大降低了部署的难度和成本,也使得一些运营商的传统优势业务可以不依赖运营商的业务平台实现(如呼叫中心等),使得运营商沦为数据通道。

#### 4.2 WebRTC带来的新机遇

WebRTC技术将改变目前的OTT竞争格局,给传统运营商应对新兴OTT(如VoIP提供者)的威胁提供新的手段。如果能够充分利用运营商的现有网络、产品以及用户资源优势、快速反应能力,也可能将WebRTC作为运营商的业务创新点和赢利点。

(1)WebRTC可以降低新业务的开发部署成本

WebRTC具有易开发、跨平台、部署成本低等优势,可以吸引众多的Web应用开发者基于运营商通信业务平台开发应用。同时,将WebRTC和运营商的平台、网络优势结合,可优化行业解决方案,帮助运营商降低终端应用开发成本,快速推出新的业务产品,并扩展和优化行业应用解决方案,补充运营商开放平台的能力。

(2)以WebRTC推动IMS/VoLTE业务发展

WebRTC技术的应用可能推进IMS业务,特别是VoLTE的部署。国内IMS网络经历多年发展,终端成本居高不下、用户迁移缓慢一直是其发展面临的两大难题。通过将IMS和WebRTC结合,可以极大地降低IMS终端的开发门槛,为用户提供基于宽带、低时延网络环境的优质多媒体解决方案。如果能够与未来LTE网络的发展策略切合,实现VoLTE业务的快速上线,将能够为运营商带来巨大的利益。

#### 4.3 运营商的应对策略

最近几年,运营商一直在学习和研究互联网的业务开发和运营模式,但受限与运营商在互联网领域的薄弱基础,一直没有有竞争力的互联网应用产品,WebRTC的应用场景可以发挥出运营商的优势,找到Telco-OTT(运营商-OTT)业务的切入点,改变目前的现状。同时也要注意,WebRTC目前仍处于发展阶段,存在的技术、监管和产品/解决方案成熟度等风险以及贡献方态度迥异可能引起技术方向的不确定性,使得运营商在积极参与WebRTC研发的同时,需维持谨慎的态度。因此,采用循序渐进、从点到面的引入策略是比较合适的选择。

建设初期可以考虑从增值业务入手,采用基于Web2 Server/网关方案,通过与现有网络(如IMS等)互通或者Telco-OTT方式,与现有业务形成互补。例如,可以通过IMS提供基础、有保障的VoLTE业务,以WebRTC方式提供Telco-OTT模式的增值业务以及扩展现有网络和业务能力。从中长期来看,可以考虑把WebRTC技术作为富媒体融合通信的基础解决方案之一,使用WebRTC native API定制开发电信运营商自己的浏览器或服务器,实现功能更复杂的融合通信产品。

从市场需求方面分析,WebRTC技术在公众客户与企业客户市场都可以有所作为。对于公众用户而言,主要从拉升流量方面考虑;对于企业用户而言,主要从更完善、更灵活、更低成本的解决方案考虑。公众市场可以考虑间接方式,即提供SDK/API分组给Web应用开发者,引入更多的开发者,将能力嵌入各种互联网应用中,以吸引用户,拉动流量持续增长。企业市场可以采用直接提供服务的模式,着力发展垂直行业市场,如基于Web语音/视频呼叫、Web会议、在线培训以及企业内部的协同业务。

## 5 结束语

目前,WebRTC技术正处于其成长期,在标准、技术实

现上都不成熟,正是运营商切入的最佳阶段。中国电信在多媒体业务领域(无论是终端还是平台方面)拥有丰富的经验,及时切入 WebRTC 的研究,引导 WebRTC 的技术发展方向,就有可能抓住新的发展机会。WebRTC 作为一项新型技术,从产生到真正产品化,还有许多细节需要完善,这也有待于进一步的观察和更深入的研究。

## 参考文献

- 1 Bergkvist A, Burnett D C, Jennings C, *et al.* WebRTC 1.0: Real-time Communication Between Browsers W3C Editor's Draft 03, 2013
- 2 Burnett D C, Narayanan A. Media Capture and Streams. W3C Editor's Draft 30, 2013
- 3 Jennings C, Rosenberg J, Uberti J, *et al.* RTCWeb Offer/Answer Protocol (ROAP), 2011
- 4 Uberti J, Jennings C. JavaScript Session Establishment Protocol (JSEP), 2012
- 5 Andersen S, Telio D A, Astrom H, *et al.* Internet Low Bit Rate Codec (iLBC). RFC3951, 2013
- 6 ITU-T Recommendation G.722. 7 kHz Audio-Coding Within 64 kbit/s, 1988
- 7 Valin J M, Vos K, Terriberry T. Definition of the Opus Audio Codec. RFC6716, 2012
- 8 Bankoski J, Koleszar J, Quillio L, *et al.* VP8 Data Format and Decoding Guide. RFC6386, 2011
- 9 Rosenberg J, Mahy R, Matthews P, *et al.* Session Traversal Utilities for NAT (STUN). RFC5389, 2008
- 10 Mahy P, Matthews P, Rosenberg J. Traversal Using Relays around NAT (TURN): Relay Extensions to Session Traversal

Utilities for NAT (STUN). RFC5766, 2010

- 11 Rosenberg J. Interactive Connectivity Establishment (ICE): a Protocol for Network Address Translator (NAT) Traversal for Offer/Answer Protocols. RFC5245, 2010

## [作者简介]



屈振华,男,博士,现就职于中国电信股份有限公司广州研究院政企客户产品开发部,主要从事多媒体实时通信、音视频编码技术、IMS 网络等方面的研究工作。

李慧云,女,硕士,中国电信股份有限公司广州研究院政企客户产品开发部工程师,主要从事终端产品研发、互联网技术研究和融合业务研发等工作。

张凌,硕士,中国电信股份有限公司广州研究院政企客户产品开发部高级工程师,主要从事移动互联网平台与终端研发工作。

杨新章,男,博士,中国电信股份有限公司广州研究院高级工程师,主要从事信息技术与产品研发工作。

龙显军,男,硕士,中国电信股份有限公司广州研究院高级工程师,主要负责融合通信与信息化、下一代移动通信等方面的技术、产品及解决方案的研发与管理工作。

(收稿日期:2013-07-10)