通信方法及装置

本申请要求于2024年3月15日提交国家知识产权局、申请号为202410310005.8、发明名称为“通信方法及装置”的中国专利申请的优先权，其全部内容通过引用结合在本申请中。

技术领域

本申请涉及通信技术领域，尤其涉及通信方法及装置。

背景技术

为了实现第五代（5th generation，5G）新通话场景的业务创新，5G 新通话在互联网协议多媒体子系统（internet protocol multimedia subsystem，IMS）网络的音频通道和视频通道之外引入了数据通道（data channel，DC），该数据通道也称为 IMS 数据通道。处于同一个 IMS 会话的终端之间可以协商数据通道，并通过协商的数据通道传输应用数据。

目前，终端可以为一个应用协商至少一个数据通道，以传输该应用的数据。IMS会话通常存在多个应用，终端采用现有的方法为这些应用协商数据通道时会出现资源开销较大的问题。

发明内容

本申请提供通信方法及装置，在创建数据通道时可以复用已创建的连接，进而降低资源开销。

为达到上述目的，本申请的采用如下技术方案：

第一方面，提供了一种通信方法，该方法可以由通信实体执行。这里的通信实体既可以指通信实体本身，也可以指通信实体中实现该方法的处理器、电路、模块、逻辑节点、芯片、或芯片系统等。可选的，该通信实体部署在终端中。

该方法包括：接收用于请求创建第一数据通道的第一请求，第一数据通道为第一通信装置（如上述终端）和第二通信装置之间的数据通道，用于承载第一应用的数据；第一数据通道与第一通信装置和第二通信装置之间的 IMS 会话相关联；若存在已创建的连接能承载第一数据通道，则发送第一请求的响应，该响应包括第一媒体描述信息，第一媒体描述信息包括第一指示信息，第一指示信息指示第一连接，第一连接为已创建的连接中第一数据通道对应的连接；或，若不存在已创建的连接能承载第一数据通道，则发送第一请求的响应并创建第二连接，响应包括第二媒体描述信息，第二媒体描述信息包括第二指示信息，第二指示信息指示第二连接，第二连接为第一数据通道对应的连接。

基于上述第一方面提供的方法，终端在创建第一数据通道之前，可以在已创建的连接中确定是否有能承载第一数据通道的连接。如果有能承载第一数据通道的连接，第一数据通道可以复用该连接，所以可以降低资源开销。

在一种可能的实现方式中，第一指示信息包括：互联网协议地址信息、用户数据报协议端口信息以及流控制传输协议端口信息。

基于上述可能的实现方式，若存在已创建的连接能承载第一数据通道，则第一数据通道可以复用互联网协议地址信息、用户数据报协议端口信息以及流控制传输协议端口信息，以降低资源开销。

在一种可能的实现方式中，该方法还包括：接收第二请求，第二请求用于请求创建第二数据通道，第二数据通道用于承载第二应用的数据，第二应用与第一应用不同；若已创建的连接不能承载第二数据通道，则发送第二请求的响应并创建第三连接，第二请求的响应包括第三媒体描述信息，第三媒体描述信息包括第二数据通道对应的第三指示信息，第三指示信息指示第三连接；或，若第一连接能承载第二数据通道，则发送第二请求的响应，第二请求的响应包括第四媒体描述信息，第四媒体描述信息包括第一指示信息；或，若第二连接能承载第二数据通道，则发送第二请求的响应，第二请求的响应包括第五媒体描述信息，第五媒体描述信息包括第二指示信息。

基于上述可能的实现方式，终端在创建第二数据通道之前，可以在已创建的连接中确定是否有能承载第二数据通道的连接，以降低资源开销。另外，若第一连接（或第二连接）能承载第二数据通道，则表示一个连接可以承载两个不同应用的数据通道，所以该连接可以传输不同应用的数据。

在一种可能的实现方式中，第二连接能承载第二数据通道，包括：第二数据通道的服务质量要求与第一数据通道的服务质量要求相同；或者，第二数据通道的服务质量要求低于第一数据通道的服务质量要求。

基于上述可能的实现方式，可以基于第二数据通道的服务质量要求与第一数据通道的服务质量要求确定第二连接能否承载第二数据通道。

在一种可能的实现方式中，第二连接能承载第二数据通道，还包括：第二数据通道为第一通信装置和第二通信装置之间的数据通道。

基于上述可能的实现方式，可以基于第二数据通道端点信息确定第二连接能否承载第二数据通道。

在一种可能的实现方式中，该方法还包括：向网络侧发送第六媒体描述信息，第六媒体描述包括第一数据通道对应的连接的信息和第二数据通道对应的连接的信息。

基于上述可能的实现方式，可以发送第六媒体描述信息，以协商第二数据通道。

在一种可能的实现方式中，第六媒体描述信息还包括第一数据通道的标识信息以及第二数据通道的标识信息。

基于上述可能的实现方式，可以确定第六媒体描述信息对应的数据通道是第一数据通道和第二数据通道。

在一种可能的实现方式中，该方法还包括：若第一请求包括第一连接的信息，则确定第一连接能承载第一数据通道；或，若第一请求不包括第一数据通道对应的连接的信息，则确定已创建的连接均不能承载第一数据通道。

基于上述可能的实现方式，可以根据第一请求是否包括第一数据通道对应的连接的信息，确定已创建的连接能否承载第一数据通道。

第二方面，提供了一种通信装置用于实现上述方法。该通信装置可以为上述第一方面中的终端。该通信装置包括实现上述方法相应的模块、单元、或手段（means），该模块、单元、或means可以通过硬件实现，软件实现，或者通过硬件执行相应的软件实现。该硬件或软件包括一个或多个与上述功能相对应的模块或单元。

在一种可能的实现方式中，该通信装置可以包括处理模块和接口模块。该处理模块，可以用于实现上述第一方面及其任意可能的实现方式中的处理功能。该处理模块例如可以为处理器。该接口模块，也可以称为接口单元，用以实现上述第一方面及其任意可能的实现方式中的发送和/或接收功能。该接口模块可以由接口电路，收发机，收发器或者通信接口构成。

在一种可能的实现方式中，接口模块包括发送模块和接收模块，分别用于实现上述第一方面及其任意可能的实现方式中的发送和接收功能。

第三方面，提供了一种通信装置，包括：处理器；用于通过执行存储器中存储的计算机程序（或计算机可执行指令），和/或通过逻辑电路，使得该通信装置执行如上述第一方面所述的方法。该通信装置可以为上述第一方面中的终端。可选的，上述处理器的数量可以是一个或多个。

在一种可能的实现方式中，该通信装置还包括存储器。

在一种可能的实现方式中，处理器和存储器集成在一起；或者，该存储器独立于该处理器。

在一种可能的实现方式中，该通信装置还包括通信接口，该通信接口用于该通信装置与其他设备进行通信，例如数据和/或信号的发送或接收。示例性的，通信接口可以是收发器、电路、总线、模块或其它类型的通信接口。

在一种可能的实现方式中，该通信装置为芯片或芯片系统。可选的，该通信装置是芯片系统时，可以由芯片构成，也可以包含芯片和其他分立器件。

第四方面，提供了一种通信装置，包括：处理器和接口电路；接口电路，用于接收计算机程序或指令并传输至处理器；处理器用于执行所述计算机程序或指令，以使该通信装置执行如上述第一方面所述的方法。该通信装置可以为上述第一方面中的终端。可选的，上述处理器的数量可以是一个或多个。

在一种可能的实现方式中，该通信装置为芯片或芯片系统。可选的，该通信装置是芯片系统时，可以由芯片构成，也可以包含芯片和其他分立器件。

第五方面，提供了一种计算机可读存储介质，该计算机可读存储介质中存储有指令，当其在计算机上运行时，使得计算机可以执行上述第一方面所述的方法。

第六方面，提供了一种包含指令的计算机程序产品，当其在计算机上运行时，使得计算机可以执行上述第一方面所述的方法。

第七方面，提供了一种通信系统，包括：至少两个用于执行上述第一方面所述方法的终端。

其中，第二方面至第七方面中任一种可能的实现方式所带来的技术效果可参见上述第一方面或第一

方面中不同可能的实现方式所带来的技术效果，此处不再赘述。

可以理解的是，在方案不矛盾的前提下，上述各个方面中的方案均可以结合。

附图说明

图 1A 为万维网实时通信（web real-time communication，WebRTC）数据通道的协议栈的示意图；

图 1B 为 IMS 会话的示意图；

图 1C 为 IMS 会话的协议栈的示意图；

图 1D 为终端 1 和终端 2 获取数据通道应用的示意图；

图 1E 为会话描述协议（session description protocol，SDP）提议方和 SDP 应答方的交互示意图；

图 1F 为 WebRTC 用户代理的架构示意图；

图 1G 为 WebRTC 信令模型的示意图；

图 1H 为 WebRTC 进行 SDP 交换的过程示意图；

图 1I 为 WebRTC 数据通道的媒体协商过程示意图；

图 2 为 IMS 数据通道的媒体协商过程示意图；

图 3A 为一种通信场景的示意图一；

图 3B 为一种通信场景的示意图二；

图 3C 为一种 DCMTSI client 的架构示意图；

图 3D 为一种终端支持的数据通道的逻辑架构示意图一；

图 3E 为一种终端支持的数据通道的逻辑架构示意图二；

图 4 为本申请提供的通信装置的硬件结构示意图；

图 5 为本申请提供的通信方法的流程示意图一；

图 6 为本申请提供的通信方法的流程示意图二；

图 7 为本申请提供的通信方法的流程示意图三；

图 8 为本申请提供的通信方法的流程示意图四；

图 9 为本申请提供的通信方法的流程示意图五；

图 10 为本申请提供的通信装置的结构示意图。

具体实施方式

在介绍本申请的技术方案之前，对本申请涉及的相关技术术语进行解释说明。可以理解的是，这些解释说明是为了让本申请更容易被理解，而不应该视为对本申请所要求的保护范围的限定。

1、终端

终端可以称为用户设备（user equipment，UE）、终端设备、终端装置、接入终端、用户单元、用户站、移动站、移动台（mobile station，MS）、移动终端（mobile terminal，MT）、远方站、远程终端、移动设备、用户终端、无线通信设备、用户代理或用户装置等，是一种具有无线或有线通信能力的设备。例如，终端可以通过空中接口与无线接入设备相连，或者终端可以通过有线接口与有线接入设备相连。从产品形态上看，终端可以包括具有通信功能的手持式设备、车载设备、可穿戴设备或计算设备等。示例性的，终端可以是手机（mobile phone）、平板电脑（pad）、带无线通信功能的电脑（如笔记本电脑、掌上电脑等）、移动互联网设备（mobile internet device，MID）、虚拟现实（virtual reality，VR）终端、增强现实（augmented reality，AR）终端、智能手表、智能手环、智能眼镜、工业控制（industrial control）中的无线终端、无人驾驶（self driving）中的无线终端、远程医疗（remote medical）中的无线终端、智能电网（smart grid）中的无线终端、运输安全（transportation safety）中的无线终端、智慧城市（smart city）中的无线终端、智慧家庭（smart home）中的无线终端、蜂窝电话、无绳电话、会话发起协议（session initiation protocol，SIP）电话、无线本地环路（wireless local loop，WLL）站、个人数字助理（personal digital assistant，PDA）、具有无线通信功能的手持设备、连接到无线调制解调器的其它处理设备、物联网（internet of things，IoT）系统中的终端、台式电脑或第三代合作伙伴计划（3rd generation partnership project，3GPP）标准规范定义的 UE 等，不予限制。

2、WebRTC

WebRTC 是一种实时通信技术，允许网络应用或者网站（或站点）在不借助中间媒介的情况下，建立浏览器之间的点对点（peer to peer，P2P）连接，实现音频流、视频流或除音频流和视频流之外的任意数据的传输。其中，上述音频流和/或视频流可通过媒体通道传输，上述数据可通过数据通道（data channel，

DC）传输，该数据通道可称为 WebRTC 数据通道。

WebRTC 数据通道是 WebRTC 用户代理（WebRTC user agent）为在浏览器之间传递音频、视频之外的任意数据而创建的通道。WebRTC 数据通道可采用流控制传输协议（stream control transmission protocol，SCTP）叠加数据报传输层安全（datagram transport layer security，DTLS），再叠加用户数据报协议（user datagram protocol，UDP）作为承载协议，并采用交互式连接建立（interactive connectivity establishment，ICE）实现网络地址翻译（network address translation，NAT）穿越，让通信双方相互感知对方的互联网协议（Internet Protocol，IP）地址和端口，例如，WebRTC 数据通道的协议栈可以如图 1A 所示。其中 SCTP 是面向连接的传输协议，其连接可以被称为耦联（association）。一个 SCTP 耦联可以包括多个单向的流（stream），各个流之间相互独立，使用流标识（stream ID）进行标记。可以理解的，每个流可以单独发送数据而不受其他流的影响。换言之，WebRTC 数据通道本质上是一个 SCTP 耦联上的两个不同方向的流，这两个流可以使用相同的 stream ID。

为了在 IMS 网络中传输音频流和视频流之外的数据，第三代合作伙伴计划（3rd generation partnership project，3GPP）为 IMS 网络引入了数通通道，称为 IMS 数据通道（IMS data channel，IMS DC）。其中，IMS 网络是一个基于 IP 承载并向用户提供各项多媒体服务的开放系统。本申请下述实施例中所述的数据通道均指 IMS 数据通道。为了更清楚地了解 IMS 数据通道，先对与 IMS 数据通道有关的通话业务以及 IMS 会话（IMS session）进行说明。

3、通话业务

通话业务是指终端以主叫身份或者被叫身份参与，经通信网络（如 IMS 网络）接续与其他一到多个终端之间进行的语音通话的业务或视频通话的业务。通话业务在时间上可以涵盖从开始拨号到通话结束之间的整个过程，也可以涵盖从开始拨号到通话结束之间的部分过程，如从参与通话业务的各方进入通话状态到通话结束之间的过程。通话业务可以是一对一的通话业务，也可以是一对多（如会议）的通话业务；本申请以一对一通话业务为例，但相关方案均可以用于一对多的通话业务。

4、IMS 会话

本申请中，IMS 会话是指在人与人、人与物或物与物之间建立的用于多媒体通信的会话，用于传输通信方之间的多媒体数据，如终端之间的数据，和/或，终端与通信网络之间的数据等。IMS 会话可以理解为多媒体通信中的通信装置之间的逻辑连接，例如，可以理解为多媒体通信中的终端之间的逻辑连接，可以传输终端之间的数据；或者可以理解为多媒体通信中的终端与通信网络之间的逻辑连接，用于传输终端和通信网络之间的数据。在具体应用中，IMS 会话可以基于 SIP/SDP 协商（或建立）。应理解，IMS 会话也可以替换成通信会话、IMS 通信会话、会话或通话业务的会话等，不予限制。

可以理解的，多媒体通信可以对应至少一个 IMS 会话。以终端 1 和终端 2 之间进行的多媒体通信为例，该多媒体通信对应一个 IMS 会话，如终端 1 和终端 2 之间的 IMS 会话。以终端 1、终端 2 和终端 3 之间进行的多媒体通信为例，该多媒体通信对应一个 IMS 会话，如终端 1、终端 2 和终端 3 之间的 IMS 会话；或者，该多媒体通信对应三个 IMS 会话，如终端 1 和终端 2 之间的 IMS 会话、终端 1 和终端 3 之间的 IMS 会话以及终端 2 和终端 3 之间的 IMS 会话。

可以理解的，为了传输各种类型的数据，IMS 会话中可建立用于传输相应类型数据的通道。示例性的，根据数据类型，通道可分为媒体通道和数据通道。在一些场景中，可以将媒体通道进一步划分为音频通道和视频通道。其中，音频通道可用于传输语音，视频通道可用于传输视频，数据通道可用于传输语音和视频之外的任意数据。当然，数据通道也可以传输语音和/或视频，不予限制。

示例性的，如图 1B 所示，终端 1 和终端 2 可通过 IMS 网络通信，终端 1 和终端 2 之间的 IMS 会话包括数据通道、音频通道和视频通道。应理解，图 1B 仅是 IMS 会话的示例，在具体应用中，IMS 会话可以包括数据通道，而不包括音频和视频通道，或者 IMS 会话包括音频通道和视频通道中的至少一个，以及数据通道，不予限制。

5、IMS 会话的协议栈

IMS 会话的协议栈用于指示适用于 IMS 会话的媒体传输协议的总和，可反映 IMS 会话中数据的传输过程。IMS 会话可传输语音、视频、文本以及除语音、视频、文本之外的数据。因此，为了匹配不同类型数据的传输要求，可为不同类型的数据的通道配置不同的协议。例如，IMS 会话的协议栈可以如图 1C 所示。在图 1C 中，音频通道、视频通道和数据通道最底层的协议是 IP，IP 之上是 UDP。对于上述通道，UDP 之上的协议有所不同，下面分别阐述。对于音频通道或视频通道，UDP 之上是实时传输协议（real-

time transport protocol，RTP）或RTP控制协议（RTP control protocol，RTCP），RTP中可封装语音、视频或文本的有效载荷（payload），再之上是会话多媒体应用（conversational multimedia application），RTCP之上也是会话多媒体应用。对于数据通道，UDP之上是DTLS协议，DTLS协议之上是SCTP，SCTP中可封装数据通道能传输的信息，如应用数据或应用程序，应用数据或应用程序之上是会话多媒体应用。

可以理解的，图1C仅是IMS会话的媒体传输协议栈的示例，在具体应用中，IMS会话的协议栈还可以是其他形式的。例如，IMS会话的协议栈可以不包括DTLS协议。

6、数据通道

本申请下述实施例所述的数据通道，在没有做特殊说明的情况下，均是指IMS网络中的数据通道，即IMS数据通道，与终端的IMS会话相关联，或者说是在该终端的IMS会话期间创建的，可以用于在该IMS会话存续期间传输与该终端相关的应用数据或应用程序（用于产生/消费应用数据）。

示例性的，根据数据通道与音/视频通话的关系，本申请实施例的数据通道至少可以分为以下类型。

（1）在一个IMS会话过程中，当该终端与其他会话参与方（可以是另一终端，也可以是网络侧网元）之间存在音频通话或视频通话的同时，还存在数据通道，此时的数据通道可以称为辅助性数据通道（auxiliary data channel），或者，依赖性数据通道（dependent data channel）。

（2）在一个IMS会话过程中，当该终端与其他会话参与方之间存在数据通道，不存在传统语音通话或视频通话，此时的数据通道可以称为独立数据通道（standalone data channel）或独立性数据通道（independent data channel）。

无论是辅助性数据通道还是独立数据通道，都可以用于传输会话参与方之间的任意类型的应用数据或应用程序。

示例性的，数据通道所传输的应用数据可以是互动应用数据（interactive application data），简称互动数据（interactive data）。互动应用数据可以是用于支持互动操作的应用数据（如用于生成图形用户界面（graphical user interface，GUI）的数据或代码/脚本，该GUI供用户进行互动操作，如“点赞”按钮、“红包”按钮），也可以是执行该互动操作所产生的应用数据（如会话参与方进行互动操作后产生的控制指令、控制信号或输出的文件，如“点赞”次数、“红包”中的金额），还可以是终端采集或者存储的数据，例如，终端通过输入/输出（input/output，I/O）模块采集的数据，例如通过摄像头拍摄的图像或视频、通过麦克风采集的语音、通过键盘输入的文字或符号或表情等、通过触摸屏上点击/滑动等动作产生的数据。

示例性的，数据通道所传输的应用数据也可以是通信增强应用数据（communication enhancement application data），简称通信增强数据（communication enhancement data）；通信增强数据可以是基于音频通话/视频通话的内容而产生的数据，如将音频通话/视频通话中的语音转换为文本，或者将视频通话中的手语转换为文本或者语音，也可以是在音频通话/视频通话基础上叠加的数据，如AR特效画面等，还可以是独立于音频通话/视频通话之外的数据，如地理位置数据、屏幕共享数据等。

示例性的，数据通道所传输的应用数据也可以是远程控制命令，则可以为用户提供远程控制服务，或者远程IOT设备连接服务。可以理解，在上述情况中，数据通道也可以用于传输远程控制命令。

示例性的，数据通道所传输的应用程序，可以是应用程序的安装包、可执行代码或脚本，该应用程序可以是互动增强型应用或通信增强型应用，用于增强传统语音通话/视频通话的互动效果/通信效果，也可以是远程控制应用/IoT通信应用，本申请不作限制。

在一些场景中，根据用途的不同，可以将辅助性数据通道或独立数据通道划分为：引导数据通道（bootstrap data channel，BDC）或应用数据通道（application data channel，ADC）。本申请中，BDC用于终端从数据通道服务器（data channel server，DCS）获取应用程序，如获取数据通道应用（DC app）。ADC用于传输应用数据，如传输通信双方运行的数据通道应用所产生的交互数据或应用数据等。可以理解的，BDC中传输的应用可以逻辑控制ADC中的应用数据的发送或接收。BDC的stream ID取值小于1000，ADC的stream ID取值大于或等于1000。

7、数据通道应用

数据通道应用可以用于通过同一个通话业务中的多个终端之间的数据通道传输应用数据。例如，数据通道应用包含以下至少一项：超文本标记语言（hypertext markup language，HTML）、JavaScript脚本、图片或层叠样式表（cascading style sheets，CSS）。数据通道应用可以描述图形化的用户界面（user interface，UI），并能够实现交互式的业务逻辑。数据通道应用可以以网络（web）页面的形式存在或运行在浏览器环境中，如以HTML网页的UI模式存在，也可以以小程序/快应用/轻应用等形式存在。

数据通道应用可以在通话建立前和通话建立后通过数据通道传递文本、图片、位置、文件等各种数据，以实现通话双方的实时数据交互，能够极大地增强通话双方的体验。以处于同一个通话业务中的终端 1 和终端 2 为例，如果终端 1 在与终端 2 的通话过程，希望向终端 2 共享自己的手机屏幕，来指导终端 2 进行手机设置，则终端 1 和终端 2 可以分别获取用于屏幕共享的数据通道应用，然后分别运行该数据通道应用与对端进行实时数据交互。

请参考图 1D，为终端 1 和终端 2 获取数据通道应用的示意图。具体过程如下：1、开发者通过线下流程完成数据通道应用的开发，然后将开发完成的数据通道应用上传到运营商的 DCS。2、DCS 将数据通道应用存储到数据通道应用仓库（data channel application repository，DCAR）。3、DCS 在需要的时候，从 DCAR 下载数据通道应用。4、终端 1 与 DCS 建立 BDC，并通过 BDC 从 DCS 获取所需的数据通道应用。5、终端 2 与 DCS 建立 BDC，并通过 BDC 从 DCS 获取所需的数据通道应用。6、终端 1 和终端 2 建立 ADC，用于传输该数据通道应用的交互数据。

可以理解的，数据通道不关注该数据通道传递的内容及通信格式，但需要通信双方对通信格式达成一致。通过互联网技术，如超文本标记语言（hypertext markup language，HTML）5 界面+JavaScript 脚本（如 Web 应用），可以在数据通道中传递多样化的应用内容，以支撑通话业务的快速创新、快速部署、快速上线。

为了使得 IMS 会话的各方对通信格式，如媒体的类型和/或编码方式达成一致，在建立 IMS 会话的过程中可以进行媒体协商，例如，可以通过 SDP 进行媒体协商。SDP 的作用是向会话参与方传送会话消息。SDP 定义了会话描述的统一格式。下面对 SDP 定义的会话描述信息进行阐述。

8、会话描述信息

本申请中，会话描述信息包括会话信息和媒体描述信息。示例性的，会话信息可以指示会话名称、会话的连接、会话的活动时间（如会话的开始时间和结束时间）和会话所有者的信息。媒体描述信息可以指示媒体类型、传输协议和媒体格式。对于 IP 多播会话，媒体描述还指示多播地址和媒体传输端口，对于 IP 单播会话，媒体描述还指示用于联系地址的媒体和传输端口的远端地址等。

示例性的，会话描述信息包括的内容可以如表 1 所示。表 1 中的“v=0”至“t=2873397496 2873404696”属于会话信息。“t=2873397496 2873404696”之后的内容属于媒体描述信息，例如，从一个“m=”开始到下一个“m=”之前所包括的内容属于一个媒体描述信息。其中，“v=0”指示协议版本。“o=mhandley 2890844526 2890842807 IN IP4 126.16.64.4”指示会话所有者的信息，如会话发起者的名称（mhandley）、主叫方的会话标识（2890844526）、主叫方会话的版本（2890842807）、主叫方的网络类型（IN）、地址类型（IP4）、会话发起方的 IP 地址（126.16.64.4）等。“s=SDP Seminar”指示会话名称。“c=IN IP4 224.2.17.12/127”指示会话的连接，是媒体流使用的 IP 地址。“t=2873397496 2873404696”指示会话的开始时间和及结束时间。“m=application 10001 UDP/DTLS/SCTP webrtc-datachannel”是数据通道的媒体描述，可以指示媒体类型（m=application）、UDP 端口号（10001）、传输协议（UDP/DTLS/SCTP）以及支持的净荷类型（webrtc-datachannel）。“c=IN IP4 192.0.2.1”指示媒体级别的连接信息。“a=sctp-port: 57483”指示 SCTP 端口号（57483）。“a=setup:actpass”指示本端点在 DTLS 协商过程中的角色（既可以发起 DTLS 建链请求，也可以接收 DTLS 建链请求）。“a=fingerprint:SHA-1\4A:AD:B9:B1:3F:82:18:3B:54:02:12:DF:3E:5D:49:6B:19:E5:7C:AB”指示 DTLS 使用的证书指纹签名。“a=tls-id:abc3de65cddef001be82”指示 DTLS 端口号。“a=dcmap:1001 subprotocol="bfcp";label="bfcp"”指示一个数据通道，"bfcp"是指数据通道应用通过该数据通道传输数据所采用的应用层协议名称。

表 1

v=0 o=mhandley 2890844526 2890842807 IN IP4 126.16.64.4 s=SDP Seminar c=IN IP4 224.2.17.12/127 t=2873397496 2873404696
…
m=application 10001 UDP/DTLS/SCTP webrtc-datachannel c=IN IP4 192.0.2.1 a=sctp-port: 57483

```
a=setup:actpass
a=fingerprint:SHA-1\4A:AD:B9:B1:3F:82:18:3B:54:02:12:DF:3E:5D:49:6B:19:E5:7C:AB
a=tls-id:abc3de65cddef001be82
a=dcmap:1001 subprotocol="bfcp";label="bfcp"
...
```

9、SDP 提议方（SDP offerer）和 SDP 应答方（SDP answerer）

为了协商 IMS 会话参与方要使用的 SDP 参数，如上述媒体描述信息，引入了 SDP 提议方和 SDP 应答方。SDP 提议方是想要创建或修改 IMS 会话的一方，SDP 应答方是响应 SDP 提议方的一方。如图 1E 所示，SDP 提议方可以发出 SDP 提议信息（SDP Offer）。SDP 提议信息可以包括一组 SDP 提议方希望使用的媒体流和编码方式，以及 SDP 提议方用于接收和发送媒体流的 IP 地址和端口号，如 SDP 提议信息可以包括如表 1 所示的内容。SDP 应答方接收 SDP 提议信息，并发送 SDP 应答信息。SDP 应答信息可以指示是否接受 SDP 提议信息中的媒体流和编码方式，以及 SDP 应答方用于发送和接收媒体流的 IP 地址和端口号。

综上所述，SDP 提议方和 SDP 应答方可以通过 SIP/SDP 媒体协商流程，确定双方传输数据使用的 IP 地址和端口以及数据通道对应的媒体信息，从而在 IMS 会话中建立一条或多条数据通道。其中，SDP 提议方和 SDP 应答方为处于同一个 IMS 会话中的两个终端；或者，SDP 提议方为 IMS 会话中的一个终端，SDP 应答方为通信网络中的网络设备，如 IMS 应用服务器（application server，AS）；或者，SDP 提议方为通信网络中的网络设备，SDP 应答方为 IMS 会话中的一个终端。

10、WebRTC 用户代理

WebRTC 用户代理可以部署在终端，用于媒体协商。如图 1F 所示，为 WebRTC 用户代理的架构示意图。WebRTC 用户代理可以通过 web 应用编程接口（application programming interface，API）层向 web 应用开发者开放 API。所以 web 应用开发者无需关心复杂的底层技术，通过了解 WebRTC 的大致原理，就可以开发出 web 应用。web 应用是指在 web 浏览器中运行的软件，可以使用标准的浏览器作为客户端。web 应用例如为一个 HTML 网页，通常以 HTML5、CSS 和 JavaScript 混合形式编写。WebRTC C++ API 层可以提供浏览器支持 WebRTC 规范而使用的 API，主要作用是把 WebRTC 的核心功能，如设备功能、音视频流数据采集等功能暴露出来，方便浏览器厂家集成到自家应用中。PeerConnection 是 WebRTC C++ API 层最核心的一个模块，用于实现 P2P 穿墙打洞、通信链路的建立和优选、流数据传输、非音视频数据传输、传输质量报告和统计等功能。会话管理（session management）/抽象信令（abstract signaling）层是 WebRTC 的会话（session）层，用于提供会话功能管理功能，例如，可创建会话、管理会话、管理上下文环境等。这一层还涉及到各种协议，如信令服务器的 SDP 协议等，主要用于信令交互和管理 RTCPeerConnection 的连接状态。音频引擎（voice engine）、视频引擎（video engine）和传输（transport）模块位于引擎层，用于做音视频的处理以及传输文件。音频采集和渲染（audio capture/render）模块、视频采集（video capture）模块以及网络 I/O（network I/O）模块位于驱动层，可以将采集的音视频传输至引擎层做进一步处理。

11、WebRTC 信令模型

WebRTC 可以采用 SDP 进行媒体协商，例如通过 SDP 提议信息和 SDP 应答信息交互 SDP 参数。协议规定 WebRTC JavaScript 应用程序（一种 web app）通过 W3C API 中指定的接口（如图 1F 中的 Web API）控制多媒体会话信令面，并假设 WebRTC JavaScript 应用程序在包含 WebRTC API 的运行环境中运行。此外，WebRTC 更多的关注媒体层面，信令面尽可能地留给应用程序。例如，WebRTC 信令模型可以如图 1G 所示。在图 1G 中，web app 之间可以进行信令交互（如发送 app specific signaling），JavaScript 会话建立协议（JavaScript session establishment protocol，JSEP）实现（例如，可以对应图 1F 中的会话管理/抽象信令层）可以负责维护信令状态机，用于指示媒体协商过程中的媒体协商状态。JSEP 实现提供 web app 所需的接口，web app 在正确的时间调用 API 可驱动信令状态机。JSEP 实现主要根据 web app 提供的信息创建 SDP，包括建立媒体路径所需的媒体类型、格式和所有相关的媒体配置参数，并处理协商好的本地和远程会话描述。不同的 web app 在信令层可以使用不同的协议，如可以使用标准化信令协议（例如，SIP），或使用自定义的信令协议，与对端 web app 进行媒体描述信息交互。例如，JSEP 实现是 WebRTC 注入到浏览器中的实现，为 web app 提供 JS API（对应图 1F 中的 Web API）。web app 为在浏览器中运行的应用，调用 JS API 来实现网络视频聊天等功能。

12、WebRTC 进行 SDP 交换的过程

图 1H 以 peer#1 和 peer#2 为例示出了 WebRTC 进行 SDP 交换的过程。具体来说，WebRTC 可以使用 RTCPeerConnection 这个类创建连接，它包含两个生成 SDP 的方法，分别是 CreateOffer()和 CreateAnswer()。前者由会话发起方（如 Peer#1）调用生成 offerSdp 并发送到信令服务器（signaling server）。后者在会话应答方（如 Peer#2）收到 offerSdp 后被调用生成 answerSdp，answerSdp 后续可以经信令服务器发送给会话发起方。RTCPeerConnection 还包含两个设置 SDP 的方法，分别是 SetLocalDescription()和 SetRemoteDescription()。前者用来设置本地的 Sdp（即 localSdp），后者用来设置接收到的对端 Sdp（即 remote）。对于会话发起方而言，localSdp 即是 offerSdp，remoteSdp 即是 answerSdp；对于会话应答方而言，localSdp 即是 answerSdp，而 remoteSdp 即是接收到的 offerSdp。上述四个方法的有序组合即可构成 WebRTC 设备之间的 SDP 交换过程。其实这也是一种协商过程，因为会话应答方需要根据发起方提供的 SDP（如 ICE 服务器信息、音视频编码信息等）决定如何回复。

13、WebRTC 数据通道的媒体协商

WebRTC 应用可以通过调用 RTCPeerConnection API 实现 WebRTC 数据通道的媒体协商。下面以终端 1（主叫终端）和终端 2（被叫终端）之间协商 WebRTC 数据通道为例进行介绍。如图 1I 所示，WebRTC 数据通道的媒体协商过程可以包括如下步骤：

S101：终端 1 中的 WebRTC 应用调用 RTCPeerConnection API，请求终端 1 中的 WebRTC 用户代理（也可以替换为 JSEP 实现）创建 RTCPeerConnection 实例。

可以理解的，当终端 1 启动 WebRTC 应用，且需要进行媒体协商时，终端 1 发起媒体协商，如调用 RTCPeerConnection API。WebRTC 用户代理创建的新的 RTCPeerConnection 实例，代表终端 1 与终端 2 的一个连接。

可以理解的，RTCPeerConnection 对应 SDP 的会话级别，一个 RTCPeerConnection 实例与一个 SCTP 耦联对应，例如，一个 RTCPeerConnection 实例可以与 SDP 中一个数据通道的媒体描述对应。一个数据通道的媒体描述包括从一个“m=application…”开始到下一个“m=…”之前的内容。

S102：终端 1 中的 WebRTC 应用调用 createDataChannel 方法请求在 RTCPeerConnection 实例（或对象）上创建 WebRTC 数据通道。

可以理解的，createDataChannel 方法还可以携带 WebRTC 数据通道的相关信息，如 WebRTC 数据通道的名称（label）。可选的，createDataChannel 方法还可以携带 WebRTC 数据通道的参数，如 reliable、ordered、maxRetransmitTime、maxRetransmits、protocol、negotiated 或 id 中的一项或多项。其中，reliable、ordered、maxRetransmitTime 和 maxRetransmits 可以用于指示传输是否是可靠的，不可靠的情况下如何重传，以及包是否要求有序到达等。protocol 用于指示 WebRTC 数据通道中传输数据的子协议，对于 WebRTC 来说是透明的。negotiated 和 id 用于指示是通过带外数据进行协商，还是通过 WebRTC 数据通道本身传输的数据进行协商。例如，当 negotiated 为真时，指示通过 SDP 进行带外数据协商，此时，id 可以提供有效的 SID 值；当 negotiated 为假时，指示建立 WebRTC 数据通道时，在底层通道链路建立后发送 Open 消息进行带内协商。

可以理解的，如果 createDataChannel 方法没有携带 WebRTC 数据通道的参数，WebRTC 用户代理可以按默认值设置参数或为 WebRTC 数据通道选择一组参数。

可以理解的，同一个 RTCPeerConnection 上的 WebRTC 数据通道可以共享同一个 SCTP 耦联，或者，同一个 RTCPeerConnection 上的 WebRTC 数据通道对应同一个 WebRTC 数据通道的媒体描述（如同一个 m 行）。因此，WebRTC 用户代理在 RTCPeerConnection 实例上创建的 WebRTC 数据通道，都可以共享一个 SCTP 偶联。换言之，WebRTC 用户代理每次在 RTCPeerConnection 上创建 WebRTC 数据通道，就会在 RTCPeerConnection 对应的 SCTP 耦联上创建一个新的 WebRTC 数据通道，如增加一个 SCTP Stream，该 SCTP Stream 可以对应到 m 行下的一个“a=dcmap”属性行。

示例性的，以 WebRTC 应用之前通过 RTCPeerConnection 创建一个 WebRTC 数据通道，该 WebRTC 数据通道的媒体描述如表 1 所示（如包括“m=application 10001 UDP/DTLS/SCTP webrtc-datachannel”至“a=dcmap:1001 subprotocol="bfcp";label="bfcp"”的内容）为例，若 WebRTC 应用又在 RTCPeerConnection 上创建一个 WebRTC 数据通道，则新建的 WebRTC 数据通道与之前的 WebRTC 数据通道共享同一个 SCTP 偶联，可以在“a=dcmap:1001 subprotocol="bfcp";label="bfcp"”之后再增加“a=dcmap”属性行，得到如表 2 所示的媒体描述。

表 2

```
…
    m=application 10001 UDP/DTLS/SCTP webrtc-datachannel
    c=IN IP4 192.0.2.1
    a=sctp-port: 57483
    a=setup:actpass
    a=fingerprint:SHA-1\4A:AD:B9:B1:3F:82:18:3B:54:02:12:DF:3E:5D:49:6B:19:E5:7C:AB
    a=tls-id:abc3de65cddef001be82
    a=dcmap:1001 subprotocol="bfcp";label="bfcp"
    a=dcmap:2001 subprotocol="msrp";label="msrp"
…
```

S103：终端 1 中的 WebRTC 应用调用 createOffer 方法，请求终端 1 中的 WebRTC 用户代理生成一个新的 SDP 描述返回给 WebRTC 应用。

可以理解的，WebRTC 用户代理还将 SDP 描述添加到 RTCPeerConnection 上。该 SDP 描述是需要与终端 2 协商的媒体信息。SDP 描述包括会话描述信息，示例性的，可以如表 3 所示。

表 3

```
    v=0
    o=mhandley 2890844526 2890842807 IN IP4 126.16.64.4
    s=SDP Seminar
    c=IN IP4 224.2.17.12/127
    t=2873397496 2873404696
…
    m=application 10001 UDP/DTLS/SCTP webrtc-datachannel
    c=IN IP4 192.0.2.1
    a=sctp-port: 57483
    a=setup:actpass
    a=fingerprint:SHA-1\4A:AD:B9:B1:3F:82:18:3B:54:02:12:DF:3E:5D:49:6B:19:E5:7C:AB
    a=tls-id:abc3de65cddef001be82
    a=dcmap:1001 subprotocol="bfcp";label="bfcp"
    a=dcmap:2001 subprotocol="msrp";label="msrp"
```

S104：终端 1 的 WebRTC 应用调用 setLocalDescription API 触发终端 1 中的 WebRTC 用户代理解析该 SDP 描述并创建接收和解码媒体的本地资源。

可以理解的，终端 1 的 WebRTC 用户代理可以触发 WebRTC 数据通道发送参数设置，进而触发 SCTP Streams 创建、设置数据源等。

S105：终端 1 的 WebRTC 应用使用信令机制（如 SIP）通过信令服务器将 SDP offer 发送给终端 2 中的 WebRTC 应用。相应的，终端 2 的 WebRTC 应用接收 SDP offer。

可以理解的，SDP offer 可以包括上述 SDP 描述。

S106：终端 2 的 WebRTC 应用调用 RTCPeerConnection API，使得终端 2 中的 WebRTC 用户代理创建 RTCPeerConnection 实例。

S107：终端 2 的 WebRTC 应用调用 setRemoteDescription API 应用收到的 SDP offer，解析该 SDP offer 并配置发送和编码的本地资源。

S108：终端 2 的 WebRTC 应用在本地 RTCPeerConnecton 对象上创建一个 RTCDataChannel 对象。如终端 2 的 WebRTC 应用调用 createDataChannel 方法在 RTCPeerConnection 对象上创建 WebRTC 数据通道。

S109：终端 2 的 WebRTC 应用调用 createAnswer API 生成 SDP answer。

其中，SDP answer 可以包括有关已连接到会话的媒体、浏览器支持的编解码器。

S110：终端 2 的 WebRTC 应用调用 setLocalDescription API 设置本端 SDP，用于配置接收和解码的本地资源。

S111：终端 2 的 WebRTC 应用通过信令服务器将 SDP Answer 返回给终端 1 的 WebRTC 应用。相应的，

终端 1 的 WebRTC 应用接收 SDP Answer。

S112：终端 1 的 WebRTC 应用调用 setRemoteDescription API 设置远端 SDP，用于配置发送和编码的本地资源。至此，初始设置完成。

S113：终端 1 的 WebRTC 用户代理和终端 2 的 WebRTC 用户代理创建 SCTP/DTLS 耦联及其相应的流。

S114：终端 1 的 WebRTC 用户代理通过 onDataChannel()事件通知终端 1 的 WebRTC 应用连接创建完成，终端 2 的 WebRTC 用户代理通过 onDataChannel()事件通知终端 2 的 WebRTC 应用连接创建完成。

S115~S118：终端 1 的 WebRTC 应用和终端 2 的 WebRTC 应用之间通过 send()接口发送媒体数据，通过 onmessage()事件通知对端接收媒体数据。

14、IMS 数据通道的媒体协商

处于同一个 IMS 会话的通信装置之间（如主叫终端和被叫终端之间，或者终端与网络侧之间）可以通过 IMS SIP/SDP 媒体协商流程，确定双方传递数据使用的 IP 地址、端口，使用的通信协议、传输层所需的参数以及 IMS 数据通道对应的描述信息，从而在 IMS 会话中建立一条或多条 IMS 数据通道。

可以理解的，数据通道应用可以通过 WebRTC API 创建 IMS 数据通道，所以 IMS 数据通道的媒体协商过程与 WebRTC 数据通道的媒体协商过程类似。以终端 1（主叫终端）和终端 2（被叫终端）之间协商 IMS 数据通道为例，终端 1 和终端 2 可以先协商 BDC，并通过 BDC 下载数据通道应用。后续，终端 1 和终端 2 运行该数据通道应用，若终端 1 的用户触发终端 1 发起媒体协商，终端 1 的数据通道应用可以依次调用 RTCPeerConnection API、createDataChannel 方法、createOffer 方法以及 setLocalDescription API，得到 SDP offer（具体的可以参考上述 S101~S104 中对应的描述）。后续，终端 1 可以在 SIP 请求，如邀请请求（INVITE）、更新请求（UPDATE）或重新邀请请求（re-INVITE）携带 SDP offer，通过 IMS 网络将上述请求路由给终端 2。终端 2 获取到 SDP offer 后，依次调用 RTCPeerConnection API、setRemoteDescription API、createDataChannel 方法、createAnswer API 以及 setLocalDescription API，得到 SDP answer（具体的可以参考上述 S106~S110 中对应的描述），并在 SIP 响应，如 18x 响应/200 OK 响应中携带 SDP answer，通过 IMS 网络将上述响应路由给终端 1，从而实现 IMS 数据通道的媒体协商。

可以理解的，在 IMS 会话创建时，可以通过 INVITE 携带发送初始生成的 SDP。后续，如果 SDP 需要更新，可以通过 UPDATE 携带更新的 SDP。在 IMS 会话创建成功之后，如果 SDP 需要更新，可以通过 re-INVITE 携带更新的 SDP。

此外，IMS 数据通道的媒体描述与 WebRTC 数据通道的媒体描述也类似。例如，IMS 数据通道的媒体描述包括的内容可以如表 4 所示。

表 4

```
c=IN IP4 192.0.2.156
...
m=application 10001  UDP/DTLS/SCTP webrtc-datachannel
c= IN IP4 192.0.2.156
a=max-message-size:1024
a=sctp-port: 57483
a=setup:actpass
a=fingerprint:SHA-1 BC:8A:99:A0:E3:28:CA:B3:09:20:1B:FD:21:D5:AC:B6:F3:5E:45:AF
a=tls-id: cd3bea56dced0f35d224
a=dcmap:7216 max-retr=5;label="low loss"
...
```

以上介绍的是一个数据通道应用触发协商 IMS 数据通道的过程。在实际应用中，IMS 会话通常存在多个数据通道应用，因此，会出现多个数据通道应用触发协商 IMS 数据通道的情况。下面以数据通道应用 1（以下简称应用 1）以及数据通道应用 2（以下简称应用 2）触发协商 IMS 数据通道为例进行阐述。如图 2 所示，应用 1 和应用 2 可以分别采用如图 1I 所示的方法协商 IMS 数据通道。因此，两个应用都有自己的会话上下文，且 WebRTC 用户代理分别为两个应用创建各自的 SCTP 耦联，两个应用分别使用不同的 SCTP 耦联发送/接收数据。例如，两个应用分别调用 createOffer 方法时，WebRTC 用户代理可以分别为两个应用创建如表 5 所示的 SDP。

表 5

应用 1 的 SDP	应用 2 的 SDP
…	…
v=0	v=0
o=mhandley 2890844526 2890842807 IN IP4 126.16.64.4	o=mhandley 2890844526 2890842807 IN IP4 126.16.64.4
s=SDP Seminar	s=SDP Seminar
t=2873397496 2873404696	t=2873397496 2873404696
m=application 10001 UDP/DTLS/SCTP webrtc-datachannel	m=application 10001 UDP/DTLS/SCTP webrtc-datachannel
c=IN IP4 192.0.2.1	c=IN IP4 192.0.2.1
a=sctp-port: 57483	a=sctp-port: 57485
a=setup:actpass	a=setup:actpass
a=fingerprint:SHA-1\4A:AD:B9:B1:3F:82:18:3B:54:02:12:DF:3E:5D:49:6B:19:E5:7C:AB	a=fingerprint:SHA-1\4A:AD:B9:B1:3F:82:18:3B:54:02:12:DF:3E:5D:49:6B:19:E5:7C:AB
a=tls-id:abc3de65cddef001be82	a=tls-id:abc3de65cddef001be82
a=mid:1	a=mid:1
a=dcmap:7216 max-retr=5;label="low loss"	a=dcmap:7384 max-retr=5;label="low latency"
…	…

上述为多个数据通道应用创建数据通道的方法会导致资源开销较大。

为了解决上述问题，本申请提供一种通信方法。在该方法中，通信实体在接收到用于请求创建第一数据通道的第一请求后，可以确定已创建的连接（如 SCTP 耦联）能否承载第一数据通道。如果存在已创建的连接能承载第一数据通道，则通信实体指示用第一连接承载第一数据通道，第一连接为已创建的连接中第一数据通道对应的连接。如果不存在已创建的连接能承载第一数据通道，则通信实体指示用新的连接承载第一数据通道，并创建新的连接。

在上述过程中，通信实体在创建数据通道之前，可以在已创建的连接中确定能承载该数据通道的连接。换言之，通过上述方法创建数据通道时，可以复用已创建的连接，所以可以降低资源开销。

可以理解的，本申请提供的方法可用于各种通信系统。例如该通信系统可以为长期演进（long term evolution，LTE）系统、5G 通信系统、无线保真（wireless fidelity，WiFi）系统、3GPP 相关的通信系统、5G 之后演进的通信系统（如：第六代（6th generation，6G）通信系统）、或多种系统融合的系统等，不予限制。其中，5G 还可以称为新无线（new radio，NR）。下面以图 3A 和图 3B 所示通信场景为例，对本申请提供的方法进行描述。图 3A 和图 3B 仅为示意图，并不构成对本申请提供的技术方案的适用场景的限定。

图 3A 和图 3B 为本申请适用的通信场景的示意图。图 3A 示出了主叫终端和被叫终端处于同一个 IMS 网络时的通信场景，图 3B 示出了主叫终端和被叫终端处于不同 IMS 网络时的通信场景。下面进行具体阐述。

在图 3A 中，终端 302 和终端 303 可通过 IMS 网络 301 通信，IMS 网络 301 即是呼叫始发网络，也是呼叫终结网络。终端 302 为主叫终端，终端 303 为被叫终端，或者终端 302 为被叫终端，终端 303 为主叫终端。IMS 网络 301 包括数据通道应用服务器（DC AS）、与 DC AS 通信连接的网络开放功能（network exposure function，NEF）实体和 DCS、与 DCS 通信连接的 IMS 归属用户服务器（home subscriber server，HSS）实体和 IMS AS、与 IMS AS 以及 IMS HSS 实体通信连接的呼叫会话控制功能（call session control function，CSCF）实体、以及与 CSCF 实体通信连接的 IMS 接入媒体网关（access media gateway，IMS-AGW）。

其中，DCS 可以包括数据通道信令功能（data channel signaling function，DCSF）实体和媒体功能（media function，MF）实体。可选的，DCS 还包括数据通道应用仓库（data channel application repository，DCAR）实体。图 3A 示出了 DCSF 和 DCAR 部署在同一个实体上，MF 单独部署在一个实体上。在具体应用中，不限制 DCSF、DCAR 和 MF 的部署方式。例如，DCSF、DCAR 和 MF 可以单独部署在不同的实体上，或者 DCSF、DCAR 和 MF 中的至少两个部署在同一个实体上。

CSCF 实体是 IMS 内部的功能实体，是整个 IMS 的核心。主要负责处理多媒体呼叫会话过程中的信令

控制。它可以管理 IMS 的用户鉴权、IMS 承载面服务质量（quality of service，QoS）、与其它网元配合进行 SIP 会话的控制，以及业务协商和资源分配等。根据功能的不同，CSCF 实体可以划分为代理 CSCF（proxy CSCF，P-CSCF）实体，查询 CSCF（interrogating CSCF，I-CSCF）实体，服务 CSCF（serving CSCF，S-CSCF）实体等。图 3A 示出了 I-CSCF 与 S-CSCF 部署在同一个实体上，P-CSCF 单独部署在一个实体上。在具体应用中，不限制 I-CSCF、S-CSCF 和 P-CSCF 的部署方式。例如，I-CSCF、S-CSCF 和 P-CSCF 可以单独部署在不同的实体上，或者 I-CSCF、S-CSCF 和 P-CSCF 中的至少两个部署在同一个实体上。

可以理解的，在图 3A 所示的通信场景中，终端 302 和终端 303 可采用本申请提供的方法协商终端 302 和终端 303 参与的 IMS 会话中的数据通道。或者，终端 302 和 IMS AS（或 DCSF 实体）可采用本申请提供的方法协商终端 302 和 IMS AS（或 DCSF 实体）参与的 IMS 会话中的数据通道。或者，终端 303 和 IMS AS（或 DCSF 实体）可采用本申请提供的方法协商终端 303 和 IMS AS（或 DCSF 实体）参与的 IMS 会话中的数据通道。或者，终端 302 和 DC AS 可采用本申请提供的方法协商终端 302 和 DC AS 参与的 IMS 会话中的数据通道。或者，终端 303 和 DC AS 可采用本申请提供的方法协商终端 303 和 DC AS 参与的 IMS 会话中的数据通道。

在图 3B 中，终端 313 和终端 314 可通过 IMS 网络 311 以及 IMS 网络 312 通信，IMS 网络 311 和 IMS 网络 312 中一个是呼叫始发网络，一个是呼叫终结网络。例如，若终端 313 为主叫终端，终端 314 为被叫终端，则 IMS 网络 311 是呼叫始发网络，IMS 网络 312 是呼叫终结网络；若终端 313 为被叫终端，终端 314 为主叫终端，则 IMS 网络 311 是呼叫终结网络，IMS 网络 312 是呼叫始发网络。IMS 网络 311 和 IMS 网络 312 包括的实体类似，例如，IMS 网络 311 或 IMS 网络 312 包括 DC AS、与 DC AS 通信连接的 NEF 实体和 DCS、与 DCS 通信连接的 IMS HSS 实体和 IMS AS、与 IMS AS 通信连接的 CSCF 实体、与 CSCF 实体通信连接的 IMS-AGW 和互连边界控制功能（interconnection border control function，IBCF）实体、以及与 IMS-AGW 通信连接的转换网关（transition gateway，TrGW）。其中，DCS 和 CSCF 实体的介绍可以参考图 3A 中对应的描述。

可以理解的，在图 3B 所示的通信场景中，终端 313 和终端 314 可采用本申请提供的方法协商终端 313 和终端 314 参与的 IMS 会话中的数据通道。或者，终端 313 和 IMS 网络 311 中的 IMS AS（或 DCSF 实体）可采用本申请提供的方法协商终端 313 和 IMS 网络 311 中的 IMS AS（或 DCSF 实体）参与的 IMS 会话中的数据通道。或者，终端 313 和 IMS 网络 312 中的 IMS AS（或 DCSF 实体）可采用本申请提供的方法协商终端 313 和 IMS 网络 312 中的 IMS AS（或 DCSF 实体）参与的 IMS 会话数据通道。或者，终端 314 和 IMS 网络 312 中的 IMS AS（或 DCSF 实体）可采用本申请提供的方法协商终端 314 和 IMS 网络 312 中的 IMS AS（或 DCSF 实体）参与的 IMS 会话中的数据通道。或者，终端 314 和 IMS 网络 311 中的 IMS AS（或 DCSF 实体）可采用本申请提供的方法协商终端 314 和 IMS 网络 311 中的 IMS AS（或 DCSF 实体）参与的 IMS 会话中的数据通道。或者，终端 313 和 DC AS 可采用本申请提供的方法协商终端 313 和 DC AS 参与的 IMS 会话中的数据通道。或者，终端 314 和 DC AS 可采用本申请提供的方法协商终端 314 和 DC AS 参与的 IMS 会话中的数据通道。

为了更清楚地了解本申请提供的通信场景，下面对图 3A 和图 3B 中各种实体的功能进行具体阐述。

DC AS 可以用于提供数据通道业务逻辑。在图 3A 或图 3B 中，DC AS 部署在 IMS 网络中，可以通过接口（比如服务化接口）和 IMS 网络中的其它实体直连。在具体应用中，DC AS 也可以不部署在 IMS 网络中，而是部署在 IMS 网络外，通过 NEF 实体（或者 IMS 边界网管）和 IMS 网络中的其它网元相连。

NEF 实体可以用于将核心网（如 5G 核心网）的各种服务安全地开放给第三方。

DCSF 实体可以提供数据通道信令控制功能。

DCAR 可以用于存储数据通道应用。例如，开发者完成数据通道应用开发后，上传到运营商的 DCS，再由 DCS 保存到 DCAR 中。在需要的时候，DCS 从 DCAR 下载数据通道应用到本地，进行后继处理。

IMS HSS 实体可以负责存放 IMS 用户签约数据。

IMS AS 是 IMS 网络中位于最上层的应用层设备，提供各种业务，包括 IMS 基本业务和补充业务、多媒体会议、融合通讯、短信网关、标准话务台等业务。IMS AS 可以与 CSCF 实体之间进行交互，实现各种网络业务的触发和执行。IMS AS 可以与终端协商数据通道。示例性的，IMS AS 可以为多媒体电话应用服务器（multiMedia telephony application server，MMTEL AS）或者电话应用服务器（telephony application server，TAS）。

MF 实体可以用于提供数据通道媒体资源管理功能。

S-CSCF 实体可以负责 IMS 用户的注册、鉴权、会话，路由和业务触发等。

I-CSCF 实体是 IMS 用户归属网络的统一入口点，可以负责分配或者查询为用户服务的 S-CSCF。

P-CSCF 实体是用户接入 IMS 网络的入口节点，可以负责 IMS 用户与归属网络之间 SIP 信令的转发。

IMS AGW 可以提供 IMS 网络接入网关和媒体网关功能。

IBCF 实体可以实现语音、视频、多媒体业务的信令互通、拓扑隐藏和 IP 安全防护功能等。

TrGW 实体可以实现语音、视频、多媒体业务的媒体互通、拓扑隐藏和 IP 安全防护功能，还支持语音编解码格式转换等功能。

可以理解的，关于终端的介绍可以参考前文对本申请涉及的技术术语中的解释说明。另外，由于本申请中的终端支持数据通道，所以本申请中的终端可以部署支持数据通道的 IMS 多媒体电话服务（multimedia telephony service for IMS，MTSI）客户端（MTSI client）。支持数据通道 MTSI client 还可以称为 DCMTSI client。DCMTSI client 的架构可以如图 3C 所示。

在图 3C 中，DCMTSI client 包括基于数据包的网络接口，以及分别与其连接的音频解码器、视频解密器、用于处理下行数据的文本媒体（以下称为文本媒体 1）、用于处理下行数据的数据通道模块（以下称为数据通道模块 1）、会话创建和控制模块、用于处理上行数据的数据通道模块（以下称为数据通道模块 2）、音频编码器、视频编码器以及用于处理上行数据的文本媒体（以下称为文本媒体 2）。其中，基于数据包的网络接口可以通过图 1C 中的协议栈实现。

其中，基于数据包的网络接口可以采用 3GPP L2 协议与 3GPP 网络通信，用于媒体传输。3GPP L2 协议包括分组数据汇聚层协议（packet data convergence protocol，PDCP）或者业务数据适配协议（service data adaptation protocol，SDAP）。例如，DCMTSI client 可以通过基于数据包的网络接口接收网络侧的媒体数据，或者向网络侧发送媒体数据。

可以理解的，音频编码器可以将麦克风采集的声音进行编码，并将编码后的音频通过基于数据包的网络接口发送，由另一个 DCMTSI client 接收。另一个 DCMTSI client 接收到编码后的音频后，可以由音频解码器解码，并通过扬声器播放。视频编码器可以将相机捕获的图像进行编码，并将编码后的图像通过基于数据包的网络接口发送，由另一个 DCMTSI client 接收。另一个 DCMTSI client 接收到编码后的图像后，可以由视频解码器解码，并在显示器上渲染图像。文本媒体 2 可以获取键盘上输入的字符或屏幕上绘制的字符，并将这些字符信息通过基于数据包的网络接口发送，由另一个 DCMTSI client 接收。另一个 DCMTSI client 接收到这些字符信息后，可以在显示器上实时渲染。此外，上述编码后的音频、上述编码后的图像和上述字符信息可以在被激活后经基于数据包的网络接口发送。激活是针对预开通的用户进行的一种操作，该操作将使得用户进入正常状态，并可以开始使用运营商提供的服务。

可以理解的，音频解码器用于解码通过基于数据包的网络接口接收的音频数据，解码后的音频可以通过扬声器播放。视频解码器用于解码通过基于数据包的网络接口接收的图像，并在显示器上渲染图像。文本媒体 1 用于处理通过基于数据包的网络接口接收的字符信息，处理后的字符可以在显示器上实时渲染。应理解，上述解码后的音频、解码后的图像以及处理后的字符可以经同步处理后，进行播放或显示。

可以理解的，对于用于实时交互的其他与多媒体电话会话相关的数据，不需要任何的编解码器，可由 DCMTSI client 生成或消费。例如，这些数据可以由数据通道模块 2 处理后经基于数据包的网络接口发送，或者这些数据可以由数据通道模块 1 控制通过扬声器播放相应的声音、通过显示器显示相应的图像或者在用户界面上呈现相应的效果。另外，用户界面可以通过下行数据通道接收交互数据，和 Web 界面以及相关脚本交互，并与用户接收的语音、视频和文本数据需要进行同步，以处理数据通道 I/O 和数据格式化。

可以理解的，为了使得数据通道应用可以在不同终端和/或不同的网络之间运行，终端需要支持相应的数据通道逻辑架构。

示例性的，如图 3D 所示，为一种终端支持的数据通道的逻辑架构示意图。在图 3D 中，终端上可以部署数据通道应用、数据通道运行环境（data channel run time）、DCMTSI client 以及人机交互界面（human machine interface）的相关功能。其中，数据通道应用中可以部署 JavaScript 应用交互式逻辑（JavaScript application interactive logic）和 HTML/CSS 文本（HTML/CSS document）。人机交互界面可以实现数据通道应用的功能，如播放声音、显示图像、采集声音和拍摄图像等。除此之外，数据通道应用还可以调用特定领域（domain specific）API 实现 AR 功能、人工智能（artificial intelligence）功能或者查

询本机系统库（native system library）。数据通道运行环境可以为数据通道应用提供 API，例如，可以提供 RTCPeerConnection API、RTCDataChannel API、RTCSignnallingService API 等，用于创建数据通道，也可用于通过数据通道发送和接收应用层数据。其中，RTCPeerConnection API 和 RTCDataChannel API 可以由 WebRTC 用户代理提供。RTCSignallingService API 可以由终端的操作系统提供。DCMTSI client 可以为数据通道提供 IMS 媒体面（media plane）的支持和 IMS 信令面（signaling plane）的支持。例如，数据通道应用可以调用 API 触发数据通道的媒体协商。数据通道运行环境接收到相应指令后，可以与 DCMTSI client 中的媒体模块（如包括数据通道媒体（data channel media）模块）和信令模块（如包括数据通道描述（data channel description）模块）交互。媒体模块和信令模块可以分别负责实现媒体的协议栈和信令的协议栈，并通过 IMS 网络与远端终端进行信令协商和媒体互通，来创建数据通道，以及通过数据通道发送和接收应用层数据。综上，数据通道运行环境可以作为数据通道应用和 DCMTSI client 的桥梁，将数据通道应用触发的信令操作传递到终端底层的信令模块进行相应的媒体协商，将远端终端发来的信令消息进行解析通知给数据通道应用。数据通道运行环境还可以将数据通道应用要发送的媒体数据通过终端底层的媒体模块发送出去，并将远端终端发来的媒体数据分发给数据通道应用。

可以理解的，数据通道运行环境、DCMTSI client 以及人机交互界面的相关功能可以是终端出厂就支持的能力，数据通道应用是终端出厂后根据需要从网络获取的。

除了图 3D 所示的架构之外，本申请还提供一种终端支持的数据通道的逻辑架构示意图，具体的，如图 3E 所示。终端上可以部署数据通道应用、中间代理（agent）、WebRTC 用户代理和 IMS 组件。其中，中间代理可以管理终端创建的数据通道对应的连接，如 SCTP 耦联。例如，数据通道应用发起创建第一数据通道的请求后，中间代理可以确定已创建的连接中是否有连接能承载第一数据通道的连接。如果有能承载第一数据通道的第一连接，中间代理向 WebRTC 用户代理请求在该第一连接中增加第一数据通道对应的流；如果没有能承载第一数据通道的连接，中间代理向 WebRTC 用户代理请求创建新连接。WebRTC 用户代理接收到中间代理的请求后，可以执行相应的操作，如在第一连接中增加第一数据通道对应的流，或者创建新连接。中间代理还可以与 IMS 组件交互，以实现 SIP 信令的发送和接收。IMS 组件可以具备 IMS 信令面的功能。

可以理解的，WebRTC 用户代理以及 IMS 组件可以是终端出厂就支持的能力，数据通道应用是终端出厂后根据需要从网络获取的。中间代理既可以是终端出厂就支持的能力，也可以是终端出厂后根据需要获取的（如从网络获取的），不予限制。

应理解，图 3A 或图 3B 仅是本申请提供的方法可应用的通信场景的示例，在具体应用中，本申请提供的方法可应用的通信场景还可以是其他形式的，例如，可以包括比图 3A 或图 3B 更多或更少的实体，不予限制。

可选的，本申请图 3A 或图 3B 中的终端也可以称之为通信装置，其可以是一个通用设备或者是一个专用设备，本申请对此不作具体限定。

可选的，本申请图 3A 或图 3B 中的终端的相关功能可以由一个设备实现，也可以由多个设备共同实现，还可以是由一个设备内的一个或多个功能模块实现，本申请对此不作具体限定。可以理解的是，上述功能既可以是硬件设备中的网络元件，也可以是在专用硬件上运行的软件功能，或者硬件与软件的结合，或者平台(例如，云平台)上实例化的虚拟化功能。

在具体实现时，图 3A 或图 3B 中的终端都可以采用图 4 所示的组成结构，或者包括图 4 所示的部件。图 4 所示为可适用于本申请的通信装置的硬件结构示意图。该通信装置 40 包括至少一个处理器 401 和至少一个通信接口 404，用于实现本申请提供的方法。该通信装置 40 还可以包括通信线路 402 和存储器 403。

处理器 401 可以是一个通用中央处理器（central processing unit，CPU），微处理器，特定应用集成电路（application-specific integrated circuit，ASIC），或一个或多个用于控制本申请方案程序执行的集成电路。

通信线路 402 可包括一通路，在上述组件之间传送信息，例如总线。

通信接口 404，用于与其他设备或通信网络通信。通信接口 404 可以是任何收发器一类的装置，如可以是以太网接口、无线接入网（radio access network，RAN）接口、无线局域网（wireless local area networks，WLAN）接口、收发器、管脚、总线、接口电路或收发电路等。

存储器 403 可以是只读存储器（read-only memory，ROM）或可存储静态信息和指令的其他类型的静态存储设备，随机存取存储器（random access memory，RAM）、高速缓存（cache）或者可存储信息和指令的其他类型的动态存储设备，也可以是电可擦可编程只读存储器（electrically erasable programmable

read-only memory，EEPROM）、只读光盘（compact disc read-only memory，CD-ROM）或其他光盘存储、光碟存储（包括压缩光碟、激光碟、光碟、数字通用光碟、蓝光光碟等）、磁盘存储介质或者其他磁存储设备、或者能够用于携带或存储具有指令或数据结构形式的期望的程序代码并能够由计算机存取的任何其他介质，但不限于此。存储器可以是独立存在，通过通信线路 402 与处理器 401 相耦合。存储器 403 也可以和处理器 401 集成在一起。本申请提供的存储器通常可以具有非易失性。

其中，存储器 403 用于存储执行本申请提供的方案所涉及的计算机执行指令，并由处理器 401 来控制执行。处理器 401 用于执行存储器 403 中存储的计算机执行指令，从而实现本申请提供的方法。或者，可选的，本申请中，也可以是处理器 401 执行本申请下述提供的方法中的处理相关的功能，通信接口 404 负责与其他设备或通信网络通信，本申请对此不作具体限定。

可选的，本申请中的计算机执行指令也可以称之为应用程序代码，本申请对此不作具体限定。

本申请中的耦合是装置、单元或模块之间的间接耦合或通信连接，可以是电性，机械或其它的形式，用于装置、单元或模块之间的信息交互。

作为一种实施例，处理器 401 可以包括一个或多个 CPU，例如图 4 中的 CPU0 和 CPU1。

作为一种实施例，通信装置 40 可以包括多个处理器，例如图 4 中的处理器 401 和处理器 407。这些处理器中的每一个可以是一个单核（single-CPU）处理器，也可以是一个多核（multi-CPU）处理器。这里的处理器可以指一个或多个设备、电路、和/或用于处理数据（例如计算机程序指令）的处理核。

作为一种实施例，通信装置 40 还可以包括输出设备 405 和/或输入设备 406。输出设备 405 和处理器 401 耦合，可以以多种方式来显示信息。例如，输出设备 405 可以是液晶显示器（liquid crystal display，LCD），发光二极管（light emitting diode，LED）显示设备，阴极射线管（cathode ray tube，CRT）显示设备，或投影仪（projector）等。输入设备 406 和处理器 401 耦合，可以以多种方式接收用户的输入。例如，输入设备 406 可以是鼠标、键盘、触摸屏设备或传感设备等。

可以理解的，图 4 中示出的组成结构并不构成对该通信装置的限定，除图 4 所示部件之外，该通信装置可以包括比图示更多或更少的部件，或者组合某些部件，或者不同的部件布置。

下面将结合附图，对本申请提供的方法进行描述。下述实施例中的各网元可以具备图 4 所示部件，不予赘述。

可以理解的是，本申请下述实施例中各个网元之间的消息名字或消息中各参数的名字等只是一个示例，具体实现中也可以是其他的名字，本申请对此不作具体限定。

可以理解的是，在本申请中，“/”可以表示前后关联的对象是一种“或”的关系，例如，A/B 可以表示 A 或 B；“和/或”可以用于描述关联对象存在三种关系，例如，A 和/或 B，可以表示：单独存在 A，同时存在 A 和 B，单独存在 B 这三种情况，其中 A，B 可以是单数或者复数。此外，类似于“A、B 和 C 中的至少一项”或“A、B 或 C 中的至少一项”的表述通常用于表示如下中任一项：单独存在 A；单独存在 B；单独存在 C；同时存在 A 和 B；同时存在 A 和 C；同时存在 B 和 C；同时存在 A、B 和 C。以上是以 A、B 和 C 共三个元素进行举例来说明该项目的可选用条目，当表述中具有更多元素时，该表述的含义可以按照前述规则获得。

为了便于描述本申请的技术方案，在本申请中，可以采用“第一”、“第二”等字样对功能相同或相似的技术特征进行区分。该“第一”、“第二”等字样并不对数量和执行次序进行限定，并且“第一”、“第二”等字样也并不限定一定不同。在本申请中，“示例性的”或者“例如”等词用于表示例子、例证或说明，被描述为“示例性的”或者“例如”的任何实施例或设计方案不应被解释为比其它实施例或设计方案更优选或更具优势。使用“示例性的”或者“例如”等词旨在以具体方式呈现相关概念，便于理解。

可以理解，说明书通篇中提到的“实施例”意味着与实施例有关的特定特征、结构或特性包括在本申请的至少一个实施例中。因此，在整个说明书各个实施例未必一定指相同的实施例。此外，这些特定的特征、结构或特性可以任意适合的方式结合在一个或多个实施例中。可以理解，在本申请的各种实施例中，各过程的序号的大小并不意味着执行顺序的先后，各过程的执行顺序应以其功能和内在逻辑确定，而不应对本申请的实施过程构成任何限定。

可以理解，在本申请中，“当……时”、“在……的情况下”、“若”以及“如果”均指在某种客观情况下会做出相应的处理，并非是限定时间，且也不要求实现时一定要有判断的动作，也不意味着存在其它限定。

可以理解，本申请中的一些可选的特征，在某些场景下，可以不依赖于其他特征，比如其当前所基于的方案，而独立实施，解决相应的技术问题，达到相应的效果，也可以在某些场景下，依据需求与其

他特征进行结合。相应的，本申请中给出的装置也可以相应的实现这些特征或功能，在此不予赘述。

可以理解的，本申请中同一个步骤或者具有相同功能的步骤或者技术特征在不同实施例之间可以互相参考借鉴。

应理解，本申请下述实施例中的通信实体可以执行本申请中的部分或全部步骤，这些步骤仅是示例，本申请还可以执行其它步骤或者各种步骤的变形。此外，各个步骤可以按照本申请呈现的不同的顺序来执行，并且有可能并非要执行本申请中的全部步骤。

可以理解的，本申请下述提供的方法中是以通信实体作为执行主体为例来示意该方法，但本申请并不限制该执行主体。例如，本申请下述实施例提供的方法中的通信实体也可以是支持该通信实体实现该方法的芯片、芯片系统、或处理器，还可以是能实现全部或部分通信实体功能的逻辑节点、逻辑模块或软件。

如图 5 所示，为本申请提供的一种通信方法，该方法可以包括如下步骤：

S501：通信实体接收第一请求。

本申请中，第一请求用于请求创建第一数据通道。第一数据通道为第一通信装置和第二通信装置之间的数据通道，该第一数据通道可以用于承载第一应用的数据。其中，第一数据通道与第一通信装置以及第二通信装置之间的 IMS 会话相关联。例如，第一数据通道为该 IMS 会话中的逻辑连接，用于在 IMS 会话存续期间传输第一应用的数据。可以理解的，第一数据通道为 ADC，第一应用为数据通道应用，第一应用部署在第一通信装置上。IMS 会话、ADC 以及数据通道应用的介绍可以参考前文对应的描述，不再赘述。

示例性的，以图 3A 所示的通信场景为例，第一通信装置为终端 302，第二通信装置为终端 303、IMS AS、DCSF 实体或 DC AS，或者第一通信装置为终端 303，第二通信装置为终端 302、IMS AS、DCSF 实体或 DC AS。

示例性的，以图 3B 所示的通信场景为例，第一通信装置为终端 313，第二通信装置为终端 314、IMS 网络 311 或 IMS 网络 312 中的 IMS AS、IMS 网络 311 或 IMS 网络 312 中的 DCSF 实体、或者 IMS 网络 311 或 IMS 网络 312 中的 DC AS。或者，第一通信装置为终端 314，第二通信装置为终端 313、IMS 网络 311 或 IMS 网络 312 中的 IMS AS、IMS 网络 311 或 IMS 网络 312 中的 DCSF 实体、或者 IMS 网络 311 或 IMS 网络 312 中的 DC AS。

本申请中，第一请求可以是由第一通信装置上的第一应用触发或发送的，相应地，第一通信装置上的 WebRTC 用户代理接收该第一请求，此时，通信实体可以为第一通信装置上的 WebRTC 用户代理，也可以是第一通信装置上的其它功能模块，不予限定。假设第一通信装置采用图 3D 所示的终端架构，那么第一请求可以是由第一应用向数据通道运行环境中的 WebRTC 用户代理发送的。假设第一通信装置采用图 3E 所示的终端架构，那么第一请求可以是由第一应用触发中间代理向 WebRTC 用户代理发送的。

可以理解的，第一通信装置在创建第一数据通道时，可以是 SDP 提议方或者 SDP 应答方。下面分别以第一通信装置为 SDP 提议方或 SDP 应答方为例，介绍通信实体接收第一请求的过程。

场景 1：第一通信装置为 SDP 提议方。

可以理解的，第一应用可以调用 API 触发第一请求。下面以下述方式 1.1~方式 1.3 为例进行介绍。

方式 1.1：第一应用调用 RTCPeerConnection API、createDataChannel 方法以及 createOffer 方法向 WebRTC 用户代理请求创建第一数据通道。

方式 1.1 的具体过程与图 1I 中的 S101~S103 类似，可以参考上述 S101~S103 中对应的介绍，此处不做赘述。此时，第一请求具体可以包括 RTCPeerConnection 命令、createDataChannel 命令或 createOffer 命令中的至少一个命令。

方式 1.2：第一应用调用 RTCPeerConnection API、createDataChannel 方法以及 createOffer 方法向 WebRTC 用户代理请求创建第一数据通道，并指示是否存在已创建的连接能承载第一数据通道。此时，第一请求具体可以包括 createDataChannel 命令或 createOffer 命令中的至少一个命令。

其中，已创建的连接可以指通信实体已创建的耦联，如 SCTP 耦联，或者 DTLS/SCTP 耦联。一个耦联可以与一组协议地址信息和协议端口信息关联。例如，一个耦联可以与 IP 地址信息、UDP 端口信息以及 SCTP 端口信息关联。因此，一个连接能承载第一数据通道可以理解为第一数据通道可以使用该连接关联的 IP 地址、UDP 端口和 SCTP 端口，或者第一数据通道对应的 IP 地址、UDP 端口和 SCTP 端口，与该连接关联的 IP 地址、UDP 端口和 SCTP 端口完全相同。一个连接不能承载第一数据通道可以理解为第一

数据通道不能使用该连接关联的 IP 地址、UDP 端口和 SCTP 端口，或者第一数据通道对应的 IP 地址、UDP 端口和 SCTP 端口，与该连接关联的 IP 地址、UDP 端口和 SCTP 端口不完全相同。应理解，本申请中一个耦联也可以与 IP 地址信息、UDP 端口信息以及 SCTP 端口信息中的一个信息或两个信息关联，不予限制。为了便于描述，本申请下述实施例以一个耦联与 IP 地址信息、UDP 端口信息以及 SCTP 端口信息关联为例进行阐述。

一种可能的设计，一个连接能承载第一数据通道，包括：第一数据通道的服务质量要求与该连接的服务质量要求相同，或者第一数据通道的服务质量要求低于该连接的服务质量要求。可选的，一个连接能承载第一数据通道，还包括：该连接对应的端点与第一数据通道对应的端点相同，也就是说，该连接对应的端点也是第一通信装置和第二通信装置，或者该连接是第一通信装置和第二通信装置协商建立的。一个连接不能承载第一数据通道，包括：第一数据通道的服务质量要求大于该连接的服务质量要求。可选的，一个连接不能承载第一数据通道，还包括：该连接对应的端点与第一数据通道对应的端点不完全相同。例如，该连接对应的端点是第一通信装置和除第二通信装置之外的通信装置。本申请中的服务质量要求（如第一数据通道的服务质量要求、连接的服务质量要求以及下述实施例中的第二数据通道的服务质量要求等）可以包括对丢包率的要求，和/或，对时延的要求等。

一种可能的实现方式，第一应用从 WebRTC 用户代理获取已创建的连接的信息，并确定已创建的连接能否承载第一数据通道。若已创建的连接能承载第一数据通道，第一应用在请求创建第一数据通道时，携带用于承载第一数据通道的连接的信息，若已创建的连接不能承载第一数据通道，第一应用在请求创建第一数据通道时，不携带第一数据通道对应的连接的信息。

示例性的，第一应用调用 RTCPeerConnection API，并调用 WebRTC 用户代理提供的用于查询已创建的连接的 API。WebRTC 用户代理向第一应用发送已创建的连接的信息。其中，已创建的连接的信息可以指示每个已创建的连接，例如，指示每个连接关联的 IP 地址、UDP 端口号和 SCTP 端口号。第一应用可以在已创建的连接中查询能承载第一数据通道的连接。如果查询到第一连接能承载第一数据通道（即第一连接为已创建的连接中第一数据通道对应的连接），第一应用可以向 WebRTC 用户代理指示第一连接。例如，第一应用在调用 createDataChannel 方法或 createOffer 方法时携带第一连接的信息，如携带第一连接的标识、第一连接关联的 IP 地址、第一连接关联的 UDP 端口号或第一连接关联的 SCTP 端口号中的至少一项。如果未查询到能承载第一数据通道的连接，第一应用在调用 createDataChannel 方法或 createOffer 方法时不携带第一数据通道对应的连接的信息。

方式 1.3：第一应用通过中间代理向 WebRTC 用户代理请求创建第一数据通道。

一种可能的实现方式，第一应用调用中间代理提供的 API，触发中间代理向 WebRTC 用户代理请求创建第一数据通道。

示例性的，第一应用调用中间代理提供的 RTCPeerConnection API 中的 createDataChannel 方法以及 createOffer 方法，向中间代理请求创建第一数据通道。中间代理接收到第一应用的请求后，确定已创建的连接能否承载第一数据通道。若已创建的连接中的第一连接能承载第一数据通道，中间代理可以向 WebRTC 用户代理请求在第一连接中增加第一数据通道对应的流。例如，中间代理调用 createDataChannel 方法以及 createOffer 方法请求在第一连接中增加第一数据通道对应的流。若已创建的连接不能承载第一数据通道，中间代理可以向 WebRTC 用户代理请求创建新连接。例如，中间代理调用 RTCPeerConnection API、createDataChannel 方法以及 createOffer 方法向 WebRTC 用户代理请求创建新连接，即第一数据通道对应的连接。此时，第一请求具体可以包括中间代理发送的 RTCPeerConnection 命令、createDataChannel 命令或 createOffer 命令中的至少一个命令。

场景 2：第一通信装置为 SDP 应答方。

可以理解的，通信实体接收到来自 SDP 提议方的 SDP 提议信息后，可以将 SDP 提议信息中与第一应用有关的 SDP 信息发送给第一应用。第一应用接收到该 SDP 信息后，可以调用 API 触发第一请求。下面以下述方式 2.1~方式 2.3 为例进行介绍。

方式 2.1：第一应用调用 RTCPeerConnection API、createDataChannel 方法以及 createAnswer 方法向 WebRTC 用户代理请求创建第一数据通道。此时，第一请求具体可以包括 RTCPeerConnection 命令、createDataChannel 命令或 createAnswer 命令中的至少一个命令。

方式 2.1 的具体过程与图 1I 中的 S106~S109 类似，可以参考上述 S106~S109 中对应的介绍，此处不做赘述。

方式 2.2：第一应用调用 RTCPeerConnection API、createDataChannel 方法以及 createAnswer 方法向 WebRTC 用户代理请求创建第一数据通道，并指示是否存在已创建的连接能承载第一数据通道。此时，第一请求具体可以包括 createDataChannel 命令或 createAnswer 命令中的至少一个命令。

可以理解的，方式 2.2 的过程与方式 1.2 的过程类似，可以参考上述方式 1.2 中对应的描述。不同的是，在方式 1.2 中，第一应用调用的是 createOffer 方法，用于创建 SDP offer，在方式 2.2 中，第一应用调用的是 createAnswer 方法，用于创建 SDP answer。

方式 2.3：第一应用通过中间代理向 WebRTC 用户代理请求创建第一数据通道。

一种可能的实现方式，第一应用调用中间代理提供的 API，触发中间代理向 WebRTC 用户代理请求创建第一数据通道。具体的，可以参考上述方式 1.3 中对应的描述。不同的是，在方式 1.2 中，第一应用和中间代理调用的是 createOffer 方法，用于创建 SDP offer，在方式 2.2 中，第一应用和中间代理调用的是 createAnswer 方法，用于创建 SDP answer。此时，第一请求具体可以包括中间代理发送的 RTCPeerConnection 命令、createDataChannel 命令或 createAnswer 命令中的至少一个命令。

一种可能的实现方式，WebRTC 用户代理接收到第一请求后，可以确定已创建的连接能否承载第一数据通道。

例如，对于上述方式 1.1 和方式 2.1，WebRTC 用户代理可以在已创建的连接中查找能承载第一数据通道的连接，如果查到第一连接能承载第一数据通道，则 WebRTC 用户代理确定第一连接能承载第一数据通道。如果未查到能承载第一数据通道的连接，则 WebRTC 用户代理确定已创建的连接均不能承载第一数据通道。

例如，对于上述方式 1.2 和方式 2.2，若第一请求包括第一连接的信息，如第一应用在调用 createDataChannel 方法时携带第一连接的信息，则 WebRTC 用户代理确定第一连接能承载第一数据通道。若第一请求不包括第一数据通道对应的连接的信息，如第一应用在调用 createDataChannel 方法时不携带第一数据通道对应的连接的信息，则 WebRTC 用户代理确定已创建的连接均不能承载第一数据通道。

例如，对于上述方式 1.3 和方式 2.3，若 WebRTC 用户代理检测到中间代理没有调用 RTCPeerConnection API，而是直接调用第一连接对应的 createDataChannel 方法，WebRTC 用户代理确定第一连接能承载第一数据通道。若 WebRTC 用户代理检测到中间代理调用 RTCPeerConnection API，WebRTC 用户代理确定已创建的连接均不能承载第一数据通道。

可以理解的，若存在已创建的连接，如上述第一连接，能承载第一数据通道，则通信实体可以指示用第一连接承载第一数据通道，例如，通信实体可以执行下述 S502A。若不存在已创建的连接能承载第一数据通道，则通信实体可以指示用新的连接承载第一数据通道并创建新的连接，例如，通信实体可以执行下述 S502B。

S502A：通信实体发送第一请求的响应。

一种可能的实现方式，WebRTC 用户代理发送第一请求的响应。其中，第一请求的响应包括第一媒体描述信息，第一媒体描述信息包括第一指示信息，第一指示信息指示第一连接。例如，第一指示信息包括 IP 地址信息、UDP 端口信息以及 SCTP 端口信息。可以理解的，第一指示信息指示的是与第一连接关联的 IP 地址信息、UDP 端口信息以及 SCTP 端口信息。

下面以上述方式 1.1~方式 2.3 为例介绍第一媒体描述信息包括的内容。

示例性的，对于上述方式 1.1、方式 1.2、方式 2.1 和方式 2.2，WebRTC 用户代理向第一应用发送第一媒体描述信息。第一媒体描述信息包括的内容可以如表 6 所示。该第一媒体描述信息中的 IP 地址为 IN IP4 192.0.2.1（也是第一连接关联的 IP 地址），UDP 端口信息为 10001（也是第一连接关联的 UDP 端口信息），SCTP 端口信息为 57483（也是第一连接关联的 SCTP 端口信息）。表 6 所示的第一媒体描述信息还包括第一数据通道的流标识（streamID）为 7384，并且第一数据通道与 application#1（即第一应用）关联。

表 6

…
m=application **10001** UDP/DTLS/SCTP webrtc-datachannel
b=AS:1000
c=**IN IP4 192.0.2.1**
a=max-message-size:100000
a=sctp-port: **57483**

```
a=setup:actpass
a=fingerprint:SHA-1 \ 4A:AD:B9:B1:3F:82:18:3B:54:02:12:DF:3E:5D:49:6B:19:E5:7C:AB
a=tls-id:abc3de65cddef001be82
a=dcmap:7384 max-retr=5;label="low latency"
a=3gpp-req-app:"application#1";7384-UE
…
```

示例性的，对于上述方式1.3和方式2.3，WebRTC用户代理向中间代理发送第一媒体描述信息。该第一媒体描述信息包括的内容可以如表7所示。该第一媒体描述信息中的IP地址为IN IP4 192.0.2.1（也是第一连接关联的IP地址），UDP端口信息为10001（也是第一连接关联的UDP端口信息），SCTP端口信息为57483（也是第一连接关联的SCTP端口信息）。表7所示的第一媒体描述信息还包括第一数据通道的流标识为7384，并且第一数据通道与application#1（即第一应用）关联。可以理解的，第一连接是在S501之前创建的连接，所以在S501之前，第一连接就有对应的数据通道，如第三数据通道，第三数据通道用于传输第三应用的数据。因此，表7所示的第一媒体描述信息还包括第三数据通道的流标识为7216，并且第三数据通道与application#3（即第三应用）关联。也就是说，第三数据通道和第一数据通道共用第一连接，或者共用一个m行（即表7所示的媒体描述信息）。中间代理接收到第一媒体描述信息后，可以向第一应用发送第七媒体描述信息。例如，中间代理删除第一媒体描述中除第一应用之外的应用的信息，得到如表8所示的第七媒体描述信息。

表7

```
…
m=application 10001 UDP/DTLS/SCTP webrtc-datachannel
c=IN IP4 192.0.2.1
b=AS:1000
a=sctp-port: 57483
a=setup: actpass
a=fingerprint:SHA-1 BC:8A:99:A0:E3:28:CA:B3:09:20:1B:FD:21:D5:AC:B6:F3:5E:45:AF
a=tls-id: cd3bea56dced0f35d224
a=dcmap:7216 max-retr=5;label="low loss"
a=dcmap:7384 max-retr=5;label="low loss"
a=3gpp-req-app:"application#3";7216-UE
a=3gpp-req-app:"application#1";7384-UE
…
```

表8

```
…
m=application 10001 UDP/DTLS/SCTP webrtc-datachannel
c=IN IP4 192.0.2.1
b=AS:1000
a=sctp-port: 57483
a=setup:actpass
a=fingerprint: SHA-1 BC:8A:99:A0:E3:28:CA:B3:09:20:1B:FD:21:D5:AC:B6:F3:5E:45:AF
a=tls-id: cd3bea56dced0f35d224
a=dcmap:7384 max-retr=5;label="low loss"
a=3gpp-req-app:"application#1";7384-UE
…
```

可以理解的，在S502A之后，通信实体可以向网络侧发送第八媒体描述信息，网络侧可以将第八媒体描述信息发送给第二通信装置。第八媒体描述信息可以包括第一数据通道对应的连接的信息和第三数据通道对应的连接的信息。第八媒体描述信息还包括第一数据通道的标识信息（如流标识）和第三数据通道的标识信息（如流标识）。例如，第八媒体描述信息包括表7所示的内容。在表7中，第一数据通道对应的连接的信息和第三数据通道对应的连接的信息都对应从“m=application **10001** UDP/DTLS/SCTP

webrtc-datachannel”到“a=tls-id: cd3bea56dced0f35d224”的内容，换言之，流标识为 7216 的数据通道（即第三数据通道）与流标识为 7384 的数据通道（即第一数据通道）可以共用从“m=application **10001** UDP/DTLS/SCTP webrtc-datachannel”到“a=tls-id: cd3bea56dced0f35d224”的内容。所以，可以减少第八媒体描述信息包括的媒体行的个数，进而降低第八媒体描述信息的信令开销。

可以理解的，对于上述场景 1，第八媒体描述信息为 SDP 提议信息，可承载于 SIP 请求，如 INVITE、UPDATE 或 re-INVITE 中。对于上述场景 2，第八媒体描述信息为 SDP 应答信息，可承载于 SIP 响应，如 18x 响应或 200 OK 响应中。

下面介绍不存在已创建的连接能承载第一数据通道的情况。

S502B：通信实体发送第一请求的响应并创建第二连接。

一种可能的实现方式，WebRTC 用户代理发送第一请求的响应。其中，第一请求的响应包括第二媒体描述信息。第二媒体描述信息包括第二指示信息。第二指示信息指示第二连接。例如，第二指示信息包括第二连接关联的 IP 地址信息、UDP 端口信息和 SCTP 信息。第二连接为第一数据通道对应的连接。

下面以上述方式 1.1~方式 2.3 为例介绍第二媒体描述信息包括的内容。

示例性的，对于上述方式 1.1、方式 1.2、方式 2.1 和方式 2.2，WebRTC 用户代理向第一应用发送第二媒体描述信息。该第二媒体描述信息包括的内容可以如表 9 所示。该第二媒体描述信息中的 IP 地址为 IN IP4 192.0.2.132（也是第二连接关联的 IP 地址），UDP 端口信息为 10011（也是第二连接关联的 UDP 端口信息），SCTP 端口信息为 57486（也是第二连接关联的 SCTP 端口信息）。表 9 所示的第二媒体描述信息还包括第一数据通道的流标识为 7388，并且第一数据通道与 application#1（即第一应用）关联。第一应用在收到第二媒体描述信息后，调用 API，如 setLocalDescription API 触发 WebRTC 用户代理解析第二媒体描述信息并创建第二连接。

表 9

```
…
m=application 10011 UDP/DTLS/SCTP webrtc-datachannel
b=AS:1000
c=IN IP4 192.0.2.132
a=max-message-size:100000
a=sctp-port: 57486
a=setup:actpass
a=fingerprint:SHA-1 \ 4A:AD:B9:B1:3F:82:18:3B:54:02:12:DF:3E:5D:49:6B:19:E5:7C:AB
a=tls-id:abc3de65cddef001be82
a=dcmap:7388 max-retr=5;label="low latency"
a=3gpp-req-app:”application#1”;7388-UE
…
```

示例性的，对于上述方式 1.3 和方式 2.3，WebRTC 用户代理向中间代理发送如表 9 所示的第二媒体描述信息。中间代理接收到第二媒体描述信息后，向第一应用发送第二媒体描述信息。第一应用接收到第二媒体描述信息后，可以调用 API 创建第二连接。例如，第一应用调用 API，如 setLocalDescription API，中间代理可以将第一应用的调用命令透传给 WebRTC 用户代理，触发 WebRTC 用户代理解析第二媒体描述信息并创建第二连接。

可以理解的，在 S502B 之后，通信实体可以向网络侧发送第九媒体描述信息，网络侧可以将第九媒体描述信息发送给第二通信装置。第九媒体描述信息可以包括第一数据通道对应的连接的信息。第九媒体描述信息还包括第一数据通道的标识信息（如流标识）。例如，第九媒体描述信息包括如表 9 所示的内容。

可以理解的，对于上述场景 1，第九媒体描述信息为 SDP 提议信息，可承载于 SIP 请求，如 INVITE、UPDATE 或 re-INVITE 中。对于上述场景 2，第九媒体描述信息为 SDP 应答信息，可承载于 SIP 响应，如 18x 响应或 200 OK 响应中。

基于图 5 所示的方法，通信实体在创建第一数据通道之前，可以在已创建的连接中确定是否有能承载第一数据通道的连接。如果有能承载第一数据通道的连接，第一数据通道可以复用该连接，所以可以降低资源开销。可以理解的，上述连接可以是 SCTP 耦联。SCTP 耦联与媒体资源、带宽资源和端口资源

（如 NAT 穿越和防火墙管理所需的端口资源）关联，所以复用 SCTP 耦联可以降低媒体资源开销、带宽资源开销以及端口资源开销。

可选的，在图 5 所示方法的一种可能的实现方式中，与第一应用不同的第二应用可以请求创建第二数据通道，通信实体可以根据已创建的连接能否承载第二数据通道来确定是否创建新连接，以降低资源开销。具体的，可以如图 6 所示，图 5 所示的方法还可以包括如下步骤：

S503：通信实体接收第二请求。

本申请中，第二请求用于请求创建第二数据通道。第二数据通道用于承载第二应用的数据。可以理解的，第二数据通道为 ADC，第二应用为数据通道应用，第二应用部署在第一通信装置上。可选的，第二数据通道为第一通信装置和第二通信装置之间的数据通道。第二数据通道与第一通信装置以及第二通信装置之间的 IMS 会话相关联。例如，第二数据通道为该 IMS 会话中的逻辑连接，用于在 IMS 会话存续期间传输第二应用的数据。IMS 会话、ADC 以及数据通道应用的介绍可以参考前文对应的描述，不再赘述。

本申请中，第二请求可以是由第一通信装置上的第二应用触发或发送的，相应地，第一通信装置上的 WebRTC 用户代理接收该第一请求，此时，通信实体可以为第一通信装置上的 WebRTC 用户代理，也可以是第一通信装置上的其它功能模块，不予限定。假设第一通信装置采用图 3D 所示的终端架构，那么第二请求可以是由第二应用向数据通道运行环境中的 WebRTC 用户代理发送的。假设第一通信装置采用图 3E 所示的终端架构，那么第二请求可以是由第二应用触发中间代理向 WebRTC 用户代理发送的。

下面分别以第一通信装置为 SDP 提议方或 SDP 应答方为例，介绍通信实体接收第二请求的过程。

场景 3：第一通信装置为 SDP 提议方。

可以理解的，第二应用可以调用 API 触发第二请求。下面以下述方式 3.1~方式 3.3 为例进行介绍。

方式 3.1：第二应用调用 RTCPeerConnection API、createDataChannel 方法以及 createOffer 方法向 WebRTC 用户代理请求创建第二数据通道。此时，第二请求具体可以包括 RTCPeerConnection 命令、createDataChannel 命令或 createOffer 命令中的至少一个命令。

方式 3.2：第二应用调用 RTCPeerConnection API、createDataChannel 方法以及 createOffer 方法向 WebRTC 用户代理请求创建第二数据通道，并指示是否存在已创建的连接能承载第二数据通道。此时，第二请求具体可以包括 createDataChannel 命令或 createOffer 命令中的至少一个命令。

方式 3.3：第二应用通过中间代理向 WebRTC 用户代理请求创建第二数据通道。此时，第二请求具体可以包括中间代理发送的 RTCPeerConnection 命令、createDataChannel 命令或 createOffer 命令中的至少一个命令。

可以理解的，上述方式 3.1~方式 3.3 的过程与场景 1 中的方式 1.1~方式 1.3 的过程类似，可以参考方式 1.1~方式 1.3 中对应的描述。

可以理解的，场景 3 可以与上述场景 1 结合，表示在创建第一数据通道和第二数据通道时，第一通信装置都是发送 SDP offer 的一方。场景 3 也可以与上述场景 2 结合，表示在创建第一数据通道时，第一通信装置是发送 SDP answer 的一方，在创建第二数据通道时，第一通信装置是发送 SDP offer 的一方。此外，当场景 3 和场景 1 结合时，上述方式 3.1~方式 3.3 可以分别与方式 1.1~方式 1.3 结合，当场景 3 和场景 2 结合时，上述方式 3.1~方式 3.3 可以分别与方式 2.1~方式 2.3 结合，不予限制。

场景 4：第一通信装置为 SDP 应答方。

可以理解的，通信实体接收到来自 SDP 提议方的 SDP 提议信息后，可以将 SDP 提议信息中与第二应用有关的 SDP 信息发送给第二应用。第二应用接收到该 SDP 信息后，可以调用 API 触发第二请求。下面以下述方式 4.1~方式 4.3 为例进行介绍。

方式 4.1：第二应用调用 RTCPeerConnection API、createDataChannel 方法以及 createAnswer 方法向 WebRTC 用户代理请求创建第二数据通道。此时，第二请求具体可以包括 RTCPeerConnection 命令、createDataChannel 命令或 createAnswer 命令中的至少一个命令。

方式 4.2：第二应用调用 RTCPeerConnection API、createDataChannel 方法以及 createAnswer 方法向 WebRTC 用户代理请求创建第二数据通道，并指示是否存在已创建的连接能承载第二数据通道。此时，第二请求具体可以包括 createDataChannel 命令或 createAnswer 命令中的至少一个命令。

方式 4.3：第二应用通过中间代理向 WebRTC 用户代理请求创建第二数据通道。此时，第二请求具体可以包括中间代理发送的 RTCPeerConnection 命令、createDataChannel 命令或 createAnswer 命令中的至少

一个命令。

可以理解的，上述方式 4.1~方式 4.3 的过程与场景 1 中的方式 2.1~方式 2.3 的过程类似，可以参考方式 2.1~方式 2.3 中对应的描述。

可以理解的，场景 4 可以与上述场景 1 结合，表示在创建第一数据通道时，第一通信装置是发送 SDP offer 的一方，在创建第二数据通道时，第一通信装置是发送 SDP answer 的一方。场景 4 也可以与上述场景 2 结合，表示在创建第一数据通道和第二数据通道时，第一通信装置都是发送 SDP answer 的一方。此外，当场景 4 和场景 1 结合时，上述方式 4.1~方式 4.3 可以分别与方式 1.1~方式 1.3 结合，当场景 4 和场景 2 结合时，上述方式 4.1~方式 4.3 可以分别与方式 2.1~方式 2.3 结合，不予限制。

可以理解的，WebRTC 用户代理接收到第二请求后，可以确定已创建的连接能否承载第二数据通道。这一过程，与 S501 中 WebRTC 用户代理接收到第一请求后，确定已创建的连接能否承载第一数据通道的过程类似，可以参考 S501 中对应的描述，不再赘述。

可以理解的，针对 S502A 和 S502B，通信实体可以有不同的操作。具体来说，对于 S502A，如果存在已创建的连接，如上述第一连接，能承载第二数据通道，则通信实体可以指示用第一连接承载第二数据通道，例如，通信实体可以执行下述 S504A，如果不存在已创建的连接能承载第二数据通道，则通信实体可以指示用新的连接承载第二数据通道并创建新的连接，例如，通信实体可以执行下述 S504C。对于 S502B，如果存在已创建的连接，如上述第二连接，能承载第二数据通道，则通信实体可以指示用第二连接承载第二数据通道，例如，通信实体可以执行下述 S504B，如果不存在已创建的连接能承载第二数据通道，则通信实体可以指示用新的连接承载第二数据通道并创建新的连接，例如，通信实体可以执行下述 S504C。下面进行具体阐述。

首先介绍第一连接能承载第二数据通道的情况。

对于 S502A，如果第一连接能承载第二数据通道，通信实体可以执行 S504A。也就是说，在本申请提供的通信方法中，通信实体可以执行 S501、S502A，S503 以及 S504A。可以理解的，第一连接能承载第二数据通道，包括：第二数据通道的服务质量要求与第一连接的服务质量要求相同，或者第二数据通道的服务质量要求低于第一连接的服务质量要求。可选的，第一连接能承载第二数据通道，还包括：第一连接对应的端点与第二数据通道对应的端点相同。下面介绍 S504A 的具体过程。

S504A：通信实体发送第二请求的响应。其中，第二请求的响应包括第四媒体描述信息。

一种可能的实现方式，WebRTC 用户代理发送第二请求的响应。上述第四媒体描述信息包括第一指示信息。第一指示信息指示第一连接。第一连接是已创建的连接中第二数据通道对应的连接。

示例性的，对于上述方式 3.1、方式 3.2、方式 4.1 和方式 4.2，WebRTC 用户代理向第二应用发送第四媒体描述信息。该第四媒体描述信息包括的内容可以如表 10 所示。该第四媒体描述信息中的 IP 地址为 IN IP4 192.0.2.1（也是第一连接关联的 IP 地址），UDP 端口信息为 10001（也是第一连接关联的 UDP 端口信息），SCTP 端口信息为 57483（也是第一连接关联的 SCTP 端口信息）。表 10 所示的第四媒体描述信息还包括第二数据通道的流标识为 7380，并且第二数据通道与 application#2（即第二应用）关联。

表 10

```
…
m=application 10001 UDP/DTLS/SCTP webrtc-datachannel
b=AS:1000
c=IN IP4 192.0.2.1
a=max-message-size:100000
a=sctp-port: 57483
a=setup:actpass
a=fingerprint:SHA-1 \ 4A:AD:B9:B1:3F:82:18:3B:54:02:12:DF:3E:5D:49:6B:19:E5:7C:AB
a=tls-id:abc3de65cddef001be82
a=dcmap:7380 max-retr=5;label="low latency"
a=3gpp-req-app:"application#2";7380-UE
…
```

示例性的，对于上述方式 3.3 和方式 4.3，WebRTC 用户代理向中间代理发送第四媒体描述信息。该第四媒体描述信息包括的内容可以如表 11 所示。该第四媒体描述信息中的 IP 地址为 IN IP4 192.0.2.1（也是

第一连接关联的 IP 地址），UDP 端口信息为 10001（也是第一连接关联的 UDP 端口信息），SCTP 端口信息为 57483（也是第一连接关联的 SCTP 端口信息）。表 11 所示的第四媒体描述信息还包括第二数据通道的流标识为 7380，并且第二数据通道与 application#2（即第二应用）关联。表 11 所示的第四媒体描述信息还包括第一数据通道的流标识 7384 以及第三数据通道的流标识为 7216，并且第一数据通道与 application#1（即第一应用）关联，第三数据通道与 application#3（即第三应用）关联。也就是说，第一数据通道、第二数据通道以及第三数据通道共用第一连接，或者共用一个 m 行（即表 11 所示的媒体描述信息）。中间代理接收到第四媒体描述信息后，可以向第二应用发送第十媒体描述信息。例如，中间代理删除第四媒体描述中除第二应用之外的应用的信息，得到如表 12 所示的第十媒体描述信息。

表 11

```
…
m=application 10001 UDP/DTLS/SCTP webrtc-datachannel
c=IN IP4 192.0.2.1
b=AS:1000
a=sctp-port: 57483
a=setup: actpass
a=fingerprint:SHA-1 BC:8A:99:A0:E3:28:CA:B3:09:20:1B:FD:21:D5:AC:B6:F3:5E:45:AF
a=tls-id: cd3bea56dced0f35d224
a=dcmap:7216 max-retr=5;label="low loss"
a=dcmap:7384 max-retr=5;label="low loss"
a=dcmap:7380 max-retr=5;label="low loss"
a=3gpp-req-app:"application#3";7216-UE
a=3gpp-req-app:"application#1";7384-UE
a=3gpp-req-app:"application#2";7380-UE
…
```

表 12

```
…
m=application 10001 UDP/DTLS/SCTP webrtc-datachannel
c=IN IP4 192.0.2.1
b=AS:1000
a=sctp-port: 57483
a=setup:actpass
a=fingerprint: SHA-1 BC:8A:99:A0:E3:28:CA:B3:09:20:1B:FD:21:D5:AC:B6:F3:5E:45:AF
a=tls-id: cd3bea56dced0f35d224
a=dcmap:7380 max-retr=5;label="low loss"
a=3gpp-req-app:"application#2";7380-UE
…
```

可以理解的，在 S504A 之后，通信实体可以向网络侧发送第十一媒体描述信息，网络侧可以将第十一媒体描述信息发送给远端端点，如第二通信装置。第十一媒体描述信息可以包括第一数据通道对应的连接的信息、第二数据通道对应的连接的信息和第三数据通道对应的连接的信息。第十一媒体描述信息还包括第一数据通道的标识信息（如流标识）、第二数据通道的标识信息（如流标识）和第三数据通道的标识信息（如流标识）。例如，第十一媒体描述信息包括表 11 所示的内容。在表 11 中，第一数据通道对应的连接的信息、第二数据通道对应的连接的信息和第三数据通道对应的连接的信息都对应从“m=application 10001 UDP/DTLS/SCTP webrtc-datachannel”到“a=tls-id: cd3bea56dced0f35d224”的内容。所以，可以减少第十一媒体描述信息包括的媒体行的个数，进而降低第十一媒体描述信息的信令开销。

可以理解的，对于上述场景 3，第十一媒体描述信息为 SDP 提议信息，可以承载于 SIP 请求，如 INVITE、UPDATE 或 re-INVITE 中。对于上述场景 4，第十一媒体描述信息为 SDP 应答信息，可以承载于 SIP 响应，如 18x 响应或 200 OK 响应中。

下面介绍第二连接能承载第二数据通道的情况。

对于S502B，如果第二连接能承载第二数据通道，通信实体可以执行S504B。也就是说，在本申请提供的通信方法中，通信实体可以执行 S501、S502B，S503 以及 S504B。可以理解的，第二连接能承载第二数据通道，包括：第二数据通道的服务质量要求与第一数据通道的服务质量要求（或第二连接的服务质量要求）相同；或者，第二数据通道的服务质量要求低于第一数据通道的服务质量要求（或第二连接的服务质量要求）。可选的，第二连接能承载第二数据通道，还包括：第二数据通道对应的端点与第一数据通道对应的端点（或第二连接对应的端点）相同，例如，第二数据通道对应的端点也是通信实体和第二通信装置。下面介绍S504B的具体过程。

S504B：通信实体发送第二请求的响应。其中，第二请求的响应包括第五媒体描述信息。

一种可能的实现方式，WebRTC用户代理发送第二请求的响应。上述第五媒体描述信息包括第二指示信息。第二指示信息指示第二连接。第二连接是已创建的连接中第二数据通道对应的连接。

示例性的，对于上述方式3.1、方式3.2、方式4.1和方式4.2，WebRTC用户代理向第二应用发送第五媒体描述信息。该第五媒体描述信息包括的内容可以如表13所示。该第五媒体描述信息中的IP地址为IN IP4 192.0.2.132（也是第二连接关联的IP地址），UDP端口信息为10011（也是第二连接关联的UDP端口信息），SCTP端口信息为57486（也是第二连接关联的SCTP端口信息）。表13所示的第五媒体描述信息还包括第二数据通道的流标识为7485，并且第二数据通道与application#2（即第二应用）关联。

表13

```
…
m=application 10011 UDP/DTLS/SCTP webrtc-datachannel
b=AS:1000
c=IN IP4 192.0.2.132
a=max-message-size:100000
a=sctp-port: 57486
a=setup:actpass
a=fingerprint:SHA-1 \ 4A:AD:B9:B1:3F:82:18:3B:54:02:12:DF:3E:5D:49:6B:19:E5:7C:AB
a=tls-id:abc3de65cddef001be82
a=dcmap:7485 max-retr=5;label="low latency"
a=3gpp-req-app:”application#2”;7485-UE
…
```

示例性的，对于上述方式3.3和方式4.3，WebRTC用户代理向中间代理发送第五媒体描述信息。第五媒体描述信息包括的内容可以如表14所示。第五媒体描述信息中的IP地址为IN IP4 192.0.2.132（也是第二连接关联的IP地址），UDP端口信息为10011（也是第二连接关联的UDP端口信息），SCTP端口信息为57486（也是第二连接关联的SCTP端口信息）。表14所示的第五媒体描述信息还包括第二数据通道的流标识为7485，并且第二数据通道与application#2（即第二应用）关联。表14所示的第五媒体描述信息还包括第一数据通道的流标识7388，并且第一数据通道与application#1（即第一应用）关联。也就是说，第一数据通道和第二数据通道共用第二连接，或者共用一个m行（即表14所示的媒体描述信息）。中间代理接收到第五媒体描述信息后，可以向第二应用发送第十二媒体描述信息。例如，中间代理删除第五媒体描述中除第二应用之外的应用的信息，得到如上述表13所示的第十二媒体描述信息。

表14

```
…
m=application 10011 UDP/DTLS/SCTP webrtc-datachannel
b=AS:1000
c=IN IP4 192.0.2.132
a=max-message-size:100000
a=sctp-port: 57486
a=setup:actpass
a=fingerprint:SHA-1 \ 4A:AD:B9:B1:3F:82:18:3B:54:02:12:DF:3E:5D:49:6B:19:E5:7C:AB
a=tls-id:abc3de65cddef001be82
a=dcmap:7388 max-retr=5;label="low latency"
```

a=dcmap:**7485** max-retr=5;label="low latency"
a=3gpp-req-app:"**application#1**";7388-UE
a=3gpp-req-app:"**application#2**";7485-UE
…

可以理解的，在 S504B 之后，通信实体可以向网络侧发送第六媒体描述信息，网络侧可以将第六媒体描述信息发送给远端端点，如第二通信装置。第六媒体描述信息可以包括第一数据通道对应的连接的信息和第二数据通道对应的连接的信息。第六媒体描述信息还包括第一数据通道的标识信息（如流标识）和第二数据通道的标识信息（如流标识）。例如，第六媒体描述信息包括表 14 所示的内容。在表 14 中，第一数据通道对应的连接的信息和第二数据通道对应的连接的信息都对应从"m=application **10011** UDP/DTLS/SCTP webrtc-datachannel"到"a=tls-id:abc3de65cddef001be82"的内容。所以，可以减少第六媒体描述信息包括的媒体行的个数，进而降低第六媒体描述信息的信令开销。

可以理解的，对于上述场景3，第六媒体描述信息为SDP提议信息，可承载于SIP请求，如INVITE、UPDATE或re-INVITE中。对于上述场景4，第六媒体描述信息为SDP应答信息，可承载于SIP响应，如18x响应或200 OK响应中。

下面介绍已创建的连接均不能承载第二数据通道的情况。

对于 S502A 和 S502B，如果已创建的连接不承载第二数据通道，通信实体可以执行 S504C。也就是说，在本申请提供的通信方法中，通信实体可以执行 S501、S502A，S503 以及 S504C；或者通信实体可以执行 S501、S502B，S503 以及 S504C。可以理解的，第二连接（或第一连接）不能承载第二数据通道，包括：第二数据通道的服务质量要求高于第二连接（或第一连接）的服务质量要求。可选的，第二连接（或第一连接）不能承载第二数据通道，还包括：第二数据通道对应的端点与第二连接（或第一连接）对应的端点不完全相同。下面介绍 S504C 的具体过程。

S504C：通信实体发送第二请求的响应并创建第三连接。其中，第二请求的响应包括第三媒体描述信息。

一种可能的实现方式，WebRTC 用户代理发送第二请求的响应。上述第三媒体描述信息包括第二数据通道对应的第三指示信息。第三指示信息指示第三连接。例如，第三指示信息包括第三连接关联的 IP 地址信息、UDP 端口信息和 SCTP 信息。第三连接为第二数据通道对应的连接。

示例性的，对于上述方式 3.1、方式 3.2、方式 4.1 和方式 4.2，WebRTC 用户代理向第二应用发送第三媒体描述信息。该第三媒体描述信息包括的内容可以如表 15 所示。该第三媒体描述信息指示的 IP 地址为 IN IP4 192.0.2.156（也是第三连接关联的 IP 地址），指示的 UDP 端口信息为 10010（也是第三连接关联的 UDP 端口信息），指示的 SCTP 端口信息为 57576（也是第三连接关联的 SCTP 端口信息）。表 15 所示的第三媒体描述信息还指示第二数据通道的流标识为 7596，并且第二数据通道与 application#2（即第二应用）关联。第二应用收到第三媒体描述信息后，调用 API，如 setLocalDescription API 触发 WebRTC 用户代理解析第三媒体描述信息并创建第三连接。

表 15

…
m=application **10010** UDP/DTLS/SCTP webrtc-datachannel
b=AS:1000
c=**IN IP4 192.0.2.156**
a=max-message-size:100000
a=sctp-port: **57576**
a=setup:actpass
a=fingerprint:SHA-1 \ 4A:AD:B9:B1:3F:82:18:3B:54:02:12:DF:3E:5D:49:6B:19:E5:7C:AB
a=tls-id:abc3de65cddef001be82
a=dcmap:**7596** max-retr=5;label="low latency"
a=3gpp-req-app:"**application#2**";7596-UE
…

示例性的，对于上述方式 3.3 和方式 4.3，WebRTC 用户代理向中间代理发送如表 15 所示的第三媒体描述信息。中间代理接收到第三媒体描述信息后，向第二应用发送第三媒体描述信息。第二应用接收到

第三媒体描述信息后，可以调用 API 创建第三连接。例如，第二应用调用 API，如 setLocalDescription API，中间代理可以将第二应用的调用命令透传给 WebRTC 用户代理，触发 WebRTC 用户代理解析第三媒体描述信息并创建第三连接。

可以理解的，在 S504C 之后，通信实体可以向网络侧发送第十三媒体描述信息，网络侧可以将第十三媒体描述信息发送给远端端点，如第二通信装置。第十三媒体描述信息可以包括第二数据通道对应的连接的信息。第十三媒体描述信息还包括第二数据通道的标识信息（如流标识）。例如，第十三媒体描述信息包括如表 15 所示的内容。

可以理解的，对于上述场景 3，第十三媒体描述信息为 SDP 提议信息，可以承载于 SIP 请求，如 INVITE、UPDATE 或 re-INVITE 中。对于上述场景 4，第十三媒体描述信息为 SDP 应答信息，可以承载于 SIP 响应，如 18x 响应或 200 OK 响应中。

可以理解的，图 5 或图 6 所示方法中的通信实体的动作可以由图 4 所示的通信装置 40 中的处理器 401 调用存储器 403 中存储的应用程序代码来执行，本申请对此不做任何限制。

以上是以通信实体的角度介绍第一应用和第二应用触发创建数据通道的过程。为了更好地理解本申请提供的方法，下面以第一终端和第二终端交互的角度介绍二者之间协商数据通道的过程。

首先，以第一终端为 SDP 提议方，第二终端为 SDP 应答方，第一终端采用方式 1.1 创建第一数据通道，采用方式3.1 创建第二数据通道，第二终端采用方式2.1 创建第一数据通道，采用方式4.1 创建第二数据通道为例进行阐述。上述第一终端和第二终端具备图 3D 所示的终端架构。

如图 7 所示，为本申请提供的又一种通信方法，该通信方法可以包括如下步骤：

S701：第一终端中的第一应用调用 RTCPeerConnection API，请求第一终端中的 WebRTC 用户代理创建 RTCPeerConnection 实例。

S702：第一终端中的第一应用调用 createDataChannel 方法请求在 RTCPeerConnection 实例（或对象）上创建数据通道。

S703：第一终端中的第一应用调用 createOffer 方法，请求第一终端中的 WebRTC 用户代理生成媒体描述返回给第一应用。

可以理解的，上述第一请求可以是 RTCPeerConnection 命令、createDataChannel 命令或 createOffer 命令中的至少一个命令。WebRTC 用户代理接收到第一请求后，可以确定已创建的连接能否承载第一数据通道。上述 S701~S703 的具体过程与上述 S101~S103 的过程类似，可以参考 S101~S103 中对应的描述。

S704：若不存在已创建的连接能承载第一数据通道，第一终端中的 WebRTC 用户代理向第一应用返回第二媒体描述信息，以指示第二连接。

示例性的，第二媒体描述信息包括的内容可以如表 9 所示。

S705：第一终端的第一应用调用 setLocalDescription API 触发 WebRTC 用户代理解析第二媒体描述信息并创建第二连接，例如，创建接收和解码媒体的本地资源、设置第一数据通道的参数等。

S706：第一终端的第一应用调用 RTCPeerSignallingService API 触发第一终端通过 IMS 网络向第二终端发送第一 SDP offer。相应的，第二终端接收第一 SDP offer。

其中，第一 SDP offer 可以包括第九媒体描述信息。

S707：第二终端的 WebRTC 用户代理通过 RTCPeerSignallingService API 将第九媒体描述信息中与第一应用有关的媒体描述信息发送给第二终端的第一应用。

示例性的，第二终端的 WebRTC 用户代理发送给第一应用的媒体描述信息可以如表 9 所示。

S708：第二终端的第一应用调用 RTCPeerConnection API，请求第二终端中的 WebRTC 用户代理创建 RTCPeerConnection 实例。

S709：第二终端的第一应用调用 setRemoteDescription API 应用接收到的媒体描述信息。

S710：第二终端中的第一应用调用 createDataChannel 方法，请求在 RTCPeerConnection 实例（或对象）上创建数据通道。

S711：第二终端的第一应用调用 createAnswer 方法，请求第二终端中的 WebRTC 用户代理生成媒体描述返回给第一应用。

可以理解的，上述 S708、S710~S711 用于第二终端中的第一应用请求创建第一数据通道，第二终端的 WebRTC 用户代理可以确定已创建的连接能否承载第一数据通道。

S712：若不存在已创建的连接能承载第一数据通道，第二终端中的 WebRTC 用户代理向第一应用返

回第十四媒体描述信息，以指示第二连接。

可以理解的，本申请中的任意一个连接（如第一连接、第二连接或第三连接等）是该连接的两个端点协商建立的。也就是说，每个端点都可以为该连接配置一组协议地址信息和协议端口信息。所以该连接可以关联两组协议地址信息和协议端口信息。以这里的第二连接为例，第二连接的两个端点为第一终端和第二终端，第一终端在上述 S704 为第二连接配置了一组协议地址信息和协议端口信息，用于第一终端通过第二连接接收/发送数据，第二终端在 S712 为第二连接配置了一组协议地址信息和协议端口信息，用于第二终端通过第二连接接收/发送数据。可以理解的，这两组协议地址可以不同，这两组协议端口号也可以不同。因此，第十四媒体描述信息包括的内容与第二媒体描述信息包括的内容类似，其中参数的具体数值可以不同，例如 IP 地址、UDP 端口号和 SCTP 端口号等不同。

S713：第二终端的第一应用调用 setLocalDescription API 触发 WebRTC 用户代理解析第十四媒体描述信息并创建第二连接，例如，创建接收和解码媒体的本地资源、设置第一数据通道的参数等。

S714：第二终端的第一应用调用 RTCPeerSignallingService API 触发第二终端通过 IMS 网络向第一终端发送第一 SDP answer。相应的，第一终端接收第一 SDP answer。

其中，第一 SDP answer 可以包括第十五媒体描述信息。第十五媒体描述信息包括第二连接的信息和第二终端创建的第一数据通道的标识信息。这里的第二连接的信息可以包括第二终端为第二连接配置的 IP 协议地址信息、UDP 端口信息和 SCTP 端口信息。

S715：第一终端的 WebRTC 用户代理通过 RTCPeerSignallingService API 将第十五媒体描述信息中与第一应用有关的媒体描述信息发送给第一终端的第一应用。

S716：第一终端的第一应用调用 setRemoteDescription API 设置第二终端为第一数据通道配置的资源。

可以理解的，在 S716 之后，第一终端和第二终端的 WebRTC 用户代理可以创建第二连接以及相应的流。第一终端和第二终端的第一应用可以通过第二连接向应用层发送数据或者接收来自应用层的数据。

S717：第一终端中的第二应用调用 RTCPeerConnection API，请求第一终端中的 WebRTC 用户代理创建 RTCPeerConnection 实例。

S718：第一终端中的第二应用调用 createDataChannel 方法请求在 RTCPeerConnection 实例（或对象）上创建数据通道。

S719：第一终端中的第二应用调用 createOffer 方法，请求第一终端中的 WebRTC 用户代理生成媒体描述返回给第二应用。

可以理解的，上述 S717~S719 用于第二应用请求创建第二数据通道，WebRTC 用户代理可以确定已创建的连接能否承载第二数据通道。

S720：若第二连接能承载第二数据通道，第一终端中的 WebRTC 用户代理向第二应用返回第五媒体描述信息，指示第二连接。

示例性的，第五媒体描述信息包括的内容可以如表 13 所示。

S721：第一终端的第二应用调用 setLocalDescription API 触发 WebRTC 用户代理应用第五媒体描述信息。

S722：第一终端的第二应用调用 RTCPeerSignallingService API 触发第一终端通过 IMS 网络向第二终端发送第二 SDP offer。相应的，第二终端接收第二 SDP offer。

其中，第二 SDP offer 可以包括第六媒体描述信息。

S723：第二终端的 WebRTC 用户代理通过 RTCPeerSignallingService API 将第六媒体描述信息中与第二应用有关的媒体描述信息发送给第二终端的第二应用。

示例性的，第二终端的 WebRTC 用户代理发送给第二应用的媒体描述信息可以如表 13 所示。

S724：第二终端的第二应用调用 RTCPeerConnection API，请求第二终端中的 WebRTC 用户代理创建 RTCPeerConnection 实例。

S725：第二终端的第二应用调用 setRemoteDescription API 应用接收到的媒体描述信息。

S726：第二终端中的第二应用调用 createDataChannel 方法，请求在 RTCPeerConnection 实例（或对象）上创建数据通道。

S727：第二终端的第二应用调用 createAnswer 方法，请求第二终端中的 WebRTC 用户代理生成媒体描述返回给第二应用。

可以理解的，上述 S724、S726~S727 用于第二应用请求创建第二数据通道，第二终端的 WebRTC 用

户代理可以确定已创建的连接能否承载第二数据通道。

S728：若第二连接能承载第二数据通道，第二终端中的 WebRTC 用户代理向第二应用返回第十六媒体描述信息，以指示第二连接。

可以理解的，第十六媒体描述信息包括的内容与第五媒体描述信息包括的内容类似，其中参数的具体数值可以不同，例如 IP 地址、UDP 端口号和 SCTP 端口号等不同。

S729：第二终端的第二应用调用 setLocalDescription API 触发 WebRTC 用户代理应用第十六媒体描述信息。

S730：第二终端的第二应用调用 RTCPeerSignallingService API 触发第二终端通过 IMS 网络向第一终端发送第二 SDP answer。相应的，第一终端接收第二 SDP answer。

其中，第二 SDP answer 可以包括第十七媒体描述信息。第十七媒体描述信息包括第二连接的信息、第二终端创建的第一数据通道的标识信息以及第二终端创建的第二数据通道的标识信息。这里的第二连接的信息可以包括第二终端为第二连接配置的 IP 协议地址信息、UDP 端口信息和 SCTP 端口信息。

S731：第一终端的 WebRTC 用户代理通过 RTCPeerSignallingService API 将第十七媒体描述信息中与第二应用有关的媒体描述信息发送给第一终端的第二应用。

S732：第一终端的第二应用调用 setRemoteDescription API 设置第二终端为第二数据通道配置的资源。

S733：第一终端的 WebRTC 用户代理更新第二连接，在第二连接中增加第二数据通道对应的流，第二终端的 WebRTC 用户代理更新第二连接，在第二连接中增加第二数据通道对应的流。

可以理解的，在 S733 之后，第一终端中的第一应用和第二应用都可以通过第二连接向应用层发送数据或者接收来自应用层的数据。类似的，第二终端中的第一应用和第二应用都可以通过第二连接向应用层发送数据或者接收来自应用层的数据。

可以理解的，上述步骤中的第一终端或者第二终端的动作可以由图 4 所示的通信装置 40 中的处理器 401 调用存储器 403 中存储的应用程序代码来执行，本申请对此不做任何限制。

基于图 7 所示的方法，第一终端和第二终端的 WebRTC 用户代理在创建数据通道之前，可以在已创建的连接中确定是否有能承载该数据通道的连接，以降低资源开销。

下面以第一终端为 SDP 提议方，第二终端为 SDP 应答方，第一终端采用方式 1.2 创建第一数据通道，第二终端采用方式 2.2 创建第一数据通道为例进行阐述。上述第一终端和第二终端具备图 3D 所示的终端架构。

如图 8 所示，为本申请提供的又一种通信方法，该通信方法可以包括如下步骤：

S801：第一终端中的第一应用调用 RTCPeerConnection API，请求第一终端中的 WebRTC 用户代理创建 RTCPeerConnection 实例。

S802：第一终端中的 WebRTC 用户代理向第一应用发送已创建的连接的信息。

示例性的，第一终端中的 WebRTC 用户代理可以向第一应用提供查询接口，用于查询已创建的连接。当第一应用调用该查询接口时，WebRTC 用户代理向第一应用返回已创建的连接的信息。已创建的连接的信息用于指示已创建的每一个连接，如指示每个连接关联的 IP 地址、UDP 端口信息和 SCTP 端口信息。

可以理解的，第一应用也可以在 S801 之前调用接口查询已创建的连接，不予限制。

S803：第一终端中的第一应用确定已创建的连接能否承载第一数据通道。

S804：若已创建的连接中的第一连接能承载第一数据通道，第一终端中的第一应用调用 createDataChannel 方法，并在调用该方法时携带第一连接的信息。

S805：第一终端中的第一应用调用 createOffer 方法，请求第一终端中的 WebRTC 用户代理生成媒体描述返回给第一应用。

可以理解的，由于第一应用调用 createDataChannel 方法时携带第一连接的信息，所以 WebRTC 用户代理确定用第一连接承载第一数据通道。应理解，在具体应用中，第一应用也可以不在调用 createDataChannel 方法时携带第一连接的信息，而是在调用 createOffer 方法时携带第一连接的信息。

S806：第一终端中的 WebRTC 用户代理向第一应用返回第一媒体描述信息，以指示第一连接。

示例性的，第一媒体描述信息包括的内容可以如表 6 所示。

S807：第一终端的第一应用调用 setLocalDescription API 触发 WebRTC 用户代理应用第一媒体描述信息。

S808：第一终端的第一应用调用 RTCPeerSignallingService API 触发第一终端通过 IMS 网络向第二终

端发送第一 SDP offer。相应的，第二终端接收第一 SDP offer。

其中，第一 SDP offer 可以包括第八媒体描述信息。

S809：第二终端的 WebRTC 用户代理通过 RTCPeerSignallingService API 将第八媒体描述信息中与第一应用有关的媒体描述信息发送给第二终端的第一应用。

示例性的，第二终端的 WebRTC 用户代理发送给第一应用的媒体描述信息可以如表 6 所示。

S810：第二终端的第一应用调用 RTCPeerConnection API，请求第二终端中的 WebRTC 用户代理创建 RTCPeerConnection 实例。

S811：第二终端的第一应用调用 setRemoteDescription API 应用接收到的媒体描述信息。

S812：第二终端中的 WebRTC 用户代理向第一应用发送已创建的连接的信息。

示例性的，第二终端中的 WebRTC 用户代理可以向第一应用提供查询接口，用于查询已创建的连接。当第一应用调用该查询接口时，WebRTC 用户代理向第一应用返回已创建的连接的信息。已创建的连接的信息用于指示已创建的每一个连接，如指示每个连接关联的 IP 地址、UDP 端口信息和 SCTP 端口信息。

可以理解的，第一应用也可以在 S811 之前调用接口查询已创建的连接，不予限制。

S813：第二终端中的第一应用确定已创建的连接能否承载第一数据通道。

S814：若已创建的连接中的第一连接能承载第一数据通道，第二终端中的第一应用调用 createDataChannel 方法，并在调用该方法时携带第一连接的信息。

可以理解的，第一连接为第二终端创建的连接中第一数据通道对应的连接。

S815：第二终端的第一应用调用 createAnswer 方法，请求第二终端中的 WebRTC 用户代理生成媒体描述返回给第一应用。

可以理解的，由于第一应用调用 createDataChannel 方法时携带第一连接的信息，所以 WebRTC 用户代理确定用第一连接承载第一数据通道。应理解，在具体应用中，第一应用也可以不在调用 createDataChannel 方法时携带第一连接的信息，而是在调用 createAnswer 方法时携带第一连接的信息。

S816：第二终端中的 WebRTC 用户代理向第一应用返回第十八媒体描述信息，以指示第一连接。

可以理解的，第十八媒体描述信息包括的内容与第一媒体描述信息包括的内容类似，其中参数的具体数值可以不同，例如 IP 地址、UDP 端口号和 SCTP 端口号等不同。

S817：第二终端的第一应用调用 setLocalDescription API 触发 WebRTC 用户代理应用第十八媒体描述信息。

S818：第二终端的第一应用调用 RTCPeerSignallingService API 触发第二终端通过 IMS 网络向第一终端发送第一 SDP answer。相应的，第一终端接收第一 SDP answer。

其中，第一 SDP answer 可以包括第十九媒体描述信息。第十九媒体描述信息包括第一连接的信息和第二终端创建的第一数据通道的标识信息。这里的第一连接的信息可以包括第二终端为第一连接配置的 IP 协议地址信息、UDP 端口信息和 SCTP 端口信息。

S819：第一终端的 WebRTC 用户代理通过 RTCPeerSignallingService API 将第十九媒体描述信息中与第一应用有关的媒体描述信息发送给第一终端的第一应用。

S820：第一终端的第一应用调用 setRemoteDescription API 设置第二终端为第一数据通道配置的资源。

S821：第一终端的 WebRTC 用户代理更新第一连接，在第一连接中增加第一数据通道对应的流，第二终端的 WebRTC 用户代理更新第一连接，在第一连接中增加第一数据通道对应的流。

可以理解的，在 S821 之后，第一终端中的第一应用可以通过第一连接向应用层发送数据或者接收来自应用层的数据，第二终端中的第一应用可以通过第一连接向应用层发送数据或者接收来自应用层的数据。

可以理解的，上述步骤中的第一终端或者第二终端的动作可以由图 4 所示的通信装置 40 中的处理器 401 调用存储器 403 中存储的应用程序代码来执行，本申请对此不做任何限制。

基于图 8 所示的方法，第一终端和第二终端的 WebRTC 用户代理可以开放接口给第一应用，供第一应用查询已创建的连接，以便第一应用确定已创建的连接能否承载自己请求创建的数据通道，从而降低资源开销。图 8 是以第一应用确定已创建的连接中的第一连接能承载自己请求创建的数据通道为例进行阐述的。可以理解的，如果第一应用确定已创建的连接不能承载自己请求创建的数据通道，WebRTC 用户代理可以生成一个全新的“m 行”（如表 9 所示的第二媒体描述信息），并返回第一应用。

下面以第一终端为 SDP 提议方，第二终端为 SDP 应答方，第一终端采用方式 1.3 创建第一数据通道，

采用方式3.3创建第二数据通道，第二终端采用方式2.3创建第一数据通道，采用方式4.3创建第二数据通道为例进行阐述。上述第一终端和第二终端具备图3E所示的终端架构。

如图9所示，为本申请提供的又一种通信方法，该通信方法可以包括如下步骤：

S901：第一终端中的第一应用调用 RTCPeerConnection API，请求第一终端中的中间代理创建RTCPeerConnection实例。

S902：第一终端中的第一应用调用createDataChannel方法请求在RTCPeerConnection实例（或对象）上创建数据通道。

S903：第一终端中的第一应用调用createOffer方法，请求第一终端中的中间代理生成媒体描述返回给第一应用。

可以理解的，RTCPeerConnection API、createDataChannel方法和createOffer方法用于请求创建第一数据通道。上述S901~S903的具体过程与上述S101~S103的过程类似，可以参考S101~S103中对应的描述。

S904：第一终端的中间代理确定已创建的连接能否承载第一数据通道。

S905：若不存在已创建的连接能承载第一数据通道，第一终端中的中间代理调用 RTCPeerConnection API，请求第一终端的WebRTC用户代理创建RTCPeerConnection实例。

S906：第一终端的中间代理调用createDataChannel方法请求在RTCPeerConnection实例（或对象）上创建数据通道。

S907：第一终端的中间代理调用 createOffer 方法，请求第一终端中的 WebRTC 用户代理生成媒体描述返回给第一应用。

可以理解的，上述第一请求可以是RTCPeerConnection命令、createDataChannel命令或createOffer命令中的至少一个命令。上述 S905~S907 的具体过程与上述 S101~S103 的过程类似，可以参考 S101~S103中对应的描述。

S908：第一终端的 WebRTC 用户代理通过中间代理向第一应用发送第二媒体描述信息，以指示第二连接。

示例性的，第二媒体描述信息包括的内容可以如表9所示。

S909：第一终端的第一应用调用setLocalDescription API触发WebRTC用户代理解析第二媒体描述信息并创建第二连接，例如，创建接收和解码媒体的本地资源、设置第一数据通道的参数等。

可以理解的，第一终端的中间代理可以将第一应用的调用命令透传给WebRTC用户代理。

S910：第一终端的第一应用调用 RTCPeerSignallingService API 触发第一终端通过 IMS 网络向第二终端发送第一SDP offer。相应的，第二终端接收第一SDP offer。

其中，第一SDP offer可以包括第九媒体描述信息。

可以理解的，S910也可以是中间代理触发的，不予限制。

S911：第二终端的 WebRTC 用户代理通过 RTCPeerSignallingService API 将第九媒体描述信息中与第一应用有关的媒体描述信息发送给第二终端的第一应用。

示例性的，第二终端的WebRTC用户代理发送给第一应用的媒体描述信息可以如表9所示。

可以理解的，第二终端的中间代理可以将WebRTC用户代理发送的信息透传给第一应用。

S912：第二终端的第一应用调用 RTCPeerConnection API，请求第二终端的中间代理创建RTCPeerConnection实例。

S913：第二终端的第一应用调用setRemoteDescription API应用接收到的媒体描述信息。

可以理解的，第一应用的调用setRemoteDescription命令可以通过中间代理透传给WebRTC用户代理。

S914：第二终端中的第一应用调用 createDataChannel 方法，请求第二终端的中间代理在RTCPeerConnection实例（或对象）上创建数据通道。

S915：第二终端的第一应用调用createAnswer方法，请求第二终端的中间代理生成媒体描述返回给第一应用。

可以理解的，上述S912、S914~S915用于第二终端的第一应用请求创建第一数据通道。

S916：第二终端的中间代理可以确定已创建的连接能否承载第一数据通道。

S917：若不存在已创建的连接能承载第一数据通道，第二终端的中间代理调用 RTCPeerConnection API，请求第二终端中的WebRTC用户代理创建RTCPeerConnection实例。

S918：第二终端的中间代理调用createDataChannel方法，请求在RTCPeerConnection实例（或对象）

上创建数据通道。

S919：第二终端的中间代理调用 createAnswer 方法，请求第二终端中的 WebRTC 用户代理生成媒体描述返回给第一应用。

S920：第二终端中的 WebRTC 用户代理通过中间代理向第一应用返回第十四媒体描述信息，以指示第二连接。

可以理解的，第十四媒体描述信息包括的内容与第二媒体描述信息包括的内容类似，其中参数的具体数值可以不同，例如 IP 地址、UDP 端口号和 SCTP 端口号等不同。

S921：第二终端的第一应用调用 setLocalDescription API 触发 WebRTC 用户代理解析第十四媒体描述信息并创建第二连接，例如，创建接收和解码媒体的本地资源、设置第一数据通道的参数等。

可以理解的，中间代理可以将第一应用的 setLocalDescription 调用命令透传给 WebRTC 用户代理。

S922：第二终端的第一应用调用 RTCPeerSignallingService API 触发第二终端通过 IMS 网络向第一终端发送第一 SDP answer。相应的，第一终端接收第一 SDP answer。

其中，第一 SDP answer 可以包括第十五媒体描述信息。第十五媒体描述信息包括第二连接的信息和第二终端创建的第一数据通道的标识信息。这里的第二连接的信息可以包括第二终端为第二连接配置的 IP 协议地址信息、UDP 端口信息和 SCTP 端口信息。

可以理解的，S922 也可以是中间代理触发的，不予限制。

S923：第一终端的 WebRTC 用户代理通过 RTCPeerSignallingService API 将第十五媒体描述信息中与第一应用有关的媒体描述信息发送给第一终端的第一应用。

可以理解的，第一终端的中间代理可以将 WebRTC 用户代理发送的信息透传给第一应用。

S924：第一终端的第一应用调用 setRemoteDescription API 设置第二终端为第一数据通道配置的资源。

可以理解的，第一应用的 setRemoteDescription 命令可以通过中间代理透传给 WebRTC 用户代理。

可以理解的，在 S924 之后，第一终端和第二终端的 WebRTC 用户代理可以创建第二连接以及相应的流。第一终端的中间代理可以通过第二连接接收数据，并将数据分发给第一应用，第二终端的中间代理可以通过第二连接接收数据，并将数据分发给第一应用。

S925：第一终端中的第二应用调用 RTCPeerConnection API，请求第一终端的中间代理创建 RTCPeerConnection 实例。

S926：第一终端中的第二应用调用 createDataChannel 方法请求在 RTCPeerConnection 实例（或对象）上创建数据通道。

S927：第一终端中的第二应用调用 createOffer 方法，请求第一终端的中间代理生成媒体描述返回给第二应用。

可以理解的，RTCPeerConnection API、createDataChannel 方法和 createOffer 方法用于请求创建第二数据通道。上述 S925~S927 的具体过程与上述 S101~S103 的过程类似，可以参考 S101~S103 中对应的描述。

S928：第一终端的中间代理确定已创建的连接能否承载第二数据通道。

S929：若第二连接能承载第二数据通道，第一终端的中间代理调用 createDataChannel 方法请求 WebRTC 用户代理在第二连接上创建数据通道。

S930：第一终端的中间代理调用 createOffer 方法，请求第一终端中的 WebRTC 用户代理生成媒体描述返回给第二应用。

S931：第一终端中的 WebRTC 用户代理向中间代理返回第五媒体描述信息。

示例性的，第五媒体描述信息包括的内容可以如表 14 所示。

S932：第一终端中的中间代理向第二应用发送第十二媒体描述信息。例如，中间代理删除第五媒体描述中除第二应用之外的应用的信息，得到如表 13 所示的第十二媒体描述信息。

S933：第一终端的第二应用调用 setLocalDescription API 触发 WebRTC 用户代理应用第五媒体描述信息。

可以理解的，第一终端的中间代理可以将第二应用的 setLocalDescription 命令透传给 WebRTC 用户代理。

S934：第一终端的第二应用调用 RTCPeerSignallingService API 触发第一终端通过 IMS 网络向第二终端发送第二 SDP offer。相应的，第二终端接收第二 SDP offer。

其中，第二 SDP offer 可以包括第六媒体描述信息。

S935：第二终端的 WebRTC 用户代理通过 RTCPeerSignallingService API 将第六媒体描述信息中与第二应用有关的媒体描述信息发送给第二终端的第二应用。

示例性的，第二终端的 WebRTC 用户代理发送给第二应用的媒体描述信息可以如表 13 所示。

可以理解的，第二终端的中间代理可以将 WebRTC 用户代理发送的信息透传给第二应用。

S936：第二终端的第二应用调用 RTCPeerConnection API，请求第二终端中的中间代理创建 RTCPeerConnection 实例。

S937：第二终端的第二应用调用 setRemoteDescription API 应用接收到的媒体描述信息。

可以理解的，第二终端的中间代理可以将第二应用的 setRemoteDescription 命令透传给 WebRTC 用户代理。

S938：第二终端中的第二应用调用 createDataChannel 方法，请求中间代理在 RTCPeerConnection 实例（或对象）上创建数据通道。

S939：第二终端的第二应用调用 createAnswer 方法，请求第二终端中的中间代理生成媒体描述返回给第二应用。

可以理解的，上述 S936、S938~S939 用于第二应用请求创建第二数据通道。

S940：第二终端的中间代理可以确定已创建的连接能否承载第二数据通道。

S941：若第二连接能承载第二数据通道，第二终端的中间代理调用 createDataChannel 方法，请求 WebRTC 用户代理在第二连接上创建数据通道。

S942：第二终端的中间代理调用 createAnswer 方法，请求第二终端中的 WebRTC 用户代理生成媒体描述返回给第二应用。

S943：第二终端中的 WebRTC 用户代理向中间代理返回第二十媒体描述信息，以指示第二连接。

可以理解的，第二十媒体描述信息包括的内容与上述第五媒体描述信息包括的内容类似，其中参数的具体数值可以不同，例如 IP 地址、UDP 端口号和 SCTP 端口号等不同。

S944：第二终端中的中间代理向第二应用返回第二十一媒体描述信息。例如，中间代理删除第二十媒体描述中除第二应用之外的应用的信息，得到第二十一媒体描述信息。

S945：第二终端的第二应用调用 setLocalDescription API 触发 WebRTC 用户代理应用第二十媒体描述信息。

可以理解的，第二终端的中间代理可以将第二应用的 setLocalDescription 命令透传给 WebRTC 用户代理。

S946：第二终端的第二应用调用 RTCPeerSignallingService API 触发第二终端通过 IMS 网络向第一终端发送第二 SDP answer。相应的，第一终端接收第二 SDP answer。

其中，第二 SDP answer 可以包括第二十二媒体描述信息。第二十二媒体描述信息包括第二连接的信息、第二终端创建的第一数据通道的标识信息以及第二终端创建的第二数据通道的标识信息。这里的第二连接的信息可以包括第二终端为第二连接配置的 IP 协议地址信息、UDP 端口信息和 SCTP 端口信息。

S947：第一终端的 WebRTC 用户代理通过 RTCPeerSignallingService API 将第二十二媒体描述信息中与第二应用有关的媒体描述信息发送给第一终端的第二应用。

S948：第一终端的第二应用调用 setRemoteDescription API 设置第二终端为第二数据通道配置的资源。

可以理解的，第一应用的 setRemoteDescription 命令可以通过中间代理透传给 WebRTC 用户代理。

S949：第一终端的 WebRTC 用户代理更新第二连接，在第二连接中增加第二数据通道对应的流，第二终端的 WebRTC 用户代理更新第二连接，在第二连接中增加第二数据通道对应的流。

可以理解的，在 S949 之后，第一终端的中间代理可以通过第二连接接收数据，并将数据分发给第一应用或第二应用，第二终端的中间代理可以通过第二连接接收数据，并将数据分发给第一应用或第二应用。

可以理解的，上述步骤中的第一终端或者第二终端的动作可以由图 4 所示的通信装置 40 中的处理器 401 调用存储器 403 中存储的应用程序代码来执行，本申请对此不做任何限制。

基于图 9 所示的方法，在创建数据通道之前，第一终端和第二终端中的中间代理可以在已创建的连接中确定是否有能承载该数据通道的连接，以降低资源开销。也就是说，上述中间代理可以具备管理连接的功能。除了管理连接的功能之外，上述中间代理还具备分发数据的功能。例如，上述中间代理可以将接收到的数据分发给相应的应用。当应用需要发送数据时，可以通过中间代理统一发送。可以理解的，

针对图 9 所示的方法，不需要更新 WebRTC 用户代理的功能，所以便于将该方法部署在终端。

本申请上文中提到的各个实施例之间在方案不矛盾的情况下，均可以进行结合，不作限制。

上述主要从各个网元之间交互的角度对本申请提供的方案进行了介绍。相应的，本申请还提供了通信装置，该通信装置可以为上述方法实施例中的通信实体，或者包含上述通信实体的装置，或者为可用于通信实体的部件。可以理解的是，上述通信实体为了实现上述功能，其包含了执行各个功能相应的硬件结构和/或软件模块。本领域技术人员应该很容易意识到，结合本文中所公开的实施例描述的各示例的单元及算法操作，本申请能够以硬件或硬件和计算机软件的结合形式来实现。某个功能究竟以硬件还是计算机软件驱动硬件的方式来执行，取决于技术方案的特定应用和设计约束条件。专业技术人员可以对每个特定的应用来使用不同方法来实现所描述的功能，但是这种实现不应认为超出本申请的范围。

本申请可以根据上述方法示例对通信实体进行功能模块的划分，例如，可以对应各个功能划分各个功能模块，也可以将两个或两个以上的功能集成在一个处理模块中。上述集成的模块既可以采用硬件的形式实现，也可以采用软件功能模块的形式实现。可以理解的是，本申请中对模块的划分是示意性的，仅仅为一种逻辑功能划分，实际实现时可以有另外的划分方式。

比如，以采用集成的方式划分各个功能模块的情况下，图 10 示出了一种通信装置 100 的结构示意图。通信装置 100 包括处理模块 1001 和接口模块 1002。处理模块 1001，也可以称为处理单元用于执行除了收发操作之外的操作，例如可以是处理电路或者处理器等。接口模块 1002，也可以称为接口单元用于执行收发操作，例如可以是接口电路，收发机，收发器或者通信接口等。

在一些实施例中，该通信装置 100 还可以包括存储模块（图 10 中未示出），用于存储程序指令和数据。

示例性地，通信装置 100 用于实现通信实体的功能。通信装置 100 例如为图 5 所示的实施例或图 6 所示的实施例所述的通信实体。

其中，接口模块 1002，用于接收第一请求。其中，第一请求用于请求创建第一数据通道，第一数据通道为通信装置 100 和第二通信装置之间的数据通道，用于承载第一应用的数据；第一数据通道与通信装置 100 和第二通信装置之间的 IMS 会话相关联。例如，接口模块 1002 可以用于执行 S501。

处理模块 1001，用于若存在已创建的连接能承载第一数据通道，控制接口模块 1002 发送第一请求的响应，该响应包括第一媒体描述信息，第一媒体描述信息包括第一指示信息，第一指示信息指示第一连接，第一连接为已创建的连接中第一数据通道对应的连接；例如，处理模块 1001 可以用于执行 S502A。或者，处理模块 1001，用于若不存在已创建的连接能承载第一数据通道，控制接口模块 1002 发送第一请求的响应并创建第二连接，该响应包括第二媒体描述信息，第二媒体描述信息包括第二指示信息，第二指示信息指示第二连接，第二连接为第一数据通道对应的连接。例如，处理模块 1001 可以用于执行 S502B。

一种可能的实现方式，第一指示信息包括：互联网协议地址信息、用户数据报协议端口信息以及流控制传输协议端口信息。

一种可能的实现方式，接口模块 1002，还用于接收第二请求，第二请求用于请求创建第二数据通道，第二数据通道用于承载第二应用的数据，第二应用与第一应用不同；若已创建的连接不能承载第二数据通道，处理模块 1001，还用于控制接口模块 1002 发送第二请求的响应并创建第三连接，第二请求的响应包括第三媒体描述信息，第三媒体描述信息包括第二数据通道对应的第三指示信息，第三指示信息指示第三连接；或，若第一连接能承载第二数据通道，处理模块 1001，还用于控制接口模块 1002 发送第二请求的响应，第二请求的响应包括第四媒体描述信息，第四媒体描述信息包括第一指示信息；或，若第二连接能承载第二数据通道，处理模块 1001，还用于控制接口模块 1002 发送第二请求的响应，第二请求的响应包括第五媒体描述信息，第五媒体描述信息包括第二指示信息。

一种可能的实现方式，第二连接能承载第二数据通道，包括：第二数据通道的服务质量要求与第一数据通道的服务质量要求相同；或者，第二数据通道的服务质量要求低于第一数据通道的服务质量要求。

一种可能的实现方式，第二连接能承载第二数据通道，还包括：第二数据通道为通信装置 100 和第二通信装置之间的数据通道。

一种可能的实现方式，接口模块 1002，还用于向网络侧发送第六媒体描述信息，第六媒体描述包括第一数据通道对应的连接的信息和第二数据通道对应的连接的信息。

一种可能的实现方式，第六媒体描述信息还包括第一数据通道的标识信息以及第二数据通道的标识

信息。

一种可能的实现方式，处理模块 1001，还用于若第一请求包括第一连接的信息，确定第一连接能承载第一数据通道；或，处理模块 1001，还用于若第一请求不包括第一数据通道对应的连接的信息，确定已创建的连接均不能承载第一数据通道。

当用于实现通信实体的功能时，关于通信装置 100 所能实现的其他功能，可参考图 5 至图 9 所示的实施例的相关介绍，不多赘述。

在一个简单的实施例中，本领域的技术人员可以想到通信装置 100 可以采用图 4 所示的形式。比如，图 4 中的处理器 401 可以通过调用存储器 403 中存储的计算机执行指令，使得通信装置 100 执行上述方法实施例中所述的方法。

示例性的，图 10 中的处理模块 1001 和接口模块 1002 的功能/实现过程可以通过图 4 中的处理器 401 调用存储器 403 中存储的计算机执行指令来实现。或者，图 10 中的处理模块 1001 的功能/实现过程可以通过图 4 中的处理器 401 调用存储器 403 中存储的计算机执行指令来实现，图 10 中的接口模块 1002 的功能/实现过程可以通过图 4 中的通信接口 404 来实现。

可以理解的是，以上模块或单元的一个或多个可以软件、硬件或二者结合来实现。当以上任一模块或单元以软件实现的时候，所述软件以计算机程序指令的方式存在，并被存储在存储器中，处理器可以用于执行所述程序指令并实现以上方法流程。该处理器可以内置于片上系统（system-on-a-chip，SoC）或 ASIC，也可是一个独立的半导体芯片。该处理器内处理用于执行软件指令以进行运算或处理的核外，还可进一步包括必要的硬件加速器，如现场可编程门阵列（field programmable gate array，FPGA）、可编程逻辑器件（programmable logic device，PLD）或者实现专用逻辑运算的逻辑电路。

当以上模块或单元以硬件实现的时候，该硬件可以是 CPU、微处理器、数字信号处理（digital signal processing，DSP）芯片、微控制单元（microcontroller unit，MCU）、人工智能处理器、ASIC、SoC、FPGA、PLD、专用数字电路、硬件加速器或非集成的分立器件中的任一个或任一组合，其可以运行必要的软件或不依赖于软件以执行以上方法流程。

可选的，本申请还提供了一种芯片系统，包括：至少一个处理器和接口，该至少一个处理器通过接口与存储器耦合，当该至少一个处理器执行存储器中的计算机程序或指令时，使得上述任一方法实施例中的方法被执行。在一种可能的实现方式中，该芯片系统还包括存储器。可选的，该芯片系统可以由芯片构成，也可以包含芯片和其他分立器件，本申请对此不作具体限定。

可选的，本申请还提供了一种计算机可读存储介质。上述方法实施例中的全部或者部分流程可以由计算机程序来指令相关的硬件完成，该程序可存储于上述计算机可读存储介质中，该程序在执行时，可包括如上述各方法实施例的流程。计算机可读存储介质可以是前述任一实施例的通信装置的内部存储单元，例如通信装置的硬盘或内存。上述计算机可读存储介质也可以是上述通信装置的外部存储设备，例如上述通信装置上配备的插接式硬盘，智能存储卡（smart media card，SMC），安全数字（secure digital，SD）卡，闪存卡（flash card）等。进一步地，上述计算机可读存储介质还可以既包括上述通信装置的内部存储单元也包括外部存储设备。上述计算机可读存储介质用于存储上述计算机程序以及上述通信装置所需的其他程序和数据。上述计算机可读存储介质还可以用于暂时地存储已经输出或者将要输出的数据。

可选的，本申请还提供了一种计算机程序产品。上述方法实施例中的全部或者部分流程可以由计算机程序来指令相关的硬件完成，该程序可存储于上述计算机程序产品中，该程序在执行时，可包括如上述各方法实施例的流程。

可选的，本申请还提供了一种计算机指令。上述方法实施例中的全部或者部分流程可以由计算机指令来指令相关的硬件（如计算机、处理器或终端等）完成。该程序可被存储于上述计算机可读存储介质中或上述计算机程序产品中。

可选的，本申请还提供了一种通信系统，包括：上述实施例中的第一终端和第二终端。

通过以上的实施方式的描述，所属领域的技术人员可以清楚地了解到，为描述的方便和简洁，仅以上述各功能模块的划分进行举例说明，实际应用中，可以根据需要而将上述功能分配由不同的功能模块完成，即将装置的内部结构划分成不同的功能模块，以完成以上描述的全部或者部分功能。

在本申请所提供的几个实施例中，应该理解到，所揭露的装置和方法，可以通过其它的方式实现。例如，以上所描述的装置实施例仅仅是示意性的，例如，所述模块或单元的划分，仅仅为一种逻辑功能划分，实际实现时可以有另外的划分方式，例如多个单元或组件可以结合或者可以集成到另一个装置，

或一些特征可以忽略，或不执行。另一点，所显示或讨论的相互之间的耦合或直接耦合或通信连接可以是通过一些接口，装置或单元的间接耦合或通信连接，可以是电性，机械或其它的形式。

所述作为分离部件说明的单元可以是或者也可以不是物理上分开的，作为单元显示的部件可以是一个物理单元或多个物理单元，即可以位于一个地方，或者也可以分布到多个不同地方。可以根据实际的需要选择其中的部分或者全部单元来实现本实施例方案的目的。

另外，在本申请各个实施例中的各功能单元可以集成在一个处理单元中，也可以是各个单元单独物理存在，也可以两个或两个以上单元集成在一个单元中。上述集成的单元既可以采用硬件的形式实现，也可以采用软件功能单元的形式实现。

以上所述，仅为本申请的具体实施方式，但本申请的保护范围并不局限于此，任何在本申请揭露的技术范围内的变化或替换，都应涵盖在本申请的保护范围之内。因此，本申请的保护范围应以所述权利要求的保护范围为准。

权 利 要 求 书

1．一种通信方法，其特征在于，所述方法包括：

接收第一请求，所述第一请求用于请求创建第一数据通道，所述第一数据通道为第一通信装置和第二通信装置之间的数据通道，用于承载第一应用的数据；所述第一数据通道与所述第一通信装置和所述第二通信装置之间的互联网协议多媒体子系统IMS会话相关联；

若存在已创建的连接能承载所述第一数据通道，则发送所述第一请求的响应，所述响应包括第一媒体描述信息，所述第一媒体描述信息包括第一指示信息，所述第一指示信息指示第一连接，所述第一连接为所述已创建的连接中所述第一数据通道对应的连接；或，

若不存在已创建的连接能承载所述第一数据通道，则发送所述第一请求的响应并创建第二连接，所述响应包括第二媒体描述信息，所述第二媒体描述信息包括第二指示信息，所述第二指示信息指示所述第二连接，所述第二连接为所述第一数据通道对应的连接。

2．根据权利要求1所述的方法，其特征在于，所述第一指示信息包括：互联网协议地址信息、用户数据报协议端口信息以及流控制传输协议端口信息。

3．根据权利要求1或2所述的方法，其特征在于，所述方法还包括：

接收第二请求，所述第二请求用于请求创建第二数据通道，所述第二数据通道用于承载第二应用的数据，所述第二应用与所述第一应用不同；

若已创建的连接不能承载所述第二数据通道，则发送所述第二请求的响应并创建第三连接，所述第二请求的响应包括第三媒体描述信息，所述第三媒体描述信息包括所述第二数据通道对应的第三指示信息，所述第三指示信息指示所述第三连接；或，

若所述第一连接能承载所述第二数据通道，则发送所述第二请求的响应，所述第二请求的响应包括第四媒体描述信息，所述第四媒体描述信息包括所述第一指示信息；或，

若所述第二连接能承载所述第二数据通道，则发送所述第二请求的响应，所述第二请求的响应包括第五媒体描述信息，所述第五媒体描述信息包括所述第二指示信息。

4．根据权利要求3所述的方法，其特征在于，所述第二连接能承载所述第二数据通道，包括：

所述第二数据通道的服务质量要求与所述第一数据通道的服务质量要求相同；或者，

所述第二数据通道的服务质量要求低于所述第一数据通道的服务质量要求。

5．根据权利要求4所述的方法，其特征在于，所述第二连接能承载所述第二数据通道，还包括：

所述第二数据通道为所述第一通信装置和所述第二通信装置之间的数据通道。

6．根据权利要求3-5中任一项所述的方法，其特征在于，所述方法还包括：

向网络侧发送第六媒体描述信息，所述第六媒体描述包括所述第一数据通道对应的连接的信息和所述第二数据通道对应的连接的信息。

7．根据权利要求6所述的方法，其特征在于，所述第六媒体描述信息还包括所述第一数据通道的标识信息以及所述第二数据通道的标识信息。

8．根据权利要求1-7中任一项所述的方法，其特征在于，所述方法还包括：

若所述第一请求包括所述第一连接的信息，则确定所述第一连接能承载所述第一数据通道；或，

若所述第一请求不包括所述第一数据通道对应的连接的信息，则确定已创建的连接均不能承载所述第一数据通道。

9．一种通信装置，其特征在于，所述通信装置包括：接口模块和处理模块；

所述接口模块，用于接收第一请求，所述第一请求用于请求创建第一数据通道，所述第一数据通道为所述通信装置和第二通信装置之间的数据通道，用于承载第一应用的数据；所述第一数据通道与所述通信装置和所述第二通信装置之间的互联网协议多媒体子系统IMS会话相关联；

所述处理模块，用于若存在已创建的连接能承载所述第一数据通道，控制所述接口模块发送所述第一请求的响应，所述响应包括第一媒体描述信息，所述第一媒体描述信息包括第一指示信息，所述第一指示信息指示第一连接，所述第一连接为所述已创建的连接中所述第一数据通道对应的连接；或，

所述处理模块，用于若不存在已创建的连接能承载所述第一数据通道，控制所述接口模块发送所述第一请求的响应并创建第二连接，所述响应包括第二媒体描述信息，所述第二媒体描述信息包括第二指示信息，所述第二指示信息指示所述第二连接，所述第二连接为所述第一数据通道对应的连接。

10. 根据权利要求 9 所述的通信装置，其特征在于，所述第一指示信息包括：互联网协议地址信息、用户数据报协议端口信息以及流控制传输协议端口信息。

11. 根据权利要求 9 或 10 所述的通信装置，其特征在于，

所述接口模块，还用于接收第二请求，所述第二请求用于请求创建第二数据通道，所述第二数据通道用于承载第二应用的数据，所述第二应用与所述第一应用不同；

若已创建的连接不能承载所述第二数据通道，所述处理模块，还用于控制所述接口模块发送所述第二请求的响应并创建第三连接，所述第二请求的响应包括第三媒体描述信息，所述第三媒体描述信息包括所述第二数据通道对应的第三指示信息，所述第三指示信息指示所述第三连接；或，

若所述第一连接能承载所述第二数据通道，所述处理模块，还用于控制所述接口模块发送所述第二请求的响应，所述第二请求的响应包括第四媒体描述信息，所述第四媒体描述信息包括所述第一指示信息；或，

若所述第二连接能承载所述第二数据通道，所述处理模块，还用于控制所述接口模块发送所述第二请求的响应，所述第二请求的响应包括第五媒体描述信息，所述第五媒体描述信息包括所述第二指示信息。

12. 根据权利要求 11 所述的通信装置，其特征在于，所述第二连接能承载所述第二数据通道，包括：

所述第二数据通道的服务质量要求与所述第一数据通道的服务质量要求相同；或者，

所述第二数据通道的服务质量要求低于所述第一数据通道的服务质量要求。

13. 根据权利要求 12 所述的通信装置，其特征在于，所述第二连接能承载所述第二数据通道，还包括：

所述第二数据通道为所述通信装置和所述第二通信装置之间的数据通道。

14. 根据权利要求 11-13 中任一项所述的通信装置，其特征在于，

所述接口模块，还用于向网络侧发送第六媒体描述信息，所述第六媒体描述包括所述第一数据通道对应的连接的信息和所述第二数据通道对应的连接的信息。

15. 根据权利要求 14 所述的通信装置，其特征在于，所述第六媒体描述信息还包括所述第一数据通道的标识信息以及所述第二数据通道的标识信息。

16. 根据权利要求 9-15 中任一项所述的通信装置，其特征在于，

所述处理模块，还用于若所述第一请求包括所述第一连接的信息，确定所述第一连接能承载所述第一数据通道；或，

所述处理模块，还用于若所述第一请求不包括所述第一数据通道对应的连接的信息，确定已创建的连接均不能承载所述第一数据通道。

17. 一种通信装置，其特征在于，包括：处理器，所述处理器与存储器耦合，所述存储器用于存储程序或指令，当所述程序或指令被所述处理器执行时，使得所述装置执行如权利要求 1 至 8 中任一项所述的方法。

18. 一种计算机可读存储介质，其上存储有计算机程序或指令，其特征在于，所述计算机程序或指令被执行时使得计算机执行如权利要求 1 至 8 中任一项所述的方法。

19. 一种计算机程序产品，所述计算机程序产品中包括计算机程序代码，其特征在于，当所述计算机程序代码在计算机上运行时，使得计算机实现权利要求 1 至 8 中任一项所述的方法。