

(12) 按照专利合作条约所公布的国际申请

(19) 世界知识产权组织
国际局

(43) 国际公布日
2019年5月31日 (31.05.2019)



(10) 国际公布号
WO 2019/100912 A1

- (51) 国际专利分类号:
H04L 7/00 (2006.01)
- (21) 国际申请号: PCT/CN2018/112595
- (22) 国际申请日: 2018年10月30日 (30.10.2018)
- (25) 申请语言: 中文
- (26) 公布语言: 中文
- (30) 优先权:
201711189288.1 2017年11月24日 (24.11.2017) CN
- (71) 申请人: 华为技术有限公司 (HUAWEI TECHNOLOGIES CO., LTD.) [CN/CN]; 中国广东省深圳市龙岗区坂田华为总部办公楼, Guangdong 518129 (CN)。

- (72) 发明人: 李从娟(LI, Congjuan); 中国广东省深圳市龙岗区坂田华为总部办公楼, Guangdong 518129 (CN)。 李晋(LI, Jin); 中国广东省深圳市龙岗区坂田华为总部办公楼, Guangdong 518129 (CN)。 李峰(LI, Feng); 中国广东省深圳市龙岗区坂田华为总部办公楼, Guangdong 518129 (CN)。
- (74) 代理人: 北京三高永信知识产权代理有限责任公司(BEIJING SAN GAO YONG XIN INTELLECTUAL PROPERTY AGENCY CO., LTD.); 中国北京市海淀区学院路蓟门里和景园A座1单元102室, Beijing 100088 (CN)。
- (81) 指定国(除另有指明, 要求每一种可提供的国家保护): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB,

(54) Title: DATA DISTRIBUTION METHOD AND DISTRIBUTION SERVER

(54) 发明名称: 数据分发方法以及分发服务器

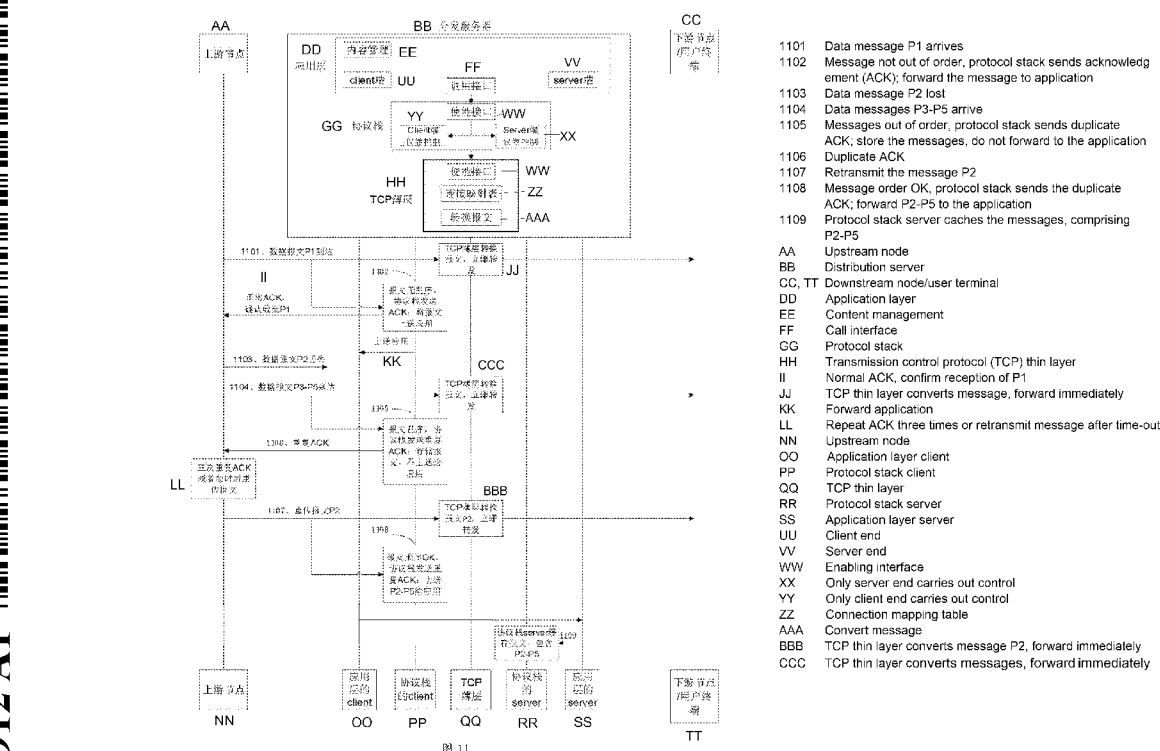


图 11

(57) Abstract: The present disclosure belongs to the field of network technology, and provided thereby are a data distribution method and distribution server. The method comprises: continuously receiving first upstream data message and a second upstream data message sent by an upstream node; copying the first upstream data message, converting a first message among two messages obtained by copying according to an established connection mapping table, sending a first downstream data message obtained by conversion to a downstream node or a user terminal, and controlling and managing a second message which is obtained by copying on the basis of

GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW。

(84) 指定国(除另有指明, 要求每一种可提供的地区保护): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), 欧亚 (AM, AZ, BY, KG, KZ, RU, TJ, TM), 欧洲 (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG)。

本国际公布:

— 包括国际检索报告(条约第21条(3))。

the first upstream data message; copying the second upstream data message, converting a first message among two messages obtained by copying according to the connection mapping table, sending a second downstream data message obtained by conversion to the downstream node or the user terminal, and controlling and managing a second message which is obtained by copying on the basis of the second upstream data message; The present disclosure reduces data distribution latency and enables low latency distribution.

(57) 摘要: 本公开提供了一种数据分发方法以及分发服务器, 属于网络技术领域。方法包括: 连续接收上游节点发送的第一上游数据报文和第二上游数据报文; 对第一上游数据报文进行拷贝处理, 根据已建立的连接映射表, 对拷贝得到的两份报文中的第一份报文进行转换处理, 将转换得到的第一下游数据报文发送给下游节点或用户终端, 并对基于第一上游数据报文拷贝得到的第二份报文进行控制管理; 对第二上游数据报文进行拷贝处理, 根据连接映射表, 对拷贝得到的两份报文中的第一份报文进行转换处理, 将转换得到的第二下游数据报文发送给下游节点或用户终端, 并对基于第二上游数据报文拷贝得到的第二份报文进行控制管理。本公开降低了数据分发时延, 实现了低时延分发。

数据分发方法以及分发服务器

本申请要求于 2017 年 11 月 24 日提交的申请号为 201711189288.1、发明名称为“数据分发方法以及分发服务器”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

本公开涉及网络技术领域，特别涉及一种数据分发方法以及分发服务器。

背景技术

分发服务器，常用于内容分发网络（Content Delivery Network, CDN）中。其中，内容分发网络是在传统的电信网络基础之上，使用多级 CDN 节点（如图 1 中的 CDN 中间节点以及 CDN 边缘节点）构建的一个 Overlay 网络。依据内容分发网络的规模不同，每一级 CDN 节点由一个或多个分发服务器组成，而各个分发服务器可基于超文本传输协议（HyperText Transfer Protocol, HTTP）或实时媒体协议（Real Time Media Protocol, RTMP）向用户终端提供诸如视频、图片、文件等数据的数据分发业务。在内容分发网络中，依据分发服务器上是否存储了用户终端请求的数据，分发服务器可提供两种分发服务，即命中服务和中转服务。其中，命中服务发生在分发服务器上存储有用户终端请求的数据的情形下，而中转服务发生在分发服务器上未存储用户终端请求的数据，还需向其他分发服务器或源站进行内容请求中转的情形下。

其中，图 2 示出了一种提供中转服务的分发服务器的内部架构示意图，该分发服务器主要包括协议栈和应用层两大块。如图 2 所示，协议栈负责因特网互联协议（Transmission Control Protocol/Internet Protocol, TCP/IP）相关的协议处理，分为 server（服务端）/client（客户端）两个模块。其中，server 负责处理与用户终端间的 TCP 连接（以下简称为下游连接），包括下游连接的建立、数据报文的分发、可靠性控制管理（比如速率控制）、与应用层的 server 之间的数据报文转交；client 负责处理与上游节点间的 TCP 连接（以下简称为上游连接），包括上游连接的建立、数据报文的接收、可靠性控制管理（比如是否乱序）、与应用层的 client 之间数据报文转交。

另外，应用层也分为 server/client 两个模块。其中，应用层的 server 负责接收用户终端的内容请求，触发应用层的 client 向上游节点请求（get）数据，接收应用层的 client 转交过来的上游节点发送的数据报文，并转交给协议栈的 server，进而将接收到的数据报文发送给下游节点或用户终端。应用层的 client 负责接收应用层的 server 的触发，发起到上游节点的内容请求，此内容请求会触发协议栈的 client 与上游节点建立连接，之后应用层的 client 接收协议栈的 client 从上游节点获取到的数据报文，并转交给应用层的 server。

简言之，在正常的报文处理流程中，一个数据报文需历经协议栈的控制管理（比如检查是否乱序），再到应用层，再由应用层转交到协议栈，再历经协议栈的控制管理（比如速率控制），之后才能由协议栈将数据报文发送给用户终端，即实现数据分发。

在实现本公开的过程中，发明人发现相关技术至少存在以下问题：在数据报文的分发过程中，数据分发与控制管理这两个过程严重的糅合在了一起，即需要在对数据报文进行控制管理后才能再进行分发，这使得控制管理过程延后了针对数据报文的分发，引入了较大时延。

发明内容

本公开实施例提供了一种数据分发方法以及分发服务器，解决了相关技术中数据分发时延较大的问题。所述技术方案如下：

第一方面，提供了一种数据分发方法，所述方法应用于分发服务器，所述方法包括：

连续接收上游节点发送的第一上游数据报文和第二上游数据报文；

对所述第一上游数据报文进行拷贝处理，根据已建立的连接映射表，对拷贝得到的两份报文中的第一份报文进行转换处理，将转换得到的第一下游数据报文发送给下游节点或用户终端，并对基于所述第一上游数据报文拷贝得到的第二份报文进行控制管理；

对所述第二上游数据报文进行拷贝处理，根据所述连接映射表，对拷贝得到的两份报文中的第一份报文进行转换处理，将转换得到的第二下游数据报文发送给所述下游节点或所述用户终端，并对基于所述第二上游数据报文拷贝得到的第二份报文进行控制管理；

其中，所述连接映射表给出了上游连接信息与下游连接信息之间的映射关系。详细来说，所述连接映射表中包括多个映射表项，而每一个映射表项中给出了上游连接的N元组信息、与所述上游连接的N元组信息匹配的下游连接的N元组信息，以及所述上游连接与所述下游连接的初始序列号差值。

此外，所述分发服务器至少包括应用层、协议栈以及数据转发层。其中，所述数据转发层负责上述的数据分发流程，而所述协议栈负责上述的控制管理流程，实现数据转发与控制管理的分离。具体地，所述协议栈包括第一使能接口，所述应用层在向所述上游节点发送用户终端的内容请求之前，会调用所述协议栈的第一使能接口。其中，所述第一使能接口携带上下游连接标识以及指示分离数据转发功能与控制管理功能的使能标志；而所述协议栈在接收到所述应用层的调用后，分离上下游连接的数据转发功能与控制管理功能。

另外，所述数据转发层包括第二使能接口，所述协议栈在接收到所述应用层的调用后，调用所述数据转发层的第二使能接口。其中，所述第二使能接口携带所述上游连接信息与所述下游连接信息；所述数据转发层在接收到所述协议栈的调用后，建立所述连接映射表。

在第一方面的第一种可能的实现方式中，所述第一上游数据报文的序号与所述第二上游数据报文的序号不连续，所述第一上游数据报文先于所述第二上游数据报文接收到。

即，出现了上游丢包，造成了数据报文丢失，导致本分发服务器没有接收到。

结合第一方面，在第一方面的第二种可能的实现方式中，所述对拷贝得到的两份报文中的第一份报文进行转换处理，包括：获取所述第一份报文中的N元组信息；基于所述第一份报文中的N元组信息，查询所述连接映射表，得到与所述第一份报文中的N元组信息匹配的目标映射表项；以所述目标映射表项中包括的目标下游连接的N元组信息，替换所述第一份报文中的N元组信息，并根据所述目标映射表项中包括的目标初始序列号差值以及所述第一份报文中携带的序列号，计算所述目标下游连接的序列号。

结合第一方面或第一方面的第一种可能的实现方式，在第一方面的第三种可能的实现

方式中，所述对基于所述第一上游数据报文拷贝得到的第二份报文进行控制管理，包括：对基于所述第一上游数据报文拷贝得到的第二份报文进行乱序检查，并向所述上游节点发送针对所述第一上游数据报文的 ACK（Acknowledgement，确认字符）报文。

结合第一方面的第三种可能的实现方式，在第一方面的第四种可能的实现方式中，所述方法还包括：将基于所述第一上游数据报文拷贝得到的第二份报文转换成第三下游数据报文；将所述第三下游数据报文存储在缓冲区中并设置超时定时器；当所述超时定时器超时，向所述下游节点或所述用户终端发送所述第三下游数据报文。

结合第一方面的第四种可能的实现方式，在第一方面的第五种可能的实现方式中，所述连接映射表中一个上游连接信息与一个下游连接信息对应，所述方法还包括：当在所述超时定时器超时之前接收到所述下游节点或所述用户终端返回的 ACK 报文时，删除所述缓冲区中存储的所述第三下游数据报文。

当上游节点重传丢失的数据报文，则本分发服务器还会执行下述转发重传的数据报文并对其进行控制管理的步骤。即

结合第一方面或第一方面的第一种可能的实现方式，在第一方面的第六种可能的实现方式中，所述方法还包括：

接收所述上游节点重传的第三上游数据报文，所述第三上游数据报文的序号与所述第一上游数据报文的序号连续；对所述第三上游数据报文进行拷贝处理，根据所述连接映射表，对拷贝得到的两份报文中的第一份报文进行转换处理，将转换得到的第四下游数据报文发送给所述下游节点或所述用户终端，并对基于所述第三上游数据报文拷贝得到的第二份报文进行控制管理。

其中，在对基于所述第三上游数据报文拷贝得到的第二份报文进行控制管理时，过程与对第一上游数据报文的过程类似。即，对基于所述第三上游数据报文拷贝得到的第二份报文进行乱序检查，并向所述上游节点发送 ACK 报文；且将基于所述第三上游数据报文拷贝得到的第二份报文转换成下游数据报文，并将得到的下游数据报文存储在缓冲区中，同时设置超时定时器；当所述超时定时器超时，向所述下游节点或所述用户终端发送存储的这一下游数据报文；当在所述超时定时器超时之前接收到所述下游节点或所述用户终端返回的 ACK 报文时，删除所述缓冲区中存储的这一下游数据报文。

结合第一方面或第一方面的第一种可能的实现方式，在第一方面的第七种可能的实现方式中，所述对基于所述第二上游数据报文拷贝得到的第二份报文进行控制管理，包括：

对基于所述第二上游数据报文拷贝得到的第二份报文进行乱序检查，并向所述上游节点发送针对所述第一上游数据报文的 ACK 报文。

结合第一方面的第七种可能的实现方式，在第一方面的第八种可能的实现方式中，所述方法还包括：在将第四下游数据报文发送给所述下游节点或所述用户终端后，将基于所述第二上游数据报文拷贝得到的第二份报文转换成第五下游数据报文；将所述第五下游数据报文存储在缓冲区中并设置超时定时器；当所述超时定时器超时，向所述下游节点或所述用户终端发送所述第五下游数据报文。

结合第一方面的第八种可能的实现方式，在第一方面的第九种可能的实现方式中，所述连接映射表中一个上游连接信息与一个下游连接信息对应，所述方法还包括：

当在所述超时定时器超时之前接收到所述下游节点或所述用户终端返回的 ACK 报文

时，删除所述缓冲区中存储的所述第五下游数据报文。

针对上游丢包导致下游丢包的情况，本分发服务器还会执行下述过程，即

结合第一方面的第六种可能的实现方式，在第一方面的第十种可能的实现方式中，所述方法还包括：若在接收到所述上游节点重传的所述第三上游数据报文之前，接收到所述下游节点或所述用户终端发送的M次重复ACK，则等待所述上游节点重传所述第三上游数据报文，所述M次重复ACK是针对所述第一上游数据报文的；在接收到重传的所述第三上游数据报文后，执行所述对所述第三上游数据报文进行拷贝处理，根据所述连接映射表，对拷贝得到的两份报文中的第一份报文进行转换处理，将转换得到的第四下游数据报文发送给所述下游节点或所述用户终端的步骤。

结合第一方面的第六种可能的实现方式，在第一方面的第十一种可能的实现方式中，所述方法还包括：

若在接收到所述上游节点重传的所述第三上游数据报文之后，接收到所述下游节点或所述用户终端发送的M次重复ACK，则将缓冲区中存储的与所述第三上游数据报文匹配的第六下游数据报文发送给所述下游节点或所述用户终端。

第二方面，提供了一种分发服务器，所述服务器用于执行上述第一方面所述的数据分发方法。

本公开实施例提供的技术方案带来的有益效果是：

针对连续接收到的两个上游数据报文，无论这两个报文的序号是否连续，本分发服务器均会将每一个上游数据报文拷贝为两份，其中一份立即进行数据转发，并对另一份执行控制管理，本公开实施例实现了数据分发与控制管理这两个过程分离，无需在对数据报文进行控制管理后再进行分发，大大缩短了数据报文转发的耗时，而且无论是否发生丢包分发服务器在接收到报文后均立即进行转发，大大降低了数据分发时延。

附图说明

图1是本公开背景技术提供的一种CDN网络的结构示意图；

图2是本公开背景技术提供的一种分发服务器的内部架构示意图；

图3是本公开实施例提供的一种CDN命中服务的场景示意图；

图4是本公开实施例提供的一种CDN中转服务的场景示意图；

图5是本公开实施例提供的数据分发方法的第一应用场景图；

图6是本公开实施例提供的数据分发方法的第一应用场景图；

图7是本公开实施例提供的数据分发方法的第一应用场景图；

图8是本公开实施例提供的一种分发服务器的内部架构示意图；

图9是本公开实施例提供的一种数据分发方法的核心处理流程图；

图10是本公开实施例提供的第一种数据分发方法的流程图；

图11是本公开实施例提供的第二种数据分发方法的流程图；

图12是本公开实施例提供的一种分发服务器的结构示意图；

图13是本公开实施例提供的第三种数据分发方法的流程图；

图14是本公开实施例提供的一种分发服务器的结构示意图。

具体实施方式

为使本公开的目的、技术方案和优点更加清楚，下面将结合附图对本公开实施方式作进一步地详细描述。

在对本公开实施例进行详细地解释说明之前，先对本公开实施例可能涉及到的一些名词术语进行解释说明。

命中服务：如图 3 所示，如果某个分发服务器已经存储有某个内容请求所请求的数据，那么这个分发服务器将直接响应这一内容请求，而不再向上一级 CDN 节点转发这一内容请求。其中，图 3 示出了 CDN 边缘节点存储有接收到的内容请求所请求的数据的情形。如图 3 所示，该 CDN 边缘节点直接响应接收到的这一内容请求，向用户终端返回响应数据。

中转服务：如图 4 所示，如果 CDN 中间节点以及 CDN 边缘节点均没有存储某个内容请求所请求的数据，那么上述 CDN 节点会将这一内容请求一直中转直至到达源站，最终由源站提供响应数据，并由各个 CDN 节点逐级进行中转，实现将响应数据分发给用户终端。其中，图 4 中的各个序号表示请求与分发的先后发生顺序。

需要说明的是，针对一般静态内容，如各个视频网站提供的视频、图片等，由于均会长时间保持不变，因此可以缓存在作为 CDN 站点的分发服务器中，如上述图 3 和图 4 中所示的 CDN 中间节点以及 CDN 边缘节点，由 CDN 站点提供命中服务。而针对视频直播（如赛事、热点事件直播等）、动态内容（比如需要源站实时生成网页等）、静态内容的首次分发，则均需上述分发服务器提供中转服务，而由源站提供命中服务。

在另一个实施例中，通常还会利用上述分发服务器来就近存储可以缓存的内容，如点播视频、静态图片等，从而就近为用户服务，降低对传输网络带宽的消耗，同时也提高用户终端的体验。再需要强调的一点是，针对虚拟现实（Virtual Reality, VR）视频、视频直播、动态内容等这些对实时性要求高的内容来说，通常利用由 CDN 中间节点以及 CDN 边缘节点组成的 CDN 网络来提供智能路由，即提供中转服务，力求低时延分发，而由源站提供命中服务。而本公开实施例提供的方案的关注焦点便在于如何实现低时延分发服务。

其中，本公开实施例为了实现低时延分发服务，提供了一种具有低时延分发功能的分发服务器，不但可以缩短数据分发的路径，降低时延，而且还可通过分离数据分发与控制管理过程，降低控制管理给数据分发带来的时延，简要说来，本公开实施例提供的分发服务器主要进行了下述改进：

(1)、新增 TCP 薄层（又称之为数据转发层），专门执行处理流程中的数据分发操作，且零等待转发。其中，TCP 薄层转发过程可简述为：

a、TCP 薄层每收到一个上游数据报文，立即依据已建立的连接映射表，来对接收到的上游数据报文进行转换处理，比如替换数据报文中的 N 元组信息及序列号；

b、转换处理结束之后，立即将转换得到的下游数据报文发送给下游节点或用户终端，零等待转发。

需要说明的是，在以上过程中均不作控制管理，比如是否有报文乱序现象发生，是否允许发送报文等，据此来避免因控制管理过程导致的延后数据分发。

(2)、为分离数据分发与控制管理，协议栈新增 socket（套接字）接口，这一接口由应用层调用。其中，关于协议栈新增的这一接口的详细解释说明，请参见后续实施例。

(3)、为支持分离数据分发与控制管理过程，协议栈在接收到应用层的调用后，需设

置传输控制协议连接（Transmission Control Protocol, TCP）仅接收数据报文供重传使用，而不再分发数据报文，即协议栈仅实现控制管理，而数据分发完全交给 TCP 薄层处理。

(3)、为支持 TCP 薄层进行数据分发，协议栈在接收到应用层的调用后，需调用 TCP 薄层的使能接口，来指示 TCP 薄层建立连接映射表，使其数据分发能力。其中，关于 TCP 薄层的这一使能接口的详细解释说明，同样请参见后续实施例。

接下来，在对本公开实施例提供的方案进行详细地解释说明之前，先对本公开实施例的系统架构或应用场景进行一下介绍。

图 5 给出了本公开实施例提供的方案的第一种应用场景。即，用户终端在发起内容请求之前，先发起到决策服务器的请求，由决策服务器决定该内容请求由什么类型的分发服务器来服务。例如源站应用厂家可按各个应用对时延的要求，将其域名划分成两类，一类是实时性域名，例如包含 `vr/live/rtmp/dynamic` 等字样的便为实时性域名，另一类是非实时性域名。一旦检测到用户终端发起的内容请求的域名是实时性域名，则直接返回具有低时延分发功能的分发服务器的网络协议（Internet Protocol, IP）地址，之后用户将发起到这一分发服务器的内容请求，而这一分发服务器将不区分互联网用户，将为接收到的所有内容请求使能低时延分发。其中，这一分发服务器再寻找上游分发节点时，依然可由决策服务器进行决定，而所有沿途的具有低时延分发功能的分发服务器默认为接收到所有内容请求开启低时延分发功能。

图 6 给出了本公开实施例提供的方案的第二种应用场景。即，由用户终端上安装的应用客户端来感知低时延分发的需求。其中，应用客户端需扩展协议字段，携带标识位表明发起的内容请求需要低时延分发，而沿途的各个分发服务器自动识别数据报文中的该协议字段，从而针对该数据报文所属的流使能低时延分发能力，继而为此流后续的数据报文提供低时延转发。

图 7 给出了本公开实施例提供的方案的第三种应用场景。即，由分发服务器自身来感知低时延分发的需求。例如，内容请求的统一资源定位符（Uniform Resource Locator, url）中携带 `dynamic/vr/live/rtmp` 等关键字，而分发服务器在感知到后自动为相应的流使能低时延分发能力，继而为此流后续的数据报文提供低时延转发。

图 8 是本公开实施例提供的一种具有低时延分发功能的分发服务器的内部架构示意图。参见图 8，该分发服务器包括应用层 801、协议栈 802 以及数据转发层 803。

其中，数据转发层（又可称之为 TCP 薄层）803 负责进行数据分发，其既可以通过软件实现，也可以通过硬件实现，本公开实施例对此不进行具体限定。数据转发层 803 包括一个使能接口 8031，主要完成以下功能：

(1) 通过使能接口 8031，接受上层的协议栈 802（或者应用层 801）的调用，来维护一个连接映射表（比如新增/删除等）。其中，连接映射表给出了上游连接信息以及下游连接信息，比如连接映射表给出了上游连接的 N 元组信息、下游连接的 N 元组信息以及上游连接的初始序列号与下游连接的初始序列号的差值，本公开实施例对此不进行具体限定。

(2) 数据转发层 803 每接收到上游节点发送的一个上游数据报文，便会获取该上游数据报文中的 N 元组信息，并基于该上游数据报文中的 N 元组信息，查询已建立的连接映射表来进行报文替换，比如替换上游数据报文中的 N 元组信息，以及修改 TCP 连接的序列号。除此之外，在进行转换处理时，还可以进行相关校验或修改其他字段，本公开实施例对此

同样不进行具体限定。

此外，协议栈 802 负责进行控制管理。如图 8 所示，协议栈除了包括 client 模块 8021 以及 server 模块 8022 之外，还包括一个使能接口 8023。其中，新增的使能接口 8023，供应用层 801 进行调用，以使能协议栈 802 分离数据分发和控制管理，剔除协议栈 802 的数据分发功能，仅保留控制管理功能。

在本公开实施例中，使能接口 8023 需携带上下游连接标识以及指示分离数据转发功能与控制管理功能的使能标志。其中，上下游连接标识如上下游连接描述符，使能标志主要用于告知协议栈 802 需分离数据分发和控制管理。

此外，协议栈 802 需剔除数据分发的功能，即在接收到应用层 801 的数据分发调用时，仅将接收到的数据报文封装成 TCP 报文，并将该 TCP 报文存储在 Buffer（缓冲区）中，但并不发送该 TCP 报文，该 TCP 报文仅供后续进行控制管理使用。针对上述情况，协议栈 802 还需重新设置超时定时器（即重传定时器），如果接收到下游节点或用户终端的正常 ACK，则删除 Buffer 中对应的 TCP 报文并更新超时定时器；当接收到下游节点或用户终端的三次重复 ACK 或者超时定时器超时，则依据拥塞控制机制，从 Buffer 中取出对应的 TCP 报文进行重传。

其中，应用层 801 包括 client 模块 8011 以及 server 模块 8012。在另一个实施例中，可能还需调用数据转发层 803 的使能接口 8031，促使数据转发层 803 维护连接映射表和使能数据分发功能。关于这部分的介绍请参见后续实施例。

本公开实施例提供的分发服务器，通过增加专门负责转换并发送数据报文的 TCP 薄层，并使能 TCP 薄层的低时延分发能力，剔除协议栈的数据转发功能，仅保留其控制管理功能，继而实现了分离协议栈的控制管理功能与数据转发功能，减少了控制管理过程对数据转发过程的影响，使得 TCP 薄层可对接收到的数据报文立即进行转换并发送，缩短了数据报文转发的路径，大大降低了时延。

下面结合上述图 8 所述的具有低时延分发功能的分发服务器的结构图，来对本公开实施例完成低时延分发的核心处理流程进行简单地总结归纳。即，图 9 示出了公开实施例提供的方案的核心处理流程图。

步骤 1、下游节点或用户终端发起到本分发服务器的请求。其中，先是到达协议栈的 server 端的连接请求，再是达到应用层的 server 端的内容请求。

步骤 2、本分发服务器检测到本地没有存储用户请求的内容之后，触发应用层的 client 端发起到上游节点的连接请求。

步骤 3、在与上游节点的连接建立完成后，应用层调用协议栈的使能接口，控制协议栈针对上下游连接进行数据转发与控制管理的分离，继而由协议栈通过调用 TCP 薄层的使能接口，来指示 TCP 薄层针对上下游连接建立连接映射表，使能低时延转发能力。

步骤 4、应用层的 client 发起到上游节点的内容请求。

步骤 5、上游节点发送的上游数据报文正常到达本分发服务器，TCP 薄层查找连接映射表，进行转换处理，比如替换上游数据报文中的 N 元组及序列号等信息，实现将上游数据转换处理成下游数据报文，然后 TCP 薄层立即将下游数据报文发送给下游节点或用户终端。

步骤 6、本分发服务器将上游数据报文上送至协议栈，协议栈负责检查数据报文是否乱序；如果数据报文无乱序，则向上游节点发送正常 ACK，确认收到该数据报文。

步骤 7、协议栈的 client 将该上游数据报文上送到应用层的 client，继而由应用层的 client 将其转交给应用层 server，之后再发送给协议栈的 server。此时，协议栈的 server 将该上游数据报文封装成 TCP 报文放置在 Buffer 中，但不发送，设置超时定时器，等待超时重传或者在接收到下游节点或用户终端返回的正常 ACK 后来删除该 TCP 报文。

步骤 8、下游节点或用户终端正常接收到下游数据报文并回复 ACK，协议栈的 server 在接收到 ACK 后，删除 Buffer 中对应的 TCP 报文并更新超时定时器。

步骤 9、若上游节点有丢包现象出现，例如上游数据报文 P2/P4/P5/P6 正常到达，P3 未到达，那么 TCP 薄层在将接收到的上游数据报文 P2/P4/P5/P6 进行转换处理后立即发送。

步骤 10、协议栈在收到 P4/P5/P6 时，发现报文乱序（P3 未收到，从 P2 跳到 P4），因此每接收到一个上游数据报文，便发送一次针对 P2 的重复 ACK。

步骤 11、上游节点在接收到三次重复 ACK 时，重传上游数据报文 P3，TCP 薄层再将 P3 进行转换处理后立即发送，此时协议栈检测到乱序消除，所以将 P2 至 P6 转交至应用层的 client，继而再转交到协议栈的 server。

此外，针对下游丢包的情况请参见后续实施例。

图 10 是本公开实施例提供的一种数据分发方法的流程图。其中，这一实施例主要给出了使能低时延分发能力以及无丢包的数据分发过程。参见图 10，本公开实施例提供的方法流程包括：

1001、下游节点/用户终端与分发服务器进行 TCP 三次握手，建立 TCP 连接。

其中，此步骤建立的 TCP 连接为下游连接。与下游节点/用户终端建立连接的对端，为分发服务器中协议栈的 server。

1002、下游节点/用户终端使用已建立好的下游连接向分发服务器发送内容请求。

其中，该内容请求最终到达分发服务器中应用层的 server。

此外，该内容请求可使用多种协议，比如 HTTP 和 RTMP。在接收到该内容请求后，应用层的 server 检查本地缓存，如果本地未存储该内容请求所请求的数据，则应用层的 server 触发协议栈的 client 向上游节点发起建立 TCP 连接请求。

1003、应用层的 client 触发协议栈的 client 向上游节点发起建立 TCP 连接请求，经过 TCP 三次握手后，上游连接建立成功。其中，上游连接与步骤 1001 中提及的下游连接对应，此处不再赘述。

需要说明的是，在应用层的 client 向上游发起内容请求之前，本公开实施例还包括先使能本分发服务器的低时延分发能力的步骤，具体参见下述步骤 1004(a)至 1004(c)。

1004(a)、应用层调用协议栈的第一使能接口，指示协议栈使能低时延分发能力。

其中，为了将协议栈新增的使能接口与后续出现的 TCP 薄层的使能接口进行区分，本公开实施例将上述两个使能接口分别以第一使能接口以及第二使能接口进行标识。

在本公开实施例中，协议栈新增的使能接口携带上下游连接标识以及指示分离数据转发功能与控制管理功能的使能标志。其中，上下游连接标识可包括上下游连接描述符 fd_client 或 fd_server，本公开实施例对此不进行具体限定。新增的第一使能接口可以使用 setsockopt API 形式，形如：`int setsockopt(int sockfd, int level, int optname, const void *optval, socklen_t optlen)`。

其中，optval 中携带需要分离控制管理与数据转发的上下游连接标识，optname 携带指

示分离数据分发与控制管理的使能标志，表示低时延分发，如可取值 TCP_ZERODELAY。其中，TCP_ZERODELAY 指代零等待转发。

1004(b)、协议栈在接收到应用层的调用后，分离上下游连接的数据转发功能与控制管理功能。

其中，在进行分离时，具体分离 fd_server 对应的 TCP 连接的数据转发功能与控制管理功能，即在此动作执行完之后再接收到数据报文，仅做 TCP 报文封装，不再发送出去，等待超时/ACK 触发控制管理处理；分离 fd_client 对应的 TCP 连接的数据转发功能与控制管理功能，即同样在此动作执行完之后再接收到数据报文，仅做 TCP 报文封装，不再发送出去，等待超时/ACK 触发控制管理处理。

需要说明的是，fd_server 对应的 TCP 连接的处理，主要是针对由上游节点转发给本节点继而要交给下游节点或用户终端的数据，这是 CDN 分发中的经典情况，因此本方案仅对此情况进行详细说明，而 fd_client 对应的 TCP 连接的处理与此相似，本方案不再另行论述。

1004(c)、协议栈在接收到应用层的调用后，调用 TCP 薄层的第二使能接口，指示 TCP 薄层建立连接映射表，使能 TCP 薄层的低时延分发能力。

在本公开实施例中，协议栈仅做分离操作还不够，还需使能 TCP 薄层的低时延分发能力，因此协议栈在接收到应用层的调用后，再调用 TCP 薄层的第二使能接口，指示 TCP 薄层建立连接映射表，使能 TCP 薄层的低时延分发能力，告知 TCP 薄层此后收到的符合连接映射条件的数据报文均按照建立的连接映射表进行转换处理，然后直接转发得到的数据报文。

其中，TCP 薄层的第二使能接口，形如以下：

```
int settcpmap(const void *client_conn_info, void *server_conn_info, bool enable);
```

其中，client_conn_info 携带上游连接信息，可包括 N 元组信息（比如源 IP/Port，目的 IP/Port）及初始序列号；server_conn_info 携带下游连接信息，可包括 N 元组信息（比如源 IP/Port，目的 IP/Port）及初始序列号；针对 enable 来说，true 表明是建立连接映射表，false 表明是删除连接映射表。

需要说明的是，TCP 薄层需要建立两条连接映射表，其中一条表达由上游向下游发送数据，另一条表达由下游向上游发送数据，二者实现方法相同。在本公开实施例中仅以由上游向下游发送数据为例进行说明。针对该种情景，连接映射表形如下述表 1：

表 1

Src	dst	seq_diff
上游连接 N 元组	下游连接 N 元组	上下游连接的初始序号差值

假设 N 元组信息为四元组信息，上游连接四元组为 (IPup, Portup, IPc, Portc,) 以及上游连接的初始序列号 Sequpinit, 下游连接四元组为 (IPs, Ports, IPdown, Portdown,), 以及下游连接的初始序列号 Seqdowninit。

其中，IPup 和 Portup 表示上游节点的侦听地址和端口，IPc 和 Portc 表示本分发服务器在发起上游连接时使用的地址和端口，IPs 和 Ports 表示本分发服务器的侦听地址和端口，IPdown 和 Portdown 表示下游节点在发起下游连接时使用的地址/端口，则由上游向下游发送数据的情况下，对应的连接映射表为可由上述表 1 替换为下述表 2：

表 2

Src	dst	seq_diff
IPup, Portup, IPc, Portc	IPs, Ports, IPdown, Portdown	Seqdowninit-Sequpinit

其中，在上述表 1 和表 2 中，Src 作为索引端，即 TCP 薄层在接收到上游数据报文时，取出其 N 元组信息与连接映射表中的 Src 进行一一比对。当出现相同的 Src 时，则使用该映射表项的 dst 及 seq_diff 做映射转换，即进行转换处理。需要说明的是，此处还可使用常用的 hash 等方法进行映射表项存储以及加快索引，本公开实施例对此不进行具体限定。

在另一个实施例中，当上游节点、下游节点/用户终端中的任一端连接关闭或发生异常时，协议栈调用新增的第一使能接口，将 enable 置为 false，指示 TCP 薄层删除上述建立的连接映射表，为此流关闭低时延分发服务。

1005、应用层的 client 使用已建立好的上游连接发起到上游节点的内容请求。

1006、上游节点使用已建立好的上游连接发送上游数据报文给本分发服务器，本分发服务器首先拷贝一份上游数据报文给 TCP 薄层，由 TCP 薄层进行低时延分发。

换一种表达方式，TCP 薄层在接收到上游节点发送的上游数据报文后，TCP 薄层根据已建立的连接映射表，对该上游数据报文进行转换处理，得到下游数据报文，并将转换后的下游连接报文立即转发出去。

其中，TCP 薄层在进行转换处理时，具体过程如下：

首先取出上游数据报文中的 N 元组信息，使用该 N 元组信息作为 key，查找已建立好的连接映射表，找出与 key 匹配的目标映射表项，之后使用目标映射表项中的 dst 和 seq_diff 对该上游数据报文做转换处理；

假设上述 N 元组信息为 (IPup, Portup, IPc, Portc)，且查找到的目标映射表项为

Src	dst	seq_diff
IPup, Portup, IPc, Portc	IPs, Ports, IPdown, Portdown	Seqdowninit-Sequpinit

则转换后的下游数据报文的 N 元组信息为 (IPs, Ports, IPdown, Portdown)，序列号为 Seqdown。其中，Seqdown=Sequp+seq_diff。

在得到下游数据报文后，TCP 薄层转发下游数据报文。换一种表达方式，TCP 薄层将转换得到的 IP 报文立即发送给下游节点或用户终端。

1007、本分发服务器再拷贝一份上游数据报文给协议栈，由协议栈进行控制管理。

需要说明的是，在这一实施例中仅考虑无丢包的数据报文按序到达的情况。由于针对该种情况协议栈判断数据报文顺序到达，即协议栈在对数据报文进行乱序检查后，发现未出现报文乱序的情况，因此协议栈发送正常 ACK 给上游节点。

1008、在数据报文按序到达后，协议栈的 client 上交上游数据报文给应用层的 client。

1009、应用层的 client 转交上游数据报文给应用层的 server。

1010、应用层的 server 转交上游数据报文给协议栈的 server。

1011、协议栈的 server 接收到应用层的 server 发送的上游数据报文后，使用下游连接信息（比如 N 元组信息、序列号等）将该上游数据报文封装成 TCP 报文，并将封装完毕后的 TCP 报文放置 Buffer 中。

其中，其中封装完毕后的 TCP 报文并不进行发送，而是仅放置在 Buffer 中。协议栈的 server 会设置超时定时器，等待下游节点/用户终端的 ACK 触发对 TCP 报文的删除处理，或

者等待超时定时器超时，如果超时定时器超时则触发重传 TCP 报文处理。

由以上列举的实施过程可知，本实施例阐述的方案创新点在于以下：

在分发服务器中新增了 TCP 薄层，低时延分发能力由新增加的 TCP 薄层实现，而为支持 TCP 薄层的低时延分发功能，改动了协议栈，剔除了协议栈的数据分发功能，继而分离了协议栈的控制管理功能与数据分发功能。其中 TCP 薄层既可软件实现，也可硬件实现。

基于此，在无丢包的数据分发过程中，这一实施例独特之处在于：

- (1)、步骤 1004 (a) 至步骤 1004 (a) 中示出的低时延分发能力的使能过程；
- (2)、步骤 1006 中在接收到数据报文后，TCP 薄层立即进行转换以及分发的过程；
- (3)、步骤 1011 中的协议栈的 server 收到应用层的 server 发送的数据报文后，仅做封装不进行转发的过程。

而现有方案在接收到数据报文后，由于会将数据报文逐层上送至应用层的 client，再转交应用层的 server，再交由协议栈的 server 发送出去。相较于现有方案，本公开实施例提供的的数据分发方法缩短了转发路径，降低了时延。此外，现有方案中当应用层将数据报文转交给协议栈的 server 后，协议栈的 server 并不一定立即进行发送，而是需要考虑发送速率等控制因素，而本方案因为 TCP 薄层仅负责数据转发，并不进行任何控制处理，所以可以降低控制处理对数据转发的影响。

本公开实施例提供的方法，通过在分发服务器中增加专门负责转换并发送数据报文的 TCP 薄层，并使能 TCP 薄层的低时延分发能力，剔除协议栈的数据转发功能，仅保留其控制管理功能，继而实现了分离协议栈的控制管理功能与数据转发功能，减少了控制管理过程对数据转发过程的影响，使得 TCP 薄层可对接收到的数据报文立即进行转换并发送，缩短了数据报文转发的路径，大大降低了时延。

图 11 是本公开实施例提供的一种数据分发方法的流程图。其中，这一实施例主要给出了上游连接控制处理过程，包括针对上游连接丢包的处理过程。参见图 11，本公开实施例提供的方法流程包括：

1101、上游节点发送的数据报文 P1 正常到达，本分发服务器将 P1 拷贝为两份，TCP 薄层在对其中一份进行转换处理后立即转发给下游节点或用户终端。

1102、协议栈对拷贝得到的另一份进行控制管理，若协议栈检查另一份数据报文无乱序，则协议栈向上游节点发送 ACK，确认收到数据报文 P1，同时将另一份数据报文中送至应用层的 client。

1103、上游节点发送的数据报文 P2 丢失。

1104、上游节点发送的数据报文 P3 正常到达，本分发服务器将 P3 拷贝为两份，TCP 薄层在对其中一份进行转换处理后立即转发给下游节点或用户终端。

1105、协议栈对拷贝得到的另一份进行控制管理，协议栈检测到另一份数据报文乱序，协议栈给上游节点发送 ACK，确认收到数据 P1。

需要说明的是，由于未接收到数据报文 P2，而是接收到了数据报文 P3，因此协议栈确定报文乱序，向上游节点还是发送针对数据报文 P1 的 ACK。此为第一次重复 ACK（针对数据报文 P1 的 ACK 在步骤 1102 中已经发送过）。

需要说明的是，由于数据报文 P2 丢失，因此协议栈的 client 存储数据报文 P3，并不上送给应用层的 client。

1106、上游节点发送的数据报文 P4 和 P5 正常到达，TCP 薄层在对数据报文 P4 和 P5 进行转换处理后立即转发给下游节点或用户终端。

由于在数据报文 P3 之后还收到了数据报文 P4 和 P5，因此还会触发针对数据报文 P1 的第二次和第三次重复 ACK。

1107、上游节点在接收到三次重复 ACK 后，重传数据报文 P2，TCP 薄层在对数据报文 P2 进行转换处理后立即转发给下游节点或用户终端。

1108、数据报文 P2 上送至协议栈，协议栈检测到数据报文无乱序，向上游节点发送 ACK，确认收到数据报文 P5，同时将数据报文 P2-P5 上交给应用层的 client。

换一种表达方式，协议栈在接收重传的数据报文 P2 后，向上游节点发送针对最后一个接收到的数据报文（即数据报文 P5）的 ACK，并将自数据报文 P2 开始至数据报文 P5 结束的 4 个数据报文中上交给应用层的 client。

1109、应用层的 client 将数据报文 P2-P5 转交给应用层的 server，应用层的 server 将数据报文 P2-P5 转交至协议栈的 server。

其中，上述数据报文 P2-P5 以 TCP 报文形式存储在协议栈的 server 的 Buffer 处，等待下游节点或用户终端 ACK、或者超时定时器超时，以触发删除 TCP 报文或重传 TCP 报文。

需要说明的是，在执行上述数据转发时，上述数据报文 P1 可称之为第一上游数据报文，针对数据报文 P1，本分发服务器转发的下游数据报文可称之为第一下游数据报文，协议栈的 Buffer 中缓存的可称之为第三下游数据报文。上述数据报文 P3 可称之为第二上游数据报文，针对数据报文 P3，本分发服务器转发的下游数据报文可称之为第二下游数据报文，协议栈的 Buffer 中缓存的可称之为第五下游数据报文。上述数据报文 P2 可称之为第三上游数据报文，针对数据报文 P2，本分发服务器转发的下游数据报文可称之为第四下游数据报文，协议栈的 Buffer 中缓存的可称之为第六下游数据报文。

另外，数据报文 P1 至 P5 上游节点是按照时间先后顺序进行发送的，且每一个数据报文对应一个序号。其中，上述第一上游数据报文的序号与第二上游数据报文的序号是不连续的。但是上述第一上游数据报文的序号与第三上游数据报文的序号是连续的。

综上所述，以上实施过程描述的是，如何确认接收到的上游数据报文，并在遇上游丢包情况时如何处理丢包。需要说明的是，本方案是在每接收到一个数据报文后立即转换并转发，而无需等到已收到丢包并检查报文无乱序后，再将期间接收到的数据报文转交至应用，进而发送给下游节点或用户终端。

在另一个实施例中，图 11 所示的实施例对报文的处理流程与图 10 所示的实施例一致，二者之间的差别在于低时延分发能力的使能过程。结合图 12，图 11 所示的实施例的低时延能力使能过程步骤如下：

(a)、应用层携带上下游连接描述符调用协议栈新增的一个使能接口，获取上下游连接信息，比如至少包括上下游连接的初始序列号。

其中，协议栈新增的这个使能接口形如：`int getsockopt(int sockfd, int level, int optname, void *optval, socklen_t*optlen);`

其中，`optname` 携带要获取上下游连接信息的标识信息，如可取值为 `TCP_CONNECTINFO`；`optval` 中包含两个结构体，结构体类型为 `connectinfo`，调用时每个 `connectinfo` 中分别携带需要分离控制管理与数据转发的上下游连接标识，如上下游连接

描述符 fd_client、fd_server。其中，connectinfo 如下：

```
struct connectinfo {
    int socketfd; //连接描述符
    struct ipports; //连接 N 元组
    int seqinit; //连接初始序列号
}
```

需要说明的是，协议栈返回时，在 seqinit 中填上 socketfd 指定的连接的初始序列号，在 ipports 中填上该连接的 N 元组，但 N 元组不是必须的，因为应用层不调用接口也可以知道 N 元组信息。

(b)、应用层携带上下游连接标识，调用协议栈新增的另一个使能接口，指示协议栈分离数据转发和控制管理功能。

其中，新增的这个使能接口可以使用 setsockopt API 形式，形如：int setsockopt(int sockfd, int level, int optname, const void *optval, socklen_t optlen)。

其中，optval 中携带需要分离控制管理与数据转发的上下游连接标识，optname 携带指示分离数据分发与控制管理的使能标志，表示低时延分发，如可取值 TCP_ZERODELAY。其中，TCP_ZERODELAY 指代零等待转发。

(c)、协议栈在接收到应用层的调用后，分离上下游连接的数据转发功能与控制管理功能。

其中，在进行分离时，具体分离 fd_server 对应的 TCP 连接的数据转发功能与控制管理功能，即在此动作执行完之后再接收到数据报文，仅做 TCP 报文封装，不再发送出去，等待超时/ACK 触发控制管理处理；分离 fd_client 对应的 TCP 连接的数据转发功能与控制管理功能，即同样在此动作执行完之后再接收到数据报文，仅做 TCP 报文封装，不再发送出去，等待超时/ACK 触发控制管理处理。

(d)、应用层携带上下游连接的 N 元组信息及初始序列号，调用 TCP 薄层的使能接口，建立连接映射表，使能 TCP 薄层的低时延分发功能。

其中，TCP 薄层的使能接口，形如以下：

```
int settcpmap(const void *client_conn_info, void *server_conn_info, bool enable);
```

其中，client_conn_info 携带上游连接信息，可包括 N 元组信息（比如源 IP/Port，目的 IP/Port）及初始序列号；server_conn_info 携带下游连接信息，可包括 N 元组信息（比如源 IP/Port，目的 IP/Port）及初始序列号；针对 enable 来说，true 表明是建立连接映射表，false 表明是删除连接映射表。

需要说明的是，TCP 薄层需要建立两条连接映射表，其中一条表达由上游向下游发送数据，另一条表达由下游向上游发送数据，二者实现方法相同。在本公开实施例申仅以由上游向下游发送数据为例进行说明。针对该种情景，连接映射表形如下所示：

Src	dst	seq_diff
上游连接 N 元组	下游连接 N 元组	上下游连接的初始序号差值

假设 N 元组信息为四元组信息，上游连接四元组为 (IPup, Portup, IPc, Portc,) 以及上游连接的初始序列号 Sequpinit, 下游连接四元组为 (IPs, Ports, IPdown, Portdown,)，以及下游连接的初始序列号 Seqdowninit。

其中，IPup 和 Portup 表示上游节点的侦听地址和端口，IPc 和 Portc 表示本分发服务器在发起上游连接时使用的地址和端口，IPs 和 Ports 表示本分发服务器的侦听地址和端口，IPdown 和 Portdown 表示下游节点在发起下游连接时使用的地址/端口，则由上游向下游发送数据的情况下，对应的连接映射表为如下所示：

Src	dst	seq_diff
IPup, Portup, IPc, Portc	IPs, Ports, IPdown, Portdown	Seqdowninit-Sequpinit

其中，在上述两个表格中，Src 作为索引端，即 TCP 薄层在接收到上游数据报文时，取出其 N 元组信息与连接映射表中的 Src 进行一一比对。当出现相同的 Src 时，则使用该映射表项的 dst 及 seq_diff 做映射转换，即进行转换处理。需要说明的是，此处还可使用常用的 hash 等方法进行映射表项存储以及加快索引，本公开实施例对此不进行具体限定。

需要说明的是，图 11 对应的实施例与图 10 对应的实施例的差异之处在于：

在图 10 对应的实施例中，协议栈仅增加一个使能接口，由应用层调用该使能接口一次告知协议栈使能低时延能力，协议栈据此分离自身的控制管理与数据转发给你，同时调用 TCP 薄层的使能接口来指示 TCP 薄层建立连接映射表和使能低时延分发能力。

在图 11 对应的实施例中，协议栈新增加了两个使能接口，应用层调用两个使能接口中的一个来获取上下游连接信息，再调用两个使能接口中的另一个来指示协议栈执行分离动作，之后再调用 TCP 薄层的使能接口来指示 TCP 薄层建立连接映射表和使能低时延分发能力

本公开实施例提供的方法，针对连续接收到的至少两个上游数据报文，无论这些报文的序号是否连续，本分发服务器均会将每一个上游数据报文拷贝为两份，其中一份立即进行数据转发，并对另一份执行控制管理，本公开实施例实现了数据分发与控制管理这两个过程分离，无需在对数据报文进行控制管理后再进行分发，大大缩短了数据报文转发的路径，而且无论是否发生丢包在接收到一个报文后均立即对该报文进行转发，大大降低了数据分发时延。通过在分发服务器中增加专门负责转换并发送数据报文的 TCP 薄层，并使能 TCP 薄层的低时延分发能力，剔除协议栈的数据转发功能，仅保留其控制管理功能，继而实现了分离协议栈的控制管理功能与数据转发功能，减少了控制管理过程对数据转发过程的影响，使得 TCP 薄层可对接收到的数据报文立即进行转换并发送，缩短了数据报文转发的路径，大大降低了时延。

图 13 是本公开实施例提供的一种数据分发方法的流程图。其中，这一实施例主要给出了下游控制处理过程，包含由上游连接丢包导致的下游连接丢包的处理过程。参见图 13，本公开实施例提供的方法流程包括：

1301、下游节点或用户终端在顺序收到数据报文 P1 后，向本分发服务器发送 ACK。

本步骤针对无丢包时处理下游 ACK 的过程。当下游节点或用户终端顺序收到数据报文 P1 后，发送 ACK 给本分发服务器，以确认收到数据报文 P1，而本分发服务器会将接收到的这一 ACK 上送给协议栈，协议栈删除 Buffer 中的数据报文 P1 以及序号在 P1 之前的其他数据报文，并更新超时定时器及其他控制参数，本公开实施例对此不进行具体限定。

1302、上游节点发送的数据报文 P2 丢失。

1303、上游节点发送的数据报文 P3-P5 到达，TCP 薄层在对数据报文 P3-P5 进行转换

处理后立即转发给下游节点或用户终端。

1304、下游节点或用户终端未接收到数据报文 P2。

针对本步骤，因为发生了上游丢包，TCP 薄层并没有接收到数据报文 P2，因此下游节点或用户终端也没有接收到数据报文 P2，当下游节点或用户终端依次接收到数据报文 P3-P5 时，会触发发送针对数据报文 P1 的 3 次重复 ACK 给本分发服务器，仅确认接收到数据报文 P1，告知数据报文 P2 丢失。

而本分发服务器在接收到三次重复 ACK 后，因为当前 Buffer 中尚未存储数据报文 P2，所以无法实现重传，因此不做任何处理，等待 TCP 薄层收到上游节点重传的数据报文 P2 转发给下游节点或用户终端。

1305、上游节点在接收到三次重复 ACK 后，重传数据报文 P2，将 P2 拷贝为两份，TCP 薄层在对其中一份进行转换处理得到下游数据报文后立即转发给下游节点或用户终端。

由于本分发服务器在数据报文 P1 之后顺序接收到数据报文 P3-P5，因此也会发送针对数据报文 P1 的 3 次重复 ACK 给上游节点，继而触发上游节点重传数据报文 P2。

1306、下游节点或用户终端接收到数据报文 P2 后，消除乱序，发送 ACK 给本分发服务器，确认接收到数据报文 P5，本分发服务器在接收到 ACK 后，协议栈删除存储的数据报文 P5 及序号在 P5 之前的数据报文，并更新超时定时器及其他控制参数。

需要说明的第一点是，针对上述步骤 1304，如果下游节点或用户终端发送的 ACK 到达时间晚，是在本分发服务器已经接收到上游节点重传的数据报文 P2 后才到达，那么本分发服务器可以再次触发重传数据报文 P2，由于 TCP 薄层已转发过一次数据报文 P2，所以下游节点或用户终端会接收到 2 份数据报文 P2，而下游节点或用户终端在接收到第 2 份数据报文 P2 时会自动丢弃，对业务并没有影响。

需要说明的第二点是，在上游丢包导致下游丢包时，本节点在接收到上游节点重传的数据报文之前，若接收到下游节点或用户终端的重复 ACK，则本节点可能无法实现重传，需等待上游节点重传数据报文到达，并由 TCP 薄层直接转发。

需要说明的第三点是，以上介绍了上游丢包导致下游丢包的情况，针对下游连接出现丢包的情况，可由协议栈的 server 负责重传，此处不再给出具体流程图进行描述。另外，如果下游丢包严重，比如链路质量太差导致丢包率高，或者下游节点的处理性能太差导致节点内丢包，那么说明下游链路或下游节点的性能无法满足当前传输数据量的要求，可触发下游节点重新发起请求，请求数据量少的内容（例如从高清视频切换到标清视频），或者触发本分发服务器建立一条新的连接映射表，依据新建立的连接映射表为慢节点提供服务。

在另一个实施例中，上述实施例中描述的连接映射表仅是一条上游连接对应一条下游连接的情况，但是在实际的应用场景中，尤其是直播场景中，可能会有多个下游节点发起多条连接到本节点（本分发服务器）请求同一个内容，此时本节点仅需发起一条上游连接即可，此时便会出现一条上游连接映射到多条下游连接的情况，其处理过程为：

本节点收到第一次请求时，依据图 10 对应的实施例中的处理，触发建立连接映射表，连接映射表如下：

Src	dst	seq_diff
IPup, Portup, IPc, Portc	IPs, Ports, IPdown1, Portdown1	Seqdowninit1-Sequpinit

本节点收到第 N (N>1) 次请求时，不再触发上游请求，但依然依据图 10 对应的实施

例中的处理，触发建立连接映射表，此时连接映射表如下：

Src	dst	seq_diff
IPup, Portup, IPC, Portc	IPs, Ports, IPdown1, Portdown1 IPs, Ports, IPdownN, PortdownN	Seqdowninit1-Sequpinit SeqdowninitN-Sequpinit

这样 TCP 薄层在接收到上游节点发送的数据报文后，会查询上述建立的连接映射表，如果查询到的目标映射表项中包含 N 个 dst 选项，则 TCP 薄层将接收到的数据转换处理 N 次，每一次转换过程与图 10 对应的实施例相似，但每一次转换依据一个不同的 dst 选项以及 seq_diff 选项，且每转换完一次发送一次。

由于增加了 TCP 薄层中一条上游连接映射到多条下游连接的处理过程，因此使得 TCP 薄层可以更好地支持直播等场景。考虑可能存在直播加入的不同步问题，有的互联网用户可能会稍微加入的晚一点，此时可由应用层筛选已缓存数据发送给后加入的互联网用户，当后加入的互联网用户与先加入的互联网用户同步后，再将后加入互联网用户的下游连接信息加入到上述连接映射表，再由 TCP 薄层统一按照前文所述的方式进行数据转发，实现同步数据；或者，还可以在 TCP 薄层或协议栈中针对直播对象设置公共 Buffer，该 Buffer 用于缓存该直播对象的已缓存数据，当有互联网用户后加入时，可由 TCP 薄层或协议栈筛选已缓存数据发送给后加入的互联网用户，同样当后加入的互联网用户与先加入的互联网用户同步后，再由 TCP 薄层统一按照前文所述的方式进行数据转发，实现同步数据。

需要说明的是，针对一条上游连接映射到多条下游连接的情况，以直播场景为例，协议栈在将接收到的上游数据报文封装为 TCP 报文并缓存在 Buffer 中后，后续过程中即便接收到下游节点或互联网用户返回的 ACK 报文，为了便于以后加入的互联网用户进行直播观看，也不会对 Buffer 中的已缓存数据进行删除。

本公开实施例提供的方法，针对连续接收到的至少两个上游数据报文，无论这些报文的序号是否连续，本分发服务器均会将每一个上游数据报文拷贝为两份，其中一份立即进行数据转发，并对另一份执行控制管理，本公开实施例实现了数据分发与控制管理这两个过程分离，无需在对数据报文进行控制管理后再进行分发，大大缩短了数据报文转发的路径，而且无论是否发生丢包在接收到一个报文后均立即对该报文进行转发，大大降低了数据分发时延。

通过在分发服务器中增加专门负责转换并发送数据报文的 TCP 薄层，并使能 TCP 薄层的低时延分发能力，剔除协议栈的数据转发功能，仅保留其控制管理功能，继而实现了分离协议栈的控制管理功能与数据转发功能，减少了控制管理过程对数据转发过程的影响，使得 TCP 薄层可对接收到的数据报文立即进行转换并发送，缩短了数据报文转发的路径，大大降低了时延。

图 14 是本公开实施例提供的一种分发服务器的结构示意图。参见图 14，该服务器包括：第一接收模块 1401、数据转发模块 1402 以及控制管理模块 1403。其中，数据转发模块 1402 用于实现前文中 TCP 薄层的数据转发功能，控制管理模块 1403 用于实现协议栈的控制管理功能，第一接收模块 1401 可对应于图 8 中接收上游节点发送的数据报文的 Buffer。

其中，第一接收模块 1401，用于连续接收上游节点发送的第一上游数据报文和第二上游数据报文，并对所述第一上游数据报文进行拷贝处理以及对所述第二上游数据报文进行

拷贝处理；

数据转发模块 1402，用于根据已建立的连接映射表，对基于所述第一上游数据报文拷贝得到的两份报文中的第一份报文进行转换处理，将转换得到的第一下游数据报文发送给下游节点或用户终端；

控制管理模块 1403，用于对基于所述第一上游数据报文拷贝得到的第二份报文进行控制管理；

数据转发模块 1402，还用于根据所述连接映射表，对基于所述第二上游数据报文拷贝得到的两份报文中的第一份报文进行转换处理，将转换得到的第二下游数据报文发送给所述下游节点或所述用户终端；

控制管理模块 1403，还用于对基于所述第二上游数据报文拷贝得到的第二份报文进行控制管理；

其中，所述连接映射表给出了上游连接信息与下游连接信息之间的映射关系。

本公开实施例提供的分发服务器，针对连续接收到的至少两个上游数据报文，无论这些报文的序号是否连续，本分发服务器均会将每一个上游数据报文拷贝为两份，其中一份立即进行数据转发，并对另一份执行控制管理，本公开实施例实现了数据分发与控制管理这两个过程分离，无需在对数据报文进行控制管理后再进行分发，大大缩短了数据报文转发的路径，而且无论是否发生丢包在接收到一个报文后均立即对该报文进行转发，大大降低了数据分发时延。

在另一个实施例中，所述第一上游数据报文的序号与所述第二上游数据报文的序号不连续，所述第一上游数据报文先于所述第二上游数据报文接收到。

在另一个实施例中，所述连接映射表包括多个映射表项，每一个映射表项中给出了上游连接的 N 元组信息、与所述上游连接的 N 元组信息匹配的下游连接的 N 元组信息，所述上游连接与所述下游连接的初始序列号差值；

所述数据转发模块 1402，还用于获取所述第一份报文中的 N 元组信息；基于所述第一份报文中的 N 元组信息，查询所述连接映射表，得到与所述第一份报文中的 N 元组信息匹配的目标映射表项；以所述目标映射表项中包括的目标下游连接的 N 元组信息，替换所述第一份报文中的 N 元组信息，并根据所述目标映射表项中包括的目标初始序列号差值以及所述第一份报文中携带的序列号，计算所述目标下游连接的序列号。

在另一个实施例中，所述控制管理模块 1403，还用于对基于所述第一上游数据报文拷贝得到的第二份报文进行乱序检查，并向所述上游节点发送针对所述第一上游数据报文的确认字符 ACK 报文。

在另一个实施例中，所述控制管理模块 1403，还用于将基于所述第一上游数据报文拷贝得到的第二份报文转换成第三下游数据报文；将所述第三下游数据报文存储在缓冲区中并设置超时定时器；当所述超时定时器超时，向所述下游节点或所述用户终端发送所述第三下游数据报文。

在另一个实施例中，所述连接映射表中一个上游连接信息与一个下游连接信息对应，所述控制管理模块 1403，还用于当在所述超时定时器超时之前接收到所述下游节点或所述用户终端返回的 ACK 报文时，删除所述缓冲区中存储的所述第三下游数据报文。

在另一个实施例中，所述第一接收模块 1401，还用于接收所述上游节点重传的第三上

游数据报文，并对所述第三上游数据报文进行拷贝处理，所述第三上游数据报文的序号与所述第一上游数据报文的序号连续；

所述数据转发模块 1402，还用于根据所述连接映射表，对拷贝得到的两份报文中的第一份报文进行转换处理，将转换得到的第四下游数据报文发送给所述下游节点或所述用户终端，并对基于所述第三上游数据报文拷贝得到的第二份报文进行控制管理。

在另一个实施例中，所述控制管理模块 1403，还用于对基于所述第二上游数据报文拷贝得到的第二份报文进行乱序检查，并向所述上游节点发送针对所述第一上游数据报文的 ACK 报文。

在另一个实施例中，所述控制管理模块 1403，还用于在数据转发模块 1402 将第四下游数据报文发送给所述下游节点或所述用户终端后，将基于所述第二上游数据报文拷贝得到的第二份报文转换成第五下游数据报文；将所述第五下游数据报文存储在缓冲区中并设置超时定时器；当所述超时定时器超时，向所述下游节点或所述用户终端发送所述第五下游数据报文。

在另一个实施例中，所述连接映射表中一个上游连接信息与一个下游连接信息对应，所述控制管理模块 1403，还用于当在所述超时定时器超时之前接收到所述下游节点或所述用户终端返回的 ACK 报文时，删除所述缓冲区中存储的所述第五下游数据报文。

在另一个实施例中，所述控制管理模块 1403，还用于若在所述第一接收模块 1401 接收到所述上游节点重传的所述第三上游数据报文之前，第二接收模块接收到所述下游节点或所述用户终端发送的 M 次重复 ACK，则等待所述上游节点重传所述第三上游数据报文，所述 M 次重复 ACK 是针对所述第一上游数据报文的；

所述数据转发模块 1402，还用于在接收到重传的所述第三上游数据报文后，执行根据所述连接映射表，对基于所述第三上游数据报文拷贝得到的两份报文中的第一份报文进行转换处理，将转换得到的第四下游数据报文发送给所述下游节点或所述用户终端的步骤。

在另一个实施例中，所述控制管理模块 1403，还用于若在所述第一接收模块 1401 接收到所述上游节点重传的所述第三上游数据报文之后，第二接收模块接收到所述下游节点或所述用户终端发送的 M 次重复 ACK，则将缓冲区中存储的与所述第三上游数据报文匹配的第六下游数据报文发送给所述下游节点或所述用户终端。

上述任一种可选方式可与本公开实施例相结合构成本公开的可选实施例。

本领域普通技术人员可以理解实现上述实施例的全部或部分步骤可以通过硬件来完成，也可以通过程序来指令相关的硬件完成，所述的程序可以存储于一种计算机可读存储介质中，上述提到的存储介质可以是只读存储器，磁盘或光盘等。

以上所述仅为本公开的可选实施例，并不用以限制本公开，凡在本公开的精神和原则之内，所作的任何修改、等同替换、改进等，均应包含在本公开的保护范围之内。

权 利 要 求 书

1、一种数据分发方法，所述方法应用于分发服务器，其特征在于，所述方法包括：

连续接收上游节点发送的第一上游数据报文和第二上游数据报文；

对所述第一上游数据报文进行拷贝处理，根据已建立的连接映射表，对拷贝得到的两份报文中的第一份报文进行转换处理，将转换得到的第一下游数据报文发送给下游节点或用户终端，并对基于所述第一上游数据报文拷贝得到的第二份报文进行控制管理；

对所述第二上游数据报文进行拷贝处理，根据所述连接映射表，对拷贝得到的两份报文中的第一份报文进行转换处理，将转换得到的第二下游数据报文发送给所述下游节点或所述用户终端，并对基于所述第二上游数据报文拷贝得到的第二份报文进行控制管理；

其中，所述连接映射表给出了上游连接信息与下游连接信息之间的映射关系。

2、根据权利要求1所述的方法，其特征在于，所述第一上游数据报文的序号与所述第二上游数据报文的序号不连续，所述第一上游数据报文先于所述第二上游数据报文接收到。

3、根据权利要求1所述的方法，其特征在于，所述连接映射表包括多个映射表项，每一个映射表项中给出了上游连接的N元组信息、与所述上游连接的N元组信息匹配的下游连接的N元组信息，所述上游连接与所述下游连接的初始序列号差值；

所述对拷贝得到的两份报文中的第一份报文进行转换处理，包括：

获取所述第一份报文中的N元组信息；

基于所述第一份报文中的N元组信息，查询所述连接映射表，得到与所述第一份报文中的N元组信息匹配的目标映射表项；

以所述目标映射表项中包括的目标下游连接的N元组信息，替换所述第一份报文中的N元组信息，并根据所述目标映射表项中包括的目标初始序列号差值以及所述第一份报文中携带的序列号，计算所述目标下游连接的序列号。

4、根据权利要求1或2所述的方法，其特征在于，所述对基于所述第一上游数据报文拷贝得到的第二份报文进行控制管理，包括：

对基于所述第一上游数据报文拷贝得到的第二份报文进行乱序检查，并向所述上游节点发送针对所述第一上游数据报文的确认字符ACK报文。

5、根据权利要求4所述的方法，其特征在于，所述方法还包括：

将基于所述第一上游数据报文拷贝得到的第二份报文转换成第三下游数据报文；

将所述第三下游数据报文存储在缓冲区中并设置超时定时器；

当所述超时定时器超时，向所述下游节点或所述用户终端发送所述第三下游数据报文。

6、根据权利要求5所述的方法，其特征在于，所述连接映射表中一个上游连接信息与一个下游连接信息对应，所述方法还包括：

当在所述超时定时器超时之前接收到所述下游节点或所述用户终端返回的ACK报文时，

删除所述缓冲区中存储的所述第三下游数据报文。

7、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：

接收所述上游节点重传的第三上游数据报文，所述第三上游数据报文的序号与所述第一上游数据报文的序号连续；

对所述第三上游数据报文进行拷贝处理，根据所述连接映射表，对拷贝得到的两份报文中的第一份报文进行转换处理，将转换得到的第四下游数据报文发送给所述下游节点或所述用户终端，并对基于所述第三上游数据报文拷贝得到的第二份报文进行控制管理。

8、根据权利要求 2 所述的方法，其特征在于，所述对基于所述第二上游数据报文拷贝得到的第二份报文进行控制管理，包括：

对基于所述第二上游数据报文拷贝得到的第二份报文进行乱序检查，并向所述上游节点发送针对所述第一上游数据报文的 ACK 报文。

9、根据权利要求 8 所述的方法，其特征在于，所述方法还包括：

在将第四下游数据报文发送给所述下游节点或所述用户终端后，将基于所述第二上游数据报文拷贝得到的第二份报文转换成第五下游数据报文；

将所述第五下游数据报文存储在缓冲区中并设置超时定时器；

当所述超时定时器超时，向所述下游节点或所述用户终端发送所述第五下游数据报文。

10、根据权利要求 9 所述的方法，其特征在于，所述连接映射表中一个上游连接信息与一个下游连接信息对应，所述方法还包括：

当在所述超时定时器超时之前接收到所述下游节点或所述用户终端返回的 ACK 报文时，删除所述缓冲区中存储的所述第五下游数据报文。

11、根据权利要求 7 所述的方法，其特征在于，所述方法还包括：

若在接收到所述上游节点重传的所述第三上游数据报文之前，接收到所述下游节点或所述用户终端发送的 M 次重复 ACK，则等待所述上游节点重传所述第三上游数据报文，所述 M 次重复 ACK 是针对所述第一上游数据报文的；

在接收到重传的所述第三上游数据报文后，执行所述对所述第三上游数据报文进行拷贝处理，根据所述连接映射表，对拷贝得到的两份报文中的第一份报文进行转换处理，将转换得到的第四下游数据报文发送给所述下游节点或所述用户终端的步骤。

12、根据权利要求 7 所述的方法，其特征在于，所述方法还包括：

若在接收到所述上游节点重传的所述第三上游数据报文之后，接收到所述下游节点或所述用户终端发送的 M 次重复 ACK，则将缓冲区中存储的与所述第三上游数据报文匹配的第六下游数据报文发送给所述下游节点或所述用户终端。

13、一种分发服务器，其特征在于，所述服务器包括：

第一接收模块，用于连续接收上游节点发送的第一上游数据报文和第二上游数据报文，并对所述第一上游数据报文进行拷贝处理以及对所述第二上游数据报文进行拷贝处理；

数据转发模块，用于根据已建立的连接映射表，对基于所述第一上游数据报文拷贝得到的两份报文中的第一份报文进行转换处理，将转换得到的第一下游数据报文发送给下游节点或用户终端；

控制管理模块，用于对基于所述第一上游数据报文拷贝得到的第二份报文进行控制管理；

所述数据转发模块，还用于根据所述连接映射表，对基于所述第二上游数据报文拷贝得到的两份报文中的第一份报文进行转换处理，将转换得到的第二下游数据报文发送给所述下游节点或所述用户终端；

所述控制管理模块，还用于对基于所述第二上游数据报文拷贝得到的第二份报文进行控制管理；

其中，所述连接映射表给出了上游连接信息与下游连接信息之间的映射关系。

14、根据权利要求 13 所述的服务器，其特征在于，所述第一上游数据报文的序号与所述第二上游数据报文的序号不连续，所述第一上游数据报文先于所述第二上游数据报文接收到。

15、根据权利要求 13 所述的服务器，其特征在于，所述连接映射表包括多个映射表项，每一个映射表项中给出了上游连接的 N 元组信息、与所述上游连接的 N 元组信息匹配的下游连接的 N 元组信息，所述上游连接与所述下游连接的初始序列号差值；

所述数据转发模块，还用于获取所述第一份报文中的 N 元组信息；基于所述第一份报文中的 N 元组信息，查询所述连接映射表，得到与所述第一份报文中的 N 元组信息匹配的目标映射表项；以所述目标映射表项中包括的目标下游连接的 N 元组信息，替换所述第一份报文中的 N 元组信息，并根据所述目标映射表项中包括的目标初始序列号差值以及所述第一份报文中携带的序列号，计算所述目标下游连接的序列号。

16、根据权利要求 13 或 14 所述的服务器，其特征在于，所述控制管理模块，还用于对基于所述第一上游数据报文拷贝得到的第二份报文进行乱序检查，并向所述上游节点发送针对所述第一上游数据报文的确认字符 ACK 报文。

17、根据权利要求 16 所述的服务器，其特征在于，所述控制管理模块，还用于将基于所述第一上游数据报文拷贝得到的第二份报文转换成第三下游数据报文；将所述第三下游数据报文存储在缓冲区中并设置超时定时器；当所述超时定时器超时，向所述下游节点或所述用户终端发送所述第三下游数据报文。

18、根据权利要求 17 所述的服务器，其特征在于，所述连接映射表中一个上游连接信息与一个下游连接信息对应，

所述控制管理模块，还用于当在所述超时定时器超时之前接收到所述下游节点或所述用户终端返回的 ACK 报文时，删除所述缓冲区中存储的所述第三下游数据报文。

19、根据权利要求 13 或 14 所述的服务器，其特征在于，所述第一接收模块，还用于接收所述上游节点重传的第三上游数据报文，并对所述第三上游数据报文进行拷贝处理，所述第三上游数据报文的序号与所述第一上游数据报文的序号连续；

所述数据转发模块，还用于根据所述连接映射表，对拷贝得到的两份报文中的第一份报文进行转换处理，将转换得到的第四下游数据报文发送给所述下游节点或所述用户终端，并对基于所述第三上游数据报文拷贝得到的第二份报文进行控制管理。

20、根据权利要求 14 所述的服务器，其特征在于，所述控制管理模块，还用于对基于所述第二上游数据报文拷贝得到的第二份报文进行乱序检查，并向所述上游节点发送针对所述第一上游数据报文的 ACK 报文。

21、根据权利要求 20 所述的服务器，其特征在于，所述控制管理模块，还用于在所述数据转发模块将第四下游数据报文发送给所述下游节点或所述用户终端后，将基于所述第二上游数据报文拷贝得到的第二份报文转换成第五下游数据报文；将所述第五下游数据报文存储在缓冲区中并设置超时定时器；当所述超时定时器超时，向所述下游节点或所述用户终端发送所述第五下游数据报文。

22、根据权利要求 21 所述的服务器，其特征在于，所述连接映射表中一个上游连接信息与一个下游连接信息对应，

所述控制管理模块，还用于当在所述超时定时器超时之前接收到所述下游节点或所述用户终端返回的 ACK 报文时，删除所述缓冲区中存储的所述第五下游数据报文。

23、根据权利要求 19 所述的服务器，其特征在于，所述控制管理模块，还用于若在所述第一接收模块接收到所述上游节点重传的所述第三上游数据报文之前，第二接收模块接收到所述下游节点或所述用户终端发送的 M 次重复 ACK，则等待所述上游节点重传所述第三上游数据报文，所述 M 次重复 ACK 是针对所述第一上游数据报文的；

所述数据转发模块，还用于在接收到重传的所述第三上游数据报文后，执行根据所述连接映射表，对基于所述第三上游数据报文拷贝得到的两份报文中的第一份报文进行转换处理，将转换得到的第四下游数据报文发送给所述下游节点或所述用户终端的步骤。

24、根据权利要求 19 所述的服务器，其特征在于，所述控制管理模块，还用于若在所述第一接收模块接收到所述上游节点重传的所述第三上游数据报文之后，第二接收模块接收到所述下游节点或所述用户终端发送的 M 次重复 ACK，则将缓冲区中存储的与所述第三上游数据报文匹配的第六下游数据报文发送给所述下游节点或所述用户终端。

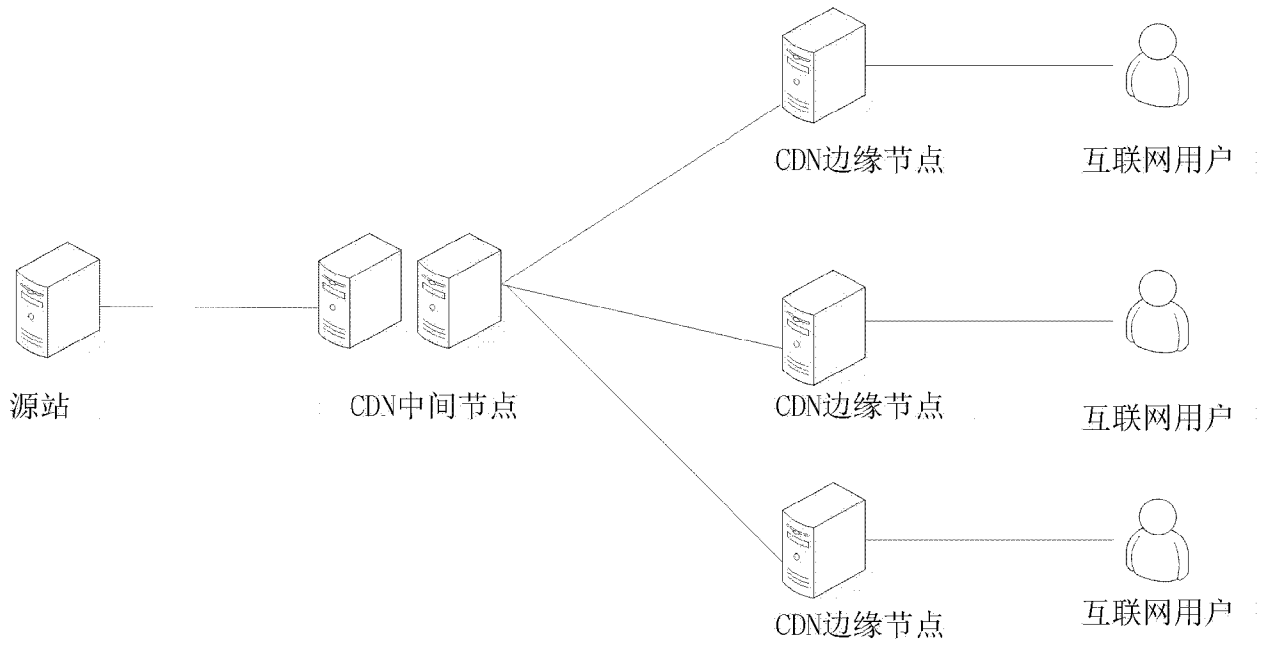


图 1

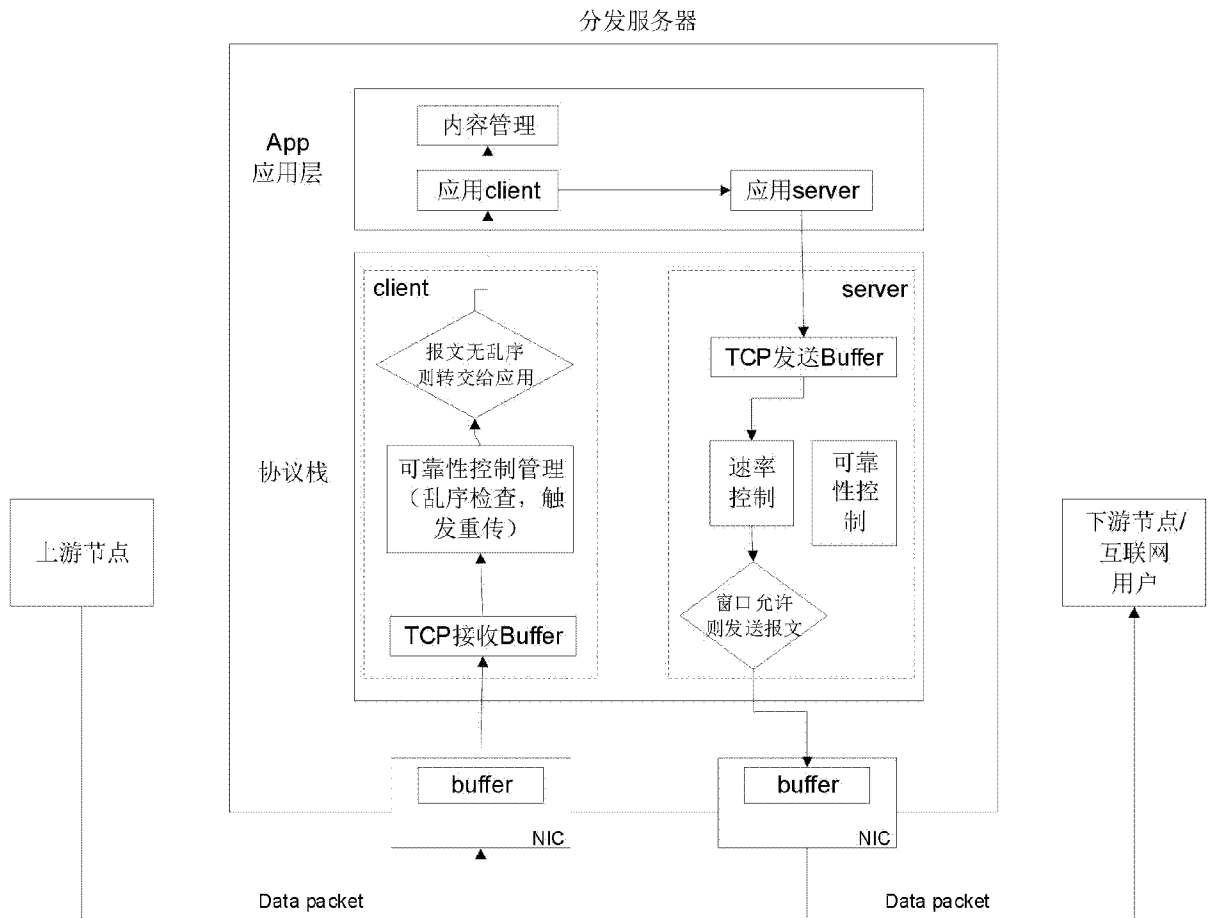


图 2

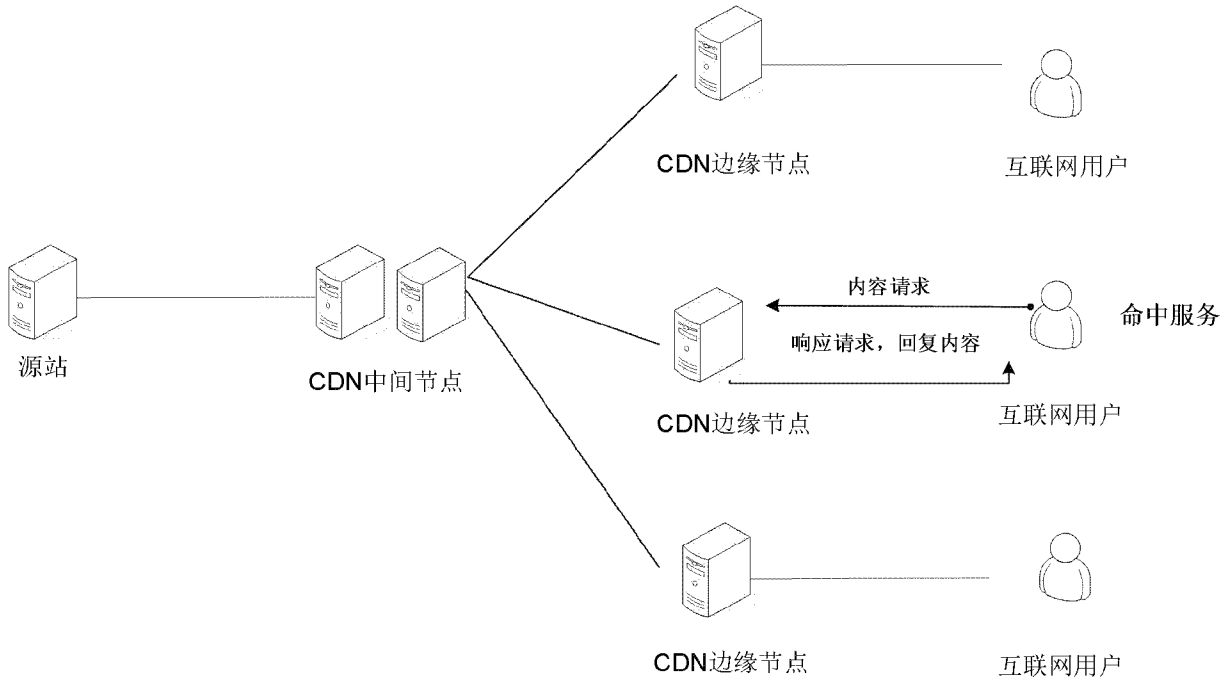


图 3

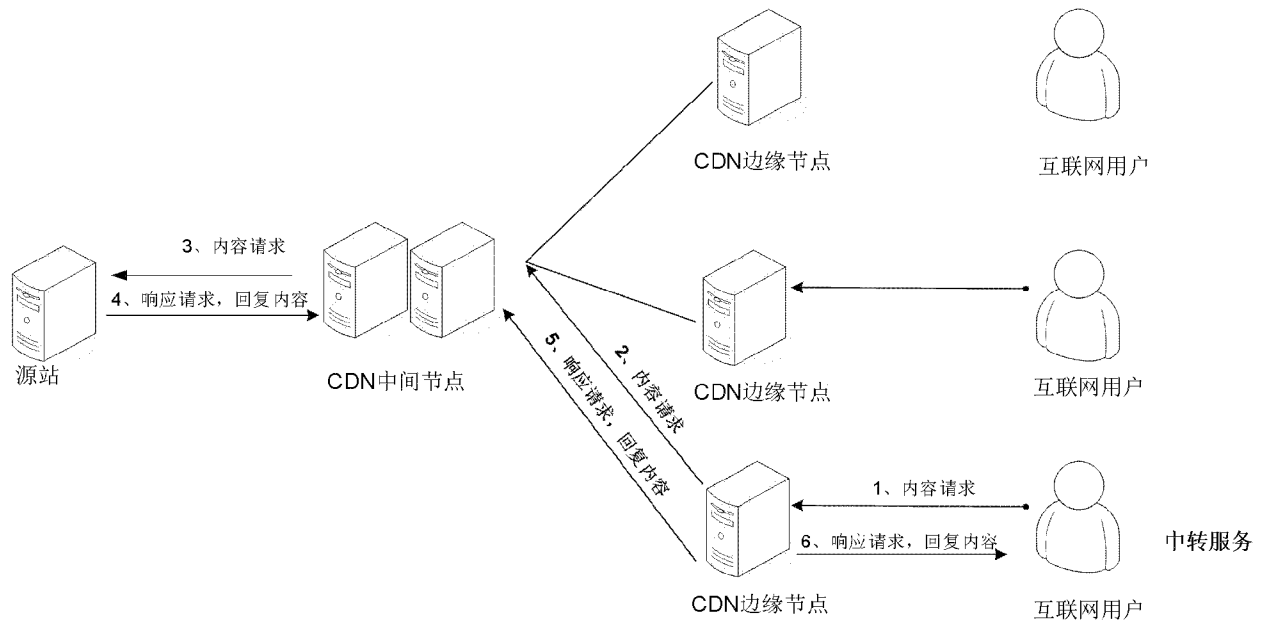


图 4

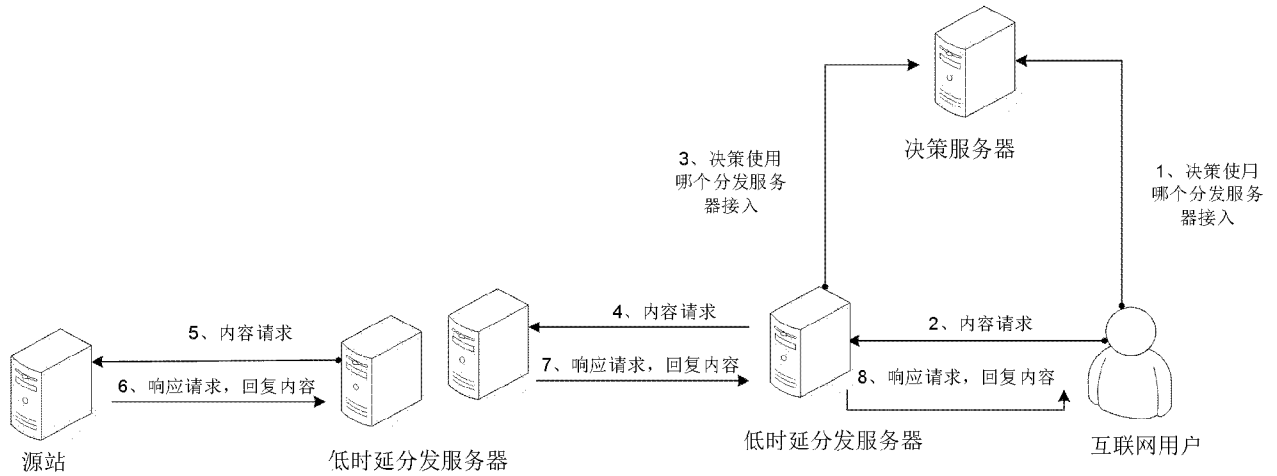


图 5

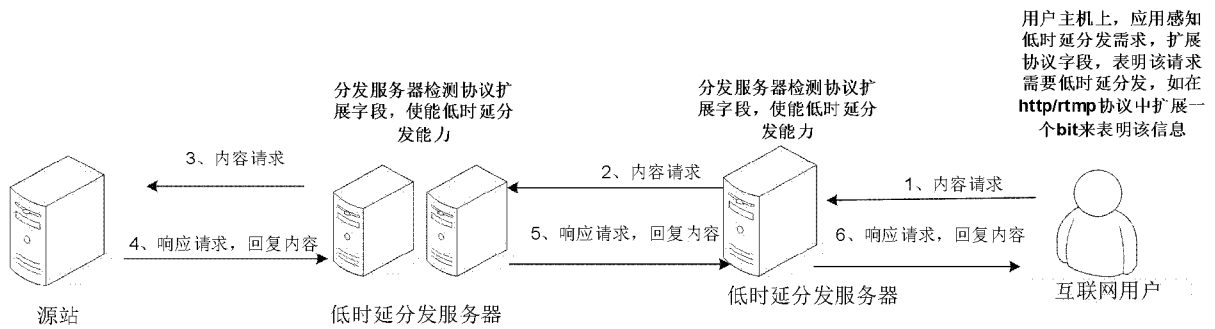


图 6

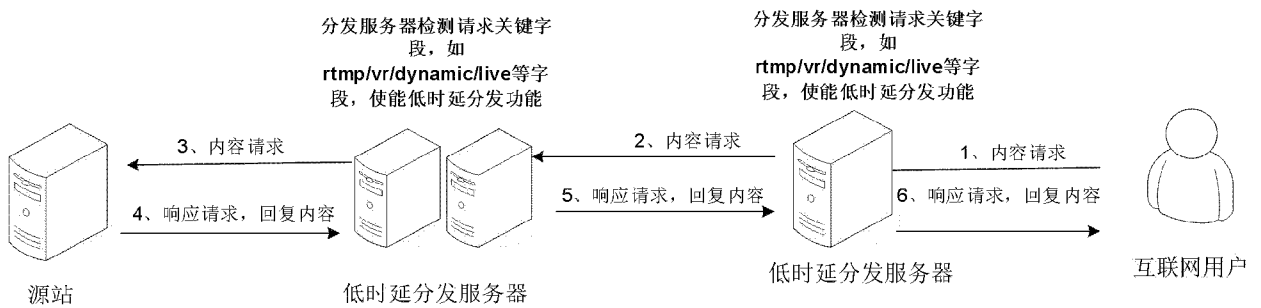


图 7

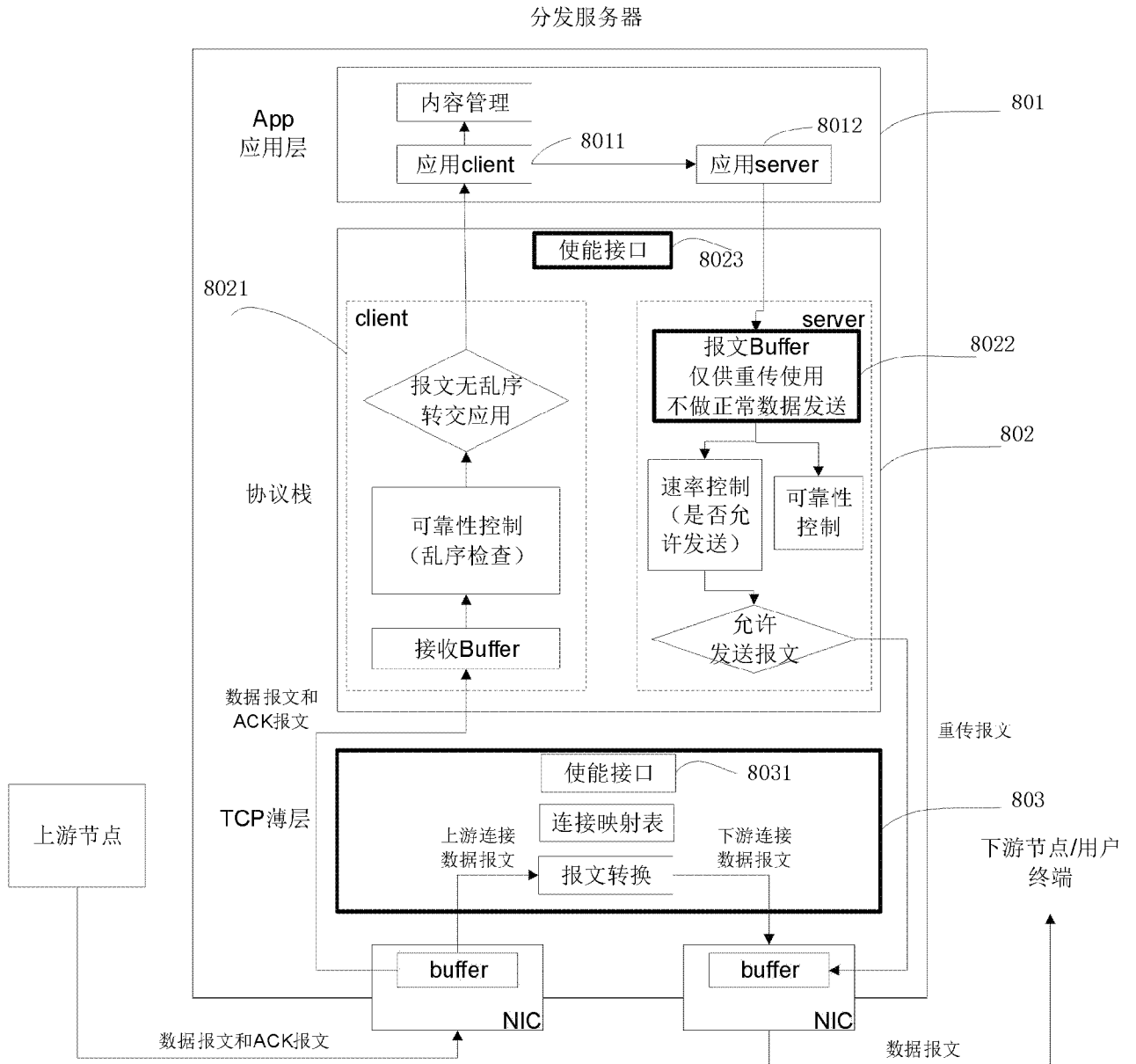


图 8

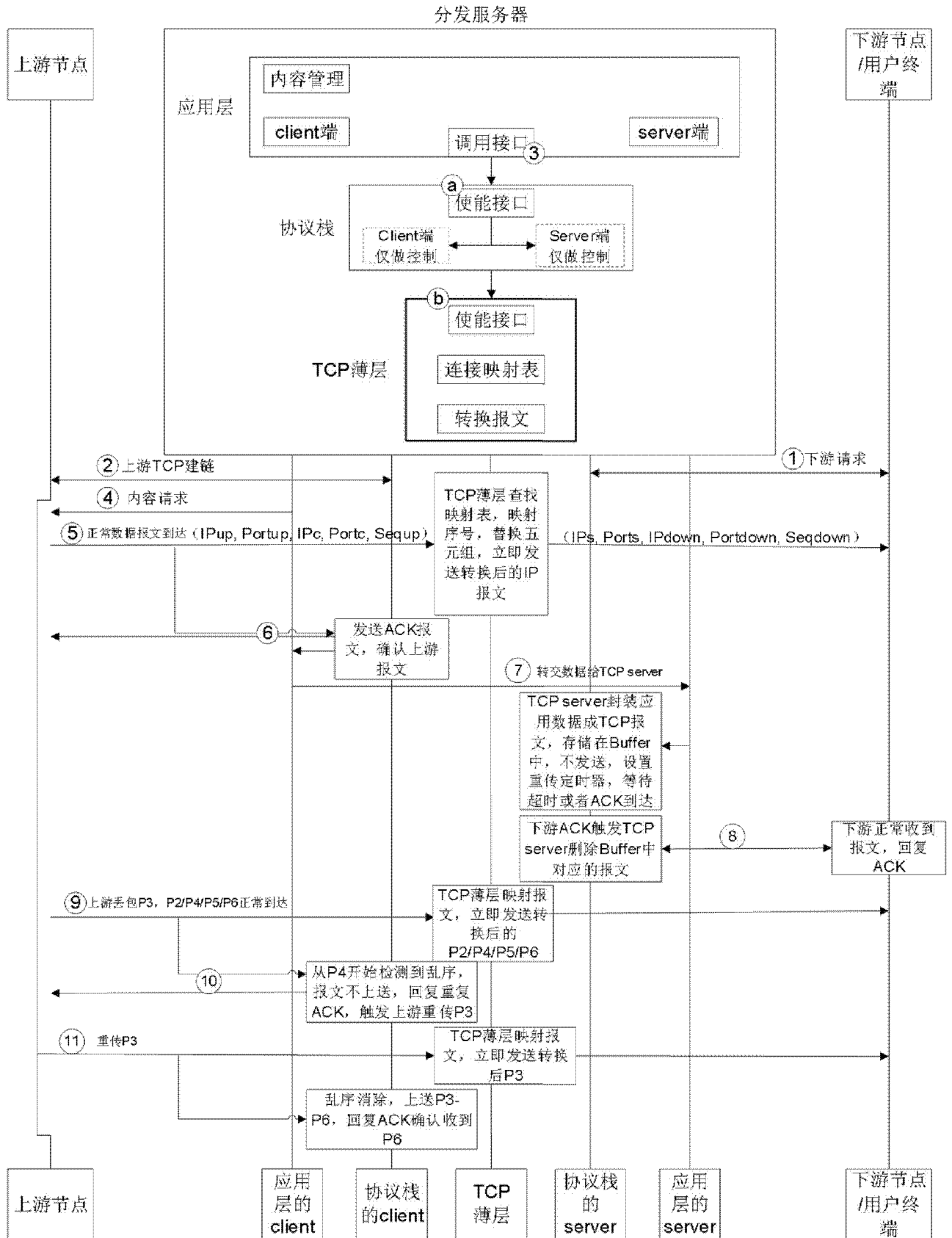


图 9

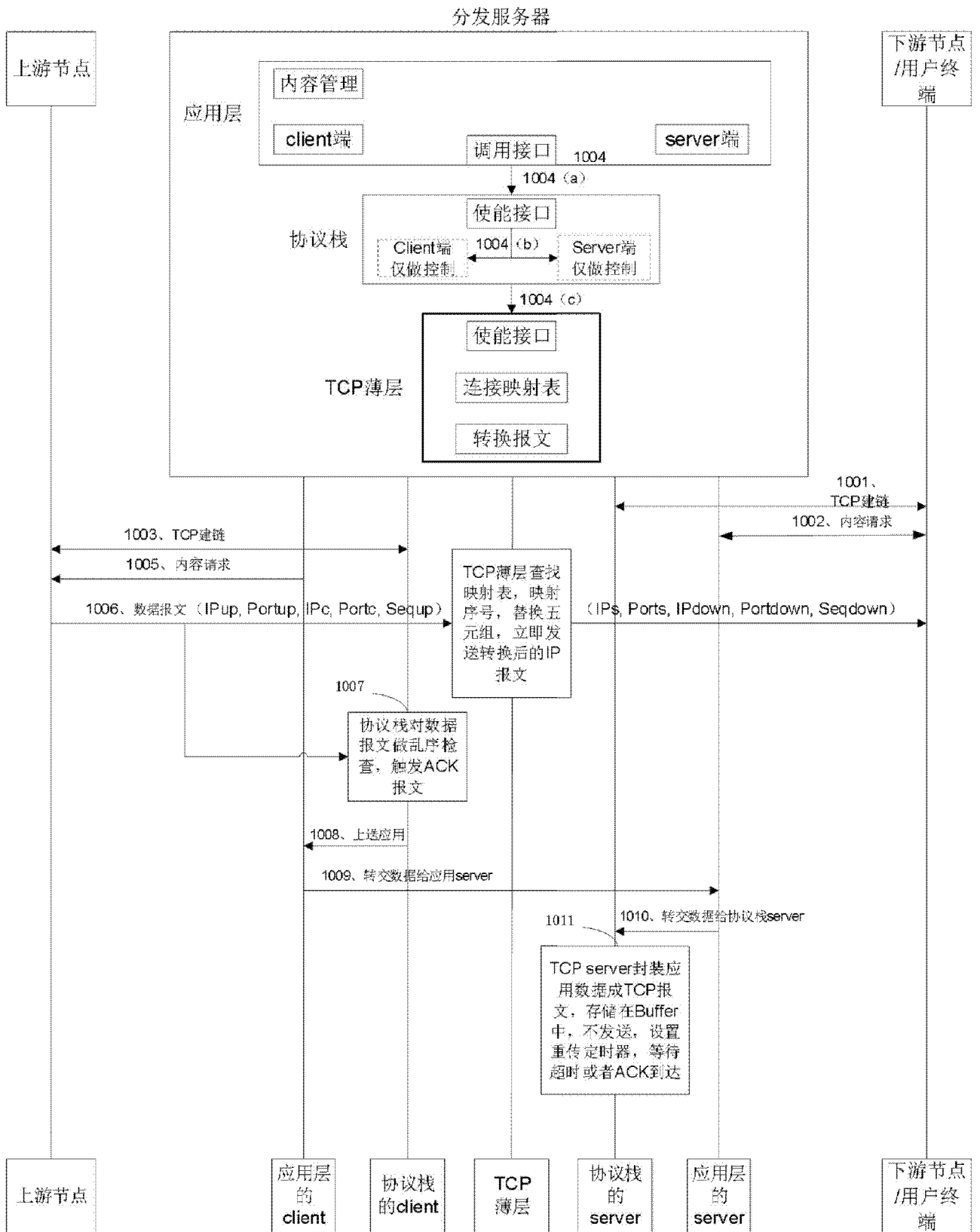


图 10

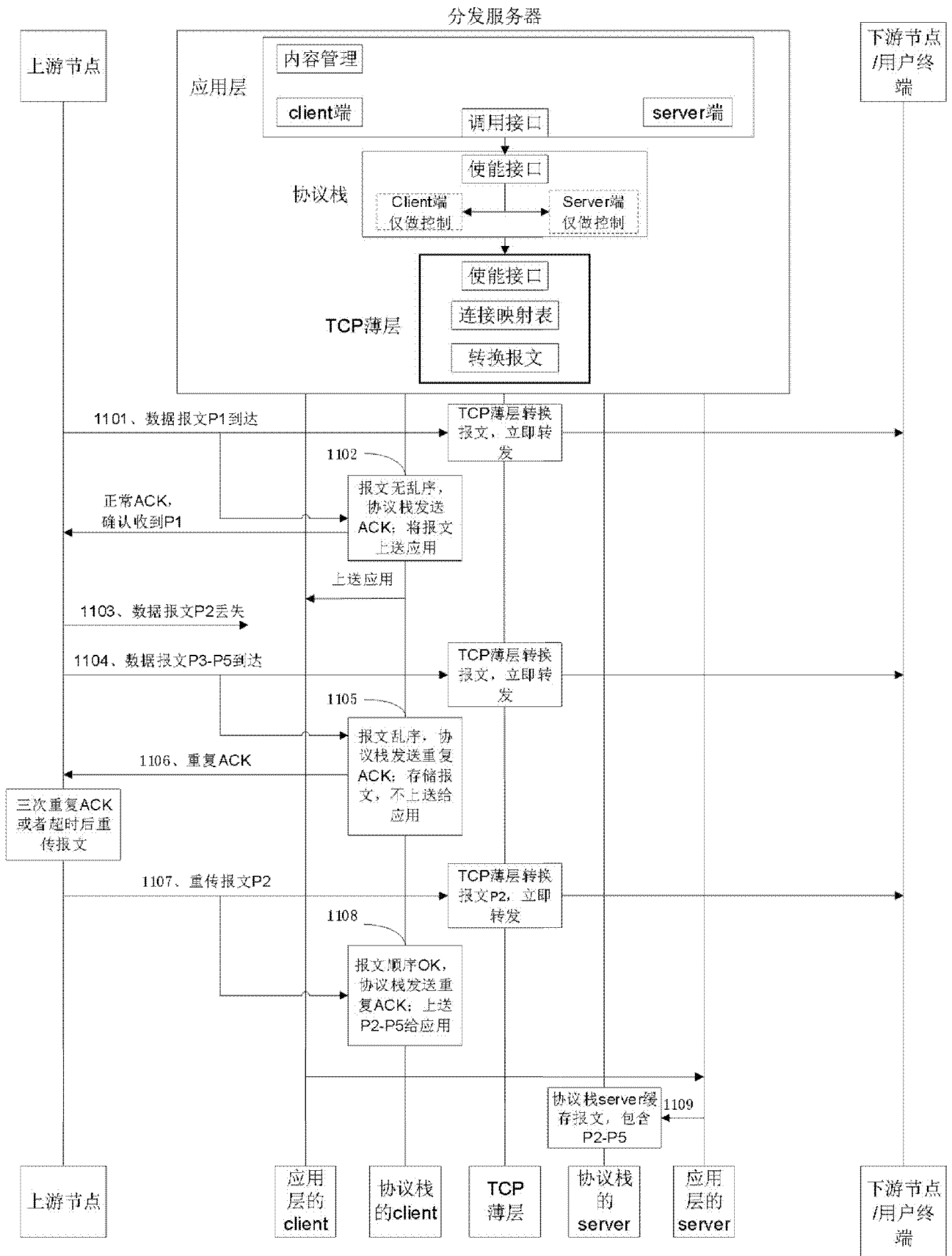


图 11

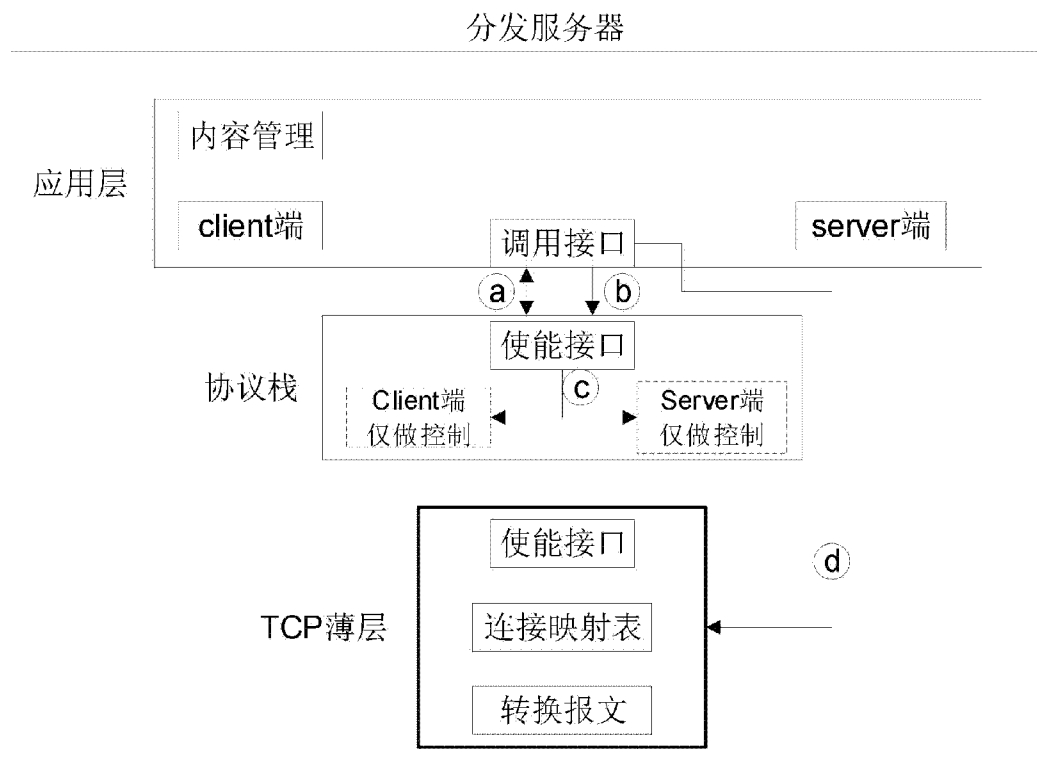


图 12

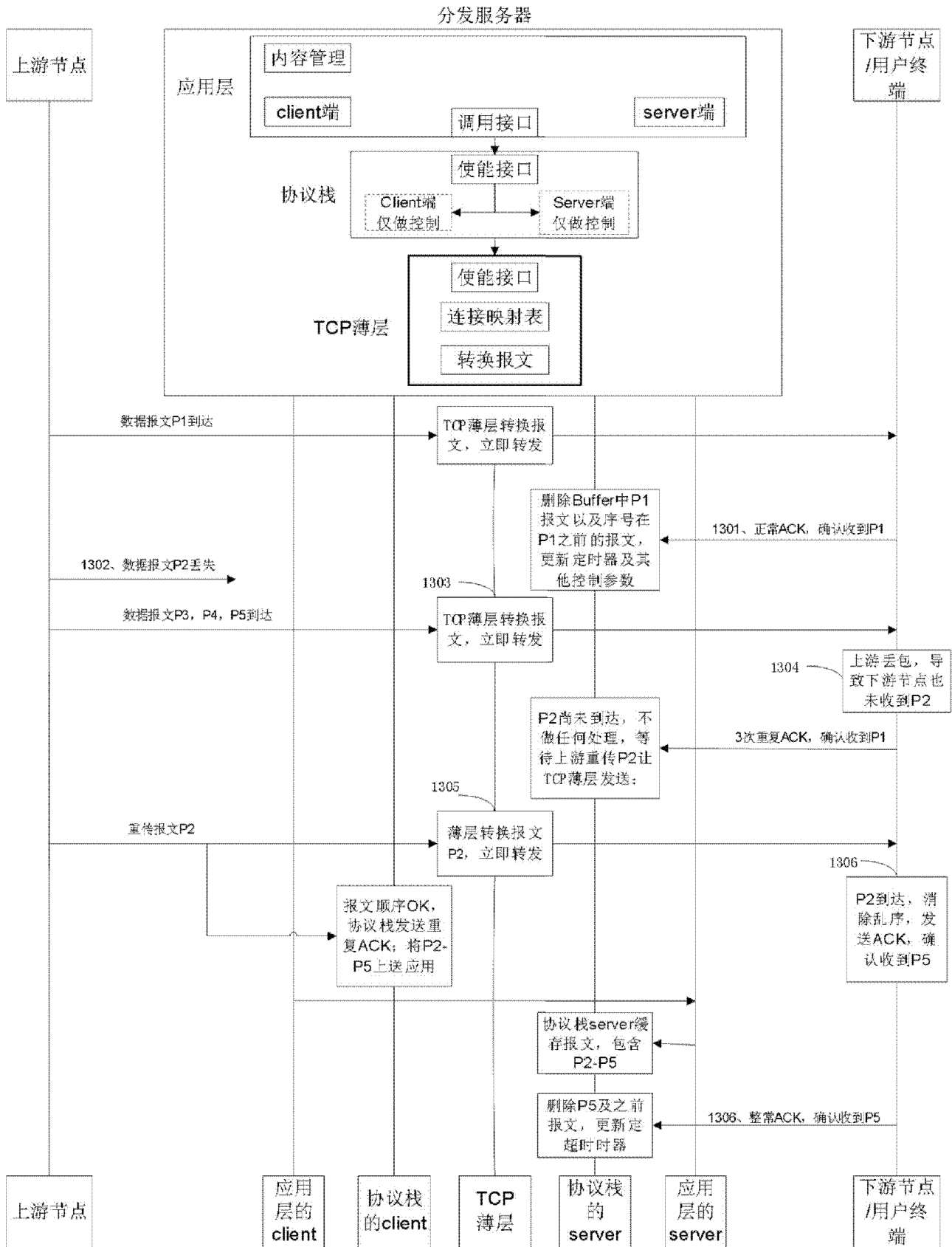


图 13

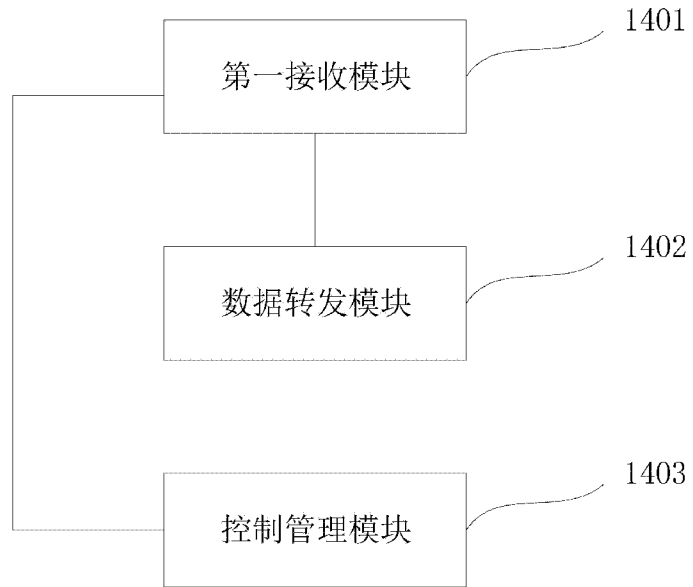


图 14

INTERNATIONAL SEARCH REPORT

International application No.

PCT/CN2018/112595

A. CLASSIFICATION OF SUBJECT MATTER

H04L 7/00(2006.01)i

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

H04L; G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

CNABS, CNTXT, VEN, DWPI, CNKI: 数据分发, 分发网络, 报文, 复制, 拷贝, 副本, 中转, 控制, 协议栈, 应用层, content delivery, CDN, message, copy, transfer, control, protocol stack, application layer

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	CN 101132269 A (ZTE CORPORATION) 27 February 2008 (2008-02-27) entire document	1-24
A	CN 103119958 A (SHARP CORPORATION) 22 May 2013 (2013-05-22) entire document	1-24
A	US 8782285 B1 (LIMELIGHT NETWORKS INC.) 15 July 2014 (2014-07-15) entire document	1-24

 Further documents are listed in the continuation of Box C. See patent family annex.

* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search

18 January 2019

Date of mailing of the international search report

29 January 2019

Name and mailing address of the ISA/CN

**National Intellectual Property Administration, PRC (ISA/
CN)
No. 6, Xitucheng Road, Jimenqiao Haidian District, Beijing
100088
China**

Facsimile No. (86-10)62019451

Authorized officer

Telephone No.

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.

PCT/CN2018/112595

Patent document cited in search report			Publication date (day/month/year)	Patent family member(s)			Publication date (day/month/year)
CN	101132269	A	27 February 2008	CN	101132269	B	23 June 2010
CN	103119958	A	22 May 2013	EP	2597869	A1	29 May 2013
				BR	112013001376	A8	17 October 2017
				JP	WO2012011450	A1	09 September 2013
				EP	2597869	A4	18 December 2013
				WO	2012011450	A1	26 January 2012
				US	2013117413	A1	09 May 2013
				BR	112013001376	A2	17 May 2016
				JP	2016181289	A	13 October 2016
US	8782285	B1	15 July 2014	None			

国际检索报告

国际申请号

PCT/CN2018/112595

<p>A. 主题的分类</p> <p>H04L 7/00 (2006.01) i</p> <p>按照国际专利分类(IPC)或者同时按照国家分类和IPC两种分类</p>														
<p>B. 检索领域</p> <p>检索的最低限度文献(标明分类系统和分类号)</p> <p>H04L; G06F</p> <p>包含在检索领域中的除最低限度文献以外的检索文献</p> <p>在国际检索时查阅的电子数据库(数据库的名称, 和使用的检索词(如使用))</p> <p>CNABS, CNTXT, VEN, DWPI, CNKI: 数据分发, 分发网络, 报文, 复制, 拷贝, 副本, 中转, 控制, 协议栈, 应用层, content delivery, CDN, message, copy, transfer, control, protocol stack, application layer</p>														
<p>C. 相关文件</p> <table border="1"> <thead> <tr> <th>类型*</th> <th>引用文件, 必要时, 指明相关段落</th> <th>相关的权利要求</th> </tr> </thead> <tbody> <tr> <td>A</td> <td>CN 101132269 A (中兴通讯股份有限公司) 2008年 2月 27日 (2008 - 02 - 27) 全文</td> <td>1-24</td> </tr> <tr> <td>A</td> <td>CN 103119958 A (夏普株式会社) 2013年 5月 22日 (2013 - 05 - 22) 全文</td> <td>1-24</td> </tr> <tr> <td>A</td> <td>US 8782285 B1 (LIMELIGHT NETWORKS INC) 2014年 7月 15日 (2014 - 07 - 15) 全文</td> <td>1-24</td> </tr> </tbody> </table>			类型*	引用文件, 必要时, 指明相关段落	相关的权利要求	A	CN 101132269 A (中兴通讯股份有限公司) 2008年 2月 27日 (2008 - 02 - 27) 全文	1-24	A	CN 103119958 A (夏普株式会社) 2013年 5月 22日 (2013 - 05 - 22) 全文	1-24	A	US 8782285 B1 (LIMELIGHT NETWORKS INC) 2014年 7月 15日 (2014 - 07 - 15) 全文	1-24
类型*	引用文件, 必要时, 指明相关段落	相关的权利要求												
A	CN 101132269 A (中兴通讯股份有限公司) 2008年 2月 27日 (2008 - 02 - 27) 全文	1-24												
A	CN 103119958 A (夏普株式会社) 2013年 5月 22日 (2013 - 05 - 22) 全文	1-24												
A	US 8782285 B1 (LIMELIGHT NETWORKS INC) 2014年 7月 15日 (2014 - 07 - 15) 全文	1-24												
<p><input type="checkbox"/> 其余文件在C栏的续页中列出。</p> <p><input checked="" type="checkbox"/> 见同族专利附件。</p>														
<p>* 引用文件的具体类型:</p> <p>“A” 认为不特别相关的表示了现有技术一般状态的文件</p> <p>“E” 在国际申请日的当天或之后公布的在先申请或专利</p> <p>“L” 可能对优先权要求构成怀疑的文件, 或为确定另一篇引用文件的公布日而引用的或者因其他特殊理由而引用的文件(如具体说明的)</p> <p>“O” 涉及口头公开、使用、展览或其他方式公开的文件</p> <p>“P” 公布日先于国际申请日但迟于所要求的优先权日的文件</p> <p>“T” 在申请日或优先权日之后公布, 与申请不相抵触, 但为了理解发明之理论或原理的在后文件</p> <p>“X” 特别相关的文件, 单独考虑该文件, 认定要求保护的发明不是新颖的或不具有创造性</p> <p>“Y” 特别相关的文件, 当该文件与另一篇或者多篇该类文件结合并且这种结合对于本领域技术人员为显而易见时, 要求保护的发明不具有创造性</p> <p>“&” 同族专利的文件</p>														
<p>国际检索实际完成的日期</p> <p>2019年 1月 18日</p>		<p>国际检索报告邮寄日期</p> <p>2019年 1月 29日</p>												
<p>ISA/CN的名称和邮寄地址</p> <p>中国国家知识产权局(ISA/CN) 中国北京市海淀区蓟门桥西土城路6号 100088</p> <p>传真号 (86-10)62019451</p>		<p>受权官员</p> <p>李晓</p> <p>电话号码 86-(010)-62412283</p>												

国际检索报告
关于同族专利的信息

国际申请号

PCT/CN2018/112595

检索报告引用的专利文件			公布日 (年/月/日)	同族专利			公布日 (年/月/日)
CN	101132269	A	2008年 2月 27日	CN	101132269	B	2010年 6月 23日
CN	103119958	A	2013年 5月 22日	EP	2597869	A1	2013年 5月 29日
				BR	112013001376	A8	2017年 10月 17日
				JP	W02012011450	A1	2013年 9月 9日
				EP	2597869	A4	2013年 12月 18日
				WO	2012011450	A1	2012年 1月 26日
				US	2013117413	A1	2013年 5月 9日
				BR	112013001376	A2	2016年 5月 17日
				JP	2016181289	A	2016年 10月 13日
US	8782285	B1	2014年 7月 15日	无			

表 PCT/ISA/210 (同族专利附件) (2015年1月)