

一种数据处理方法、装置、设备、计算机可读存储介质及计算机程序产品

相关申请的交叉引用

5 本申请基于申请号为 202111360129.X、申请日为 2021 年 11 月 17 日的中国专利申请提出，并要求该中国专利申请的优先权，该中国专利申请的全部内容在此引入本申请作为参考。

技术领域

本申请涉及计算机应用领域中的数据处理技术，尤其涉及一种数据处理方法、装置、设备、计算机可读存储介质及计算机程序产品。

10 背景技术

随着计算机通信技术的快速发展，对数据传输的要求越来越高；为了提高数据传输效率和稳定性，可以采用多网络通道方式进行数据传输。

15 一般来说，采用多网络通道方式进行数据传输时，常常存在网络通道不能满足传输需求的情况（比如，网络通道中断等），此时，发送端需要对该网络通道中所传输的数据元是否被接收端全部接收进行确认并重传，而接收端针对数据元需要等待接收未接收到的剩余数据；如此，多网络通道传输过程中，数据元传输的时间成本较高，影响了多网络通道传输的数据处理效率。

发明内容

20 本申请实施例提供一种数据处理方法、装置、设备、计算机可读存储介质及计算机程序产品，能够在多网络通道传输的过程中提升数据处理的效率。

本申请实施例的技术方案是这样实现的：

本申请实施例提供一种数据处理方法，所述方法由接收设备执行，所述方法包括：通过至少两个网络通道接收发送设备发送的数据元，其中，所述数据元包括交付顺序；

25 当接收到的第 n 个所述数据元为待判断数据元时，基于所述交付顺序，获取接收到的第 n 个所述数据元的后序数据元，其中，所述待判断数据元的状态为不完整状态，且所述待判断数据元的交付类型为可丢类型，所述状态是基于是否接收到所述数据元的全部数据确定的，所述交付类型基于以下两种信息确定：后处理时被其他所述数据元依赖的程度和对后处理结果的影响程度，n 为正整数；

30 当所述后序数据元包括目标数据元时，确定第 n 个所述数据元的交付结果为丢弃，其中，所述目标数据元的所述状态为完整状态，且所述目标数据元与第 n 个所述数据元在数据类型上相同，以及所述目标数据元为独立数据元，所述数据类型是基于所述数据元的作用确定的，所述独立数据元在后处理时对其他所述数据元的依赖程度小于指定程度。

35 本申请实施例还提供一种数据处理方法，所述方法由发送设备执行，所述方法包括：

通过至少两个网络通道向接收设备发送数据元, 其中, 所述数据元包括交付顺序;
在所述数据元的发送过程中, 响应于切换指令, 获取第 n 个所述数据元的交付类型,
其中, 所述切换指令用于对传输第 n 个所述数据元的所述网络通道进行切换, 所述交付
类型基于以下两种信息确定: 后处理时被其他所述数据元依赖的程度和对后处理结果的
影响程度, n 为正整数;

5 当第 n 个所述数据元的所述交付类型为可丢类型, 且后序数据元包括目标数据元时,
取消向所述接收设备发送第 n 个所述数据元, 其中, 所述后序数据元为已发送的所述交
付顺序在第 n 个所述数据元之后的所述数据元, 所述目标数据元的状态为完整状态, 且
所述目标数据元与第 n 个所述数据元在数据类型上相同, 以及所述目标数据元为独立数
10 据元, 所述状态是基于所述接收设备是否接收到所述数据元的全部数据确定的, 所述数
据类型是基于所述数据元的作用确定的, 所述独立数据元在后处理时对其他所述数据元
的依赖程度小于指定程度。

本申请实施例提供一种第一数据处理装置, 所述第一数据处理装置包括:

15 数据接收模块, 配置为通过至少两个网络通道接收发送设备发送的数据元, 其中,
所述数据元包括交付顺序;

交付处理模块, 配置为当接收到的第 n 个所述数据元为待判断数据元时, 基于所述
交付顺序, 获取接收到的第 n 个所述数据元的后序数据元, 其中, 所述待判断数据元的
状态为不完整状态, 且所述待判断数据元的交付类型为可丢类型, 所述状态是基于是否
接收到所述数据元的全部数据确定的, 所述交付类型基于以下两种信息确定: 后处理时
20 被其他所述数据元依赖的程度和对后处理结果的影响程度, n 为正整数;

交付处理模块, 还配置为当所述后序数据元包括目标数据元时, 确定第 n 个所述数
据元的交付结果为丢弃, 其中, 所述目标数据元的所述状态为完整状态, 且所述目标数
据元与第 n 个所述数据元在数据类型上相同, 以及所述目标数据元为独立数据元, 所述
数据类型是基于所述数据元的作用确定的, 所述独立数据元在后处理时对其他所述数
25 据元的依赖程度小于指定程度。

本申请实施例提供一种第二数据处理装置, 所述第二数据处理装置包括:

数据发送模块, 配置为通过至少两个网络通道向接收设备发送数据元, 其中, 所述
数据元包括交付顺序;

30 通道切换模块, 配置为在所述数据元的发送过程中, 响应于切换指令, 获取第 n 个
所述数据元的交付类型, 其中, 所述切换指令用于对传输第 n 个所述数据元的所述网络
通道进行切换, 所述交付类型基于以下两种信息确定: 后处理时被其他所述数据元依赖
的程度和对后处理结果的影响程度, n 为正整数;

切换处理模块, 配置为当第 n 个所述数据元的所述交付类型为可丢类型, 且后序数
据元包括目标数据元时, 取消向所述接收设备发送第 n 个所述数据元, 其中, 所述后序
数据元为已发送的所述交付顺序在第 n 个所述数据元之后的所述数据元, 所述目标数据
元的状态为完整状态, 且所述目标数据元与第 n 个所述数据元在数据类型上相同, 以及
所述目标数据元为独立数据元, 所述状态是基于所述接收设备是否接收到所述数据元的
全部数据确定的, 所述数据类型是基于所述数据元的作用确定的, 所述独立数据元在后
35 处理时对其他所述数据元的依赖程度小于指定程度。

40 本申请实施例提供一种用于数据处理的接收设备, 所述接收设备包括:

第一存储器, 用于存储计算机可执行指令;

第一处理器, 用于执行所述第一存储器中存储的计算机可执行指令时, 实现本申请
实施例提供的应用于接收设备的数据处理方法。

本申请实施例提供一种用于数据处理的发送设备, 所述发送设备包括:

第二存储器，用于存储计算机可执行指令；

第二处理器，用于执行所述第二存储器中存储的计算机可执行指令时，实现本申请实施例提供的应用于发送设备的数据处理方法。

本申请实施例提供一种计算机可读存储介质，存储有计算机可执行指令，所述计算机可执行指令用于被第一处理器执行时，实现本申请实施例提供的应用于接收设备的数据处理方法；或者，所述计算机可执行指令用于被第二处理器执行时，实现本申请实施例提供的应用于发送设备的数据处理方法。

本申请实施例提供一种计算机程序产品，包括计算机程序或计算机可执行指令，所述计算机程序或计算机可执行指令被第一处理器执行时，实现本申请实施例提供的应用于接收设备的数据处理方法；或者，所述计算机程序或计算机可执行指令被第二处理器执行时，实现本申请实施例提供的应用于发送设备的数据处理方法。

本申请实施例至少具有以下有益效果：在多网络通道传输的过程中，当因网络通道切换导致接收端当前接收的数据元（第n个数据元）接收了数据元的部分数据（称为不完整状态）时，如果当前接收的数据元为可丢类型，表明当前接收的数据元在后处理时被其他数据元依赖的程度和对后处理结果的影响程度均低于指定值；接着，如果确定交付顺序在当前接收的数据元之后的后序数据元中，包括目标数据元（能够独立完成与当前接收的数据元作用相同的后处理的数据元）时，发送设备直接进行后序未发送数据元的发送，接收设备将当前接收的数据元进行丢弃处理，如此，降低了发送设备确认重传的时长和接收设备等待数据的时长；从而，能够降低网络通道切换时的数据元传输时间成本，提升多网络通道传输过程中的数据处理效率。

附图说明

- 图1是本申请实施例提供的数据处理系统的架构示意图；
- 图2是本申请实施例提供的图1中的终端的组成结构示意图；
- 图3是本申请实施例提供的图1中的服务器的组成结构示意图；
- 图4是本申请实施例提供的数据处理方法的流程示意图一；
- 图5是本申请实施例提供的一种示例性的多网络通道切换的示意图；
- 图6a是本申请实施例提供的一种示例性的数据处理方法的交互图；
- 图6b是本申请实施例提供的一种示例性的云游戏应用的系统架构图；
- 图6c是本申请实施例提供的一种示例性的通信模型架构图；
- 图6d是本申请实施例提供的数据处理方法的流程示意图二；
- 图6e是本申请实施例提供的数据处理方法的流程示意图三；
- 图6f是本申请实施例提供的数据处理方法的流程示意图四；
- 图7是本申请实施例提供的示例性的接收到的数据元的存储示意图；
- 图8是本申请实施例提供的云游戏应用场景的示意图；
- 图9是本申请实施例提供的示例性的数据处理的流程示意图一；
- 图10是本申请实施例提供的示例性的数据处理的流程示意图二。

具体实施方式

为了使本申请的目的、技术方案和优点更加清楚，下面将结合附图对本申请作进一步地详细描述，所描述的实施例不应视为对本申请的限制，本领域普通技术人员在没有做出创造性劳动前提下所获得的所有其它实施例，都属于本申请保护的范围。

在以下的描述中，涉及到“一些实施例”，其描述了所有可能实施例的子集，但是可以理解，“一些实施例”可以是所有可能实施例的相同子集或不同子集，并且可以在不冲突的情况下相互结合。

在以下的描述中，所涉及的术语“第一\第二\第三”仅仅是区别类似的对象，不代表针对对象的特定排序，可以理解地，“第一\第二\第三”在允许的情况下可以互换特定的顺序或先后次序，以使这里描述的本申请实施例能够以除了在这里图示或描述的以外的顺序实施。

除非另有定义，本申请实施例所使用的所有的技术和科学术语与属于本申请的技术领域的技术人员通常理解的含义相同。本申请实施例中所使用的术语只是为了描述本申请实施例的目的，不是旨在限制本申请。

对本申请实施例进行进一步详细说明之前，对本申请实施例中涉及的名词和术语进行说明，本申请实施例中涉及的名词和术语适用于如下的解释。

1) 云计算 (Cloud Computing)，是一种计算模式，通过将计算任务分布在大量计算机构成的资源池上，使各种应用系统能够根据需要获取计算力、存储空间和信息服务；其中，为资源池提供资源的网络被称为“云”，“云”中的资源在使用者看来是可以无限扩展的，并且可以随时获取，按需使用，随时扩展，按使用付费。

2) 云游戏 (Cloud Gaming)，又可分为游戏点播 (Gaming on Demand)，是一种以云计算技术为基础的在线游戏技术。云游戏技术使得图形处理与数据运算能力低于指定能力的轻端设备 (Thin Client) 能够流畅地运行游戏。在云游戏场景下，游戏并不在玩家游戏终端运行，而是在云端服务器中运行，并由云端服务器将游戏场景渲染为音视频流，通过网络传输给玩家游戏终端。玩家游戏终端的图形运算与数据处理能力低于指定能力，也能通过基本的流媒体播放能力与获取玩家输入指令并发送给云端服务器的能力来运行游戏。本申请实施例提供的数据处理方法，可以应用于云游戏的多网络通道传输的场景中。

3) 网络通道，用于传输数据，可以是基于网络和/或传输协议确定出的传输通道；其中，网络比如为第四代移动通信网络 (4G, The 4th Generation)，第五代移动通信网络 (5G, The 5th Generation)，各频段的无线相容性认证 (Wi-Fi) 网络；传输协议比如为传输控制协议 (TCP, Transmission Control Protocol)，用户数据报协议 (UDP, User Datagram Protocol)。

4) 多网络通道传输 (Multichannel Transmission)，是指在多个网络通道上同时传输多个数据元的处理，该多个数据元可以是同一网络会话 (Session) 中的；本申请实施例中采用至少两个网络通道传输数据元的过程即为多网络通道传输。

5) 多网络通道切换，是指由于网络原因等导致网络会话中的数据元从一条网络通道切换至另一网络通道进行传输的过程。

本申请实施例提供一种数据处理方法、装置、设备、计算机可读存储介质及计算机程序产品，能够提升多网络通道传输过程中的数据处理效率。下面说明本申请实施例提供的发送设备和接收设备的示例性应用，本申请实施例提供的发送设备和接收设备可以实施为智能手机、智能手表、笔记本电脑、平板电脑、台式计算机、智能电视、机顶盒、智能车载设备、便携式音乐播放器、个人数字助理、专用消息设备、便携式游戏设备和智能音箱等各种类型的终端，也可以实施为服务器，而服务器可以是独立的物理服务器，也可以是多个物理服务器构成的服务器集群或者分布式系统，还可以是提供云服务、云数据库、云计算、云函数、云存储、网络服务、云通信、中间件服务、域名服务、安全服务、内容分发网络 (CDN, Content Delivery Network)、以及大数据和人工智能平台等基础云计算服务的云服务器；以及可以实施为终端和服务器。下面，将说明发送设备实

施为服务器、而接收设备实施为终端时的示例性应用。

参见图 1，图 1 是本申请实施例提供的数据处理系统的架构示意图；如图 1 所示，为支撑一个数据处理应用，在数据处理系统 100 中，终端 200（称为接收设备，示例性地示出了终端 200-1 和终端 200-2）和服务 5 器 300（称为发送设备）通过至少两个网络通道 400 进行数据传输。另外，该数据处理系统 100 中还包括数据库 500，用于向服务器 300 提供数据支持；并且，图 1 中示出的为数据库 500 独立于服务器 300 的一种情况，此外，数据库 500 还可以集成在服务器 300 中，本申请实施例对此不限定。

终端 200，用于通过至少两个网络通道 400 接收服务器 300 发送的数据元，其中，数据元包括交付顺序；当接收到的第 n 个数据元为待判断数据元时，基于交付顺序，获取接收到的第 n 个数据元的后序数据元，其中，待判断数据元的状态为不完整状态，且待判断数据元的交付类型为可丢类型，状态是基于是否接收到数据元的全部数据确定的，交付类型基于以下两种信息确定：后处理时被其他数据元依赖的程度和对后处理结果的影响程度， n 为正整数；当后序数据元包括目标数据元时，确定第 n 个数据元的交付结果为丢弃，其中，目标数据元的状态为完整状态，且目标数据元与第 n 个数据元在数据类型上相同，以及目标数据元为独立数据元，数据类型是基于数据元的作用确定的，独立数据元在后处理时对其他数据元的依赖程度小于指定程度。还用于依次对数据元进行交付和后处理，以播放音视频（比如，终端 200-1 和终端 200-2 中渲染的游戏画面）。10

服务器 300，用于通过至少两个网络通道向接收设备发送数据元；在数据元的发送过程中，响应于切换指令，获取第 n 个数据元的交付类型，其中，切换指令用于对传输第 n 个数据元的网络通道进行切换；当第 n 个数据元的交付类型为可丢类型，且后序数据元包括目标数据元时，取消向接收设备发送第 n 个数据元。20

需要说明的是，发送设备和接收设备是相对的；在一个应用场景中的发送设备，在另一个应用场景中可以为接收设备。

在一些实施例中，终端 200 和服务器 300 可以通过有线或无线通信方式进行直接或间接地连接，以及终端 200 和服务器 300 在通过网络连接时，该网络可以是广域网或者局域网，又或者是二者的组合，本申请实施例中不作限制。25

参见图 2，图 2 是本申请实施例提供的图 1 中的终端的组成结构示意图，图 2 所示的终端 200 包括：至少一个第一处理器 210、第一存储器 250、至少一个第一网络接口 220 和第一用户接口 230。终端 200 中的各个组件通过第一总线系统 240 耦合在一起。可理解，第一总线系统 240 用于实现这些组件之间的连接通信。第一总线系统 240 除包括数据总线之外，还包括电源总线、控制总线和状态信号总线。但是为了清楚说明起见，在图 2 中将各种总线都标为第一总线系统 240。30

第一处理器 210 可以是一种集成电路芯片，具有信号的处理能力，例如通用处理器、数字信号处理器（DSP，Digital Signal Processor），或者其他可编程逻辑器件、分立门或者晶体管逻辑器件、分立硬件组件等，其中，通用处理器可以是微处理器或者任何常规的处理器等。35

第一用户接口 230 包括使得能够呈现媒体内容的一个或多个第一输出装置 231，包括一个或多个扬声器和/或一个或多个视觉显示屏。第一用户接口 230 还包括一个或多个第一输入装置 232，包括有助于用户输入的用户接口部件，比如键盘、鼠标、麦克风、触屏显示屏、摄像头、其他输入按钮和控件。40

第一存储器 250 可以是可移除的，不可移除的或其组合。示例性的硬件设备包括固态存储器，硬盘驱动器，光盘驱动器等。第一存储器 250 可选地包括在物理位置上远离第一处理器 210 的一个或多个存储设备。

第一存储器 250 包括易失性存储器或非易失性存储器，也可包括易失性和非易失性

存储器两者。非易失性存储器可以是只读存储器 (ROM, Read Only Memory), 易失性存储器可以是随机存取存储器 (RAM, Random Access Memory)。本申请实施例描述的第一存储器 250 旨在包括任意适合类型的存储器。

5 在一些实施例中, 第一存储器 250 能够存储数据以支持各种操作, 这些数据的示例包括程序、模块和数据结构或者其子集或超集, 下面示例性说明。

第一操作系统 251, 包括用于处理各种基本系统服务和执行硬件相关任务的系统程序, 例如框架层、核心库层、驱动层等, 用于实现各种基础业务以及处理基于硬件的任务;

10 第一网络通信模块 252, 用于经由一个或多个 (有线或无线) 第一网络接口 220 到达其他电子设备, 示例性的第一网络接口 220 包括: 蓝牙、Wi-Fi、和通用串行总线 (USB, Universal Serial Bus) 等;

第一呈现模块 253, 用于经由一个或多个与第一用户接口 230 相关联的第一输出装置 231 (例如, 显示屏、扬声器等) 使得能够呈现信息 (例如, 用于操作外围设备和显示内容和信息的用户接口);

15 第一输入处理模块 254, 用于对一个或多个来自一个或多个第一输入装置 232 之一的一个或多个用户输入或互动进行检测以及翻译所检测的输入或互动。

20 在一些实施例中, 本申请实施例提供的第一数据处理装置可以采用软件方式实现, 图 2 示出了存储在第一存储器 250 中的第一数据处理装置 255, 其可以是程序和插件等形式的软件, 包括以下软件模块: 数据接收模块 2551 和交付处理模块 2552, 这些模块是逻辑上的, 因此根据所实现的功能可以进行任意的组合或进一步拆分。将在下文中说明各个模块的功能。

25 在一些实施例中, 本申请实施例提供的第一数据处理装置可以采用硬件方式实现, 作为示例, 本申请实施例提供的第一数据处理装置可以是采用硬件译码处理器形式的第一处理器, 其被编程以执行本申请实施例提供的应用于接收设备的数据处理方法, 例如, 硬件译码处理器形式的第一处理器可以采用一个或多个应用专用集成电路 (ASIC, Application Specific Integrated Circuit)、DSP、可编程逻辑器件 (PLD, Programmable Logic Device)、复杂可编程逻辑器件 (CPLD, Complex Programmable Logic Device)、现场可编程门阵列 (FPGA, Field-Programmable Gate Array) 或其他电子元件。

30 参见图 3, 图 3 是本申请实施例提供的图 1 中的服务器的组成结构示意图, 图 3 所示的服务器 300 包括: 至少一个第二处理器 310、第二存储器 350 和至少一个第二网络接口 320。服务器 300 中的各个组件通过第二总线系统 340 耦合在一起。可理解, 第二总线系统 340 用于实现这些组件之间的连接通信。第二总线系统 340 除包括数据总线之外, 还包括电源总线、控制总线和状态信号总线。但是为了清楚说明起见, 在图 3 中将各种总线都标为第二总线系统 340。

35 第二处理器 310 可以是一种集成电路芯片, 具有信号的处理能力, 例如通用处理器、数字信号处理器, 或者其他可编程逻辑器件、分立门或者晶体管逻辑器件、分立硬件组件等, 其中, 通用处理器可以是微处理器或者任何常规的处理器等。

40 第二存储器 350 可以是可移除的, 不可移除的或其组合。示例性的硬件设备包括固态存储器, 硬盘驱动器, 光盘驱动器等。第二存储器 350 可选地包括在物理位置上远离第二处理器 310 的一个或多个存储设备。

第二存储器 350 包括易失性存储器或非易失性存储器, 也可包括易失性和非易失性存储器两者。非易失性存储器可以是只读存储器, 易失性存储器可以是随机存取存储器。本申请实施例描述的第二存储器 350 旨在包括任意适合类型的存储器。

在一些实施例中, 第二存储器 350 能够存储数据以支持各种操作, 这些数据的示例

包括程序、模块和数据结构或者其子集或超集，下面示例性说明。

第二操作系统 351，包括用于处理各种基本系统服务和执行硬件相关任务的系统程序，例如框架层、核心库层、驱动层等，用于实现各种基础业务以及处理基于硬件的任务；

5 第二网络通信模块 352，用于经由一个或多个（有线或无线）第二网络接口 320 到达其他电子设备，示例性的第二网络接口 320 包括：蓝牙、Wi-Fi、和通用串行总线等；

在一些实施例中，本申请实施例提供的第二数据处理装置可以采用软件方式实现，图 3 示出了存储在第二存储器 350 中的第二数据处理装置 355，其可以是程序和插件等形式的软件，包括以下软件模块：数据发送模块 3551、通道切换模块 3552 和切换处理
10 模块 3553，这些模块是逻辑上的，因此根据所实现的功能可以进行任意的组合或进一步拆分。将在下文中说明各个模块的功能。

在一些实施例中，本申请实施例提供的第二数据处理装置可以采用硬件方式实现，作为示例，本申请实施例提供的第二数据处理装置可以是采用硬件译码处理器形式的第二处理器，其被编程以执行本申请实施例提供的应用于发送设备的数据处理方法，例如，
15 硬件译码处理器形式的第二处理器可以采用一个或多个应用专用集成电路、DSP、可编程逻辑器件、复杂可编程逻辑器件、现场可编程门阵列或其他电子元件。

在一些实施例中，终端或服务器可以通过运行计算机程序来实现本申请实施例提供的数据处理方法。举例来说，计算机程序可以是操作系统中的原生程序或软件模块；可以是本地（Native）应用程序（APP，Application），即需要在操作系统中安装才能运行的程序，如游戏 APP、邮箱 APP、文件管理 APP 或者即时通信 APP；也可以是小程序，
20 即只需要下载到浏览器环境中就可以运行的程序；还可以是能够嵌入至任意 APP 中的小程序。总而言之，上述计算机程序可以是任意形式的应用程序、模块或插件。

下面，将结合本申请实施例提供的发送设备和接收设备的示例性应用和实施，说明本申请实施例提供的数据处理方法。

25 参见图 4，图 4 是本申请实施例提供的数据处理方法的流程示意图一，将结合图 4 示出的步骤进行说明。

S401、发送设备通过至少两个网络通道向接收设备发送数据元。

在本申请实施例中，发送设备和接收设备之间通过多网络通道实现数据的传输，并且传输的单位数据为数据元（比如，一帧音频，一帧视频，等等），以及多网络通道即
30 为至少两个网络通道。这里，当发送设备通过至少两个网络通道向接收设备发送数据元时，相应地，接收设备通过至少两个网络通道接收发送设备发送的数据元。

需要说明的是，发送设备通过向接收设备传输数据，以在接收设备进行数据处理来实现应用功能；比如，服务器（发送设备）向终端（接收设备）发送云游戏对应的音视频数据或直播对应的音视频数据，以在终端播放音视频。这里，发送设备将所传输的数据通过拆分为数据元在至少两个网络通道中传输，数据元包括交付顺序，该交付顺序是接收设备交付该数据元的顺序，其中，交付用于完成数据的后处理以实现应用功能，比如，交付用于完成解码以实现音视频的播放，其中，后处理为交付后的处理，比如，解码、渲染，等等；也就是说，通过至少两个网络通道所传输的数据是具有连贯性的数据元序列。这里，交付顺序可以是由发送设备携带在数据元中的数据。

40 还需要说明的是，发送设备可以是实时地向接收设备发送数据元，当然也可以是非实时地向接收设备发送数据元，本申请实施例对此不作限定；在本申请实施例中，当采用多网络通道传输是为了提升数据元的传输效率时，通过多网络通道实时地发送数据元。

S402、发送设备在数据元的发送过程中，响应于切换指令，获取第 n 个数据元的交付类型。

在本申请实施例中，发送设备在通过多网络通道向接收设备发送数据元的过程中，由于用于发送的网络具有不定性；因此，当因为网络等原因触发了网络通道切换事件时，发送设备也就获得了切换指令；此时，发送设备响应于该切换指令，进行网络通道的切换，并获取切换指令所指示切换网络通道的数据元，也就获得了第 n 个数据元。

5 需要说明的是，网络通道切换事件是指切换网络通道的事件，包括触发条件和切换策略；其中，触发条件为网络通道的连接状态、传输速度和适用场景等中的至少一种，切换策略是指切换网络通道的规则，用于确定传输数据元的新的网络通道，比如，优先级。这里，切换后的网络通道是指切换指令指示的用于传输数据元的新的网络通道；第 n 个数据元为接收到切换指令时待切换的网络通道中正传输的数据元；切换指令用于对
10 传输第 n 个数据元的网络通道进行切换， n 为正整数，比如，1，9，22，等等；交付类型是基于后处理时被其他数据元依赖的程度和对后处理结果的影响程度确定的，用于确定第 n 个数据元的处理方式是确认并重传还是丢弃并直接传输后序的数据元。

示例性地，参见图 5，图 5 是本申请实施例提供的一种示例性的多网络通道切换的示意图；如图 5 所示，发送设备 5-1 通过多网络通道 5-2 向接收设备 5-3 发送数据，其中，所发送的数据为数据元序列，数据元序列包括数据元 5-41 至数据元 5-48，多网络
15 通道 5-2 包括网络通道 5-21 至网络通道 5-24。这里，数据元 5-45 至数据元 5-48 为未发送的数据元，数据元 5-41 和数据元 5-42 均通过网络通道 5-22 传输，数据元 5-43 和数据元 5-44 均通过网络通道 5-21 传输；传输一段时间之后，接收设备 5-3 接收到了通过网络通道 5-22 传输来的数据元 5-41 和数据元 5-42，而发送设备 5-1 此时由于网络原因
20 需要进行网络通道的切换，将正在传输的数据元 5-43 和数据元 5-44 的传输由网络通道 5-21 切换至网络通道 5-22，此时，发送设备获取数据元 5-43 和数据元 5-44 分别对应的交付类型，以基于获得的交付类型确定网络通道切换后的处理方式。

S403、当第 n 个数据元的交付类型为可丢类型，且后序数据元包括目标数据元时，发送设备取消向接收设备发送第 n 个数据元。

25 在本申请实施例中，交付类型包括可丢类型和非可丢类型，其中，可丢类型是指后处理时被其他数据元依赖的程度和对后处理结果的影响程度均较小（比如，被其他数据元依赖的程度小于第一程度阈值，对后处理结果的影响程度小于第二程度阈值）；可丢类型是指后处理时被其他数据元依赖的程度和对后处理结果的影响程度中的至少一种较大（比如，被其他数据元依赖的程度大于第一程度阈值，和/或，对后处理结果的影响
30 程度大于第二程度阈值）。

首先，当发送设备确定第 n 个数据元的交付类型为可丢类型时，表明接收设备即使未完全接收到该数据元的数据，将该数据元丢弃，也不影响其他数据元的后处理。

接着，发送设备判断已发送的交付顺序在第 n 个所述数据元之后的后序数据元中是否包括目标数据元，目标数据元的状态为完整状态，且目标数据元与第 n 个数据元在数据
35 类型上相同，以及目标数据元为独立数据元；如果包括，表明即使发送设备因为网络通道切换未完成第 n 个数据元的全部数据的发送而不再进行确认重传时，接收设备也能够通过丢弃第 n 个数据元，继续依据后序数据元进行交付并完成后处理；因此，发送设备在确定第 n 个数据元的交付类型为可丢类型、且后序数据元包括目标数据元时，向接收设备发送未发送的数据元，并取消向接收设备发送第 n 个后序数据元；也就是说，发
40 送设备取消第 n 个数据元的全部数据是否被接收设备接收的确认处理，以及取消重传第 n 个数据元的处理。

需要说明的是，状态是基于是否接收到了数据元的全部数据确定的，包括不完整状态和完整状态，其中，完整状态是指接收到了数据元的全部数据的状态，不完整状态是指接收到了数据元的部分数据的状态。数据类型是基于数据元的作用确定的，从而，数

据元的作用相同则数据元的数据类型相同。独立数据元为后处理时对其他数据元的依赖程度小于指定程度的数据元，即独立数据元是依据自身数据进行后处理的数据元，比如，音频帧数据，关键帧（帧内预测（Intra-Prediction，I）帧）数据，等等。后序数据元为交付顺序在第 n 个数据元之后的至少一个数据元，比如为第 $n+1$ 个数据元、第 $n+2$ 个数据元、...、第 $n+i$ 个数据元， i 为大于等于 0 的正整数。

S404、当接收设备接收到的第 n 个数据元为待判断数据元时，基于交付顺序，获取接收到的第 n 个数据元的后序数据元。

在本申请实施例中，接收设备通过至少两个网络通道接收发送设备发送的数据元，并对接收到的数据元按交付顺序进行交付，以对数据元执行后处理。

这里，接收设备对每个接收到的数据元的状态进行判断，以确定是否接收到了数据元的全部数据，如果确定当前接收到的数据元仅接收了数据元的部分数据，则确定当前接收到的数据元的状态为不完整状态，其中，当前接收到的数据元即为接收到的第 n 个数据元。在确定第 n 个数据元的状态为不完整状态时，接收设备对第 n 个数据元的交付类型进行确定，当确定第 n 个数据元的交付类型为可丢类型时，表明不完整状态的第 n 个数据元在后处理时被其他数据元依赖的程度小于第一程度阈值，以及对后处理结果的影响程度均小于第二程度阈值，是可丢弃的，此时，接收到的第 n 个数据元为待判断数据元；其中，待判断数据元的状态为不完整状态，且待判断数据元的交付类型为可丢类型。在确定不完整状态的第 n 个数据元的交付类型为可丢类型时，接收设备还要对接收到的后序数据元进行判断，以确定是否对第 n 个数据元进行丢弃。

需要说明的是，发送设备在发送数据元时，可以将数据元的交付类型携带在数据元的数据中进行发送；如此，接收设备通过读取数据元的数据，就能确定数据元的交付类型；发送设备也可以基于对数据元的数据读取结果进行判断，以确定数据元的交付类型；等等，本申请实施例对此不作限定。

这里，当接收设备确定第 n 个数据元接收到了数据元的全部数据时，可以确定第 n 个数据元的状态为完整状态，此时，交付第 n 个数据元以进行后处理；也就是说，当接收设备接收到的第 n 个数据元为完整状态时，确定第 n 个数据元的交付结果包括交付处理。

S405、当后序数据元包括目标数据元时，接收设备确定第 n 个数据元的交付结果为丢弃。

在本申请实施例中，接收设备获取接收到的交付顺序在第 n 个数据元之后的数据元，并确定获得的后序数据元中是否包括目标数据元，其中，目标数据元的状态为完整状态，且目标数据元与第 n 个数据元在数据类型上相同，以及目标数据元为独立数据元；也就是说，接收设备确定后序数据元中是否包括能够跳过第 n 个数据元继续进行相同作用的后处理的数据元；当确定包括时，接收设备确定第 n 个数据元的处理方式丢弃，即交付结果为丢弃，不进行交付，也就是说，针对第 n 个数据元未接收到的数据，接收设备不再等待接收数据。

需要说明的是，后序数据元可以为与第 n 个数据元在交付顺序上相邻的数据元序列，也可以为与第 n 个数据元在交付顺序上不相邻的数据元序列；以及数据元序列中各数据元之间的交付顺序可以是相邻的，也可以是不相邻的；本申请实施例对此不作限定。

示例性地，参见图 6a，图 6a 是本申请实施例提供的一种示例性的数据处理方法的交互图；如图 6a 所示，该示例性的数据处理方法描述了云游戏场景中，云游戏服务器（称为发送设备）向游戏客户端（称为接收设备）发送云游戏音视频拆分的数据元的数据处理过程，包括 S601 至 S606，下面对各步骤分别进行说明。

S601、云游戏服务器通过至少两个网络通道向游戏客户端发送云游戏音视频拆分的

数据元。

需要说明的是，云游戏服务器采用多网络通道的传输方式向游戏客户端发送云游戏音视频拆分的数据元。

5 S602、云游戏服务器在数据元的发送过程中，响应于切换指令，获取待切换的网络通道所传输的数据元（称为第 n 个数据元）的交付类型。

需要说明的是，当云游戏服务器基于预先设置的网络通道切换策略确定进行网络通道切换时，生成切换指令，用于对网络通道进行切换。

10 S603、当待切换的网络通道所传输的数据元的交付类型为可丢类型，且后序数据元包括状态为完整状态并与待切换的网络通道所传输的数据元在数据类型上相同的独立数据元时，云游戏服务器取消向游戏客户端发送待切换的网络通道所传输的数据元。

需要说明的是，在云游戏服务器侧，如果进行了网络通道的切换，则判断待切换的网络通道中正在传输的数据元之后是否包括能够独立完成相同类型的后处理的数据元（比如，能够独立渲染游戏画面的关键帧），如果包括，则取消向游戏客户端发送待切换的网络通道所传输的数据元，并基于切换后的网络通道直接进行未发送的数据元的发送，降低确认重传的资源消耗。

15 S604、当游戏客户端当前接收到的数据元的状态为不完整状态、且交付类型为可丢类型时，获取接收到的后序数据元。

这里，当前接收到的数据元即为待切换的网络通道所传输的数据元。

20 S605、当后序数据元包括状态为完整状态、且与待切换的网络通道所传输的数据元在数据类型上相同的独立数据元时，游戏客户端确定当前接收到的数据元的交付结果为丢弃。

需要说明的是，在游戏客户端侧，如果确认因云游戏服务器侧进行了网络通道的切换导致接收到的待切换的网络通道所传输的数据元不完整，并确认第 n 个数据元为可丢类型，以及确认后序数据元中包括能够独立完成相同类型的后处理的数据元，则丢弃当前接收到的数据元，继续进行后续相同类型的数据元的后处理；如此，降低了接收不完整状态的数据元的等待时间，提升了云游戏的渲染效果。

25 还需要说明的是，图 6a 是基于云游戏应用场景的系统架构实现的；下面说明云游戏应用场景的系统架构；参见图 6b，图 6b 是本申请实施例提供的一种示例性的云游戏应用的系统架构图；如图 6b 所示，该示例性的云游戏应用的系统架构中，包括游戏客户端 6-1 和云游戏服务器 6-2，游戏客户端 6-1 用于响应于游戏用户针对虚拟场景界面所执行的操作，向云游戏服务器 6-2 发送游戏指令 6-3。此时，云游戏服务器 6-2 基于游戏指令 6-3 进行渲染，得到云游戏音视频 6-4，并将游戏音视频 6-4 拆分为数据元（示例性地示出了数据元 6-41 至数据元 6-48）通过多网络通道 6-5（示例性地示出了网络通道 6-51 和网络通道 6-52）进行传输（数据元 6-41 至数据元 6-43 通过网络通道 6-51 传输，数据元 6-44 至数据元 6-48 通过网络通道 6-52 传输）。

30 这里，当前的数据元的传输情况为：云游戏服务器 6-2 在网络通道 6-52 上正传输数据元 6-44 至数据元 6-48；游戏客户端 6-1 已接收到数据元 6-41，正在接收通过网络通道 6-51 传输的数据元 6-42，云游戏服务器 6-2 侧还包括待进行传输数据元 6-43；此时，云游戏服务器 6-2 触发了网络通道切换事件，确定将网络通道 6-51 上传输的数据元切换至网络通道 6-52 上进行传输，则正在传输数据元的 6-42（称为第 n 个数据元）通过图 6a 示出的数据处理方法进行处理，并且数据元 6-43 将通过网络通道 6-52 进行传输。

40 其中，图 6a 和图 6b 中，云游戏服务器向游戏客户端进行数据传输时，是基于图 6c 示出的通信模型架构图实现的；如图 6c 所示，示例性地示出了通信模型 6-6 包括会话层 6-61、传输层 6-62、网络层 6-63、数据链路层 6-64 和物理层 6-65。其中，传输层 6-62

包括多个网络通道(示例性地示出了 TCP 协议对应的网络通道 6-51 和 UDP 协议对应的网络通道 6-52), 以使本申请实施例实现多个网络通道传输。这里, 当云游戏服务器向游戏客户端传输由云游戏音视频拆分的数据元时, 第一层物理层 6-65, 用于实现机械、电子和定时接口通信信道上的原始比特流传输; 第二层数据链路层 6-64, 用于物理寻址, 同时将原始比特流转变为逻辑传输线路。第一层与第二层通常归属于网卡和网线; 第三层网络层 6-63, 用于控制子网的运行, 如逻辑编址、分组传输和路由选择等。第四层传输层 6-62, 用于向游戏客户端传输数据元, 以及接收游戏客户端的应答。第五层会话层 6-61, 用于与游戏客户端建立会话和管理会话, 向传输层 6-62 的多个网络通道分发数据元。

可以理解的是, 在多网络通道传输的过程中, 当因网络通道切换导致接收端当前接收的数据元接收了数据元的部分数据时, 如果当前接收的数据元为可丢类型, 表明当前接收的数据元在后处理时被其他数据元依赖的程度和对后处理结果的影响程度均低于指定值; 接着, 如果确定交付顺序在当前接收的数据元的交付顺序之后的后序数据元中, 包括目标数据元时, 发送设备直接进行后序未发送数据元的发送, 接收设备将当前接收的数据元进行丢弃处理, 如此, 降低了发送设备确认重传的时长和接收设备等待数据的时长; 从而, 能够降低网络通道切换时的数据元传输时间成本, 提升多网络通道传输的过程中的数据元处理效率。

参见图 6d, 图 6d 是本申请实施例提供的数据处理方法的流程示意图二; 如图 6d 所示, 在本申请实施例中, S404 之后还包括 S406 和 S407; 也就是说, 接收设备基于交付顺序, 获取接收到的第 n 个数据元的后序数据元之后, 该数据处理方法还包括 S406 和 S407, 下面对各步骤分别进行说明。

S406、当后序数据元中的第 $n+1$ 个数据元被接收到全部数据时, 接收设备确定第 $n+1$ 个数据元的状态为完整状态。

S407、当处于完整状态的第 $n+1$ 个数据元满足第一条条件时, 确定后序数据元包括目标数据元。

在本申请实施例中, 接收设备在确定第 n 个数据元为待判断数据元时, 对接收到的第 $n+1$ 个数据元进行获取, 并对第 $n+1$ 个数据元的状态、数据类型和是否为独立数据元分别进行判断, 并在确定第 $n+1$ 个数据元的状态为完整状态、数据类型与第 n 个数据元的数据类型相同、以及是独立数据元时, 确定处于完整状态的第 $n+1$ 个数据元满足第一条条件, 此时确定后序数据元包括目标数据元。易知, 后序数据元包括第 $n+1$ 个数据元, 满足第一条条件包括: 与第 n 个数据元在数据类型上相同, 并且属于独立数据元。

需要说明的是, 发送设备在确定对传输中的第 n 个数据元所在的网络通道进行切换, 并确定第 n 个数据元为可丢类型之后, 也可以采用 S406 和 S407 描述的过程, 来确定后序数据元是否包括目标数据元。

继续参见图 6d, S406 之后还包括 S408 和 S409; 也就是说, 接收设备确定第 $n+1$ 个数据元的状态为完整状态之后, 该数据处理方法还包括 S408 和 S409, 下面对各步骤分别进行说明。

S408、当处于完整状态的第 $n+1$ 个数据元为满足第二条条件时, 或者, 当处于完整状态的第 $n+1$ 个数据元满足第三条条件时, 获取后序数据元中的第 $n+2$ 个数据元的状态。

S409、当第 $n+2$ 个数据元的状态为完整状态、且第 $n+2$ 个数据元满足第一条条件时, 接收设备确定后序数据元包括目标数据元。

在本申请实施例中, 当接收设备确定第 $n+1$ 个数据元的状态为完整状态时, 如果接收设备确定第 $n+1$ 个数据元为与第 n 个数据元在数据类型上不同的独立数据元(表明处于完整状态的第 $n+1$ 个数据元满足第二条条件), 或者确定第 $n+1$ 个数据元为与第 n 个数

据元在数据类型上相同的关联数据元(表明处于完整状态的第 $n+1$ 个数据元满足第三条条件), 获取后序数据元中的第 $n+2$ 个数据元, 对第 $n+2$ 个数据元的状态、数据类型和是否为独立数据元分别进行判断, 并在确定第 $n+2$ 个数据元的状态为完整状态、数据类型与第 n 个数据元的数据类型相同、以及是独立数据元时, 确定处于完整状态的第 $n+2$ 个所述数据元满足第一条条件, 此时确定后序数据元包括目标数据元。易知, 后序数据元包括第 $n+2$ 个数据元; 满足第二条条件包括: 与第 n 个数据元在数据类型上不同, 并且属于独立数据元; 满足第三条条件包括: 与第 n 个数据元在数据类型上相同, 并且属于关联数据元。

这里, 关联数据元在后处理时对其他数据元的依赖程度大于等于指定程度, 即关联数据元是结合其他数据元进行后处理的数据元, 与独立数据元是相对的。

需要说明的是, 发送设备在确定是针对传输中的第 n 个数据元所在的网络通道进行切换, 并确定第 n 个数据元为可丢类型之后, 也可以采用 S406、S408 和 S409 描述的过程, 来确定后序数据元包括目标数据元。

参见图 6e, 图 6e 是本申请实施例提供的数据处理方法的流程示意图三; 如图 6e 所示, 在本申请实施例中, S404 中接收设备基于交付顺序, 获取接收到的第 n 个数据元的后序数据元之后, 该数据处理方法还包括 S410, 下面对各步骤分别进行说明。

S410、当后序数据元中的第 $n+1$ 个数据元为待判断数据元时, 接收设备获取交付顺序在第 $n+1$ 个数据元之后的数据元, 并基于获得的数据元确定第 n 个数据元和第 $n+1$ 个数据元的交付结果。

在本申请实施例中, 接收设备在确定后序数据元中的第 $n+1$ 个数据元的状态为不完整状态、且交付类型为可丢类型时, 也就确定了第 $n+1$ 个数据元为待判断数据元; 此时, 接收设备采用类似于 S404 和 S405 的描述对第 $n+1$ 个数据元进行交付。

需要说明的是, 发送设备也可以在确定后序数据元中的第 $n+1$ 个数据元的状态为不完整状态、且交付类型为可丢类型时, 通过获取交付顺序在第 $n+1$ 个数据元的交付顺序之后的数据元, 并基于获得的数据元确定是否不再发送第 n 个数据元和第 $n+1$ 个数据元分别对应的剩余未发送的数据。

在本申请实施例中, 当接收设备确定第 n 个数据元之前均是可丢类型的不完整状态的数据元时, 在确定后序数据元中包括目标数据元时, 可将第 n 个数据元之前的可丢类型的不完整状态的数据元进行丢弃。

继续参见图 6e, 如图 6e 所示, 在本申请实施例中, S404 中接收设备基于交付顺序, 获取接收到的第 n 个数据元的后序数据元之后, 该数据处理方法还包括 S411, 下面对各步骤分别进行说明。

S411、当后序数据元不包括目标数据元时, 接收设备确定第 n 个数据元的交付结果为等待接收数据。

在本申请实施例中, 当后序数据元包括状态为完整状态的数据元、且后序数据元不包括与第 n 个数据元在数据类型上相同的独立数据元时, 表明后续数据元不包括目标数据元; 从而, 接收设备确定后序数据元不包括能够跳过第 n 个数据元实现与第 n 个数据元相同后处理结果的数据元, 此时, 接收设备确定第 n 个数据元的交付结果为等待接收剩余未接收到的数据。

需要说明的是, 发送设备在确定后序数据元不包括目标数据元时, 针对第 n 个数据元的剩余未发送的数据进行确认并发送。

继续参见图 6e, 如图 6e 所示, 在本申请实施例中, 接收设备通过至少两个网络通道接收发送设备发送的数据元之后, 该数据方法还包括 S412, 下面对该步骤进行说明。

S412、当接收设备接收到的第 n 个数据元为等待数据元时, 确定第 n 个数据元的交

付结果为等待接收数据。

在本申请实施例中，当接收设备接收到的第 n 个数据元的状态为不完整状态、且第 n 个数据元的交付类型为非可丢类型时，表明第 n 个数据元为等待数据元；由于第 n 个数据元是非可丢类型，因此，丢弃第 n 个数据元则在后处理时无法被其他数据元依赖和/或对后处理结果的影响程度大于第二程度阈值；故接收设备等待接收第 n 个数据元的剩余未接收到的数据。

需要说明的是，接收设备对每个不完整状态的非可丢类型的数据元，均确定对应的交付结果为等待接收数据；从而，获取后序数据元时如果遇到了不完整状态、且第 n 个数据元的交付类型为非可丢类型的数据元时，确定该数据元的交付结果为等待接收数据。

继续参见图 6c，如图 6c 所示，S402 之后还包括 S413 和 S414；也就是说，发送设备获取第 n 个数据元的交付类型之后，该数据处理方法还包括 S413 和 S414，下面对各步骤分别进行说明。

S413、当第 n 个数据元的交付类型为非可丢类型时，发送设备获取第 n 个数据元的状态。

在本申请实施例中，发送设备在针对传输第 n 个数据元的网络通道进行切换时，先判断第 n 个数据元的交付类型，如果确定第 n 个数据元的交付类型为非可丢类型，则需要通过确认机制确定接收设备是否接收到了第 n 个数据元的全部数据，即判断第 n 个数据元的状态。

需要说明的是，非可丢类型是指后处理时被其他数据元依赖的程度和对后处理结果的影响程度中的至少一种高于阈值的类型。

S414、当第 n 个数据元的状态为不完整状态时，发送设备通过切换后的网络通道向接收设备发送第 n 个数据元对应的未传输数据。

需要说明的是，发送设备通过确认机制确定接收设备接收到了第 n 个数据元的部分数据时，确定第 n 个数据元的状态为不完整状态；另外，发送设备在利用确认机制确定第 n 个数据元的状态时，还确定了第 n 个数据元的剩余未发送的数据，这里称为未传输数据；从而，此时，发送设备通过切换后的网络通道向接收设备发送第 n 个数据元对应的未传输数据。

继续参见图 6c，如图 6c 所示，在本申请实施例中，S402 之后还包括 S415；也就是说，发送设备获取第 n 个数据元的交付类型之后之前，该数据处理方法还包括 S415，下面对该步骤进行说明。

S415、当第 n 个数据元的交付类型为可丢类型时，从交付顺序在第 n 个数据元之后的未发送数据元中，选择待处理数据元，其中，待处理数据元与第 n 个数据元在数据类型上相同，且待处理数据元在交付顺序上与第 n 个数据元最近；基于独立数据元对待处理数据元进行处理，得到待发送数据元；向接收设备发送待发送数据元。

需要说明的是，发送设备在进行了网络通道的切换之后，如果确定第 n 个数据元的交付类型为可丢类型，可以在交付顺序在第 n 个数据元之后的未发送的数据元中，将与第 n 个数据元在数据类型上相同、且交付顺序最近的数据元，处理为独立数据元，以使接收设备根据接收到的后序数据元中包括目标数据元，进而丢弃第 n 个数据元。

相应地，在本申请实施例中，接收设备获得的后序数据元中的目标数据元，可以是发送设备直接发送的，还可以是发送设备执行针对性处理后再发送的，本申请实施例对此不作限定；其中，针对性处理用于将与第 n 个数据元在数据类型上相同、且交付顺序最近的数据元处理为独立数据元。

可以理解的是，在多网络通道的传输过程中，发送设备在执行了网络通道的切换时，能够主动生成与第 n 个数据元在数据类型上相同的独立数据元，作为接收设备丢弃第 n

个数据元的丢弃信号,使得接收设备丢掉第 n 个数据元并继续交付后序数据元,降低了数据处理的重复确认时长,提高了多网络通道传输过程中的数据处理效率。

在本申请实施例中,数据元包括视频帧数据,视频帧数据包括帧内预测帧数据和帧间预测帧数据,其中,帧内预测帧数据为独立数据元,且帧内预测帧数据的交付类型为可丢类型,以及帧内预测帧数据在后处理时基于自身数据渲染帧图像,帧间预测帧数据为关联数据元,且帧间预测帧数据的交付类型为可丢类型,以及帧间预测帧数据在后处理时基于自身数据和帧内预测帧数据渲染帧图像,帧内预测帧数据和帧间预测帧数据在数据类型上相同。

需要说明的是,当通过多网络通道传输的数据为视频数据时,数据元包括视频帧数据;因此,当第 n 个数据元为可丢类型时,表明第 n 个数据元为帧间预测帧数据,比如,单向预测(Prediction, P)帧数据,双向预测(Bi-prediction, B)帧数据;从而,后序数据元所包括的完整状态的、且与第 n 个数据元在数据类型上相同的独立数据元为帧内预测帧数据,比如, I 帧数据。

在本申请实施例中,数据元还包括音频帧数据,此时,通过多网络通道传输的数据为音视频数据。这里,音频帧数据为独立数据元,且音频帧数据的交付类型为可丢类型,以及音频帧数据和视频帧数据在数据类型上不同。即后序数据元所包括的完整状态的、且与第 n 个数据元在数据类型上不同的独立数据元为音频帧数据。

参见图 6f,图 6f 是本申请实施例提供的数据处理方法的流程图四;如图 6f 所示,在本申请实施例中,接收设备通过至少两个网络通道接收发送设备发送的数据元之后,该数据处理方法还包括 S416 和 S417,下面对各步骤分别进行说明。

S416、当接收设备接收到的数据元与第一数据元集合中的数据元在交付顺序上相邻时,将接收到的数据元确定为第一数据元集合中的数据元,并从第二数据元集合中,获取与接收到的数据元在交付顺序上相邻的至少一个数据元,以及将相邻的至少一个数据元确定为第一数据元集合中的数据元。

S417、当接收设备接收到的数据元与第一数据元集合中的数据元在交付顺序上不相邻时,将接收到的数据元确定为第二数据元集合中的数据元。

需要说明的是,接收设备将通过多网络通道接收的数据元分类存储,一类存储的为待交付的且在交付顺序上相邻(称为连续)的数据元,另一类存储的为与第一数据元集合中的数据元在交付顺序上不相邻(称为不连续的)的数据元。

相应地,在本申请实施例中,S404 中接收设备基于交付顺序,获取接收到的第 n 个数据元的接收到的后序数据元,包括:接收设备基于交付顺序,从第一数据元集合和第二数据元集合中,获取接收到的第 n 个数据元的后序数据元。

需要说明的是,接收设备按照交付顺序从第一数据元集合中获取与最近交付的数据元在交付顺序上相邻的数据元。如果交付过程中确定存在数据元的状态为不完整状态,则从第一数据元集合和第二数据元集合中获取接收到的后序数据元,以确定该不完整状态的数据元是等待接收数据还是丢弃。

示例性地,参见图 7,图 7 是本申请实施例提供的示例性的接收到的数据元的存储示意图;如图 7 所示,当第一数据元集合和第二数据元集合均通过队列存储时,队列 7-1 用于存储第一数据元集合中的数据元,队列 7-2 用于存储第二数据元集合中的数据元;这里,当数据元 7-31 至数据元 7-36 依次对应的交付顺序为 1 至 6,以及数据元 7-31 至数据元 7-33 通过网络通道 A 传输,数据元 7-34 至数据元 7-36 通过网络通道 B 传输,而网络通道 B 的传输速度快于网络通道 A 的传输速度时,接收设备先接收到了数据元 7-31 和数据元 7-32、以及数据元 7-34 至数据元 7-36;此时,接收设备将数据元 7-31 和数据元 7-32 存储至队列 7-1 中,而将数据元 7-34 至数据元 7-36 存储至队列 7-2 中;后

面接收到数据元 7-33 时, 将数据元 7-33 存储至队列 7-1 中, 并确定队列 7-2 中的数据元 7-34 至数据元 7-36 与数据元 7-33 在交付顺序上相邻, 从而, 将队列 7-2 中的数据元 7-34 至数据元 7-36 移至队列 7-1 中存储。其中, 头节点用于指向队列中的第一个数据元。

可以理解的是, 本申请实施例中接收设备通过第一数据元集合和第二数据元集合接收通过多网络通道传输过来的数据元, 由于数据元包括交付顺序, 需要按照交付顺序依次交付数据元; 因此, 接收设备将接收到的数据元分连续和不连续进行存储, 使得交付时从第一数据元集合中能够快速获得待交付的数据元, 提升数据处理的效率。

下面, 将说明本申请实施例在一个实际的应用场景中的示例性应用。该示例性应用描述了云游戏应用场景中, 云端游戏服务器 (比如, 图 1 中的服务器 300) 通过多网络通道向用户客户端 (用户客户端可以是运行在终端中的客户端, 其中, 终端比如为图 1 中的终端 200) 传输云游戏音视频数据的过程。

需要说明的是, 云游戏是通过大于指定码率的码率传输音视频数据, 以实现虚拟场景画面的呈现, 来形成如同本地游戏一样的沉浸式体验, 因此, 云游戏的效果取决于音视频传输和显示的流畅度。

参见图 8, 图 8 是本申请实施例提供的云游戏应用场景的示意图; 如图 8 所示, 云端游戏服务器 8-1 接收用户客户端 8-2 发送的游戏指令 8-3, 并响应于游戏指令 8-3, 向用户客户端 8-2 发送音视频数据 8-4 (称为传输的数据)。

参见图 9, 图 9 是本申请实施例提供的示例性的数据处理的流程示意图一; 如图 9 所示, 该示例性的数据处理的流程包括 S901 至 S912, 执行主体为用户客户端, 下面对

S901、判断待交付数据元 (称为第 n 个数据元) 是否完整。如果是则执行 S902, 如果否则执行 S903。

需要说明的是, 该示例性的数据处理开始时, 执行 S901。这里, 用户客户端通过两个队列 (比如, 图 7 中的队列 7-1 和队列 7-2) 接收云端游戏服务器通过多网络通道发送来的数据元。从而, 用户客户端先从存储了与最近交付的数据元在交付顺序上相邻的数据元的队列中 (比如, 图 7 中的队列 7-1), 获取交付顺序最近的一个数据元, 也就得到了待交付数据元; 并判断待交付数据元是否完整 (即为检测数据元的状态), 以基于完整性的判断结果确定是否交付该待交付数据元。

S902、交付待交付数据元。之后从队列中删除待交付数据元, 并结束交付待交付数据元的数据处理流程。

需要说明的是, 用户客户端通过对交付后的数据元进行解码, 实现音视频的播放。

S903、判断待交付数据元是否是可丢类型。如果否则执行 S904, 如果是则执行 S905。

需要说明的是, 发送设备在通过多网络通道传输数据元时, 将数据元对应的类型和交付顺序携带在了数据元中。其中, 数据元的类型包括可丢类型和非可丢类型, 当通过多网络通道进行数据传输的数据元为帧数据时, 如果视频帧包括 I 帧和 P 帧, 则数据元为 I 帧数据 (称为帧内预测帧数据) 和音频帧数据时均是非可丢类型, 数据元为 P 帧数据 (帧间预测帧数据) 是可丢类型; 这是因为 I 帧数据可以直接解码获得一个完整的画面, 而 P 帧数据和 B 帧数据则需要前后参考信息才可以获得完整画面, 因此, 在接收到完整 I 帧数据时可以放弃之前的 P 帧数据和 B 帧数据, 通过解码 I 帧数据获得游戏画面。

S904、等待接收待交付数据元的未传输数据。进而根据接收到的数据执行 S901。

需要说明的是, 如果不完整的待交付数据元为 I 帧数据或音频帧数据, 则不完整的待交付数据元是不可丢类型; 由于 I 帧数据丢失后影响后面数据元的解码 (称为后处理), 音频帧数据丢失后影响音频的渲染, 因此, 不能丢弃, 而是继续等待接收数据。

S905、依次获取后面的数据元 (称为后序数据元)。之后执行 S906 和 S912。

需要说明的是，用户客户端通过扫描两个队列中的数据元，获取交付顺序最近的一个数据元之后的数据元，也就得到了后面的数据元。

S906、判断是否获取到不完整的非可丢类型的数据元。如果是则执行 S907，如果否则执行 S908。

5 这里，用户客户端扫描队列获取后面的数据元的过程中，如果未扫描到不完整的非可丢类型的数据元，而首先扫描到了一个完整的数据元，则判断该数据元的类型；如果扫描到了不完整的非可丢类型的数据元，停止扫描，等待接收数据。

S907、等待接收数据。并基于接收的数据执行 S908。

S908、确定获取到的第一个完整的数据元的类型。

10 S909、类型为 I 帧时，丢弃后面的第一个完整的数据元之前的数据元，交付后面的第一个完整的数据元。之后从队列中删除待交付数据元，并结束交付待交付数据元的数据处理流程。

需要说明的是，类型包括数据类型上是否相同的结果和是独立数据元还是关联数据元的结果。类型为 I 帧，表明后面的第一个完整的数据元是指与待交付数据元（P 帧数据）在数据类型上相同的独立数据元，交付该后面的第一个完整的数据元，并丢弃该后面的第一个完整的数据元之前的数据元。

S910、类型为 P 帧或音频帧时，继续获取再后面的数据元。

需要说明的是，类型为 P 帧，表明后面的第一个完整的数据元是指与待交付数据元（P 帧数据）在数据类型上相同的关联数据元；类型为音频帧，表明后面的第一个完整的数据元是指与待交付数据元（P 帧数据）在数据类型上不同的独立数据元；此时，用户客户端还需要进一步判断再后面的数据元。

S911、当确定再后面的数据元包括完整的类型为 I 帧的数据元时，丢弃待交付数据元。结束交付待交付数据元的数据处理流程。

25 S912、当确定后面的数据元均不完整，等待接收数据。进而根据接收到的数据执行 S908。

相应地，在本申请实施例中，云端游戏服务器在切换网络通道时，先确定切换的网络通道所传输的数据元的交付类型为可丢类型，即确定是 P 帧数据，不再进行确认重传，而是在确定后面的数据元包括完整的类型为 I 帧的数据元时，直接发送后面的数据元。而如果云端游戏服务器确定切换的网络通道所传输的数据元的交付类型为非可丢类型，即确定是 I 帧数据或音频帧数据时，执行确认重传处理。

30 示例性地，参见图 10，图 10 是本申请实施例提供的示例性的数据处理的流程示意图二；如图 10 所示，该示例性的数据处理的流程包括 S1001 至 S1007，执行主体为云端游戏服务器，下面对各步骤分别进行说明。

S1001、获取切换指令。

35 需要说明的是，网络通道的切换可以是云端游戏服务器侧主动触发的，此时，获取到的切换指令是由云端游戏服务器侧自身生成的；也就是说，网络通道的切换是云端游戏服务器自身的决策结果。

S1002、响应于切换指令，确定待切换的网络通道中正传输的数据元（称为第 n 个数据元）。

40 S1003、判断正传输的数据元是否是可丢类型。如果是则执行 S1004，如果否则执行 S1005。

S1004、当确定后面的数据元包括完整的 I 帧数据时，直接传输交付顺序在正传输的数据元之后的待传输的数据元。

需要说明的是，后面的数据元中所包括的完整的 I 帧数据，可以是云端游戏服务器

在切换网络通道时，不再继续采用 P 帧数据的编码方式进行，而是及时地通过 I 帧数据的编码方式进行视频帧编码获得的。如此，实现主动发送 I 帧数据触发用户客户端侧丢弃不完整的 P 帧数据，使得用户客户端能够主动结束画面卡顿，从而，降低画面卡顿时长。

- 5 可以理解的是，云端游戏服务器在网络通道切换时不再进行确认重传，减少了确认和重传的资源消耗。

S1005、确认正传输的数据元中需要重传的数据。

S1006、重传需要重传的数据。

S1007、传输交付顺序在正传输的数据元之后的待传输的数据元。

- 10 可以理解的是，用户客户端和云端游戏服务器通过丢弃不完整的 P 帧数据，能够提高云游戏应场景在复杂网络环境下采用多网络通道传输的音视频的稳定性和流畅性。也就是说，通过对不同数据形成不同的处理策略（丢弃或等待），从而简化双端在网络通道切换时的收发逻辑，减少网络通道切换时恢复画面所需的时长，提升数据处理效率。

- 下面继续说明本申请实施例提供的第一数据处理装置 255 的实施为软件模块的示例性结构，在一些实施例中，如图 2 所示，存储在所述第一存储器 250 的第一数据处理装置 255 中的软件模块可以包括：

数据接收模块 2551，配置为通过至少两个网络通道接收发送设备发送的数据元，其中，所述数据元包括交付顺序；

- 20 交付处理模块 2552，配置为当接收到的第 n 个所述数据元为待判断数据元时，基于所述交付顺序，获取接收到的第 n 个所述数据元的后序数据元，其中，所述待判断数据元的状态为不完整状态，且所述待判断数据元的交付类型为可丢类型，所述状态是基于是否接收到所述数据元的全部数据确定的，所述交付类型基于以下两种信息确定：后处理时被其他所述数据元依赖的程度和对后处理结果的影响程度，n 为正整数；

- 25 交付处理模块 2552，还配置为当所述后序数据元包括目标数据元时，确定第 n 个所述数据元的交付结果为丢弃，其中，所述目标数据元的所述状态为完整状态，且所述目标数据元与第 n 个所述数据元在数据类型上相同，以及所述目标数据元为独立数据元，所述数据类型是基于所述数据元的作用确定的，所述独立数据元在后处理时对其他所述数据元的依赖程度小于指定程度。

- 30 在本申请实施例中，所述交付处理模块 2552，还配置为当所述后序数据元中的第 n+1 个所述数据元被接收到全部数据时，确定所述第 n+1 个所述数据元的所述状态为所述完整状态；当处于所述完整状态的第 n+1 个所述数据元满足第一条件时，确定所述后序数据元包括所述目标数据元，其中，满足所述第一条件包括：与第 n 个所述数据元在所述数据类型上相同，并且属于所述独立数据元。

- 35 在本申请实施例中，所述交付处理模块 2552，还配置为当处于所述完整状态的第 n+1 个所述数据元满足第二条件时，或者，当处于所述完整状态的第 n+1 个所述数据元满足第三条件时，获取所述后序数据元中的第 n+2 个所述数据元的所述状态，其中，满足所述第二条件包括：与第 n 个所述数据元在所述数据类型上不同，并且属于所述独立数据元；满足所述第三条件包括：与第 n 个所述数据元在所述数据类型上相同，并且属于关联数据元；所述关联数据元在后处理时对其他所述数据元的依赖程度大于等于所述指定程度；当第 n+2 个所述数据元的所述状态为所述完整状态、且第 n+2 个所述数据元满足所述第一条件时，确定所述后序数据元包括所述目标数据元。

- 40 在本申请实施例中，所述交付处理模块 2552，还配置为当所述后序数据元中的第 n+1 个所述数据元为所述待判断数据元时，获取所述交付顺序在第 n+1 个所述数据元之后的所述数据元，并基于获得的所述数据元确定第 n 个所述数据元和所述第 n+1 个所述

数据元的所述交付结果。

在本申请实施例中，所述交付处理模块 2552，还配置为当所述后序数据元不包括所述目标数据元时，确定第 n 个所述数据元的所述交付结果为等待接收数据。

在本申请实施例中，所述交付处理模块 2552，还配置为当接收到的第 n 个所述数据元为等待数据元时，确定第 n 个所述数据元的所述交付结果为等待接收数据，其中，所述等待数据元的所述状态为所述不完整状态，且所述等待数据元的所述交付类型为非可丢类型。

在本申请实施例中，所述数据元包括视频帧数据，所述视频帧数据包括帧内预测帧数据和帧间预测帧数据，其中，所述帧内预测帧数据为所述独立数据元，且所述帧内预测帧数据的所述交付类型为非可丢类型，以及所述帧内预测帧数据在后处理时基于自身数据渲染帧图像，所述帧间预测帧数据为关联数据元，且所述帧间预测帧数据的所述交付类型为所述可丢类型，以及所述帧间预测帧数据在后处理时基于自身数据和所述帧内预测帧数据渲染帧图像，所述帧内预测帧数据和所述帧间预测帧数据在所述数据类型上相同。

在本申请实施例中，所述数据元还包括音频帧数据，其中，所述音频帧数据为所述独立数据元，且所述音频帧数据的所述交付类型为所述非可丢类型，以及所述音频帧数据和所述视频帧数据在所述数据类型上不同。

在本申请实施例中，所述数据接收模块 2551，还配置为当接收到的所述数据元与最近交付的所述数据元在所述交付顺序上相邻时，将接收到的所述数据元确定为第一数据元集合中的所述数据元，并从第二数据元集合中，获取与接收到的所述数据元在所述交付顺序上相邻的至少一个所述数据元，以及将相邻的至少一个所述数据元确定为所述第一数据元集合中的所述数据元；当接收到的所述数据元与最近交付的所述数据元在所述交付顺序上不相邻时，将接收到的所述数据元确定为所述第二数据元集合中的所述数据元。

在本申请实施例中，所述数据接收模块 2551，还配置为基于所述交付顺序，从所述第一数据元集合和所述第二数据元集合中，获取接收到的第 n 个所述数据元的所述后序数据元。

本申请实施例提供了一种计算机程序产品或计算机程序，该计算机程序产品或计算机程序包括计算机指令，该计算机指令存储在计算机可读存储介质中。接收设备的第一处理器从计算机可读存储介质读取该计算机指令，第一处理器执行该计算机指令，使得该接收设备执行本申请实施例上述的接收设备侧的数据处理方法。

下面继续说明本申请实施例提供的第二数据处理装置 355 的实施为软件模块的示例性结构，在一些实施例中，如图 3 所示，存储在第二存储器 350 的第二数据处理装置 355 中的软件模块可以包括：

数据发送模块 3551，配置为通过至少两个网络通道向接收设备发送数据元，其中，所述数据元包括交付顺序；

通道切换模块 3552，配置为在所述数据元的发送过程中，响应于切换指令，获取第 n 个所述数据元的交付类型，其中，所述切换指令用于对传输第 n 个所述数据元的所述网络通道进行切换，所述交付类型基于以下两种信息确定：后处理时被其他所述数据元依赖的程度和对后处理结果的影响程度， n 为正整数；

切换处理模块 3553，配置为当第 n 个所述数据元的所述交付类型为可丢类型，且后序数据元包括目标数据元时，取消向所述接收设备发送第 n 个所述数据元，其中，所述后序数据元为已发送的所述交付顺序在第 n 个所述数据元之后的所述数据元，所述目标数据元的状态为完整状态，且所述目标数据元与第 n 个所述数据元在数据类型上相同，

以及所述目标数据元为独立数据元,所述状态是基于所述接收设备是否接收到所述数据元的全部数据确定的,所述数据类型是基于所述数据元的作用确定的,所述独立数据元在后处理时对其他所述数据元的依赖程度小于指定程度。

在本申请实施例中,所述切换处理模块 3553,还配置为当第 n 个所述数据元的所述交付类型为可丢类型时,获取第 n 个所述数据元的所述状态;当第 n 个所述数据元的所述状态为不完整状态时,通过切换后的所述网络通道向所述接收设备发送第 n 个所述数据元对应的未传输数据。

在本申请实施例中,所述切换处理模块 3553,还配置为当第 n 个所述数据元的所述交付类型为可丢类型时,从所述交付顺序在第 n 个所述数据元之后的未发送数据元中,选择待处理数据元,其中,所述待处理数据元与第 n 个所述数据元在数据类型上相同,且所述待处理数据元在所述交付顺序上与第 n 个所述数据元最近;基于所述独立数据元对所述待处理数据元进行处理,得到待发送数据元;向所述接收设备发送所述待发送数据元。

本申请实施例提供了一种计算机程序产品或计算机程序,该计算机程序产品或计算机程序包括计算机指令,该计算机指令存储在计算机可读存储介质中。发送设备的第二处理器从计算机可读存储介质读取该计算机指令,第二处理器执行该计算机指令,使得该发送设备执行本申请实施例上述的发送设备侧的数据处理方法。

本申请实施例提供一种存储有计算机可执行指令的计算机可读存储介质,其中存储有计算机可执行指令,当计算机可执行指令被第一处理器执行时,将引起第一处理器执行本申请实施例提供的接收设备侧的数据处理方法;当计算机可执行指令被第二处理器执行时,将引起第二处理器执行本申请实施例提供的发送设备侧的数据处理方法;例如,如图 4 示出的数据处理方法。

在一些实施例中,计算机可读存储介质可以是 FRAM、ROM、PROM、EPROM、EEPROM、闪存、磁表面存储器、光盘、或 CD-ROM 等存储器;也可以是包括上述存储器之一或任意组合的各种设备。

在一些实施例中,计算机可执行指令可以采用程序、软件、软件模块、脚本或代码的形式,按任意形式的编程语言(包括编译或解释语言,或者声明性或过程性语言)来编写,并且其可按任意形式部署,包括被部署为独立的程序或者被部署为模块、组件、子例程或者适合在计算环境中使用的其它单元。

作为示例,计算机可执行指令可以但不一定对应于文件系统中的文件,可以可被存储在保存其它程序或数据的文件的一部分,例如,存储在超文本标记语言(HTML, Hyper Text Markup Language)文档中的一个或多个脚本中,存储在专用于所讨论的程序的单个文件中,或者,存储在多个协同文件(例如,存储一个或多个模块、子程序或代码部分的文件)中。

作为示例,计算机可执行指令可被部署为在一个电子设备上执行(此时,这一个电子设备即接收设备或发送设备),或者在位于一个地点的多个电子设备上执行(此时,位于一个地点的多个电子设备即接收设备或发送设备),又或者,在分布在多个地点且通过通信网络互连的多个电子设备上执行(此时,分布在多个地点且通过通信网络互连的多个电子设备即接收设备或发送设备)。

可以理解的是,在本申请实施例中,涉及到事件等相关的数据,当本申请实施例运用到具体产品或技术中时,需要获得用户许可或者同意,且相关数据的收集、使用和处理需要遵守相关国家和地区的相关法律法规和标准。

综上所述,本申请实施例在多网络通道传输的过程中,当因网络通道切换导致接收端当前接收的数据元接收了数据元的部分数据时,如果当前接收的数据元为可丢类型,

表明当前接收的数据元在后处理时被其他数据元依赖的程度和对后处理结果的影响程度均较低；接着如果确定当前接收的数据元的交付顺序之后的后序数据元中，包括能够独立完成与当前接收的数据元作用相同的后处理的数据元时，发送设备直接进行后序数据元的发送，接收设备将当前接收的数据元进行丢弃处理，如此，降低了发送设备确认重传的时长和接收设备等待数据的时长；因此，能够降低网络通道切换时的数据元传输时间成本和网络通道切换的复杂度，提升多网络通道传输的过程中的数据元处理效率；另外，当传输的数据元为音频帧数据和视频帧数据、又或者为视频帧数据时，能够提升视频画面的渲染流畅度，提升视频或音视频的播放效果。

以上所述，仅为本申请的实施例而已，并非用于限定本申请的保护范围。凡在本申请的精神和范围之内所作的任何修改、等同替换和改进等，均包含在本申请的保护范围之内。

权利要求书

- 1、一种数据处理方法，所述方法由接收设备执行，所述方法包括：
通过至少两个网络通道接收发送设备发送的数据元，其中，所述数据元包括交付顺序；
- 5 当接收到的第 n 个所述数据元为待判断数据元时，基于所述交付顺序，获取接收到的第 n 个所述数据元的后序数据元，其中，所述待判断数据元的状态为不完整状态，且所述待判断数据元的交付类型为可丢类型，所述状态是基于是否接收到所述数据元的全部数据确定的，所述交付类型基于以下两种信息确定：后处理时被其他所述数据元依赖的程度和对后处理结果的影响程度， n 为正整数；
- 10 当所述后序数据元包括目标数据元时，确定第 n 个所述数据元的交付结果为丢弃，其中，所述目标数据元的所述状态为完整状态，且所述目标数据元与第 n 个所述数据元在数据类型上相同，以及所述目标数据元为独立数据元，所述数据类型是基于所述数据元的作用确定的，所述独立数据元在后处理时对其他所述数据元的依赖程度小于指定程度。
- 15 2、根据权利要求 1 所述的方法，其中，所述基于所述交付顺序，获取接收到的第 n 个所述数据元的后序数据元之后，所述方法还包括：
当所述后序数据元中的第 $n+1$ 个所述数据元被接收到全部数据时，确定所述第 $n+1$ 个所述数据元的所述状态为所述完整状态；
当处于所述完整状态的第 $n+1$ 个所述数据元满足第一条件时，确定所述后序数据元
- 20 包括所述目标数据元，其中，满足所述第一条件包括：与第 n 个所述数据元在所述数据类型上相同，并且属于所述独立数据元。
- 3、根据权利要求 2 所述的方法，其中，所述确定所述第 $n+1$ 个所述数据元的所述状态为所述完整状态之后，所述方法还包括：
当处于所述完整状态的第 $n+1$ 个所述数据元满足第二条件时，或者，当处于所述完
- 25 整状态的第 $n+1$ 个所述数据元满足第三条件时，获取所述后序数据元中的第 $n+2$ 个所述数据元的所述状态，其中，满足所述第二条件包括：与第 n 个所述数据元在所述数据类型上不同，并且属于所述独立数据元；满足所述第三条件包括：与第 n 个所述数据元在所述数据类型上相同，并且属于关联数据元；所述关联数据元在后处理时对其他所述数据元的依赖程度大于等于所述指定程度；
- 30 当第 $n+2$ 个所述数据元的所述状态为所述完整状态、且第 $n+2$ 个所述数据元满足所述第一条件时，确定所述后序数据元包括所述目标数据元。
- 4、根据权利要求 1 所述的方法，其中，所述基于所述交付顺序，获取接收到的第 n 个所述数据元的后序数据元之后，所述方法还包括：
当所述后序数据元中的第 $n+1$ 个所述数据元为所述待判断数据元时，获取所述交付
- 35 顺序在第 $n+1$ 个所述数据元之后的所述数据元，并基于获得的所述数据元确定第 n 个所述数据元和所述第 $n+1$ 个所述数据元的所述交付结果。
- 5、根据权利要求 1 所述的方法，其中，所述基于所述交付顺序，获取接收到的第 n 个所述数据元的后序数据元之后，所述方法还包括：
当所述后序数据元不包括所述目标数据元时，确定第 n 个所述数据元的所述交付结
- 40 果为等待接收数据。
- 6、根据权利要求 1 所述的方法，其中，所述通过至少两个网络通道接收发送设备发送的数据元之后，所述方法还包括：

当接收到的第 n 个所述数据元为等待数据元时, 确定第 n 个所述数据元的所述交付结果为等待接收数据, 其中, 所述等待数据元的所述状态为所述不完整状态, 且所述等待数据元的所述交付类型为非可丢类型。

5 7、根据权利要求 1 至 6 任一项所述的方法, 其中, 所述数据元包括视频帧数据, 所述视频帧数据包括帧内预测帧数据和帧间预测帧数据, 其中, 所述帧内预测帧数据为所述独立数据元, 且所述帧内预测帧数据的所述交付类型为非可丢类型, 以及所述帧内预测帧数据在后处理时基于自身数据渲染帧图像, 所述帧间预测帧数据为关联数据元, 且所述帧间预测帧数据的所述交付类型为所述可丢类型, 以及所述帧间预测帧数据在后处理时基于自身数据和所述帧内预测帧数据渲染帧图像, 所述帧内预测帧数据和所述帧间预测帧数据在所述数据类型上相同。

8、根据权利要求 7 所述的方法, 其中, 所述数据元还包括音频帧数据, 其中, 所述音频帧数据为所述独立数据元, 且所述音频帧数据的所述交付类型为所述非可丢类型, 以及所述音频帧数据和所述视频帧数据在所述数据类型上不同。

9、根据权利要求 1 至 6 任一项所述的方法, 其中, 所述通过至少两个网络通道接收发送设备发送的数据元之后, 所述方法还包括:

当接收到的所述数据元与最近交付的所述数据元在所述交付顺序上相邻时, 将接收到的所述数据元确定为第一数据元集合中的所述数据元, 并从第二数据元集合中, 获取与接收到的所述数据元在所述交付顺序上相邻的至少一个所述数据元, 以及将相邻的至少一个所述数据元确定为所述第一数据元集合中的所述数据元;

20 当接收到的所述数据元与最近交付的所述数据元在所述交付顺序上不相邻时, 将接收到的所述数据元确定为所述第二数据元集合中的所述数据元;

所述基于所述交付顺序, 获取接收到的第 n 个所述数据元的后序数据元, 包括:

基于所述交付顺序, 从所述第一数据元集合和所述第二数据元集合中, 获取接收到的第 n 个所述数据元的所述后序数据元。

25 10、一种数据传输方法, 所述方法由发送设备执行, 所述方法包括:

通过至少两个网络通道向接收设备发送数据元, 其中, 所述数据元包括交付顺序;

30 在所述数据元的发送过程中, 响应于切换指令, 获取第 n 个所述数据元的交付类型, 其中, 所述切换指令用于对传输第 n 个所述数据元的所述网络通道进行切换, 所述交付类型基于以下两种信息确定: 后处理时被其他所述数据元依赖的程度和对后处理结果的影响程度, n 为正整数;

35 当第 n 个所述数据元的所述交付类型为可丢类型, 且后序数据元包括目标数据元时, 取消向所述接收设备发送第 n 个所述数据元, 其中, 所述后序数据元为已发送的所述交付顺序在第 n 个所述数据元之后的所述数据元, 所述目标数据元的状态为完整状态, 且所述目标数据元与第 n 个所述数据元在数据类型上相同, 以及所述目标数据元为独立数据元, 所述状态是基于所述接收设备是否接收到所述数据元的全部数据确定的, 所述数据类型是基于所述数据元的作用确定的, 所述独立数据元在后处理时对其他所述数据元的依赖程度小于指定程度。

11、根据权利要求 10 所述的方法, 其中, 所述获取第 n 个所述数据元的交付类型之后, 所述方法还包括:

40 当第 n 个所述数据元的所述交付类型为非可丢类型时, 获取第 n 个所述数据元的所述状态;

当第 n 个所述数据元的所述状态为不完整状态时, 通过切换后的所述网络通道向所述接收设备发送第 n 个所述数据元对应的未传输数据。

12、根据权利要求 10 所述的方法，其中，所述获取第 n 个所述数据元的交付类型之后，所述方法还包括：

当第 n 个所述数据元的所述交付类型为可丢类型时，从所述交付顺序在第 n 个所述数据元之后的未发送数据元中，选择待处理数据元，其中，所述待处理数据元与第 n 个所述数据元在数据类型上相同，且所述待处理数据元在所述交付顺序上与第 n 个所述数据元最近；

基于所述独立数据元对所述待处理数据元进行处理，得到待发送数据元；

向所述接收设备发送所述待发送数据元。

13、一种第一数据处理装置，所述第一数据处理装置包括：

数据接收模块，配置为通过至少两个网络通道接收发送设备发送的数据元，其中，所述数据元包括交付顺序；

交付处理模块，配置为当接收到的第 n 个所述数据元为待判断数据元时，基于所述交付顺序，获取接收到的第 n 个所述数据元的后序数据元，其中，所述待判断数据元的状态为不完整状态，且所述待判断数据元的交付类型为可丢类型，所述状态是基于是否接收到所述数据元的全部数据确定的，所述交付类型基于以下两种信息确定：后处理时被其他所述数据元依赖的程度和对后处理结果的影响程度，n 为正整数；

交付处理模块，还配置为当所述后序数据元包括目标数据元时，确定第 n 个所述数据元的交付结果为丢弃，其中，所述目标数据元的所述状态为完整状态，且所述目标数据元与第 n 个所述数据元在数据类型上相同，以及所述目标数据元为独立数据元，所述数据类型是基于所述数据元的作用确定的，所述独立数据元在后处理时对其他所述数据元的依赖程度小于指定程度。

14、一种第二数据处理装置，所述第二数据处理装置包括：

数据发送模块，配置为通过至少两个网络通道向接收设备发送数据元，其中，所述数据元包括交付顺序；

通道切换模块，配置为在所述数据元的发送过程中，响应于切换指令，获取第 n 个所述数据元的交付类型，其中，所述切换指令用于对传输第 n 个所述数据元的所述网络通道进行切换，所述交付类型基于以下两种信息确定：后处理时被其他所述数据元依赖的程度和对后处理结果的影响程度，n 为正整数；

切换处理模块，配置为当第 n 个所述数据元的所述交付类型为可丢类型，且后序数据元包括目标数据元时，取消向所述接收设备发送第 n 个所述数据元，其中，所述后序数据元为已发送的所述交付顺序在第 n 个所述数据元之后的所述数据元，所述目标数据元的状态为完整状态，且所述目标数据元与第 n 个所述数据元在数据类型上相同，以及所述目标数据元为独立数据元，所述状态是基于所述接收设备是否接收到所述数据元的全部数据确定的，所述数据类型是基于所述数据元的作用确定的，所述独立数据元在后处理时对其他所述数据元的依赖程度小于指定程度。

15、一种用于数据处理的接收设备，所述接收设备包括：

第一存储器，用于存储计算机可执行指令；

第一处理器，用于执行所述第一存储器中存储的计算机可执行指令时，实现权利要求 1 至 9 任一项所述的数据处理方法。

16、一种用于数据处理的发送设备，所述发送设备包括：

第二存储器，用于存储计算机可执行指令；

第二处理器，用于执行所述第二存储器中存储的计算机可执行指令时，实现权利要求 10 至 12 任一项所述的数据处理方法。

17、一种计算机可读存储介质，存储有计算机可执行指令，其中，所述计算机可执行指令用于被第一处理器执行时，实现权利要求 1 至 9 任一项所述的方法；或者，所述计算机可执行指令用于被第二处理器执行时，实现权利要求 10 至 12 任一项所述的数据处理方法。

- 5 18、一种计算机程序产品，包括计算机程序或计算机可执行指令，所述计算机程序或计算机可执行指令被第一处理器执行时，实现权利要求 1 至 9 任一项所述的数据处理方法；或者，所述计算机程序或计算机可执行指令被第二处理器执行时，实现权利要求 10 至 12 任一项所述的数据处理方法。