(19)中华人民共和国国家知识产权局

(12)发明专利

(10)授权公告号 CN 103116552 B
(45)授权公告日 2017.03.15

(21)申请号 201310085354.6

(22)申请日 2013.03.18

(65)同一申请的已公布的文献号
申请公布号 CN 103116552 A

(43)申请公布日 2013.05.22

(73)专利权人 华为技术有限公司
地址 518129 广东省深圳市龙岗区坂田华为总部办公楼

(72)发明人 王兴勇 杨军

(74)专利代理机构 北京永新同创知识产权代理有限公司 11376
代理人 钟胜光

(51)Int.Cl.
G06F 12/0871(2016.01)
G06F 3/06(2006.01)

(56)对比文件
WO 2008039527 A3,2008.07.24,
CN 1635579 A,2005.07.06,

审查员 鱼冰

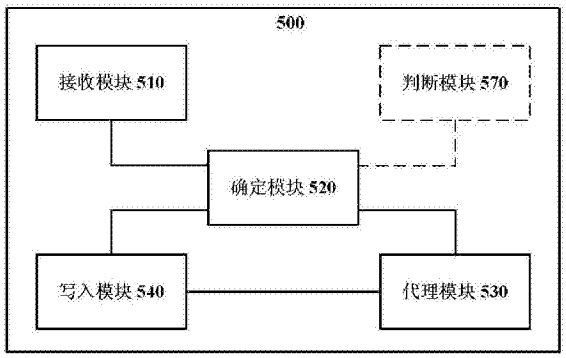
权利要求书5页 说明书10页 附图4页

(54)发明名称

用于在分布式存储系统中分配存储空间的方法和装置

(57)摘要

本发明涉及一种用于在通信网络的分布式存储系统中分配存储空间的方法和装置，其中，该装置包括：接收模块，用于接收针对文件的数据写请求，所述数据写请求包含要写入一个或多个存储设备的数据，其中，所述一个或多个存储设备是由存储设备服务器来管理的；确定模块，用于确定是否需要为所述数据的至少一部分分配空闲存储空间；代理模块，用于如果确定需要为所述数据的所述至少一部分分配空闲存储空间，则向所述存储设备服务器申请空闲逻辑管理单元chunk，申请得到的空闲chunk包含的存储空间不小于存储所述数据的所述至少一部分所需的存储空间；写入模块，用于向所述存储设备服务器发起写请求，以将所述数据的所述至少一部分写入所述一个或多个存储设备。



CN 103116552 B

1.一种用于在分布式存储系统中分配存储空间的方法，所述分布式存储系统包括存储设备服务器和Cache客户端，所述方法包括步骤：

接收来自用户的针对文件的数据写请求，所述数据写请求包含要写入一个或多个存储设备的数据，其中，所述一个或多个存储设备是由所述存储设备服务器来管理的；

确定是否需要为所述数据的至少一部分分配空闲存储空间；

如果确定需要为所述数据的所述至少一部分分配空闲存储空间，则向所述存储设备服务器申请空闲逻辑管理单元chunk，申请得到的空闲chunk包含的存储空间不小于存储所述数据的所述至少一部分所需的存储空间；其中，所述chunk是所述存储设备服务器的逻辑管理单元，包含一个或多个物理存储单元；

向所述存储设备服务器发起写请求，以将所述数据的所述至少一部分写入所述一个或多个存储设备；

其中，申请得到的空闲chunk包含的存储空间是连续存储空间；

其中，所述方法由所述Cache客户端、或系统中的其它硬件和/或软件组件、装置或实体来执行。

2.如权利要求1所述的方法，其中，所述确定是否需要为所述数据的至少一部分分配空闲存储空间的步骤包括：

确定是否已在所述一个或多个存储设备中为所述数据的所述至少一部分分配了chunk；

如果确定没有为所述数据的所述至少一部分分配chunk，则检查已为所述文件分配的chunk，以确定已为所述文件分配的chunk中是否有足够的空闲存储空间以用于存储所述数据的所述至少一部分。

3.如权利要求2所述的方法，其中，所述写入步骤包括：

如果确定已为所述数据的所述至少一部分分配了chunk，则向所述存储设备服务器发起写请求，以将所述数据的所述至少一部分写入已分配的chunk。

4.如权利要求2所述的方法，其中，还包括：

如果确定有足够的空闲存储空间以用于存储所述数据的所述至少一部分，则选择已为所述文件分配的chunk中的一个或多个chunk的空闲存储空间以用于存储所述数据的所述至少一部分；

记录所述数据的所述至少一部分与所述一个或多个chunk的对应关系；

并且其中，所述写请求用于将所述数据的所述至少一部分写入所选择的空闲存储空间。

5.如权利要求2所述的方法，其中，还包括：

如果确定没有足够的空闲存储空间以用于存储所述数据的所述至少一部分，则确定需要为所述数据的所述至少一部分分配空闲存储空间。

6.如权利要求1所述的方法，其中，还包括：

选择申请得到的空闲chunk中的一个或多个空闲chunk的空闲存储空间以用于存储所述数据的所述至少一部分；

记录所述数据的所述至少一部分与所述一个或多个空闲chunk的对应关系；

并且其中，所述写请求用于将所述数据的所述至少一部分写入所选择的空闲存储空

间。

7.如权利要求4或6所述的方法，其中，所选择的空闲存储空间是连续的空闲存储空间。

8.如权利要求3、4和6中的任一项所述的方法，其中，还包括：

确定所述数据是否全部被写入所述一个或多个存储设备；

如果确定所述数据没有被全部写入，则确定是否需要为所述数据的未写入部分的至少一部分分配空闲存储空间；

如果确定需要为所述数据的所述未写入部分的所述至少一部分分配空闲存储空间，则向所述存储设备服务器申请另外的空闲chunk，申请得到的另外的空闲chunk包含的存储空间不小于存储所述数据的所述未写入部分的所述至少一部分所需的存储空间；

向所述存储设备服务器发起另一写请求，以将所述数据的所述未写入部分的所述至少一部分写入所述一个或多个存储设备。

9.如权利要求1-6中的任一项所述的方法，其中，所述数据的所述至少一部分是与所述数据相对应的分片中的一个或多个分片。

10.如权利要求9所述的方法，其中，所述写请求是以并发方式向所述存储设备服务器发起的。

11.一种用于在分布式存储系统中分配存储空间的方法，所述分布式存储系统包括：

存储设备服务器，用于对所述分布式存储系统中的一个或多个存储设备进行管理；

Cache客户端，用于根据本地用户的数据文件访问请求对所述分布式存储系统中的存储空间的分配进行二次管理；

所述方法包括：

由所述存储设备服务器从所述Cache客户端接收对一个或多个空闲逻辑管理单元chunk的申请，其中，所述申请是响应于对第一用户数据的一部分的写请求，所申请的一个或多个空闲chunk包含的存储空间大于所述第一用户数据的大小，所述第一用户数据的大小是由所述Cache客户端响应于所述写请求根据应用层导出的；

响应于接收到所述申请，由所述存储设备服务器采用预定策略向所述Cache客户端分配所述一个或多个存储设备中的所申请的一个 或多个空闲chunk，其中，所分配的一个或多个空闲chunk能够用于对整个所述第一用户数据和至少第二用户数据的一部分进行连续存储；

其中，所述方法由所述存储设备服务器或用于管理存储设备的其它硬件和/或软件组件、装置或实体来执行。

12.如权利要求11所述的方法，其中，所述预定策略包括首次匹配策略和最佳匹配策略中的至少一个策略。

13.如权利要求11所述的方法，其中，还包括：

接收针对文件的chunk释放请求；

响应于接收到所述chunk释放请求，释放与所述文件相对应的chunk。

14.如权利要求13所述的方法，其中，还包括：

使所释放的chunk与其周围的空闲chunk合并，以组成更大的连续chunk。

15.一种用于在分布式存储系统中分配存储空间的装置，包括：

接收模块，用于接收来自用户的针对文件的数据写请求，所述数据写请求包含要写入

一个或多个存储设备的数据，其中，所述一个或多个存储设备是由所述分布式存储系统中的存储设备服务器来管理的；

确定模块，用于确定是否需要为所述数据的至少一部分分配空闲存储空间；

代理模块，用于如果确定需要为所述数据的所述至少一部分分配空闲存储空间，则向所述存储设备服务器申请空闲逻辑管理单元chunk，申请得到的空闲chunk包含的存储空间不小于存储所述数据的所述至少一部分所需的存储空间；其中，所述chunk是存储设备服务器的逻辑管理单元，包含一个或多个物理存储单元；

写入模块，用于向所述存储设备服务器发起写请求，以将所述数据的所述至少一部分写入所述一个或多个存储设备；

其中，申请得到的空闲chunk包含的存储空间是连续存储空间；

其中，所述装置是所述分布式存储系统中的Cache客户端或是被配置在通信网络的节点中的其它装置。

16.如权利要求15所述的装置，其中，所述确定模块进一步用于：

确定是否已在所述一个或多个存储设备中为所述数据的所述至少一部分分配了chunk；

如果确定没有为所述数据的所述至少一部分分配chunk，则检查已为所述文件分配的chunk，以确定已为所述文件分配的chunk中是否有足够的空闲存储空间以用于存储所述数据的所述至少一部分。

17.如权利要求16所述的装置，其中，所述写入模块进一步用于：

如果确定已为所述数据的所述至少一部分分配了chunk，则向所述存储设备服务器发起写请求，以将所述数据的所述至少一部分写入已分配的chunk。

18.如权利要求16所述的装置，其中，所述代理模块进一步用于：

如果确定有足够的空闲存储空间以用于存储所述数据的所述至少一部分，则选择已为所述文件分配的chunk中的一个或多个chunk的空闲存储空间以用于存储所述数据的所述至少一部分；

记录所述数据的所述至少一部分与所述一个或多个chunk的对应关系；

并且其中，所述写请求用于将所述数据的所述至少一部分写入所选择的空闲存储空间。

19.如权利要求16所述的装置，其中，所述确定模块进一步用于：

如果确定没有足够的空闲存储空间以用于存储所述数据的所述至少一部分，则确定需要为所述数据的所述至少一部分分配空闲存储空间。

20.如权利要求15所述的装置，其中，所述代理模块进一步用于：

选择申请得到的空闲chunk中的一个或多个空闲chunk的空闲存储空间以用于存储所述数据的所述至少一部分；

记录所述数据的所述至少一部分与所述一个或多个空闲chunk的对应关系；

并且其中，所述写请求用于将所述数据的所述至少一部分写入所选择的空闲存储空间。

21.如权利要求18或20所述的装置，其中，所选择的空闲存储空间是连续的空闲存储空间。

22.如权利要求17、18和20中的任一项所述的装置，其中，还包括：

判断模块，用于确定所述数据是否全部被写入所述一个或多个存储设备；

并且其中，所述确定模块进一步用于：如果确定所述数据没有被全部写入，则确定是否需要为所述数据的未写入部分的至少一部分分配空闲存储空间；

并且其中，所述代理模块进一步用于：如果确定需要为所述数据的所述未写入部分的所述至少一部分分配空闲存储空间，则向所述存储设备服务器申请另外的空闲chunk，申请得到的另外的空闲chunk包含的存储空间不小于存储所述数据的所述未写入部分的所述至少一部分所需的存储空间；

并且其中，所述写入模块进一步用于：向所述存储设备服务器发起另一写请求，以将所述数据的所述未写入部分的所述至少一部分写入所述一个或多个存储设备。

23.如权利要求15-20中的任一项所述的装置，其中，所述数据的所述至少一部分是与所述数据相对应的分片中的一个或多个分片。

24.如权利要求23所述的装置，其中，所述写请求是以并发方式向所述存储设备服务器发起的。

25.一种用于在分布式存储系统中分配存储空间的装置，所述分布式存储系统包括：

Cache客户端，用于根据本地用户的数据文件访问请求对所述分布式存储系统中的存储空间的分配进行二次管理；

所述装置包括：

接收模块，用于从所述Cache客户端接收对一个或多个空闲逻辑管理单元chunk的申请，其中，所述申请是响应于对第一用户数据的一部分的写请求，所申请的一个或多个空闲chunk包含的存储空间大于所述第一用户数据的大小，所述第一用户数据的大小是由所述Cache客户端响应于所述写请求根据应用层导出的；

分配模块，用于响应于接收到所述申请，采用预定策略向所述Cache客户端分配一个或多个存储设备中的所申请的一个或多个空闲chunk，其中，所分配的一个或多个空闲chunk能够用于对整个所述第一用户数据和至少第二用户数据的一部分进行连续存储；

其中，所述装置是所述分布式存储系统中的一个存储设备服务器或是被配置在通信网络的节点中的其它装置。

26.如权利要求25所述的装置，其中，所述预定策略包括首次匹配策略和最佳匹配策略中的至少一个策略。

27.如权利要求25所述的装置，其中，

所述接收模块进一步用于：接收针对文件的chunk释放请求；

并且其中，所述装置还包括：

释放模块，用于响应于接收到所述chunk释放请求，释放与所述文件相对应的chunk。

28.如权利要求27所述的装置，其中，所述释放模块进一步用于：

使所释放的chunk与其周围的空闲chunk合并，以组成更大的连续chunk。

29.一种高速缓冲存储器Cache客户端，包括：

存储器，用于存储可执行指令；以及

处理器，用于根据所述可执行指令执行权利要求1-7中的任意一项所包括的步骤。

30.如权利要求29所述的Cache客户端，其中，当所述处理器用于执行权利要求4或6中

的步骤时，所选择的空闲存储空间是连续的空闲存储空间。

31.如权利要求29所述的Cache客户端，其中，所述数据的所述至少一部分是与所述数据相对应的分片中的一个或多个分片。

32.一种用于管理一个或多个存储设备的存储设备服务器，包括：

存储器，用于存储可执行指令；以及

处理器，用于根据所述可执行指令执行权利要求11-14中的任意一项所包括的步骤。

用于在分布式存储系统中分配存储空间的方法和装置

技术领域

[0001] 本发明涉及通信网络中的存储技术，尤其涉及一种用于在分布式存储系统中分配存储空间的方法和装置。

背景技术

[0002] 随着3G网络的大规模应用、智能手机的普及、移动多媒体和移动互联网业务的兴起，移动宽带(MBB)数据业务正面临着快速增长，这对网间流量、用户体验质量(QoE)等提出了新的挑战。

[0003] 为了提高用户的QoE，降低网间流量和对服务器的冲击，在通信网络的若干节点中利用分布式存储系统是一种不错的选择。这种分布式存储系统的一个示例是分布式Cache(高速缓冲存储器)系统，该系统通过在边缘节点和/或骨干节点处部署Cache，可以将内容缓存到靠近用户的位置。

[0004] 对于分布式Cache系统而言，Cache的读写效率和访问并发性对于性能提升的影响很大。如果能够根据数据的相关性，连续地分配数据的存储空间以保证数据在磁盘上连续存储，则对Cache的读写效率有比较可观的提升。

[0005] 现有的分布式存储系统一般采用以下两种方式来分配存储空间。一种方式是按需分配，其仅根据要写入的数据的大小来分配足够的存储空间，而不保证数据在磁盘上的连续存储。另一种方式是服务器集中分配，其需要一个服务器完全负责磁盘空间的分配和管理，从而容易导致单点瓶颈问题，限制了系统的规模和吞吐量。

[0006] 图1示出了一种现有的服务器集中分配的分布式存储系统100的示意图。图1所示的分布式存储系统中包括存储服务器端101和若干客户端105。该存储服务器端101包括预分配描述符管理模块102、写请求处理模块103以及ext3本地文件系统104。

[0007] 图1所示的分布式存储系统100按照以下步骤来实现用户数据写入：客户端105向存储服务器端101发送针对某个目标文件的写请求；写请求处理模块103根据该写请求中所包含的信息，获取目标文件相关信息(例如文件名)并打开目标文件；预分配描述符管理模块102根据目标文件相关信息，为目标文件初始化一个块预分配描述符，并将该块预分配描述符缓存在存储服务器端101的内存中；存储服务器端101中的ext3本地文件系统104根据该块预分配描述符，为目标文件预留相应的数据块；存储服务器端101在完成对目标文件的写入操作之后关闭该目标文件；在存储服务器端101关闭目标文件之后，该目标文件的上述预分配描述符继续缓存在内存中。

[0008] 然而，图1所示的分布式存储系统存在以下缺点。

[0009] (1)由于存储服务器端集中地负责磁盘空间的分配和管理，并且客户端直接向该服务器发起写请求，因此当存在很多客户端时或者当客户端的读写请求频繁时，会增加存储服务器端的负担(例如单点瓶颈问题)。

[0010] (2)存储服务器端在存储客户端的写请求时，不是立即将该数据写入磁盘，而是先将要写入同一数据对象的数据预先缓存在系统内存中，直到数据累积到一定长度或用户最

后一次写请求时，才进行写入操作。由于系统内存的总量是有限的，系统的并发性不是很好，同时数据容易丢失。

[0011] (3)以文件为粒度进行存储，并且通常为需要存储的数据直接向存储服务器申请物理存储单元(block)，而未基于条带化存储做优化。

发明内容

[0012] 考虑到现有技术的上述缺点，本发明提供了用于在分布式存储系统中分配存储空间的技术方案(包括方法和装置等)。利用本发明的技术方案，可以在克服现有技术的上述缺点的基础上，保证用户访问相关性大的数据连续存储并改善分布式系统中随机存储的读写性能问题。

[0013] 在一个方面，本发明提供了一种于在分布式存储系统中分配存储空间的方法。该方法包括步骤：接收针对文件的数据写请求，所述数据写请求包含要写入一个或多个存储设备的数据，其中，所述一个或多个存储设备是由存储设备服务器来管理的；确定是否需要为所述数据的至少一部分分配空闲存储空间；如果确定需要为所述数据的所述至少一部分分配空闲存储空间，则向所述存储设备服务器申请空闲逻辑管理单元chunk，申请得到的空闲chunk包含的存储空间不小于存储所述数据的所述至少一部分所需的存储空间；向所述存储设备服务器发起写请求，以将所述数据的所述至少一部分写入所述一个或多个存储设备。

[0014] 在一个实施例中，该方法的所述确定步骤可以包括：确定是否已在所述一个或多个存储设备中为所述数据的所述至少一部分分配了chunk；如果确定没有为所述数据的所述至少一部分分配chunk，则检查已为所述文件分配的chunk，以确定已为所述文件分配的chunk中是否有足够的空闲存储空间以用于存储所述数据的所述至少一部分。优选地，该方法还可以包括：如果确定有足够的空闲存储空间以用于存储所述数据的所述至少一部分，则选择已为所述文件分配的chunk中的一个或多个chunk的空闲存储空间以用于存储所述数据的所述至少一部分；记录所述数据的所述至少一部分与所述一个或多个chunk的对应关系；并且其中，所述写请求用于将所述数据的所述至少一部分写入所选择的空闲存储空间。并且，优选地，所选择的空闲存储空间是连续的空闲存储空间。

[0015] 在另一个方面，本发明提供了一种用于在分布式存储系统中分配存储空间的方法。该方法包括步骤：接收对一个或多个空闲逻辑管理单元chunk的申请；响应于接收到所述申请，采用预定策略从一个或多个存储设备中分配一个或多个空闲chunk，使得所述一个或多个空闲chunk包含的存储空间是连续存储空间。

[0016] 在一个实施例中，该方法还可以包括：接收针对文件的chunk释放请求；响应于接收到所述chunk释放请求，释放与所述文件相对应的chunk。优选地，该方法还可以包括：使所释放的chunk与其周围的空闲chunk合并，以组成更大的连续chunk。

[0017] 在另一个方面，本发明提供了一种用于在分布式存储系统中分配存储空间的装置。该装置包括：接收模块，用于接收针对文件的数据写请求，所述数据写请求包含要写入一个或多个存储设备的数据，其中，所述一个或多个存储设备是由存储设备服务器来管理的；确定模块，用于确定是否需要为所述数据的至少一部分分配空闲存储空间；代理模块，用于如果确定需要为所述数据的所述至少一部分分配空闲存储空间，则向所述存储设备服

务器申请空闲逻辑管理单元chunk，申请得到的空闲chunk包含的存储空间不小于存储所述数据的所述至少一部分所需的存储空间；写入模块，用于向所述存储设备服务器发起写请求，以将所述数据的所述至少一部分写入所述一个或多个存储设备。

[0018] 在另一个方面，本发明提供了一种用于在分布式存储系统中分配存储空间的装置。该装置包括：接收模块，用于接收对一个或多个空闲逻辑管理单元chunk的申请；分配模块，用于响应于接收到所述申请，采用预定策略从一个或多个存储设备中分配一个或多个空闲chunk，使得所述一个或多个空闲chunk包含的存储空间是连续存储空间。

[0019] 在另一个方面，本发明提供了一种高速缓冲存储器Cache客户端。该Cache客户端包括：存储器，用于存储可执行指令；以及处理器，用于根据所述可执行指令执行根据前述第一个方面的方法所包括的步骤。

[0020] 在另一个方面，本发明提供了一种用于管理一个或多个存储设备的存储设备服务器。该存储设备服务器包括：存储器，用于存储可执行指令；以及处理器，用于根据所述可执行指令执行根据前述第二个方面的方法所包括的步骤。

[0021] 在另一个方面，本发明提供了一种机器可读介质，其上存储有可执行指令。当所述可执行指令被执行时，使得机器执行根据前述第一个方面或第二个方面的方法所包括的步骤。

[0022] 由上述内容可见，本发明的方面可以实现以下有益技术效果，并解决现有技术中存在的相应技术问题。

[0023] 在本发明的技术方案中，由于在需要为数据的至少一部分分配空闲存储空间的情况下申请一个或多个chunk（chunk是存储设备服务器的逻辑管理单元，其可以包含一个或多个物理存储单元，并且申请得到的一个或多个chunk所对应的连续存储空间可以位于磁盘阵列的不同磁盘上），因此有利于有效地实现数据的条带化存储，从而提升数据并发读取性能，增加系统的吞吐量。

[0024] 另外，在本发明的技术方案中，由于可以在为数据的至少一部分申请存储空间之前对已针对文件分配chunk中的空闲存储空间进行判断，并根据判断结果分别执行申请空闲chunk的步骤或者选择空闲存储空间的步骤（该判断步骤、申请步骤和选择步骤形成对服务器端存储的二次管理），因此有利于提高存储空间分配效率，降低突发访问请求对存储设备服务器的性能冲击，缓冲频繁读写请求对服务器端造成的负担（例如，减轻了单点瓶颈问题），并提高磁盘空间利用率。

[0025] 此外，在本发明的技术方案中，由于为数据的至少一部分申请的空闲chunk所包含的存储空间可以是连续存储空间，并且可以为数据的至少一部分选择连续的空闲存储空间，因此有助于高效地将针对同一被访问文件请求写入的数据存储到连续空间（例如，使来自不同用户的随机数据按单一用户方式存储到连续空间），减少磁头频繁移动次数，并且有利于结合文件系统的预取特性来提高读写效率。

[0026] 此外，在本发明的技术方案中，由于可以在释放chunk时，使所释放的chunk与其周围的空闲chunk合并以组成更大的连续chunk，因此有利于服务器端主动发起空闲chunk清理、碎片整理等，以满足后续磁盘空间的连续、合理且高效的分配。

附图说明

[0027] 本发明的其它特点、特征、优点和益处通过以下结合附图的详细描述将变得更加显而易见。其中：

[0028] 图1示出了一种现有的服务器集中分配的分布式存储系统的示意图；

[0029] 图2示出了根据本发明的实施例的示例性分布式存储系统的示意图；

[0030] 图3示出了按照本发明一个实施例的、用于在分布式存储系统中分配存储空间的方法的流程图；

[0031] 图4示出了按照本发明一个实施例的、用于在分布式存储系统中分配存储空间的方法的流程图；

[0032] 图5示出了按照本发明一个实施例的、用于在分布式存储系统中分配存储空间的装置的示意图；

[0033] 图6示出了按照本发明一个实施例的、用于在分布式存储系统中分配存储空间的装置的示意图；

[0034] 图7示出了按照本发明一个实施例的Cache客户端的示意图；

[0035] 图8示出了按照本发明一个实施例的、用于管理一个或多个存储设备的存储设备服务器的示意图。

具体实施方式

[0036] 在按照本发明实施例所提出的用于在分布式存储系统中分配存储空间的方案中，利用了用户数据访问的相关性，并考虑分布式系统条带化存储的需求，提出了按需要申请chunk的方式，从而保证了用户访问相关性大的数据连续存储。本发明所提出的一些方案可以通过形成对服务器端存储的二次管理，提高存储空间分配效率并缓解服务器处的单点瓶颈问题。并且，本发明所提出的一些方案可以申请包含连续存储空间的空闲chunk并且/或者选择连续空闲存储空间，从而有助于为同一文件的数据分配连续存储空间，减少磁头移动次数并提高读写效率。另外，本发明所提出的一些方案可以在释放chunk时使所释放的chunk与其周围的空闲chunk合并以组成更大的连续chunk，从而有助于满足后续磁盘空间的连续、合理且高效的分配。

[0037] 下面，将结合附图详细描述本发明的各个实施例。

[0038] 现在参见图2，其示出了根据本发明的实施例的示例性分布式存储系统200的示意图。

[0039] 如图2所示，该分布式存储系统200可以包括Cache客户端203和存储设备服务器集合210，其中，该存储设备服务器集合210可包括一个或多个存储设备服务器(例如，存储设备服务器210_1到210_n)，每个存储设备服务器对相应的存储设备集合(包含一个或多个存储设备)进行管理。例如，存储设备服务器210_1可以对存储设备集合213(包含存储设备213_1到213_m)进行管理。

[0040] Cache客户端203可以接收用户对文件的访问请求，例如数据写请求、数据删除请求或者数据读取请求等。通过Cache客户端与存储设备服务器集合之间的消息传递，可以实现用户对文件的访问请求，例如向该文件写入数据、删除该文件中的数据或者读取该文件中的数据等。

[0041] 现在参见图3，其示出了按照本发明一个实施例的、用于在分布式存储系统中分配

存储空间的方法的流程图。图3所示的方法可以由图2中的Cache客户端203执行。在该实施例中，该方法用于在例如根据应用层的决定对数据进行分片的情况下，在分布式存储系统中为请求写入的数据分配存储空间。以下结合图2中的分布式存储系统200来描述图3的方法。

[0042] 如图3所示，在步骤S301，Cache客户端203可以接收来自用户的数据写请求。在一个示例中，该数据写请求可以包含要写入存储设备集合213的数据（下文称为“用户数据”）、指向被访问文件的统一资源标识符/统一资源定位符（URI/URL）、该用户数据在被访问文件中的偏移（下文称为“用户数据偏移”）。根据例如应用层的决定，该用户数据可以被分成（或者对应于）一个或多个分片。

[0043] 在步骤S304，在接收到数据写请求之后，Cache客户端203可以根据数据写请求来计算被访问文件的标识符（即，该被访问文件的文件ID），该文件ID用于在系统中唯一地标识该被访问的文件。在一个示例中，可以利用消息摘要算法第五版（MD5）来计算出该文件ID。

[0044] 在步骤S307，在计算出文件ID之后，Cache客户端203将该文件ID映射到被访问的文件。例如，根据该文件ID，获得该文件所在的目录信息以及该文件的文件名等。

[0045] 另外，在步骤S307，Cache客户端203还可以根据用户数据偏移和用户数据大小来分析该用户数据所对应的分片。例如，可以按照（式1）来计算用户数据所对应的分片数量：

[0046] $N_F = S_D / S_F$ （式1）

[0047] 其中，N_F表示上述分片数量（其为大于或等于1的正整数），S_D表示用户数据的大小（其通常保存在元数据中），S_F表示分片大小。分片大小可以由例如应用层决定，其可以是固定长度（如4K字节、16K字节、64K字节等），也可以是可变长度（如1到2秒的视频内容）。

[0048] 另外，可以按照（式2）来计算用户数据所对应的分片中的每一个分片的标识符（即分片ID）：

[0049] $ID_i = i + (offset / S_F)$ （式2）

[0050] 其中，i表示该分片为用户数据中的第i个分片（i=1、2、…、N_F）（下文称为“分片i”），ID_i表示该用户数据中的分片i的分片ID，offset表示上述用户数据偏移。

[0051] 在分析用户数据的分片之后，Cache客户端203可以对这些分片中的每个分片执行以下步骤S310至步骤S325，直到将该用户数据的所有分片存储到对象存储设备集合213中为止。在一个示例中，可以通过设定计数器（例如，步骤S308、S309和S328）来实现对所有分片（第一个分片到第N_F个分片）的处理。下文针对分片i描述图3中的步骤S310至步骤S325。

[0052] 在步骤S310，Cache客户端203确定存储设备集合213中是否存在与分片i相对应的逻辑管理单元（下文简称为“chunk”，其是存储设备服务器的管理单元，可以包含一个或多个物理存储单元（下文简称为“block”），chunk的大小由文件系统决定）。例如，对于所有已存储在存储设备集合213中的分片（下文称为“已存储分片”），Cache客户端203可以维持分片存储资源表。该分片存储资源表可以记录，例如，被访问文件的每个已存储分片的分片ID与存储设备集合213的用于存储该已存储分片的chunk之间的对应关系。此外，该分片存储资源表还可以记录已存储分片的分片ID与用于存储该已存储分片的block的地址、数量之间的对应关系。根据该分配存储资源表，Cache客户端203可以查找存储设备集合213中是否存在与ID_i相对应的chunk；如果存在，那么还可以进一步确定该chunk中用于存储分片i的

block的地址和数量。

[0053] 如果在步骤S310确定存在与分片i相对应的chunk，则转向步骤S325；否则，前进到步骤S313。

[0054] 在步骤S313，Cache客户端203根据该分片i的大小来计算存储该分片所需的block的数量N_b。

[0055] 在步骤S316，在计算出N_b之后，Cache客户端203检查存储设备集合213中的存储情况，以确定已为被访问文件分配的chunk中是否有足够的空闲block来存储分片i。例如，可以判断空闲block的数量是否大于或等于N_b。如果有足够的空闲block，则转向步骤S322；否则，前进到步骤S319。

[0056] 在步骤S319，Cache客户端203向存储设备服务器210_1申请chunk(例如，可以一次申请一个或多个chunk)。一般而言，Cache客户端203所申请的这些空闲chunk包含数量大于N_b的空闲block。优选地，这些空闲chunk可以对应于连续的存储空间。对于磁盘阵列，这些连续存储空间可以位于磁盘阵列上的不同磁盘上(例如，适用于并发访问的情况)。优选地，这些空闲chunk还可以与先前针对该被访问文件所分配的chunk一起对应于连续的存储空间。

[0057] 在步骤S322，在申请得到空闲chunk(步骤S319)之后，或者在确定已为被访问文件分配的chunk具有足够的空闲block(步骤S316)之后，Cache客户端203选择相应chunk中的一个或多个chunk的空闲block，以用于存储该分片i。优选地，可以选择连续的空闲block。

[0058] 在步骤S323，在为分片i选择空闲block之后，Cache客户端203可以记录ID_i与前述一个或多个chunk的对应关系，例如，将该对应关系加入上述分片存储资源表。在一个示例中，可以在分片存储资源表中记录ID_i和用于存储分片i的block的地址及数量。

[0059] 在步骤S325，Cache客户端203向存储设备服务器210_1发起chunk写请求。在一个示例中，可以通过并发读写模块以并发方式发起该写请求。一般地，响应于接收到该写请求，存储设备服务器210_1可以将分片i写入相应的存储设备。应当注意的是，虽然图2中步骤S323在步骤S325之前，但是本领域技术人员应当明白，二者的执行顺序可以互换，或者可以同时执行。

[0060] 如上所述，可以通过对用户数据的所有分片(第一个分片到第N_F个分片)执行步骤S310至步骤S325，使整个用户数据被写入相应的存储设备。

[0061] 本领域技术人员应当理解，虽然根据图3的实施例，向存储设备服务器申请多个空闲chunk的条件是已为被访问文件分配的chunk中没有足够的空闲block，但是本发明不限于此。例如，该条件可以是已为被访问文件分配的chunk中没有足够的空闲存储空间(而不限于根据block来判断)，或者，该条件可以是首次请求写入当前分片，等等。

[0062] 本领域技术人员应当理解，虽然在图3的实施例中以单个分片为粒度进行存储，但是本发明不限于此。例如，可以以分片组(包含多个分片)为粒度进行存储，或者，还可以以按其它方式指定的用户数据的至少一部分为粒度进行存储，等等。

[0063] 本领域技术人员应当理解，虽然图3的实施例以一种计数器方式实现对用户数据各部分的存储，但是本发明不限于此。例如，可以以其他计数器方式、或者以其他适合的方式来实现对整个用户数据的存储。

[0064] 本领域技术人员应当理解，图3的实施例可以以逐个分片的方式发起chunk写请

求，也可以以多个分片并发的方式发起chunk写请求。例如，可以针对多个分片并发地执行图3中的步骤S310-S323，并且以并发方式为这些分片向存储设备服务器发起chunk写请求。这种并发写入的方式可以有利于提高写入效率。

[0065] 本领域技术人员应当理解，虽然图3的实施例涉及根据分片ID来查询是否存在与某个分片相对应的chunk，但是本发明不限于此。例如，可以根据分片的偏移来进行查询，或者可以根据分片的其他标识来进行查询，等等。

[0066] 本领域技术人员应当理解，虽然图3的实施例中根据URI/URL并使用文件ID来确定被访问的文件，但是本发明不限于此。例如，可以根据系统定义的其它方式来确定被访问的文件。

[0067] 本领域技术人员应当理解，虽然图3的实施例中的方法可以由Cache客户端执行，但是本发明并不限于此。该方法还可以由系统中的其他硬件和/或软件组件、装置或者实体来执行，例如，配置在网络节点中的某个装置，等等。

[0068] 现在参见图4，其示出了按照本发明一个实施例的、用于在分布式存储系统中分配存储空间的方法的流程图。图4所示的方法可以由图2中的存储设备服务器210_1执行，也可以由用于管理存储设备的其他硬件和/或软件组件、装置或者实体来执行。以下结合图2中的分布式存储系统200来描述图4的方法。

[0069] 当Cache客户端203向存储设备服务器210_1申请空闲chunk时，存储设备服务器210_1可以执行下述步骤S401-S404，以便针对该申请分配空闲chunk。

[0070] 如图4所示，在步骤S401，存储设备服务器210_1可以从Cache客户端203接收对一个或多个空闲chunk的申请。

[0071] 在步骤S404，存储设备服务器210_1可以响应于接收到所述申请，采用预定策略从其管理的存储设备集合213中分配一个或多个空闲chunk，使得所分配的空闲chunk包含的存储空间是连续存储空间。例如，预定策略可以包括首次匹配策略、最佳匹配策略、其它策略及其组合。

[0072] 在某些情况下，例如，当用户确认不再使用某个文件时，或者当系统终止对该文件的使用时，可以执行下述步骤S407-S413以释放资源。

[0073] 在步骤S407，存储设备服务器210_1从Cache客户端203接收针对文件的chunk释放请求。

[0074] 在步骤S410，存储设备服务器210_1响应于接收到该chunk释放请求，释放与该文件相对应的chunk，例如，解除对相应chunk资源的分配，或者将相应chunk资源重新定义为空闲chunk资源，等等。并且，存储设备服务器210_1还可以使被释放的chunk与其周围的空闲chunk合并，以组成更大的连续资源块(步骤S413)。

[0075] 现在参见图5，其示出了按照本发明一个实施例的用于在分布式存储系统中分配存储空间的装置的示意图。本领域技术人员应当理解，图5所示的装置500可以是图2的Cache客户端203，或者，也可以是被配置在通信网络的某个节点中的其他装置；并且，装置500可以利用软件、硬件或软硬件结合的方式来实现。以下结合图2中的分布式存储系统200来描述图5的装置500。

[0076] 如图5所示，装置500可以包括接收模块510、确定模块520、代理模块530和写入模块540。

[0077] 其中，接收模块510用于接收针对文件的数据写请求，该数据写请求包含要写入存储设备集合213的数据。确定模块520用于确定是否需要在存储设备集合213中为该数据的至少一部分（例如，与该数据相对应的分片中的一个或多个分片）分配空闲存储空间。代理模块530用于如果确定需要为该数据的该至少一部分分配空闲存储空间，则向存储设备服务器210_1申请空闲chunk。一般而言，所申请的空闲chunk包含的存储空间不小于存储该数据的该至少一部分所需的存储空间。优选地，所申请的空闲chunk包含的存储空间可以是连续存储空间。写入模块540用于向存储设备服务器210_1发起写请求，以将该数据的该至少一部分写入存储设备集合230。

[0078] 在一个示例中，确定模块520可以进一步用于：确定是否已在存储设备集合213中为该数据的该至少一部分分配了chunk；如果确定没有为该数据的该至少一部分分配chunk，则检查已为该文件分配的chunk，以确定已为该文件分配的chunk中是否有足够的空闲存储空间以用于存储该数据的该至少一部分。

[0079] 在该示例中，写入模块540可以进一步用于：如果确定已为该数据的该至少一部分分配了chunk，则向存储设备服务器210_1发起写请求，以将该数据的该至少一部分写入已分配的chunk。

[0080] 在该示例中，代理模块530可以进一步用于：如果确定有足够的空闲存储空间以用于存储该数据的该至少一部分，则选择已为该文件分配的chunk中的一个或多个chunk的空闲存储空间以用于存储该数据的该至少一部分。并且，代理模块530可以进一步用于记录该数据的该至少一部分与这一个或多个chunk的对应关系。并且，上述写请求可以用于将该数据的该至少一部分写入所选择的空闲存储空间。优选地，所选择的空闲存储空间可以是连续的空闲存储空间。

[0081] 在该示例中，确定模块520可以进一步用于：如果确定没有足够的空闲存储空间以用于存储该数据的该至少一部分，则确定需要为该数据的该至少一部分分配空闲存储空间。

[0082] 在另一个示例中，代理模块530可以进一步用于：选择申请得到的空闲chunk中的一个或多个空闲chunk的空闲存储空间以用于存储该数据的该至少一部分。并且，代理模块530可以进一步用于：记录该数据的该至少一部分与这一个或多个空闲chunk的对应关系。并且，上述写请求可以用于将该数据的该至少一部分写入所选择的空闲存储空间。优选地，所选择的空闲存储空间可以是连续的空闲存储空间。

[0083] 在又一个示例中，装置500还可以包括判断模块570，该判断模块570用于确定所述数据是否全部被写入存储设备集合213。并且，确定模块520可以进一步用于：如果确定该数据没有被全部写入，则确定是否需要为该数据的未写入部分的至少一部分分配空闲存储空间。并且，代理模块530可以进一步用于：如果确定需要为该数据的该未写入部分的该至少一部分分配空闲存储空间，则向存储设备服务器申请另外的空闲逻辑管理单元chunk，所申请到的这些另外的空闲chunk包含的存储空间不小于存储该数据的该未写入部分的该至少一部分所需的存储空间。

[0084] 本领域技术人员应当理解，虽然图5中的各个模块是分立的，但是本发明不限于此。例如，这些模块中的多个模块（例如，确定模块520、代理模块530）可以结合在一个模块中。

[0085] 本领域技术人员应当理解，虽然图5仅示出了七个模块，但是本发明不限于此。例如，装置500还可以包括有助于实现分布式存储的其他模块（例如，分片管理模块）。该分片管理模块可以维护分片的状态，并且执行分片查找等功能。

[0086] 现在参见图6，其示出了按照本发明一个实施例的用于在分布式存储系统中分配存储空间的装置的示意图。本领域技术人员应当理解，图6所示的装置600可以是图2的存储设备服务器集合210中的一个存储设备服务器（例如存储设备服务器210_1），或者，也可以是被配置在通信网络的某个节点中的其他装置；并且，装置600可以利用软件、硬件或软硬件结合的方式来实现。以下结合图2中的分布式存储系统200来描述图6的装置600。

[0087] 如图6所示，装置600可以包括接收模块610、分配模块620、释放模块630。

[0088] 其中，接收模块610用于从Cache客户端203接收对一个或多个空闲chunk的申请。分配模块620用于响应于接收到该申请，采用预定策略从存储设备集合213中分配一个或多个空闲chunk，使得所分配的空闲chunk包含的存储空间是连续存储空间。优选地，预定策略可以包括首次匹配策略和最佳匹配策略中的至少一个策略。

[0089] 此外，接收模块610可以进一步用于接收针对文件的chunk释放请求。释放模块630用于响应于接收到该chunk释放请求，释放与该文件相对应的chunk。优选地，释放模块可以进一步用于使所释放的chunk与其周围的空闲chunk合并，以组成更大的连续chunk。

[0090] 现在参见图7，其示出了按照本发明一个实施例的Cache客户端700的示意图。该Cache客户端700可以是图2中的Cache客户端203。以下结合图2中的分布式存储系统200来描述Cache客户端700。

[0091] 如图7所示，Cache客户端700可以包括用于存储可执行指令的存储器710和处理器720。

[0092] 其中，处理器720可以根据存储器710所存储的可执行指令执行以下步骤：接收针对文件的数据写请求，该数据写请求包含要写入存储设备集合213的数据；确定是否需要在存储设备集合213中为该数据的至少一部分（例如，与该数据相对应的分片中的一个或多个分片）分配空闲存储空间；如果确定需要为该数据的该至少一部分分配空闲存储空间，则向存储设备服务器210_1申请空闲逻辑管理单元chunk；向存储设备服务器210_1发起写请求，以将该数据的该至少一部分写入存储设备230。一般而言，所申请的空闲chunk包含的存储空间不小于存储该数据的该至少一部分所需的存储空间。优选地，这多个空闲chunk包含的存储空间可以是连续存储空间。

[0093] 此外，前述确定步骤可以包括：确定是否已在存储设备集合213中为该数据的该至少一部分分配了chunk；如果确定没有为该数据的该至少一部分分配chunk，则检查已为该文件分配的chunk，以确定已为该文件分配的chunk中是否有足够的空闲存储空间以用于存储该数据的该至少一部分。

[0094] 此外，前述写入步骤可以包括：如果确定已为该数据的该至少一部分分配了chunk，则向该存储设备服务器发起写请求，以将该数据的该至少一部分写入已分配的chunk。

[0095] 此外，处理器720还可以根据存储器710所存储的可执行指令执行以下步骤：如果确定有足够的空闲存储空间以用于存储该数据的该至少一部分，则选择已为该文件分配的chunk中的一个或多个chunk的空闲存储空间以用于存储该数据的该至少一部分；记录该数

据的该至少一部分与这一个或多个chunk的对应关系；其中，上述写请求用于将该数据的该至少一部分写入所选择的空闲存储空间。

[0096] 或者，处理器720还可以根据存储器710所存储的可执行指令执行以下步骤：如果确定没有足够的空闲存储空间以用于存储该数据的该至少一部分，则确定需要为该数据的该至少一部分分配空闲存储空间。在该情况下，处理器720还可以根据存储器710所存储的可执行指令执行以下步骤：选择申请得到的空闲chunk中的一个或多个空闲chunk的空闲存储空间以用于存储该数据的该至少一部分；记录该数据的该至少一部分与这一个或多个空闲chunk的对应关系；其中，上述写请求可以用于将该数据的该至少一部分写入所选择的空闲存储空间。

[0097] 现在参见图8，其示出了按照本发明一个实施例的、用于管理一个或多个存储设备的存储设备服务器800的示意图。该存储设备服务器800可以是图2中的存储设备服务器210_1，并且，其所管理的一个或多个存储设备可以是图2中的存储设备集合213。以下结合图2中的分布式存储系统200来描述存储设备服务器800。

[0098] 如图8所示，存储设备服务器800可以包括用于存储可执行指令的存储器810和处理器820。

[0099] 其中，处理器820可以根据存储器810所存储的可执行指令执行以下步骤：从Cache客户端203接收对一个或多个空闲chunk的申请；响应于接收到该申请，采用诸如首次匹配策略、最佳匹配策略、其它策略及其组合之类的预定策略从其管理的存储设备集合213中分配一个或多个空闲chunk，使得所分配的空闲chunk包含的存储空间是连续存储空间。

[0100] 此外，处理器820还可以根据存储器810所存储的可执行指令执行以下步骤：从Cache客户端203接收针对文件的chunk释放请求；响应于接收到该chunk释放请求，释放与该文件相对应的chunk。

[0101] 此外，处理器820还可以根据存储器810所存储的可执行指令执行以下步骤：使所释放的chunk与其周围的空闲chunk合并，以组成更大的连续chunk。

[0102] 本发明的一个实施例提供一种机器可读介质，其上存储有可执行指令，当该可执行指令被执行时，使得机器执行前述处理器720或处理器820所执行的步骤。

[0103] 本领域技术人员应当理解，本发明的各个实施例可以在不偏离发明实质的情况下做出各种变形和改变，因此，本发明的保护范围应当由所附的权利要求书来限定。

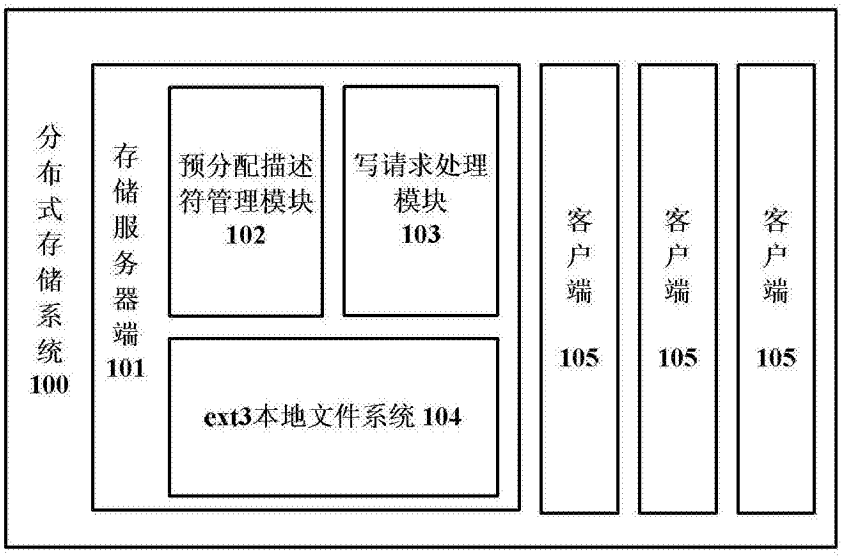

分布式存储系统 100
存储服务器端 101
预分配描述符管理模块 102
写请求处理模块 103
ext3本地文件系统 104
客户端 105
客户端 105
客户端 105

图1

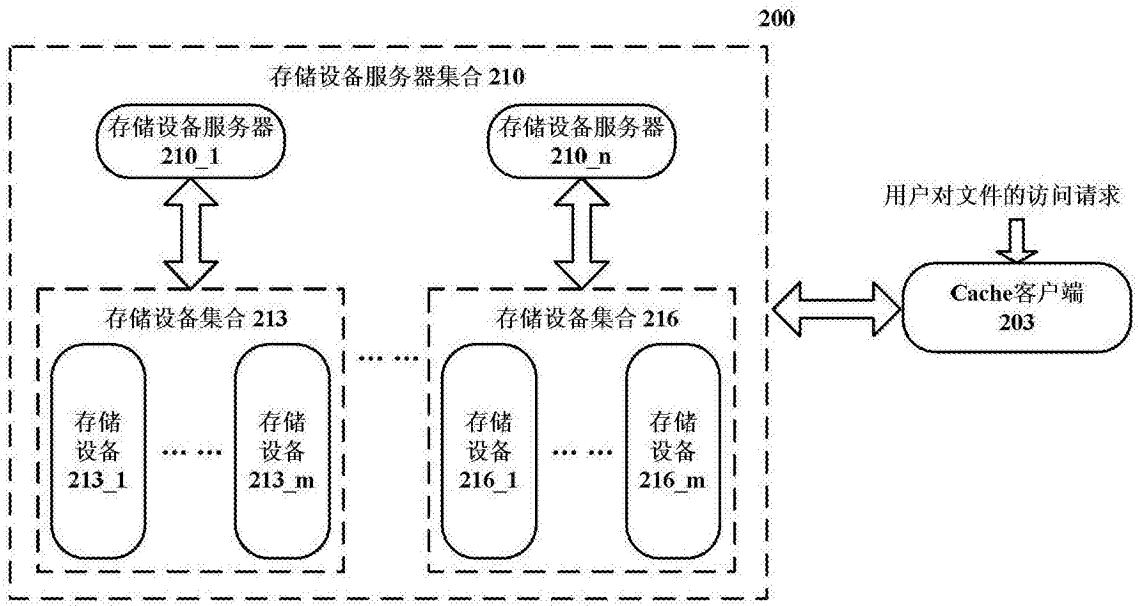

200
存储设备服务器集合 210
存储设备服务器 210_1
存储设备服务器 210_n
用户对文件的访问请求
Cache客户端 203
存储设备集合 213
存储设备集合 216
存储设备 213_1
存储设备 213_m
存储设备 216_1
存储设备 216_m

图2

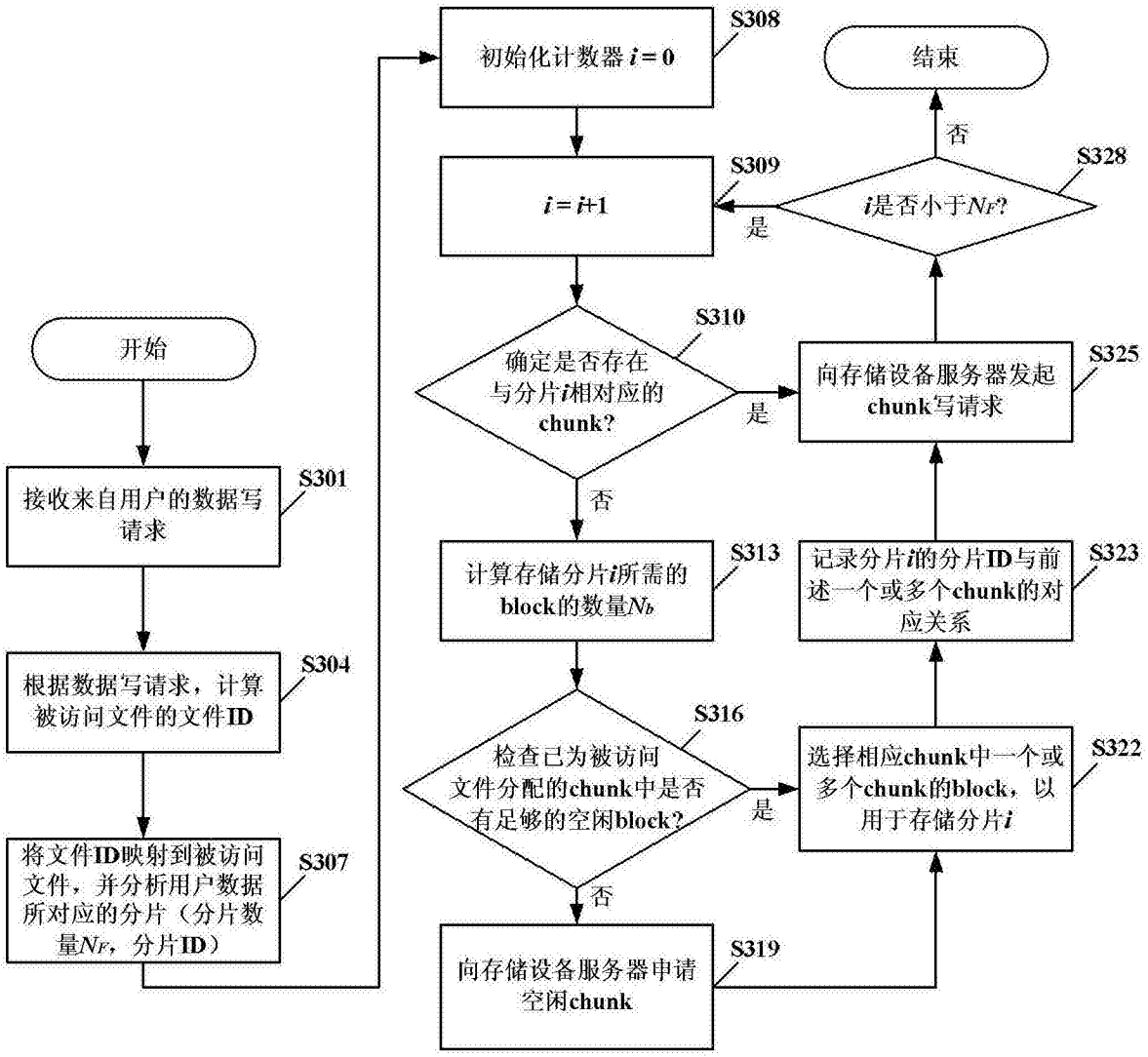

开始
接收来自用户的数据写请求
S301
根据数据写请求，计算被访问文件的文件ID
S304
将文件ID映射到被访问文件，并分析用户数据所对应的分片（分片数量N_F，分片ID）
S307
初始化计数器 $i=0$
S308
$i=i+1$
S309
确定是否存在与分片i相对应的chunk?
S310
是
否
计算存储分片i所需的block的数量N_b
S313
检查已为被访问文件分配的chunk中是否有足够的空闲block?
S316
是
否
向存储设备服务器申请空闲chunk
S319
选择相应chunk中一个或多个chunk的block，以用于存储分片i
S322
记录分片i的分片ID与前述一个或多个chunk的对应关系
S323
向存储设备服务器发起chunk写请求
S325
i是否小于N_F?
S328
是
否
结束

图3

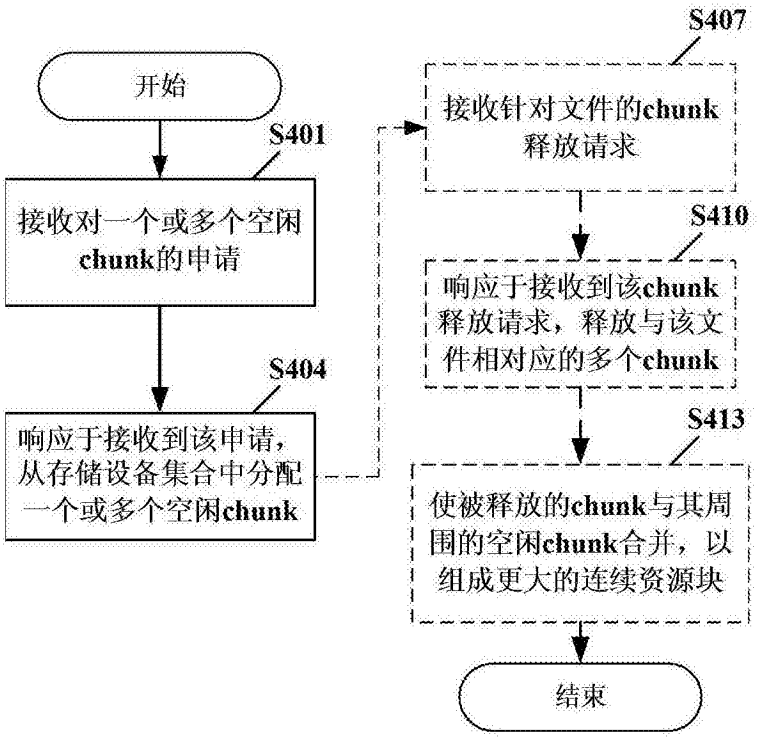

开始
S401
接收对一个或多个空闲chunk的申请
S404
响应于接收到该申请，从存储设备集合中分配一个或多个空闲chunk
S407
接收针对文件的chunk释放请求
S410
响应于接收到该chunk释放请求，释放与该文件相对应的多个chunk
S413
使被释放的chunk与其周围的空闲chunk合并，以组成更大的连续资源块
结束

图4

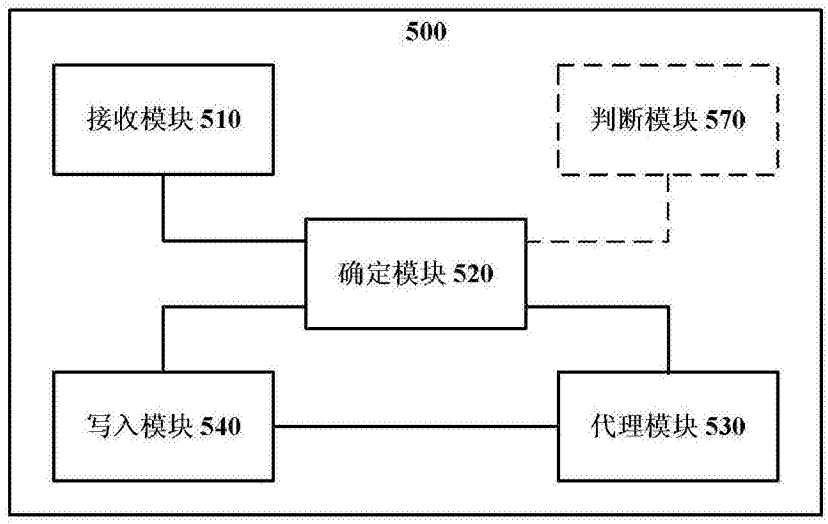

500
接收模块 510
判断模块 570
确定模块 520
写入模块 540
代理模块 530

图5

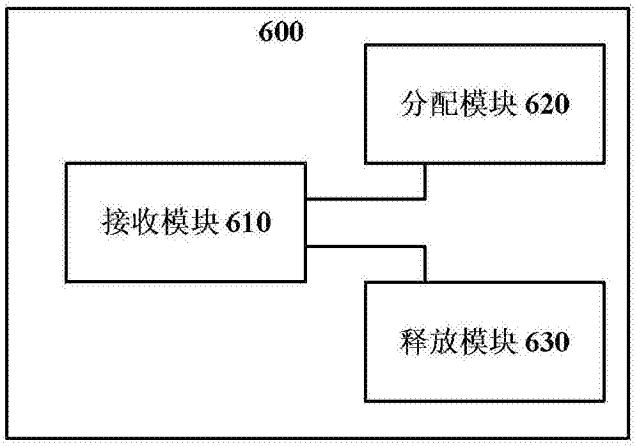

600
分配模块 620
接收模块 610
释放模块 630

图6

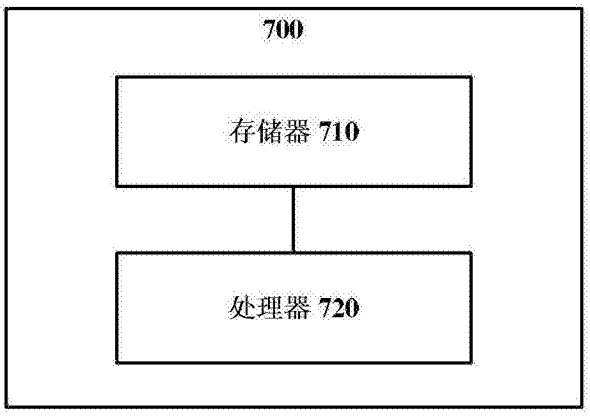

700
存储器 710
处理器 720

图7

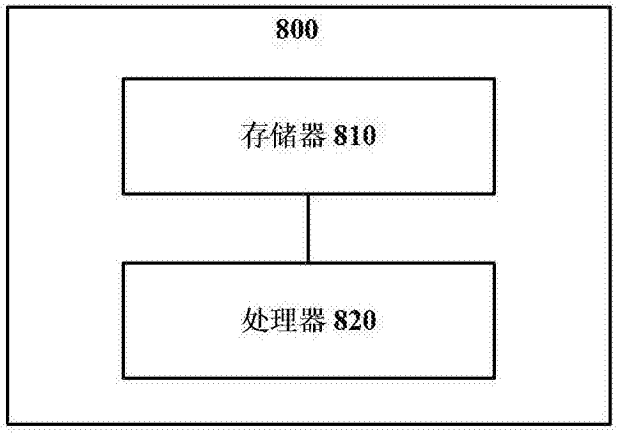

800
存储器 810
处理器 820

图8