



Learning WebRTC

# Learning WebRTC 中文版

## 用WebRTC开发交互实时通信应用

[美] Dan Ristic 著  
宋晓薇 王雅琼 丁坚 刘振涛 译



中国工信出版集团



电子工业出版社  
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY  
<http://www.phei.com.cn>

**Learning WebRTC**

# Learning WebRTC 中文版

[美] Dan Ristic 著

宋晓薇 王雅琼 丁坚 刘振涛 译

电子工业出版社

Publishing House of Electronics Industry

北京•BEIJING

## 内 容 简 介

WebRTC 是一个支持网络浏览器进行实时语音对话或视频对话的软件架构。

本书使用形象的案例介绍,逐步深入地阐述了 WebRTC 的幕后工作原理。通过阅读本书,读者可以快速、有效地掌握创建一个 WebRTC 应用所必需的知识,包括获取用户设备信息、创建 WebRTC 应用的客户端和服务端、连接用户并发送数据、文件共享、数据信息安全和性能优化。

本书适合有一定 HTML 和 JavaScript 经验,希望了解 WebRTC 并想学习实时通信工作原理的开发者参考阅读。

Copyright © 2015 Packt Publishing. First published in the English language under the title 'learning WebRTC'.

本书简体中文版专有出版权由 Packt Publishing 授予电子工业出版社。未经许可,不得以任何方式复制或抄袭本书的任何部分。专有出版权受法律保护。

版权贸易合同登记号 图字:01-2015-6648

### 图书在版编目(CIP)数据

Learning WebRTC 中文版 / (美)丹·里斯蒂克(Dan Ristic)著;宋晓薇等译. —北京:电子工业出版社, 2016.7

ISBN 978-7-121-28817-3

I. ①L… II. ①丹… ②宋… III. ①移动终端—应用程序—程序设计 IV. ①TN929.53

中国版本图书馆 CIP 数据核字(2016)第 111861 号

策划编辑:张春雨

责任编辑:张春雨

印 刷:北京中新伟业印刷有限公司

装 订:北京中新伟业印刷有限公司

出版发行:电子工业出版社

北京市海淀区万寿路 173 信箱 邮编:100036

开 本:787×980 1/16 印张:10.5 字数:240.8 千字

版 次:2016 年 7 月第 1 版

印 次:2016 年 7 月第 1 次印刷

定 价:65.00 元

凡所购买电子工业出版社图书有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系,联系及邮购电话:(010) 88254888, 88258888。

质量投诉请发邮件至 [zltz@phei.com.cn](mailto:zltz@phei.com.cn), 盗版侵权举报请发邮件至 [dbqq@phei.com.cn](mailto:dbqq@phei.com.cn)。

本书咨询联系方式:010-51260888-819 [faq@phei.com.cn](mailto:faq@phei.com.cn)。

# 译者序

WebRTC 是谷歌的一个网页间实时通信的开源技术，它无须插件，就可以实现浏览器之间的交流功能，这意味着，用户无须安装任何东西，就可以在浏览器里面进行实时的声音和视频通话。当然，它不仅仅可以嵌入实时的音频和视频，同时还有文件共享等功能。在没有 WebRTC 之前，如果你想要在浏览器内进行通话或者共享文件，通常需要安装一个非常大的程序或者 Flash。而有了 WebRTC，用户直接单击已经开发好的网页应用，无须其他操作，即可进行视频会议。

当然，WebRTC 是一个非常年轻的技术，还有很多成长的空间，它的崛起势必会给互联网带来巨大的冲击。目前来看，Chrome 和 Firefox 浏览器已经对它有了很好的支持，而像 IE 等浏览器，还需要安装一些开源的插件来支持 WebRTC。

本书作者通过对网络视频原理的讲解，结合一些实例，非常详细地介绍了 WebRTC 在网页应用中的每一个细节。相信通过阅读本书，你将可以创建一个非常好用的 WebRTC 应用。

特别感谢寸志的引荐让我们得以参与本书的翻译，以及在整个翻译过程中提供的指导和帮助。同时也感谢这段时间团队 4 人之间的相互帮助，我们力求能正确传达作者的意思，但纰漏在所难免，存在的不足敬请广大读者批评指正。

# 关于作者

Dan Ristic

他是一位前端工程师，同时也是一名开放网络（Open Web）布道师。他致力于研究前沿技术，并以独到的创见推进 Web 领域的发展。他曾在位于美国亚利桑那州的先进技术大学（University of Advancing Technology）研习工程知识，对 Web 领域满怀热情，离开后便一直从事 Web 应用开发相关的工作。

他目前在旧金山地区生活工作，在索尼国际娱乐网络公司担任高级软件工程师一职，负责管理前端应用架构，为数百万用户提供 PlayStation 应用商店服务。业余时间里，他喜欢徒步旅行、探险、开发项目或出席一些活动。

---

感谢我的家人，无论我做什么事情，你们始终支持并鼓励着我。

同样感谢我的朋友们，在与你们的交流中我获益匪浅，让这本书更加充实可靠。

---

# 关于审校者

Roy Binux 是一位软件工程师，同时也是一名开源软件开发者。在实际工作中，他致力于研究爬虫系统和信息提取技术，也会出于兴趣构建一些有趣的在线程序。他乐于尝试新技术，努力优化着从 Web 平台获取知识的过程，并借助网络的力量开发一些有用的工具。他的大部分作品已开源，你可以在 <http://github.com/binux> 上找到相关信息。

Tsahi Levent-Levi 是一位独立的 WebRTC 分析师及顾问。

他在电信、VoIP 及 3G 通信产业等领域有着 15 年的经验，在这期间他担任过工程师、经理、市场营销及首席技术官等职位。他也是一名企业家、独立分析师及顾问，协助其他公司将其电信领域技术转化为相应的商业战略。

他拥有计算机科学硕士学位和工商管理硕士学位，擅长创业及战略制定，拥有三个 3G-324M 和 VoIP 相关的专利，同时还是 IMTC（国际多媒体通信协会）多个活动小组的主席。

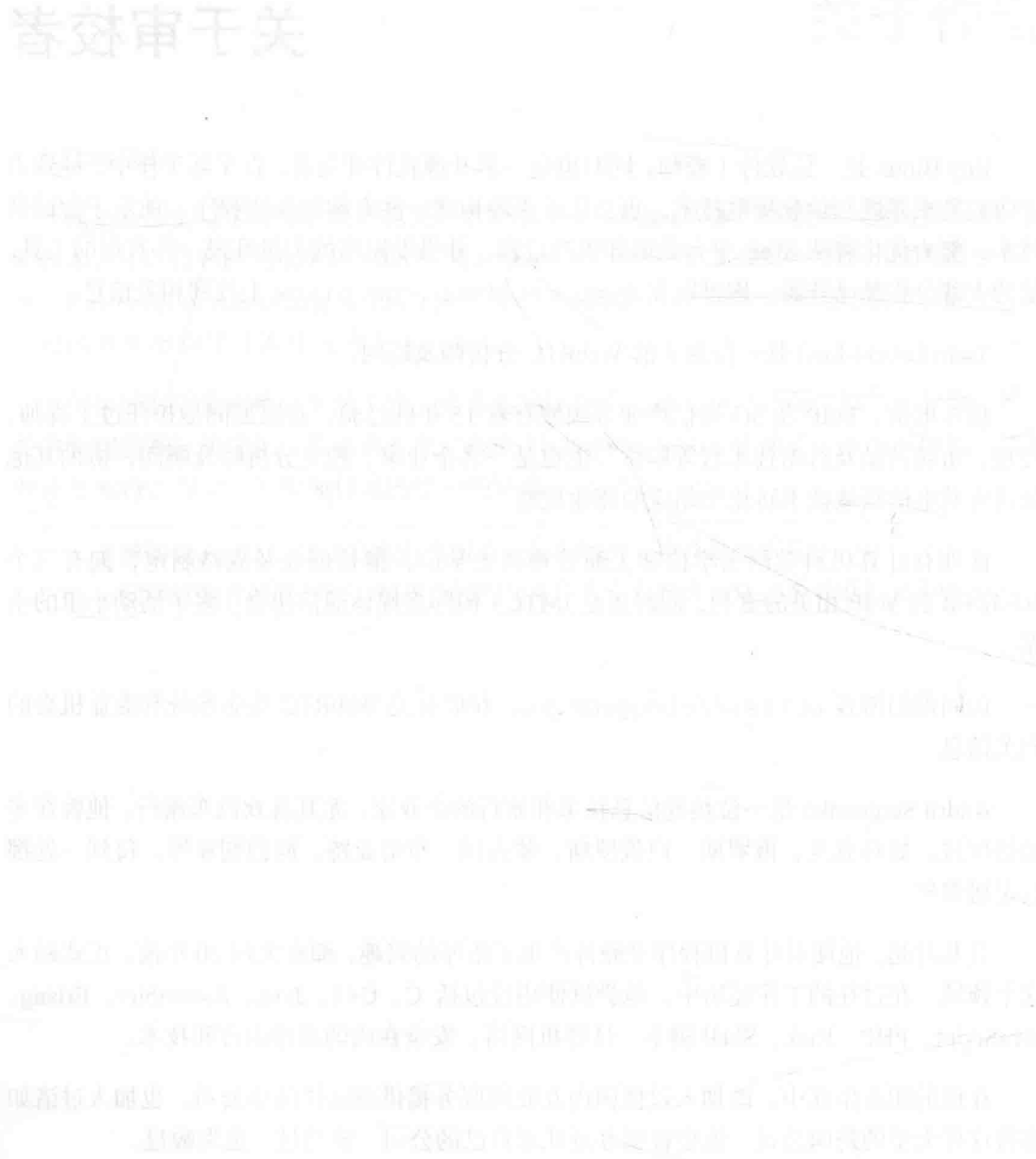
访问他的博客 <https://bloggeek.me>，获取有关 WebRTC 生态系统和商业机会的相关信息。

Andrii Sergiienko 是一位热爱信息技术和旅行的企业家，尤其喜欢汽车旅行。他曾在多地居住过，如乌克兰、俄罗斯、白俄罗斯、蒙古国、布里亚特、西伯利亚等，每到一处都会定居多年。

自儿时起，他便对计算机程序和硬件产生了浓厚的兴趣，而在大约 20 年前，正式踏入这个领域。在过往的工作经历中，他尝试使用过包括 C、C++、Java、Assembler、Erlang、JavaScript、PHP、Riak、Shell 脚本、计算机网络、安全在内的多种语言和技术。

在他的职业生涯中，既加入过像国内互联网服务提供商这样的小公司，也加入过诸如惠普这样大型的跨国公司。他也曾创办过几家自己的公司，成功过，也失败过。

现在，他致力于 Oslikas 的成长，这家新公司也是他一手创办的，总部位于爱沙尼亚，专注于现代 IT 技术及解决方案，他们开发了一套用于创建富媒体 WebRTC 应用和服务的全栈框架。访问 <http://www.oslikas.com> 浏览更多有关 Oslikas 的信息。



# 前言

第一次写 HTML 代码的时候，我非常兴奋。我在键盘上输入的每一个字符都会作为指令告诉计算机该怎么处理。我很高兴我能创造出一些东西并用这些指令表达我的创造力。当我保存我的劳动成果，打开浏览器然后显示页面，这一切让我心生敬畏。当看到自己的名字以大号粗体字展现在一个叫作 GIF 的动态图片上时，心里像是有一团窜动的火焰。

web 由来已久，事实上它不仅是一个创造新事物的地方，也是展示和分享创意给他人的平台。这个表达创意的强大平台驱动着 web 以更快的速度发展。由于 web 越来越普及，我们整个生活都依赖于它。网站支持着你的邮件、娱乐、银行账号、法律文件和税收，甚至连这本书的部分撰写也是使用了 web 工具。人们想把生活搬到 web 上的需求促进了 API 的快速有力发展，例如 WebRTC。

WebRTC 是 web 平台的最实质的加法指令之一。它带来了一整套的新技术，如摄像头、流动数据，甚至整个网络的协议栈。令人吃惊的是，不仅看到大量的工作进入 Web RTC API，而且任何应用开发者使用它都是完全免费的。

WebRTC 的目的是实现实时通信大众化。早些时候，创建一个较小的视频通信应用需要花费几个月时间，即使是做一个很小的应用程序都需要一些定制工程。但是，现在只要花费一半甚至更少的时间。同时它也把开源社区带入实时通信领域。你可以查看全世界的其他的 WebRTC 例子，通过搜索一些源代码看看这些应用是怎么实现的。

正是这些创造性的表达方式和网络带来的自由促成了本书的出世。我非常高兴能有机会通过我的作品把这些东西带给大家，我希望能激励其他人像我当初一样，同样由于被激励做了第一个网页。写这本书是我工作中最重要也是最困难的事之一，但我非常感激在这过程中所学习到的东西。

如果你正在寻找能创建一个新的最简单的实时通信的经验，并想分享给他人，可以读这本书。这本书像其他技术书籍一样，只是引导人们在 web 上创造一些更好的东西的途径。你不仅能学会怎样使用 WebRTC，也能够做到举一反三。这本书不仅是一个学习工具，而

且也是对有激情地创造一些事情的激励。

## 这本书包含了什么

第 1 章, 开启 WebRTC 之旅, 囊括了 WebRTC 怎么以网页为基础, 来实现音频和视频的通信。可以在你的浏览器上运行一个 WebRTC 应用。

第 2 章, 获得 User Media, 介绍了创建通信应用程序的第一步、网络摄像头和麦克风的获取。这一章还包括如何使用媒体和流 API 捕捉你的信息, 我们也开始开发建设基础通信的例子。

第 3 章, 创建简单的 WebRTC 应用, 包括了对第一个 WebRTC-RTC 对等连接的介绍。同时看到 WebRTC 里面的复杂结构以及开始使用 API 后我们可以期待些什么, 来奠定创建一个 WebRTC 应用程序的基础。

第 4 章, 创建信令服务器, 包括创建我们自己的信令服务器, 来帮助客户在互联网上找到彼此的步骤。这也包括在 WebRTC 上信令是如何工作, 以及如何运用到我们的示例应用程序中去的深层信息。

第 5 章, 把客户端连接到一起, 包括我们的信令服务器的实际使用, 也包括使用 WebRTC API、媒体捕捉和在前面章节用来创建我们示例应用的信令服务器, 来成功连接两个用户。

第 6 章, 使用 WebRTC 发送数据, 包括了 RTCDataChannel 介绍, 以及它是如何用于两个对等体之间的原始数据发送的。这一章通过为客户添加文本聊天来详细说明我们的示例。

第 7 章, 文件共享, 通过观察如何在两个对等体之间的共享文件, 阐述了发送原始数据的概念。这将展示音频和视频共享以外的 WebRTC 的许多用途。

第 8 章, 高安全性与大规模优化, 包括高级的主题, 比如如何提供一个大规模的 WebRTC 应用程序。我们还关注安全和其他公司使用的性能优化。

附录, 自测题答案, 包括每一章最后自测题的答案。

## 你需要准备什么

本书中所有的例子都是建立在 web 标准上的。由于 WebRTC 规范是相当新的，建议运行示例时使用更新过的浏览器，优先使用最新的 Firefox 和 Chrome 浏览器。

所有的服务器代码均使用 Node.js 编写。Node.js 框架大多数在 Windows、Linux 和 MacOSX 的机器内运行。

你可以使用任何支持 JavaScript 和 HTML 代码的文本编辑器。

## 这本书是写给谁的

你应该有一些使用 HTML 和 JavaScript 构建 web 应用程序的经验。也许，你现在正在构建应用程序，或有一个利用用户间音频和视频交流力量来构建新的应用程序的想法。你可能还需要通过在用户之间转移高性能数据来实现应用程序。

你应该牢固地掌握编程概念和网络基础，但是这本书是写给新入门的网络工程师的。提到的概念很深入，并且是一点一点地深入，而不是一开始就进入高级话题。你可能完全不知道 WebRTC 或者只听说过一点，而且想学习实时通信的内部工作原理。

## 约定

本书中使用了很多格式的文本，以区分各种不同的信息。这里我们举例说明这些格式，并解释它们的含义。

文本、数据库表名、文件夹名称、文件名、文件扩展名、路径名、伪 URL、用户输入和 Twitter 中的代码，格式是这样的：“一件事需要注意的是，当我们从 `theirConnection` 得到一个 ICE 候选人，我们将其添加到连接，反之亦然。”

代码块的格式设置如下：

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8" />
```

```
<title>Learning WebRTC - Chapter 4: Creating a
    RTCPeerConnection</title>
</head>
<body>
    <div id=""container"">
        <video id=""yours"" autoplay></video>
        <video id=""theirs"" autoplay></video>
    </div>
    <script src=""main.js""></script>
</body>
</html>
```

所有命令行输入或输出写为如下：

```
> 1 + 1
2
> var hello = "world";
undefined
> "Hello" + hello;D
'Helloworld'
```

新术语和重要词汇以黑体显示。在屏幕上看到的词，例如，在菜单或对话框上，你这样表示：“现在你可以单击 **Capture** 按钮，捕获出现在屏幕上的一帧视频。”



警告或重要提示以这种方式表示。



技巧和窍门像这样表示。

## 下载示例代码

你可以从 <http://www.broadview.com.cn> 下载所有已购买的博文视点书籍的示例代码文件。

## 勘误表

虽然我们已经尽力谨慎地确保内容的准确性，但错误仍然存在。如果你发现了书中的错误，包括正文和代码中的错误，请告诉我们，我们会非常感激。这样，你不仅帮助了其他读者，也帮助我们改进后续的出版。如发现任何勘误，可以在博文视点网站相应图书的页面提交勘误信息。一旦你找到的错误被证实，你提交的信息就会被接受，我们的网站也会发布这些勘误信息。你可以随时浏览图书页面，查看已发布的勘误信息。

# 目录

|                                   |    |
|-----------------------------------|----|
| 前言 .....                          | XI |
| 1 开启 WebRTC 之旅 .....              | 1  |
| 音视频通信领域的发展现状 .....                | 1  |
| 在 web 平台传输音频和视频 .....             | 2  |
| 捕捉摄像头和麦克风 .....                   | 3  |
| 音频及视频的编解码 .....                   | 3  |
| 传输层 .....                         | 4  |
| 会话 (Session) 管理 .....             | 4  |
| 创建 web 标准 .....                   | 5  |
| 浏览器支持 .....                       | 5  |
| Chrome、Firefox 和 Opera 的兼容性 ..... | 5  |
| 安卓操作系统的兼容性 .....                  | 6  |
| 苹果操作系统兼容性 .....                   | 6  |
| IE 的兼容性 .....                     | 6  |
| 在浏览器中使用 WebRTC .....              | 6  |
| 支持 WebRTC 的应用 .....               | 7  |
| 自测题 .....                         | 7  |
| 小结 .....                          | 8  |
| 2 获取用户媒体 .....                    | 9  |
| 访问媒体设备 .....                      | 9  |
| 配置静态服务器 .....                     | 10 |
| 创建我们的首个媒体流页面 .....                | 11 |

|                                |           |
|--------------------------------|-----------|
| 限制媒体流 .....                    | 14        |
| 限制视频捕捉.....                    | 15        |
| 多设备处理 .....                    | 18        |
| 创建一个拍照室应用.....                 | 20        |
| 修改媒体流 .....                    | 23        |
| 自测题 .....                      | 25        |
| 小结 .....                       | 26        |
| <b>3 创建简单的 WebRTC 应用 .....</b> | <b>27</b> |
| 理解 UDP 传输协议和实时传输 .....         | 27        |
| WebRTC API .....               | 30        |
| RTCPeerConnection 对象.....      | 30        |
| 信号传递和交涉.....                   | 31        |
| 会话描述协议（SDP） .....              | 33        |
| 找到一条清晰的路线到其他用户.....            | 35        |
| 创建一个基本的 WebRTC 应用.....         | 38        |
| 创建一个 RTCPeerConnection.....    | 38        |
| 建立 SDP OFFER 和返回 .....         | 43        |
| 寻找 ICE 候选路径 .....              | 44        |
| 加入流和打磨.....                    | 46        |
| 运行你的第一个 WebRTC 应用.....         | 47        |
| 自测题 .....                      | 51        |
| 小结 .....                       | 52        |
| <b>4 创建信令服务器.....</b>          | <b>53</b> |
| 构建信令服务器 .....                  | 53        |
| 搭建开发环境 .....                   | 55        |
| 获得一个连接 .....                   | 56        |
| 测试我们的服务器.....                  | 58        |
| 识别用户 .....                     | 59        |
| 发起通话 .....                     | 62        |

|                               |           |
|-------------------------------|-----------|
| 呼叫应答 .....                    | 63        |
| 处理 ICE 候选路径 .....             | 64        |
| 呼叫挂断 .....                    | 65        |
| 完成信令服务器 .....                 | 66        |
| 在实际应用中发送信令 .....              | 70        |
| WebSockets 的困境 .....          | 70        |
| 连接其他服务 .....                  | 71        |
| 自测题 .....                     | 72        |
| 小结 .....                      | 73        |
| <b>5 把客户端连接到一起 .....</b>      | <b>75</b> |
| 客户端应用 .....                   | 75        |
| 创建页面 .....                    | 77        |
| 获取一个连接 .....                  | 79        |
| 登录到应用程序 .....                 | 81        |
| 开始一个对等连接 .....                | 82        |
| 发起通话 .....                    | 85        |
| 检测通信 .....                    | 87        |
| 挂断电话 .....                    | 88        |
| 一个完整的 WebRTC 客户端 .....        | 89        |
| 改进应用程序 .....                  | 95        |
| 自测题 .....                     | 95        |
| 小结 .....                      | 96        |
| <b>6 使用 WebRTC 发送数据 .....</b> | <b>97</b> |
| 流控制传输协议和数据传输 .....            | 97        |
| RTCDataChannel 对象 .....       | 99        |
| 数据通道选项 .....                  | 101       |
| 发送数据 .....                    | 101       |
| 加密与安全 .....                   | 102       |
| 添加文字聊天 .....                  | 103       |

|                           |            |
|---------------------------|------------|
| 用例 .....                  | 107        |
| 自测题 .....                 | 108        |
| 小结 .....                  | 109        |
| <b>7 文件共享 .....</b>       | <b>111</b> |
| 使用文件 API 拾取文件 .....       | 112        |
| 准备我们的页面 .....             | 114        |
| 获取对文件的引用 .....            | 121        |
| 文件分块 .....                | 122        |
| 使文件分块可读 .....             | 124        |
| 文件读取与发送 .....             | 126        |
| 在“另一端”组合文件块 .....         | 128        |
| 向用户展示进度 .....             | 130        |
| 自测题 .....                 | 131        |
| 小结 .....                  | 132        |
| <b>8 高安全性与大规模优化 .....</b> | <b>133</b> |
| 保护信令服务器 .....             | 133        |
| 使用编码 .....                | 134        |
| 使用 OAuth 提供者 .....        | 134        |
| 支持移动设备 .....              | 136        |
| 网格网络简介 .....              | 138        |
| 网格类型 .....                | 139        |
| 网格网络的缺陷 .....             | 143        |
| 更多用户的视频会议 .....           | 144        |
| 视频会议的未来 .....             | 146        |
| 自测题 .....                 | 146        |
| 小结 .....                  | 147        |
| <b>附录 自测题答案 .....</b>     | <b>149</b> |

# 开启 WebRTC 之旅

音频和视频可以在互联网上传输，诸如 Netflix 和 Pandora 这样的 web 应用每天都向成千上万的用户传输着音视频内容。然而，在如今的 web 平台上尚未形成一套成熟的实时通信解决方案，如果 Facebook 这样的网站要在浏览器中加入视频通信的功能，通常会引导用户安装一个插件来实现。于是，WebRTC（web 实时通信）技术就应运而生了。

在本章中，我们将介绍 WebRTC 的基础知识，包括：

- 音频和视频领域的发展现状
- WebRTC 对音视频领域的影响
- WebRTC 主要特性及使用方法

## 音视频通信领域的发展现状

人们在很久以前就利用各式各样的技术和工具进行通信，就以移动电话运营商为例，大型电话公司建立起大规模的音频通信网络，为全球成千上万的用户提供音频通信服务，让每个人的语音都能够传播到世界各地。

后来，视频通信也随之流行起来。基于苹果的 FaceTime、Google 的 Hangouts 还有 Skype 的视频通话这些工具背后的技术，用户可以很方便地与其他人进行视频对话。开发者们为了将用户体验优化到极致，通过大量的技术手段保障视频质量，针对每一种问题都提出了相应的工程化解决方案，例如：减少丢包、断网恢复、及时响应用户网络变化等。