(19)中华人民共和国国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 106202139 A
(43)申请公布日 2016.12.07

(21)申请号 201610313237.4

(22)申请日 2016.05.12

(30)优先权数据
14/727,478 2015.06.01 US

(71)申请人 阿里巴巴集团控股有限公司
地址 英属开曼群岛大开曼资本大厦一座四层847号邮箱

(72)发明人 李舒 牛功彪

(74)专利代理机构 北京市清华源律师事务所 11441
代理人 沈泳 王永秀

(51)Int.Cl.
*G06F 17/30*(2006.01)
*G06F 11/10*(2006.01)

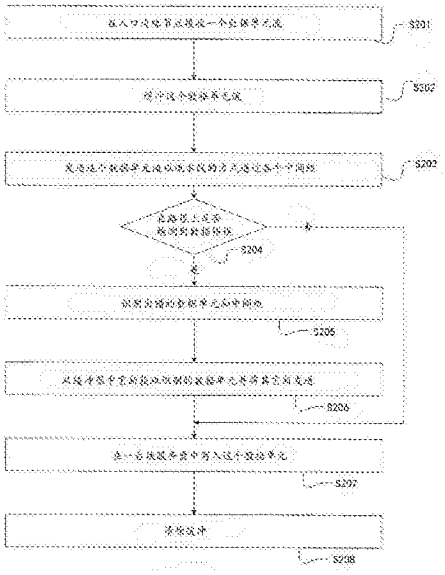
权利要求书3页 说明书8页 附图5页

(54)发明名称

通过缓冲入口数据增强云存储系统中数据一致性的数据存储方法和设备

(57)摘要

本申请提供了通过缓冲入口数据增强云存储系统中数据一致性的数据存储方法和设备。在本申请的技术方案中，云系统利用位于入口级服务器的非易失性缓冲器来缓冲所接收的输入数据。数据从入口级经由数据路径上的各个处理级传输，直到被写入云系统的目标存储设备。数据按照基于事件的时间表以流水线的方式传输通过各个级。各个级都能够从上一个级对这个数据进行接收和/或处理，验证其数据一致性，以及将其发送到下一级。如果在数据路径上检测到数据错误，则从非易失性缓冲器中恢复所标识的数据，将其插入数据流中并重新发送到这条数据路径上。



CN 106202139 A

1.一种通过缓冲入口数据增强云存储系统中数据一致性的数据存储方法，所述方法包括：

在所述云存储系统的入口节点处接收数据单元流，其中，所述云存储系统包括数据路径，所述数据路径包括所述入口节点、中间节点以及目标存储节点；

在所述入口节点处缓冲所述数据单元流；

在将所述数据单元流存储于所述目标存储节点之前，将所述数据单元流依次发送至所述中间节点，以流水线的方式进行缓存；以及

在接收到针对所标识的数据单元在所述数据路径上检测到错误的信号之后，从所述入口节点重新发送所标识的数据单元到所述数据路径上。

2.根据权利要求1所述的方法，其特征在于，所述重新发送包括：

推迟发送已经计划好在第一时刻发送至所述数据路径的下一个数据单元；

在所述第一时刻从所述入口节点发送所标识的数据单元。

3.根据权利要求1所述的方法，其特征在于，一个数据单元被缓存在第一中间节点中，直到发生前一个数据单元被在所述数据路径上与所述第一中间节点相邻的第二中间节点成功接收的事件为止。

4.根据权利要求1所述的方法，其特征在于，所述错误发生在下列过程中：在所述数据路径上的所标识的中间节点处接收所标识的数据单元的过程；在所标识中间节点处理所标识的数据单元的过程；或者从所标识的中间节点发送所标识的数据单元的过程。

5.根据权利要求1所述的方法，其特征在于，还包括：当所述数据单元流被成功存储于所述目标存储节点时，覆盖所述入口节点中的所述单元数据流。

6.根据权利要求1所述的方法，其特征在于，还包括：当多个后续的数据单元流被成功存储于所述目标存储节点时，从所述入口节点覆盖所述单元数据流。

7.根据权利要求1所述的方法，其特征在于，还包括：如果接收额外的数据单元导致所述入口节点处的缓冲溢出，则向用户发送表明所述云存储系统中的数据传输被延迟的消息。

8.根据权利要求1所述的方法，其特征在于，所述缓冲包括在非易失性存储器中以先进先出的方式缓冲所述数据单元流直到检测到所述错误。

9.一种通过缓冲入口数据增强云存储系统中数据一致性的数据存储设备，所述设备包括：

处理器；

与所述处理器相连接的通信电路，其中，所述通信电路还经由网络与所述云系统相连接；

与所述处理器相连接的存储器，所述存储器包含所述处理器可执行的指令，其中，所述指令实施包括以下步骤的方法：

使所述云系统的缓冲器存储由所述云系统接收的输入数据；

从所述云系统的数据路径接收错误信号，所述错误信号指示了在所述输入数据经由所述数据路径向目标存储服务器传输的过程中的数据一致性；以及

响应于所述错误信号，使所述缓冲器向所述目标存储服务器重新发送所述输入数据。

10.根据权利要求9所述的设备，其特征在于，所述缓冲器被部署在所述云系统的入口

节点处;所述入口节点设置用于接收从互联网传输来的所述输入数据;所述输入数据的所述传输包括在所述入口节点与所述目标存储服务器之间以流水线的方式穿过所述数据路径上的中间节点的传输。

11.根据权利要求10所述的设备,其特征在于,所述缓冲器包括桶型移位器;所述缓冲器的深度与所述数据路径上的所述中间节点的数目相关;所述方法还包括:接收所述输入数据已经被成功存储于所述目标存储服务器的确认;以及从所述缓冲器中移除所述输入数据。

12.根据权利要求10所述的设备,其特征在于,所述中间节点中的各个中间节点被设置为:

从所述数据路径上的上游中间节点接收所述输入数据的第一数据单元,并生成安全接收事件的信号;

处理所述第一数据单元;

验证所述第一数据单元的数据一致性;并且

如果数据一致性验证成功,则将所述第一数据单元发送至所述数据路径上的下游中间节点,并生成安全通过事件的信号;

如果检测到数据不一致,则生成错误事件的信号。

13.根据权利要求9所述的设备,其特征在于,所述缓冲器包括混合双列直插存储模块。

14.根据权利要求12所述的设备,其特征在于,所述方法还包括:

基于所述错误信号,标识所述输入数据的数据单元;

在第一时刻推迟发送已经计划好要发送到所述数据路径上的下一个数据单元;

通知所述入口节点在所述第一时刻在所述数据路径上从所述缓冲器重新发送所标识的数据单元。

15.根据权利要求10所述的设备,其特征在于,所述方法还包括:

如果接收额外的数据单元导致所述缓冲器的缓冲溢出,则向与所述云系统相连接的用户设备发送消息,所述消息表明所述云系统中的数据传输被延迟。

16.一种非瞬时性计算机可读存储介质,包含指令集;所述指令集在由处理设备执行时,使所述处理设备实施通过缓冲入口数据增强云存储系统中数据一致性的数据存储方法:

在所述云存储系统的入口节点处接收数据单元流,其中,所述云存储系统包括数据路径,所述数据路径包括所述入口节点、中间节点和目标存储节点;

在所述入口节点处缓冲所述数据单元流;

在将所述数据单元流存储于所述目标存储节点之前,将所述数据单元流依次发送至所述中间节点,以流水线的方式进行缓存;以及

在接收到针对所标识的数据单元在所述数据路径上检测到错误的信号之后,从所述入口节点重新发送所标识的数据单元到所述数据路径上。

17.根据权利要求16所述的非瞬时性计算机可读存储介质,其特征在于,所述重新发送包括:

在第一时刻推迟发送已经计划好发送至所述数据路径的下一个数据单元;

在所述第一时刻从所述入口节点发送所标识的数据单元。

18.根据权利要求16所述的非瞬时性计算机可读存储介质，其特征在于，各个数据单元被缓存在第一中间节点中，直到发生前一个数据单元被所述数据路径上与所述第一中间节点相邻的第二中间节点成功接收的事件为止。

19.根据权利要求16所述的非瞬时性计算机可读存储介质，其特征在于，所述方法还包括：当两个以上后续的数据单元流被成功存储于所述目标存储节点时，从所述入口节点覆盖所述单元数据流。

20.根据权利要求16所述的非瞬时性计算机可读存储介质，其特征在于，所述缓冲包括在非易失性存储器中缓冲所述数据单元流，并且所述方法还包括：如果接收额外的数据单元导致所述入口节点处的缓冲溢出，则向用户设备发送表明所述云存储系统中的数据传输被延迟的消息。

# 通过缓冲入口数据增强云存储系统中数据一致性的数据存储方法和设备

[0001] 本申请要求于2015年6月1日提交美国专利商标局、申请号为US14727478、发明名称为“ENHANCING DATA CONSISTENCY IN CLOUD STORAGE SYSTEM BY ENTRANCE DATA BUFFERING”的美国发明专利申请的优先权，其全部内容通过引用结合在本申请中。

## 技术领域

[0002] 本申请涉及云存储系统领域，具体涉及通过缓冲入口数据增强云存储系统中数据一致性的数据存储方法。本申请还涉及通过缓冲入口数据增强云存储系统中数据一致性的数据存储设备以及非瞬时性计算机可读存储介质。

## 背景技术

[0003] 云存储系统为用户在互联网上存储数字数据充当虚拟池。云存储系统包含多个物理存储服务器(通常位于多个位置)，一般由一家网络服务公司所拥有和管理。这些云存储供应商负责使数据保持为可用和可存取，并负责物理环境的保护和运行。个人和机构用户从这些供应商购买或租赁存储容量来存储数据或应用。

[0004] 通常在用户数据真正被写入一个云存储系统的块服务器之前，需要通过一个很长的数据路径。这个数据路径一般包括数个层，每一个层都可能有对数据进行缓冲、缓存和/或处理的多个级，例如压缩和加密。众所周知，数据一致性对云存储服务至关重要。但是，在数据传输、缓存和处理的过程中，硬件、软件和通信中的各种问题都可能造成数据的不一致，从而导致数据不能如实和安全地写入块服务器的存储盘。例如，由于软件或缓存的漏洞、非正常系统行为、系统电力故障、内存的位翻转错误、通信干扰等原因，用户数据可能会被无意地更改。

[0005] 一种常规的解决方法是，使用基于各种一致性模型的软件来控制数据的一致性。遗憾的是，这种软件倾向于产生错误的一致性结果，通常不可靠，并且 软件自身也易于引起数据错误。此外，这种方法还对由于数据中心的意外断电或重启、硬盘驱动程序或固件的漏洞、硬盘控制器的问题等造成的数据错误不起作用。

[0006] 另一种常规的解决方法依赖元数据来恢复不一致的数据。但是，在文件系统或元数据中的指定路径出现问题的情况下，元数据本身无法使用，不能用来进行数据恢复。

## 发明内容

[0007] 因此，提供一种在云存储系统的数据传输过程中保持数据一致性可靠而有效的机制是有利的。

[0008] 本申请所公开的实施方式在云存储系统的入口级(例如，前端服务器)采用非易失性缓冲器来缓冲接收到的输入数据。在被写入云存储系统的目标存储盘之前，数据经过一条从入口级贯穿各个处理级的数据路径。各个级都能够从上一级接收数据单元，缓存和/或处理该数据单元，验证数据一致性，并发送至下一级。数据按照基于事件的时间表，以流水

线的方式传输通过各个级。如果在数据路径上发现与数据单元的数据一致性有关的错误，则向入口级发送一个恢复该数据单元的请求。作为回应，从入口级的缓冲器中获取接收到的数据单元，并插入到数据流中，重新发送到上述数据路径上。

[0009] 入口级的缓冲器可以实现为使用可靠的非易失性存储模块的日志结构的缓冲器或者桶型移位器。可以选择缓冲器的深度以匹配数据路径上的级数。

[0010] 由于最初的输入数据存储于云存储系统入口处的非常可靠的缓冲器中，所以在数据路径上发现数据不一致时，可以获取最初的数据并重新发送。因此，本公开可以更好地保证写入和存储在云存储系统中的数据与接收到的数据具有一致性。不仅如此，无需在云系统中引入复杂昂贵的硬件设备就可以更好地增强云系统中的数据一致性。

[0011] 本申请的一个实施方式提供了一种在云存储系统中存储数据的计算机实现的方法，该方法包括：在云存储系统的入口边缘节点接收数据单元流；其中，该云存储系统包括数据路径，该数据路径包含上述入口节点、中间节点以及目标存储节点。这些数据单元在入口节点处被缓冲，并依次发送至中间节点以流水线的方式进行缓存，直至它们存储于目标存储节点。一旦所标识的数据单元 接收到在数据路径上检测到错误的信号，就从入口节点重新发送所标识的数据单元到上述数据路径上。

[0012] 发明内容部分对本申请的技术方案进行了必要的简化、概括并省略了细节。本领域技术人员应当认识到发明内容仅为说明性的，绝非要限制本申请。在权利要求书中单独定义的本申请的其他方面、创新特点和优点，将通过下面的非限制性详细说明而变得明了。

## 附图说明

[0013] 参照附图和下面的详细描述有助于更好地理解本申请提供的各实施方式；附图中相同的标号表示相同的要素：

[0014] 图1A例示了本申请实施方式的示例性云系统，该云系统能够在输入数据被写入存储服务器之前在入口边缘处缓冲这个输入数据；

[0015] 图1B例示了本申请实施方式的云存储系统中的示例数据路径；

[0016] 图2例示了本申请实施方式的示例性计算机实现的传输输入数据以存储于云系统中的处理过程流程图；

[0017] 图3是例示了在本申请实施方式的示例性云系统中进行数据传输过程中级状态时间变化图；

[0018] 图4A是例示了本申请实施方式的从入口节点处的缓冲器重新发送不一致的数据单元的级状态时间变化图；

[0019] 图4B是例示了本申请实施方式的从入口节点重新发送不一致的数据单元并插入到数据流中的级状态时间变化图；

[0020] 图5A是例示了本申请实施方式的一组数据单元出现错误并从入口节点重新发送的级状态时间变化图；

[0021] 图5B是例示了本申请实施方式的一组数据单元出现错误并从入口节点重新发送以及被插入数据流的级状态时间变化图；

[0022] 图6A是例示了本申请实施方式的位于云系统入口级的示例桶型移位缓冲器的级状态时间变化图；

[0023] 图6B例示了本申请实施方式的将输入数据推入位于云系统入口级的示例桶型移位缓冲器的顺序；

[0024] 图6C例示了本申请实施方式的将接收的数据推入位于云系统入口级的日志结构缓冲器的顺序；

[0025] 图7例示了本申请实施方式的被配置为对云存储系统中的数据传输进行管理的示例性计算系统。

## 具体实施方式

[0026] 下面将详细说明本申请的优选实施方式，附图中例示了优选实施方式的实施例。虽然将结合这些优选实施方式来描述本申请，但是应当理解，并不是要将本申请限于这些实施方式。相反，本申请旨在涵盖可能包含在权利要求所限定的本申请的主旨和范围内的另选例、变形例和等同形式。此外，在下面详细描述的本申请的实施方式中，阐述的大量具体细节是为了提供对本申请的透彻理解。然而，本领域技术人员应当认识到，本申请的实现不依赖于这些具体细节。在其他实例中，并未详细描述公知的方法、过程、组件和电路，是为了不对本申请的实施方式产生不必要的影响。本申请的附图大体是示意性的，并非按照比例绘制，尤其是有些尺寸为了表示清楚而被放大。类似的是，尽管附图中的视图为了便于说明而通常示出了类似的取向，但是途中的描述对于大多数部件是任意的。一般来说，本申请可以在任意取向上操作。

[0027] 然而，所有这些和类似术语都要和适当的物理量结合，只是适用于这些量的方便的标签。除非从以下说明中明显得出的相反意思，应该认为贯穿本申请，利用诸如“处理”或“存取”或“执行”或“存储”或“呈现”此类术语的讨论是指计算机系统或者类似电子计算设备的对计算机系统的寄存器中的表示为物理(电子)量的数据进行操纵并转换为计算机系统存储器或寄存器或者其他这种信息存储装置、传输或客户设备内类似表示为物理量的其他数据的动作和处理。当某个部件出现在数个实施方式中时，使用相同标号来表示该部件与初始实施方式中的相同。总的来说，本公开的实施方式提供了一种云存储系统，其使用非易失性缓冲器在数据路径的入口处存储原始输入数据的副本。在最终被写入目标存储设备之前，输入数据会经过数据路径上的多个中间级。如果在传输过程中发现某一数据单元的数据不一致，则从缓冲器获取这个数据单元并重新发送到上述数据路径上。

[0028] 在本文中，除非特别说明，“级”和“节点”这两个术语可以互换使用。

[0029] 图1A例示了本申请实施方式的示例性网络100和云系统110，其能够在输入数据101被写入存储服务器之前在云系统110的入口处缓冲输入数据101。用户数据101来源于用户终端设备130并通过互联网120传输到云系统110进行存储。互联网服务供应商140控制用户对互联网120的访问。

[0030] 云系统110包括位于系统入口处的前端服务器111(或者入口边缘节点)、多个中间服务器112至115以及块服务器116A至116C。在被写入块服务器的非瞬时性存储介质之前，输入数据需要传输过这些中间服务器112至115。在此例中，中间服务器包括防火墙服务器112、应用交付控制器(ADC)113、认证服务器114，文件服务器115，等等。但是，需要指出，本公开并不受限于云系统的功能、组成部件、基础设施或体系结构，也不受限于传输和存储到云系统的数据的类型。服务器112至115通过网络相互通信，这个网络可以是私有网络、公共

网络、无线网、广域网、局域网、内联网、互联网、蜂窝网络，或者上述网络的组合。

[0031] 本申请的云系统110的边缘节点(在此为前端服务器111)包括缓冲器117，缓冲器117能够在原始输入数据101到达该云系统的边缘时存储其副本。在确认这个原始数据已经被正确地写入块服务器(例如，116A至116C)之前，原始数据一直保存在缓冲器117中。如果检测到某个数据单元的数据不一致，则指示前端服务器111从缓冲器117中恢复这个数据单元的原始副本并通过数据路径重新发送这个数据副本。数据路径上的每个服务器111至116C均被设置为从前一级的服务器接收数据单元、缓存和/或处理该数据单元，然后将该数据单元传递至后一级的服务器。由于每一个数据单元在通过中间级服务器时均进行了验证，所以可在写入块服务器之前捕获并恢复任何潜在的数据错误。因此，可以如实将输入数据101存储于云存储系统110中。

[0032] 图1B例示了本申请实施方式的云存储系统中的示例性数据路径150。数据路径150包括入口级(并未清楚示出)，这个入口级用于在云系统的入口边缘处接收输入数据。数据路径150还包括一系列级151至156，这些级能够在数据被物理写入存储设备156之前进行缓存和处理。例如，级151至156对应图1A中的各个服务器112至116C。

[0033] 在本例中，输入数据以数据单元为单位、以流水线的方式传输通过各个级151至156。位于入口级(并未清楚示出)的缓冲器177在云系统最初接收到输入数据161时对其进行存储。在数据传输过程中的某一特定时间，流水线的各 个级(级151至156)缓存了不同的数据单元，这一点将在下面进行更详细的描述。当确认数据161已经成功地并正确地写入目标存储器156(参见反馈线164)时，这个数据就可以从缓冲器177中清除。

[0034] 提高云系统数据一致性的一种常规方法是在系统中各个级的缓存中使用比回写更加可靠的直写。但是，这种方法不可避免且不期望地增加了数据传输延迟。另一种方法是在各个级使用非易失性存储器来缓存数据，但这种方法会增加相当高的系统成本。本申请的实施方式通过在输入数据被成功写入目标设备之前保持其原始副本，不需要进行昂贵的硬件升级和对数据路径上各个级进行专门配置，即可提高数据的一致性。

[0035] 输入数据中包含的每个数据单元都沿着数据路径150先后通过各个级151至156。每当数据单元通过一个级时，都验证离开这个级的数据单元是否与进入这个级的数据单元相匹配。这种验证可以由云系统的各个级或主控制器使用本领域公知的各种适当的方法来执行。如果在某一级检测到了数据不一致，则标识该数据单元并生成一个消息，指示缓冲器177重新发送所标识的数据单元到数据路径上。

[0036] 在一些实施方式中，每个级都配置成验证数据一致性，例如，使用循环冗余校验(CRC)。如果检测到数据单元以不期望的方式被改变，则该级报告一个错误，这个错误将传送给入口级以重新发送该数据单元。

[0037] 本申请的实施方式在数据沿着数据路径经过各个级时验证数据的一致性，从而保证最终写入目标存储设备的数据是没有错误的。由于在入口级一直可以获取原始输入数据，所以能够有利地捕获到由于数据路径上任何类型的行为所引起的数据错误，并通过重新发送该数据单元来进行恢复。

[0038] 本申请不限于云系统数据路径上的数据不一致的具体原因。当数据单元经过特定级时，在数据接收、缓存、处理和传输等行为过程中都可能产生数据错误。可能产生数据错误的原因有：意外断电、硬件或软件漏洞、系统不正常动作、内存位翻转、通信干扰，等。

[0039] 图2是本申请实施方式的示例性计算机实现的处理过程S200的流程图，该处理过程S200传输输入数据以在云系统中存储。在步骤S201，在云系统的入口级处接收数据单元流。入口级包括缓冲器，该缓冲器用于在步骤S202缓冲接收到的数据。在步骤S203，这些数据单元从入口级被发送并以流水线的方式通过 各个中间级。

[0040] 在每个级将数据单元传递给下一个级之前，在步骤S204验证这个数据单元的数据一致性。如果没有错误，则可以将这个数据单元传递给下一个级。如果检测到数据错误，则在步骤S205，例如根据中间级的标识和报错的时间，来识别该数据单元。作为回应，在步骤S206生成一个重新发送该数据单元的请求，并指示入口级从缓冲器重新发送所标识的数据单元。在数据单元成功穿过所有级之后，在步骤S207该数据单元被写入块服务器进行存储。如果确认流中的所有数据单元均具有一致性并被正确地写入到块服务器，则在步骤S208可以清除或覆盖缓冲器中维护的数据副本。

[0041] 图3是例示了本申请实施方式的云系统的数据传输过程中的级状态时间变化图。在本例中，从时刻T1到T14，数据单元A到H以流水线的方式先后经过云系统的七个连续级，并且没有检测到数据错误。例如，在时刻T1，数据单元A缓冲于第一级(级1)，而其他级不包含数据；在时刻T2，数据单元A缓冲于级2，而数据单元B缓冲于级1，如此继续。

[0042] 需要指出，由于各个级的处理时间不同以及各个级之间的传输延迟不同，时刻T1到T14之间的间隔并非必然相等。在某些实施方式中，数据在各个级之间的传输可以由预先定义的事件触发。例如，响应于确认下一个级(级2)已经成功将前一个数据单元(数据C)传递到再下一个级(级3)而将数据单元(比如，数据D)从一个级(比如，级1)传输到下一个级(比如，级2)。这可以防止级2中的数据C在由级3成功接收之前被数据D覆盖。出于各种目的，也可以定义各种其他事件来触发在各个级之间的数据传输。在一些实施方式中，使用握手协议，使得各个级能够就其数据状态彼此进行交流。例如，一个级可以向它的最后一个级发送通知，表明它准备好接收下一个数据单元。

[0043] 如果任何一个级检测到当前存储的数据单元的数据不一致，则该级向入口节点，例如直接或者通过云系统的中央控制器，发送一个恢复请求。作为应答，从缓冲器中获取入口级所接收的数据单元并在经由级1到7的数据路径上重新发送。图4A是本申请实施方式的从入口节点的缓冲器重新发送不一致数据单元的级状态时间变化图。与图3所示的例子不同，在时刻T9，级6的数据D被判定为与最初接收的版本不一致，因此在时刻T10未传递给级7(在T10，级7的状态为:无)。而事实上，在T10，数据D将从入口级的缓冲器中获取并重新进入从级1开始的数据路径。如图所示，数据D重新进入数据路径并不影响后续 数据单元E到H在路径中的传输。在本例中，当检测到错误(不一致数据D)时，在最后一个数据单元H之后不会有新数据进入数据路径。

[0044] 在一些实施方式中，当重新发送数据时，入口级可以暂停接受新的输入数据。这将有效避免入口处的数据流量堵塞以及缓冲器的溢出，并保证数据单元按照其被接收的顺序从缓冲器中清空。

[0045] 图4B是例示了本申请实施方式的不一致数据单元从入口节点重新发送并插入到数据流中的级状态时间变化图。不同于图4A中的例子，在检测到数据D不一致的时刻(T9)，级1被数据I占据，因此无法接收已经重新发送的数据D。数据D因此在时刻T10被插入到数据I与数据J之间并进入级1，这推迟了数据J和其之后的数据单元(未图示)的传输。

[0046] 图5A是例示了本申请实施方式的一组数据单元出现错误并从入口节点重新发送的级状态时间变化图。在本例中，在时刻T9所有级都丢失了数据(数据C到H)，比如，由于系统电力故障。假设电力在T10恢复，然后数据C到H从入口级中的缓冲器重新发送，并按照云系统接收它们的顺序重新进入数据路径。在数据H于时刻T15重新进入数据路径之前，任何新的数据(未图示)都将推迟。

[0047] 图5B是例示了本申请实施方式的一组数据单元出现错误并从入口节点重新发送并插入到数据流中的级状态时间变化图。图5B示出，在T9，数据C到I丢失，并从T10开始从入口级的缓冲器中按顺序恢复。如果没有数据错误，数据J就该在时刻T10进入该路径。由于数据错误，数据C到I被插入到数据J和K之前，这延迟了J和K的传输。如图所示，如果级7为目标存储设备，则数据A到K会按照它们被云系统接收的顺序写入级7。

[0048] 本申请不受限于入口级处的缓冲器的存储类型、容量、电路设计或任何其他配置因素。在一些实施方式中，缓冲器实现为桶型移位器，以先进先出的方式(FIFO)缓冲输入数据。缓冲器的深度最好匹配数据路径上的级的数目，以防止缓冲溢出。在一些实施方式中，如果因为被缓冲的数据还未成功写入目标存储设备，缓冲器无法接受新的数据(比如，在一段延时内)，则入口级可以暂时停止接受新的数据。可以向用户设备发送一个消息以通知这种延迟。

[0049] 图6A是例示了在本申请实施方式的云系统的入口级处的示例桶型移位缓冲器中的数据状态时间变化图。参考图6A描述的数据路径与图3中的数据路径的 结构相同。在这个简化实例中，数据路径有七个级，缓冲器相应地有七个数据地址。数据单元A到G按顺序分别被推送到地址1到7。任何已经成功写入目标存储设备的数据都可以从缓冲器中清除，为新的输入数据留出空间。

[0050] 返回参考图3。在T7，数据A被成功写入目标级7(比如，块服务器的存储盘)。从这一刻起，数据A不再需要存储于入口级缓冲器中。参考图6A。在时刻T8，新的数据H覆盖了数据A并直到时刻T14一直保留在地址1中，因为在数据H之后没有其他数据。数据单元B到G分别在时刻T9到T14从缓冲器中清除。

[0051] 图6B例示了本申请实施方式的将输入数据推入云系统入口级的示例性桶型移位缓冲器的时序。输入数据单元以地址递增的方式被推入缓冲器。因为缓冲器的深度与数据路径上的级的数目相匹配，所以在最后一个地址中的数据用完之后，可以清空缓冲器。

[0052] 在一些实施方式中，入口级缓冲器可以实现为日志结构缓冲器，例如使用闪速存储器。图6C例示了本申请实施方式的将输入数据推入位于入口级的日志结构缓冲器的时序。固态硬盘(SSD)的一个平面可以被设置为循环日志缓冲器，其中数据被推入到递增的地址，每次读取的地址都对应于当前指针的位置加上一个偏移量。这样，存储在缓冲器中的数据不必在被写入目标存储设备后立即被清除，这将有助于延长SSD的寿命。在新的数据进来时，它可以被存储于缓冲器中的下一个空位，直到存满整个平面。有利的是，SSD缓冲器的大存储容量可以显著降低数据擦除的频率并避免向云系统传输的数据流的中断，同时也具有非常出色的成本效益。

[0053] 在一个实施方式中，日志结构的SSD缓冲器被设置为保持写入放大率(写入放大率＝1)并禁止垃圾收集、磨损均衡和预留空间的操作。因此，输入数据按照它们被接收的顺序写入缓冲器。结果，如果特定级经常引起数据不一致并从缓冲器进行数据恢复，则可以将对

缓冲器的读取权限限制到相对较小的地址范围。

[0054] 在一些其他实施方式中，云系统的入口级处的缓冲器可以使用混合式双直列存储模块(DIMM)来实现，其组合了动态随机存取存储器(DRAM)、闪速存储器和超级电容器。DRAM具有很大的容量，可以划分出空间作为缓冲器来使用。缓冲器的大小可以根据数据路径上包含的级的数目和各个数据单元的大 小来确定。闪速存储器一般都存在与区块擦除和存储磨损有关的问题，因而可以专用于在电路故障时存储数据。例如，断电时，超级电容器提供电力将数据从DRAM缓冲器传送到单层单元(SLC)闪速存储器。电力恢复时，从闪速存储器读取所传送的数据，进行传输(或重新传输)到数据路径上。

[0055] 混合式DIMM可以将不同种类的存储器芯片集成为多芯片包，诸如NOR闪速存储器和静态随机存取存储器(SRAM)的组合、NOR和NAND闪速存储器和SRAM的组合、NAND和DRAM的组合，或者任何适当的组合。在另一些其他实施方式中，入口级缓冲器包括非易失性DIMM(NVDIMM)，例如，使用相变存储器(PCM)作为存储介质。

[0056] 在本申请的云系统中，可以用集中方式来管理(参考图1A-7所描述的)数据传输/恢复调度和数据路径上各个级之间的通信。例如，主控制服务器收集和维护与各个级和入口级缓冲器的状态有关的信息。根据该信息，主控制服务器标识不一致的数据单元和发出报告的级，确定适当的数据传输和恢复时间表，修改数据传输顺序，并生成让各个级相应执行的指令。

[0057] 在一些其他实施方式中，可以用分布方式来控制数据传输/恢复时间表，其中各个级彼此之间相互通信并根据握手协议进行工作。例如，一个级可以将不一致的数据单元直接报告给入口级，这将提醒入口级确定一个合适的时隙并重新发送该数据单元到数据路径上。每个级都与相邻的级交流数据接收和发送事件。还可以使用任何其他适合的控制结构，都不会超出本申请的范围，例如，集中式和分布式控制的组合，或层级控制结构。不仅如此，可以在各个级或中央控制器中执行验证数据一致性的不同处理。

[0058] 图7例示了本申请实施方式的被配置为对云存储系统中的数据传输进行管理的示例性计算系统800。计算系统700可以对应于云系统的主控制服务器。计算系统700包括处理器701、系统存储器702、图形处理单元703、输入输出接口704和网络电路705，以及存储在存储器702中的操作系统706和应用软件710。软件710包括数据传输管理程序720，这个程序具有数据调度模块721、错误数据识别模块722、指令生成模块723、事件管理模块724、数据状态图模块725、一致性验证模块726、消息生成模块727等。

[0059] 在被CPU 701执行时，数据传输管理程序720控制云系统内数据路径上的各个数据单元的传输以保证数据一致性。事件管理模块724从每个级接收与数 据接收、发送、验证、查错等事件有关的信息。基于相应的事件，数据调度模块721确定适当的时间在数据路径上的不同级之间传输各个数据单元。如果需要进行数据恢复，则数据调度模块721确定用来重新发送所恢复的数据单元的时隙，并相应地推迟后续的数据单元，如参照3A到4B更详细描述的那样。

[0060] 数据状态图模块725跟踪数据单元在每一时刻在每一级中的标识。响应于特定级报告的数据错误信号，错误数据识别模块722通过查询数据状态图来标识数据单元和发出报告的级。相应地，指令生成模块723向入口级发出获取和重新发送所标识的数据单元的指令。在数据于不同级服务器中被缓存和/或处理之后，验证模块726验证数据一致性。如果在

数据路径上数据传输延迟了一段延长时间，则消息生成模块727生成一个消息，通知数据发送方停止发送新的数据。

[0061] 数据传输管理程序720被设置为执行参照图1到7更详细描述的其他功能，可以包括本领域公知的各种其他组件和功能。本领域技术人员应该理解，程序720可以使用本领域公知的任何一种或者多种适当的编程语言来编写，比如C、C++、Java、Python、Perl、TCL等。

[0062] 尽管本文公开了某些优选实施方式和方法，但本领域技术人员通过以上公开容易想到，在不脱离本申请的主旨和范围的情况下，可以对这种实施方式和方法进行变动和修改。因此本申请的保护范围应当仅被限定为所附权利要求书和适用法律的规定和原则所需要的程度。

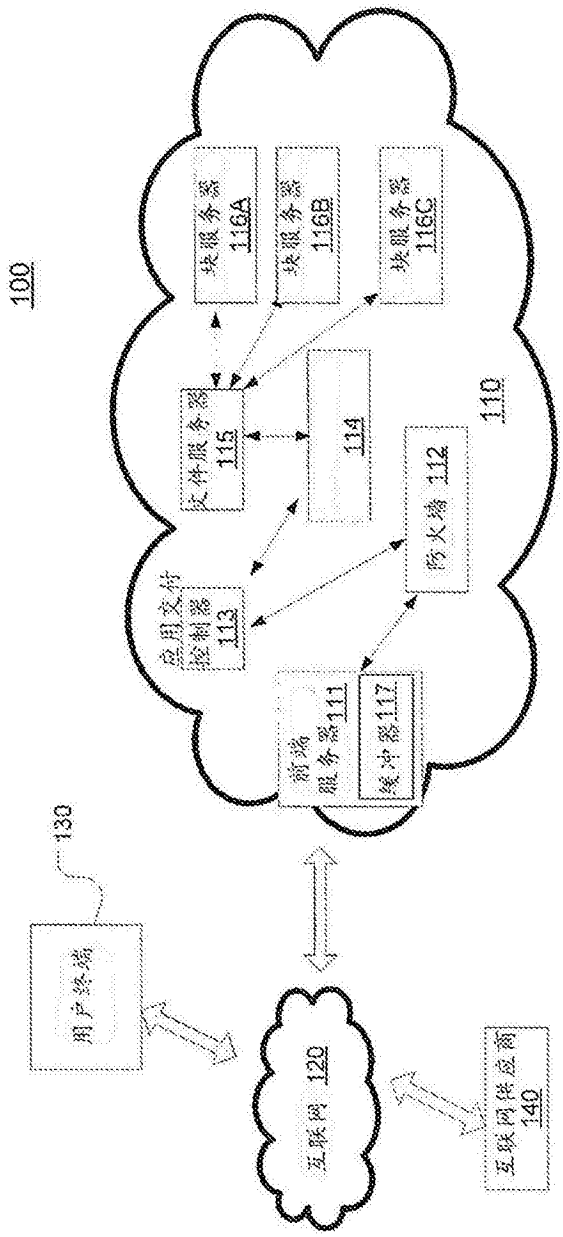

100
块服务器 116A
块服务器 116B
块服务器 116C
文件服务器 115
114
110
应用文件控制器 113
防火墙 112
前端服务器 111
缓冲器 117
130
用户终端
互联网 120
互联网供应商 140

图1A

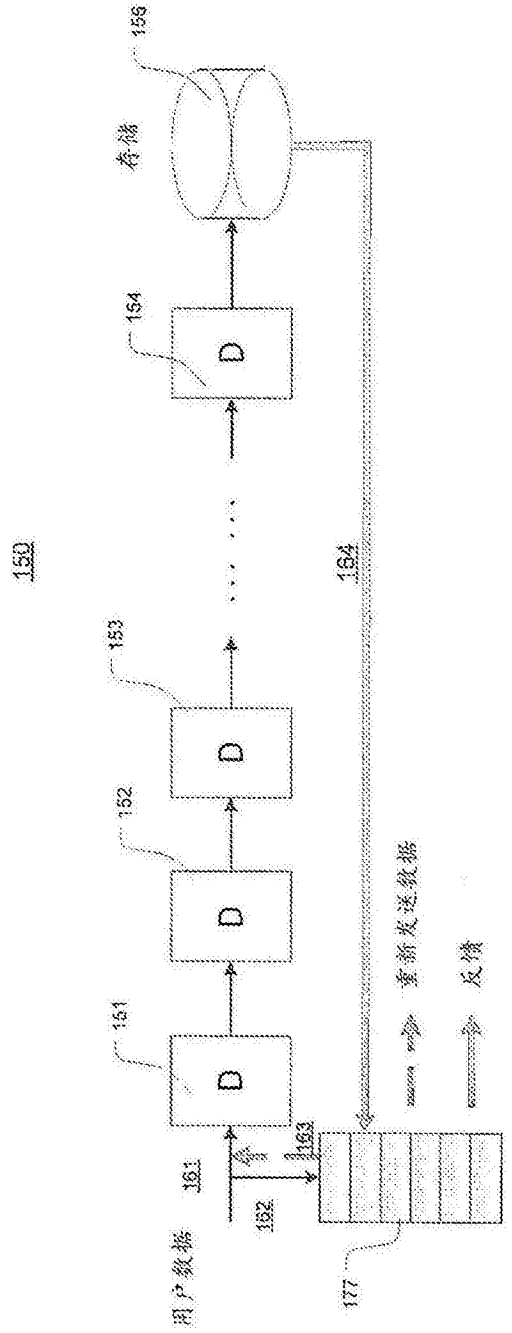

150
156
存储
164
D
153
152
151
163
161
162
用户数据
177
重新发送数据
反馈

图1B

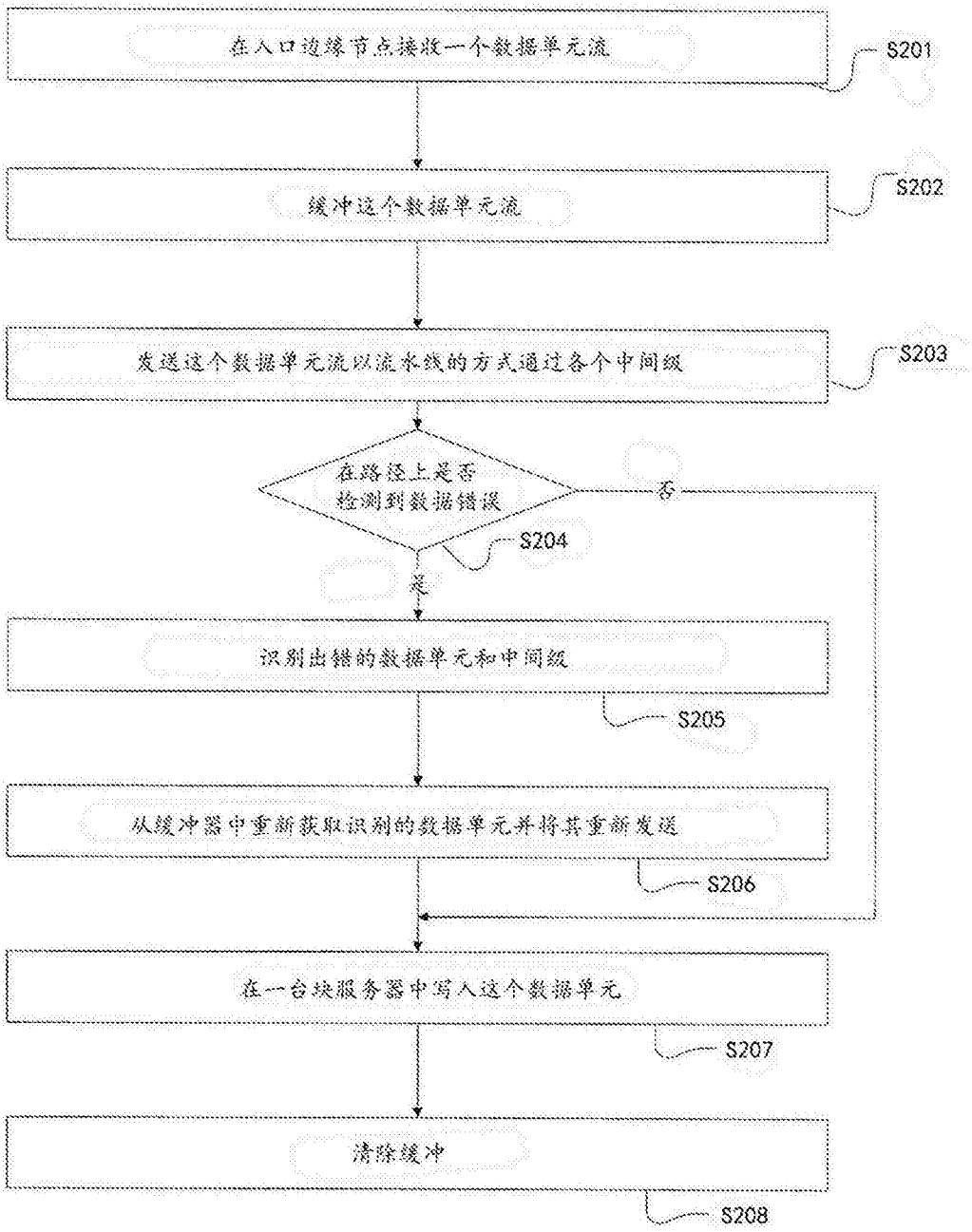

在入口边缘节点接收一个数据单元流
S201
缓冲这个数据单元流
S202
发送这个数据单元流以流水线的方式通过各个中间级
S203
在路径上是否检测到数据错误
否
S204
是
识别出错的数据单元和中间级
S205
从缓冲器中重新获取识别的数据单元并将其重新发送
S206
在一台块服务器中写入这个数据单元
S207
清除缓冲
S208

图2

| | T1 | T2 | T3 | T4 | T5 | T6 | T7 | T8 | T9 | T10 | T11 | T12 | T13 | T14 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 级1 | A | B | C | D | E | F | G | H | | | | | | |
| 级2 | | A | B | C | D | E | F | G | H | | | | | |
| 级3 | | | A | B | C | D | E | F | G | H | | | | |
| 级4 | | | | A | B | C | D | E | F | G | H | | | |
| 级5 | | | | | A | B | C | D | E | F | G | H | | |
| 级6 | | | | | | A | B | C | D | E | F | G | H | |
| 级7 | | | | | | | A | B | C | D | E | F | G | H |

图3

| | T1 | T2 | T3 | T4 | T5 | T6 | T7 | T8 | T9 | T10 | T11 | T12 | T13 | T14 | T15 | T16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 级1 | A | B | C | D | E | F | G | H | | D | | | | | | |
| 级2 | | A | B | C | D | E | F | G | H | | D | | | | | |
| 级3 | | | A | B | C | D | E | F | G | H | | D | | | | |
| 级4 | | | | A | B | C | D | E | F | G | H | | D | | | |
| 级5 | | | | | A | B | C | D | E | F | G | H | | D | | |
| 级6 | | | | | | A | B | C | D | E | F | G | H | | D | |
| 级7 | | | | | | | A | B | C | N/A | E | F | G | H | | D |

图4A

| | T1 | T2 | T3 | T4 | T5 | T6 | T7 | T8 | T9 | T10 | T11 | T12 | T13 | T14 | T15 | T16 | T17 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 级1 | A | B | C | D | E | F | G | H | I | D | J | | | | | | |
| 级2 | | A | B | C | D | E | F | G | H | | D | J | | | | | |
| 级3 | | | A | B | C | D | E | F | G | H | | D | J | | | | |
| 级4 | | | | A | B | C | D | E | F | G | H | | D | J | | | |
| 级5 | | | | | A | B | C | D | E | F | G | H | | D | J | | |
| 级6 | | | | | | A | B | C | D | E | F | G | H | | D | J | |
| 级7 | | | | | | | A | B | C | N/A | E | F | G | H | | D | J |

图4B

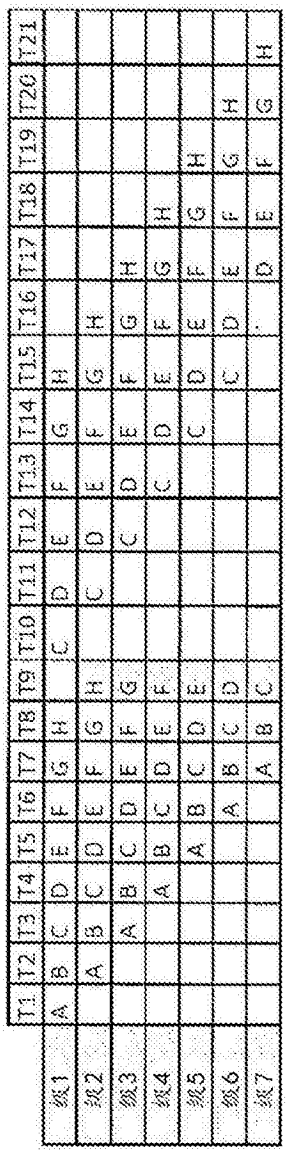


图5A

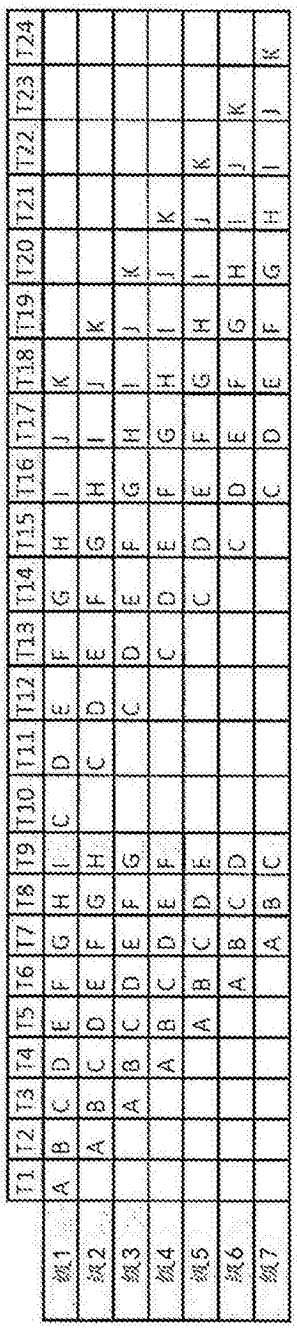


图5B

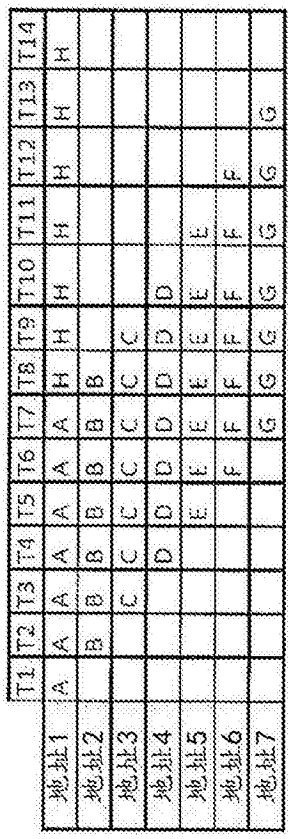


图6A

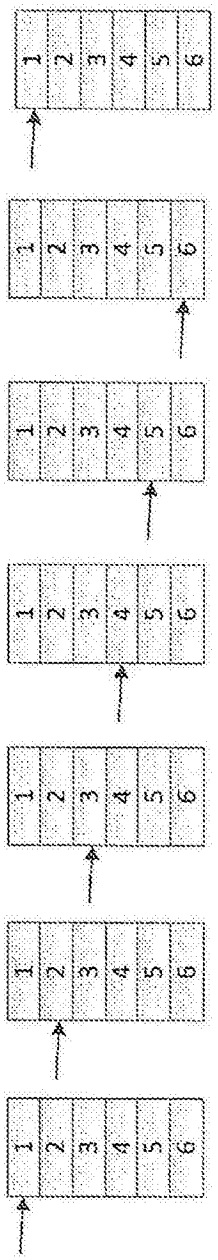


图6B

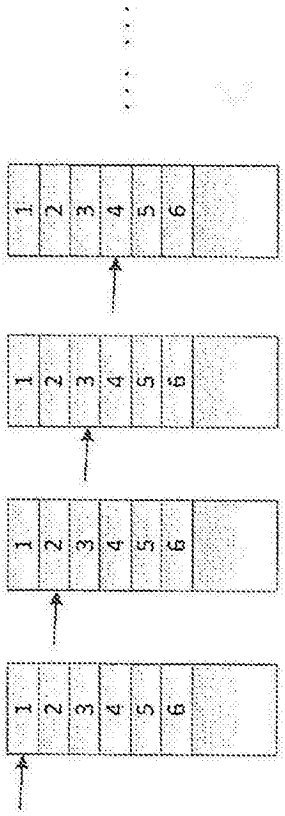

1
2
3
4
5
6

图6C

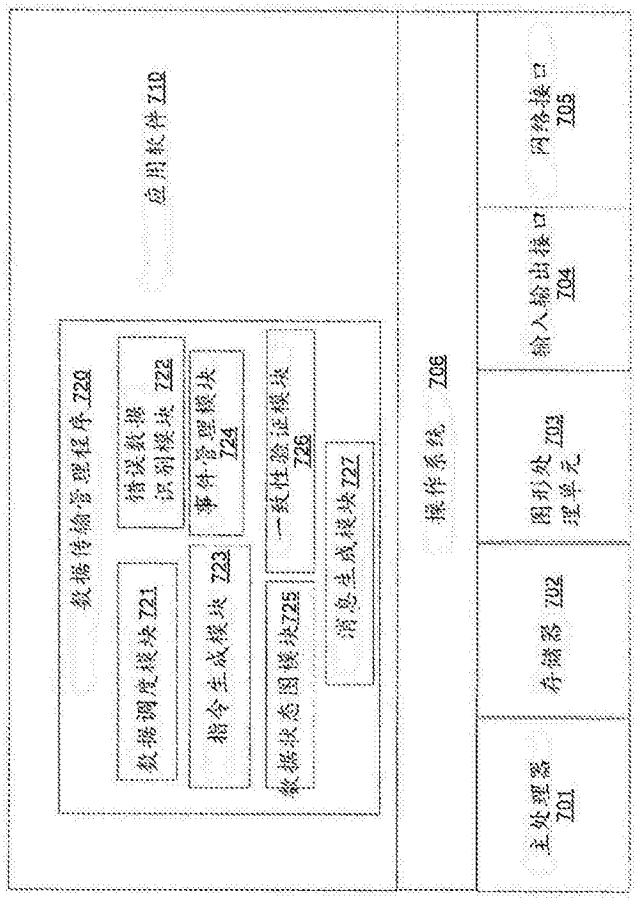

应用软件 Z10
数据传输管理程序 Z20
数据调度模块 Z21
错误数据识别模块 Z22
指令生成模块 Z23
事件管理模块 Z24
数据状态图模块Z25
一致性验证模块 Z26
消息生成模块 Z27
操作系统 Z06
主处理器 Z01
存储器 Z02
图形处理单元 Z03
输入输出接口 Z04
网络接口 Z05

图7