

一种数据处理方法装置及系统

技术领域

本发明涉及存储领域，特别涉及数据处理方法、装置及系统。

背景技术

基于对象的存储系统（Object-based Storage System）是一种分布式存储系统，由存储服务器和基于对象的存储设备（Object-based Storage Device, OSD）组成。基于对象的存储系统也可以称为对象存储系统，基于对象的存储设备也可以称为对象存储设备。在对象存储系统中，以对象作为最基本的存储内容单元。数据可以是文件或者卷。以文件为例，文件被拆分成分片，文件分片有属性信息，文件分片、文件分片的元数据、文件分片的属性可以共同组成了一个对象，对象存储在多个 OSD 中。

对象存储系统提供快照（Snapshot）功能。快照是关于指定数据集的拷贝，该拷贝标记了相应数据在某个时间点（拷贝开始的时间点）的映像。

以文件为例，在快照后，如果对整个文件或者文件的部分数据进行修改，需要把修改数据存入存储系统中。现有技术使用对象 ID 作为对象的唯一标识，如果同一个文件更新，被更新的数据需要以新的对象 ID 存储到存储设备中。如果文件频繁更新，对象 ID 的总数变得非常庞大，占用了较多的存储空间从而增加了系统资源的损耗。

发明内容

本发明提供一种数据管理技术，可以减少对象 ID 的总数，节约对象 ID 占用存储空间。

第一方面，本发明实施例提供一种数据存储方法，包括：对象存储设备 OSD 接收客户服务器发送的条带写请求，所述条带写请求中携带待写条带、待写条带版本号、待写条带偏移量、以及所述待写条带的对象 ID，其中，所述待写条带版本号与所述待写条带所属文件或者卷的最近一次快照

的快照 ID 对应, 所述待写条带偏移量描述所述待写条带在所属对象中的位置, 所述待写条带的对象 ID 是所述待写条带所属对象的 ID; 所述 OSD 将所述待写条带写入由所述对象 ID、所述待写条带版本号以及所述待写条带偏移量确定的存储位置。

第二方面, 本发明实施例提供一种数据存储方法, 包括: 对象存储设备 OSD 接收客户服务器发送的条带写请求, 所述条带写请求中携带待写条带、待写条带版本号、待写条带偏移量、以及所述待写条带的对象 ID, 其中, 所述待写条带版本号与所述待写条带所属文件或者卷的最近一次快照的快照 ID 对应, 所述待写条带偏移量描述所述待写条带在所属对象中的位置, 所述待写条带的对象 ID 是所述待写条带所属对象的 ID; 所述 OSD 判断由所述待写条带版本号和所述对象 ID 确定的对象是否已备份: 如果已备份, 则所述 OSD 将所述待写条带写入由所述对象 ID、所述待写条带版本号以及所述待写条带偏移量确定的存储位置; 如果未备份, 则所述 OSD 使用所述待写条带建立一个拼接对象, 然后把所述拼接对象写入由所述待写条带版本号和所述对象 ID 确定的存储位置。

第三方面, 本发明实施例提供一种数据存储方法, 包括: 对象存储设备 OSD 接收客户服务器发送的条带写请求, 所述条带写请求中携带待写条带、待写条带版本号、待写条带偏移量、以及所述待写条带的对象 ID, 其中, 所述待写条带版本号与所述待写条带所属文件或者卷的最近一次快照的快照 ID 对应, 所述待写条带偏移量描述所述待写条带在所属对象中的位置, 所述待写条带的对象 ID 是待写条带所属对象的 ID; 所述 OSD 判断由所述待写条带版本号、所述待写条带的对象 ID 以及所述待写条带偏移量确定的条带是否已备份; 如果已备份, 则将所述待写条带写入由所述待写条带版本号、所述待写条带的对象 ID 以及所述待写条带偏移量确定的存储位置; 如果未备份, 则将所述 OSD 中初始版本对象中位于所述待写条带偏移量、大小是所述待写条带大小的数据备份到由所述待写条带版本号、所述待写条带偏移量以及所述待写条带的对象 ID 确定的存储位置, 其中, 所述

初始版本对象的对象 ID 和所述待写条带的对象 ID 相同, 所述初始版本对象的版本号是初始版本号; 把所述待写条带写入由所述待写条带的对象 ID、所述初始版本号以及所述待写条带偏移量确定的存储位置。

第四方面, 本发明实施例提供一种数据存储方法, 包括: 对象存储设备 OSD 接收客户服务器发送的条带写请求, 所述条带写请求中携带待写条带、待写条带版本号、待写条带偏移量、以及所述待写条带的对象 ID, 其中, 所述待写条带版本号与所述待写条带所属文件或者卷的最近一次快照的快照 ID 对应, 所述待写条带偏移量描述所述待写条带在所属对象中的位置, 所述待写条带的对象 ID 是所述待写条带所属对象的 ID; 所述 OSD 判断由所述待写条带版本号和所述对象 ID 确定的对象是否已备份: 如果已备份, 则所述 OSD 将所述待写条带写入由所述对象 ID、所述对象的版本号以及所述待写条带偏移量确定的存储位置; 如果未备份, 则将所述 OSD 中初始版本对象中的数据备份到由所述待写条带版本号以及所述对象 ID 确定的存储位置, 其中, 所述初始版本对象的对象 ID 和所述待写条带的对象 ID 相同, 所述初始版本对象的版本号是初始版本号; 所述 OSD 将所述待写条带写入由所述对象 ID、所述初始版本号以及所述待写条带偏移量确定的存储位置。

第五方面, 本发明实施例提供一种读数据方法, 包括: 对象存储设备 OSD 接收客户服务器发送的读条带请求, 所述读条带请求中携带待读条带大小、待读条带偏移量、待读条带版本号以及待读条带的对象 ID, 其中, 所述待读条带版本号与所述待读条带所属文件或者卷的最近一次快照的快照 ID 对应, 所述待读条带的对象 ID 是所述待读条带所属对象的 ID; 所述 OSD 判断由所述对象 ID、所述待读条带版本号以及所述待读条带偏移量所确定的条带是否已备份: 如果已备份, 则读取由所述对象 ID、所述待读条带版本号、所述待读条带偏移量以及所述待读条带大小所确定的数据, 把读取的数据作为待读条带发送给所述客户服务器; 如果未备份, 则从对象 ID 和所述待读条带的对象 ID 相同、版本号和所述待读条带的版本号不同的

对象中，按照对象的快照时间由晚到早的顺序，逐个对象进行查找，直至找到在所述待读条带偏移量的存储位置存储有有效数据的对象，把找到的有效数据作为待读条带发送给所述客户服务器，其中，所述对象的版本号和所述对象生成之前，所属文件或者卷的最近一次快照的快照 ID 对应。

第六方面，本发明实施例提供一种读数据方法，包括：所述 OSD 接收所述客户服务器发送的读条带请求，所述读条带请求中携带待读条带大小、待读条带偏移量、待读条带版本号以及待读条带的对象 ID，其中，所述待读条带版本号与所述待读条带所属文件或者卷的最近一次快照的快照 ID 对应，所述待读条带的对象 ID 是所述待读条带所属对象的 ID；所述 OSD 判断由所述对象 ID、所述待读条带版本号确定的对象是否已备份：如果已备份，则读取由所述对象 ID、所述待读条带版本号、所述待读条带偏移量以及所述待读条带大小所确定的数据，把读取的数据作为待读条带发送给所述客户服务器；如果未备份，则从对象 ID 和所述待读条带的对象 ID 相同、版本号和所述待读条带的版本号不同的对象中，按照对象的快照时间由晚到早的顺序，逐个对象进行查找，直至找到在所述待读条带偏移量的存储位置存储有有效数据的对象，把找到的有效数据作为待读条带发送给所述客户服务器，其中，所述对象的版本号和所述对象生成之前，所属文件或者卷的最近一次快照的快照 ID 对应。

第七方面，本发明实施例提供一种数据处理装置，包括：条带请求接收模块，用于接收客户服务器发送的条带写请求，所述条带写请求中携带待写条带、待写条带版本号、待写条带偏移量、以及所述待写条带的对象 ID，其中，所述待写条带版本号与所述待写条带所属文件或者卷的最近一次快照的快照 ID 对应，所述待写条带偏移量描述所述待写条带在所属对象中的位置，所述待写条带的对象 ID 是所述待写条带所属对象的 ID；条带存储模块，用于将所述待写条带写入由所述对象 ID、所述待写条带版本号以及所述待写条带偏移量确定的存储位置。

第八方面，本发明实施例提供一种数据处理装置，包括：条带请求接

收模块，用于接收客户服务器发送的条带写请求，所述条带写请求中携带待写条带、待写条带版本号、待写条带偏移量、以及所述待写条带的对象 ID，其中，所述待写条带版本号与所述待写条带所属文件或者卷的最近一次快照的快照 ID 对应，所述待写条带偏移量描述所述待写条带在所属对象中的位置，所述待写条带的对象 ID 是所述待写条带所属对象的 ID；条带存储模块，用于判断由所述待写条带版本号和所述对象 ID 确定的对象是否已备份：

如果已备份，则所述条带存储模块还用于将所述待写条带写入由所述对象 ID、所述待写条带版本号以及所述待写条带偏移量确定的存储位置；如果未备份，则所述条带存储模块还用于使用所述待写条带建立一个拼接对象，然后把所述拼接对象写入由所述待写条带版本号和所述对象 ID 确定的存储位置。

第九方面，本发明实施例提供一种数据处理装置，包括：条带请求接收模块，用于接收客户服务器发送的条带写请求，所述条带写请求中携带待写条带、待写条带版本号、待写条带偏移量、以及所述待写条带的对象 ID，其中，所述待写条带版本号与所述待写条带所属文件或者卷的最近一次快照的快照 ID 对应，所述待写条带偏移量描述所述待写条带在所属对象中的位置，所述待写条带的对象 ID 是待写条带所属对象的 ID；条带存储模块，用于判断由所述待写条带版本号、所述待写条带的对象 ID 以及所述待写条带偏移量确定的条带是否已备份；如果已备份，则将所述待写条带写入由所述待写条带版本号、所述待写条带的对象 ID 以及所述待写条带偏移量确定的存储位置；如果未备份，则将所述数据存储装置中初始版本对象中位于所述待写条带偏移量、大小是所述待写条带大小的数据备份到由所述待写条带版本号、所述待写条带偏移量以及所述待写条带的对象 ID 确定的存储位置，其中，所述初始版本对象的对象 ID 和所述待写条带的对象 ID 相同，所述初始版本对象的版本号是初始版本号；把所述待写条带写入由所述待写条带的对象 ID、所述初始版本号以及所述待写条带偏移量确定的

存储位置。

第十方面，本发明实施例提供一种数据处理装置，包括：条带请求接收模块，用于接收客户服务器发送的条带写请求，所述条带写请求中携带待写条带、待写条带版本号、待写条带偏移量、以及所述待写条带的对象 ID，其中，所述待写条带版本号与所述待写条带所属文件或者卷的最近一次快照的快照 ID 对应，所述待写条带偏移量描述所述待写条带在所属对象中的位置，所述待写条带的对象 ID 是所述待写条带所属对象的 ID；条带存储模块，用于判断由所述待写条带版本号和所述对象 ID 确定的对象是否已备份：如果已备份，则将所述待写条带写入由所述对象 ID、所述对象的版本号以及所述待写条带偏移量确定的存储位置；如果未备份，则将初始版本对象中的数据备份到由所述待写条带版本号以及所述对象 ID 确定的存储位置，其中，所述初始版本对象的对象 ID 和所述待写条带的对象 ID 相同，所述初始版本对象的版本号是初始版本号；条带存储模块，还用于将所述待写条带写入由所述对象 ID、所述初始版本号以及所述待写条带偏移量确定的存储位置。

第十一方面，本发明实施例提供一种数据处理装置，包括：条带请求接收模块，用于接收客户服务器发送的读条带请求，所述读条带请求中携带待读条带大小、待读条带偏移量、待读条带版本号以及待读条带的对象 ID，其中，所述待读条带版本号与所述待读条带所属文件或者卷的最近一次快照的快照 ID 对应，所述待读条带的对象 ID 是所述待读条带所属对象的 ID；条带读取模块，用于判断由所述对象 ID、所述待读条带版本号以及所述待读条带偏移量所确定的条带是否已备份：如果已备份，则读取由所述对象 ID、所述待读条带版本号、所述待读条带偏移量以及所述待读条带大小所确定的数据，把读取的数据作为待读条带发送给所述客户服务器；如果未备份，则从对象 ID 和所述待读条带的对象 ID 相同、版本号和所述待读条带的版本号不同的对象中，按照对象的快照时间由晚到早的顺序，逐个对象进行查找，直至找到在所述待读条带偏移量的存储位置存储有有

效数据的对象，把找到的有效数据作为待读条带发送给所述客户服务器，其中，所述对象的版本号和所述对象生成之前，所属文件或者卷的最近一次快照的快照 ID 对应。

第十二方面，本发明实施例提供一种数据处理装置，包括：条带请求接收模块，用于接收客户服务器发送的读条带请求，所述读条带请求中携带待读条带大小、待读条带偏移量、待读条带版本号以及待读条带的对象 ID，其中，所述待读条带版本号与所述待读条带所属文件或者卷的最近一次快照的快照 ID 对应，所述待读条带的对象 ID 是所述待读条带所属对象的 ID；条带读取模块，用于判断由所述对象 ID、所述待读条带版本号确定的对象是否已备份：如果已备份，则读取由所述对象 ID、所述待读条带版本号、所述待读条带偏移量以及所述待读条带大小所确定的数据，把读取的数据作为待读条带发送给所述客户服务器；如果未备份，则从对象 ID 和所述待读条带的对象 ID 相同、版本号和所述待读条带的版本号不同的对象中，按照对象的快照时间由晚到早的顺序，逐个对象进行查找，直至找到在所述待读条带偏移量的存储位置存储有有效数据的对象，把找到的有效数据作为待读条带发送给所述客户服务器，其中，所述对象的版本号和所述对象生成之前，所属文件或者卷的最近一次快照的快照 ID 对应。

第十三方面，本发明实施例提供一种数据存储系统，包括客户服务器和对象存储设备，所述客户服务器用于：接收文件写请求，所述文件写请求携带待写数据、待写数据偏移量、以及文件的名称，所述待写数据是所述文件的一部分；所述客户服务器根据所述文件的名称获得文件编号 FID，根据 FID 查询所述文件的元数据，获得所述文件的版本号，将所述文件的版本号作为所述待写条带版本号，其中，所述文件的版本号与所述文件的最近一次快照的快照 ID 对应；所述客户服务器按照所述待写数据偏移量以及所述待写数据的大小，把所述待写数据拆分成包括所述待写条带的多个条带，确定所述待写条带所属对象的 ID，以及获得所述待写条带偏移量；创建条带写请求发送给所述对象存储设备；所述对象存储设备用于：接收

所述条带写请求，所述条带写请求中携带待写条带、待写条带版本号、待写条带偏移量、以及所述待写条带的对象 ID，其中，所述待写条带版本号与所述待写条带所属文件的最近一次快照的快照 ID 对应，所述待写条带偏移量描述所述待写条带在所属对象中的位置，所述待写条带的对象 ID 是所述待写条带所属对象的 ID；所述 OSD 将所述待写条带写入由所述对象 ID、所述待写条带版本号以及所述待写条带偏移量确定的存储位置。

第十四方面，本发明实施例提供一种数据存储系统，包括所述客户服务器和对象存储设备，客户服务器用于：接收卷写请求，所述卷写请求携带有待写数据、待写数据偏移量以及卷的编号 ID，所述待写数据是所述卷的一部分；根据所述卷的 ID 查询所述卷的元数据，获得所述卷的版本号，其中，所述卷的版本号与所述卷的最近一次快照的快照 ID 对应；按照所述待写数据偏移量以及所述待写数据的大小，把所述待写数据段拆分成包括待写条带的多个条带，确定所述待写条带所属对象的 ID，以及获得待写条带偏移量；创建条带写请求发送给所述对象存储设备；所述对象存储设备用于：接收所述条带写请求，所述条带写请求中携带所述待写条带、待写条带版本号、待写条带偏移量、以及所述待写条带的对象 ID，其中，所述卷的版本号为所述待写条带版本号，所述待写条带偏移量描述所述待写条带在所属对象中的位置，所述待写条带的对象 ID 是所述待写条带所属对象的 ID；所述 OSD 将所述待写条带写入由所述对象 ID、所述待写条带版本号以及所述待写条带偏移量确定的存储位置。

第十五方面，本发明实施例提供一种数据存储系统，包括客户服务器和对象存储设备，所述客户服务器用于：接收文件写请求，所述文件写请求携带待写数据、待写数据偏移量、以及文件的名称，所述待写数据是所述文件的一部分；所述客户服务装置根据所述文件的名称获得文件编号 FID，根据 FID 查询所述文件的元数据，获得所述文件的版本号，其中，所述文件的版本号与所述文件的最近一次快照的快照 ID 对应；

所述客户服务器按照所述待写数据偏移量以及所述待写数据的大小，把所述待写数据拆分成包括所述待写条带的多个条带，确定所述待写条带所属对象的 ID，以及获得所述待写条带偏移量；创建所述条带写请求发送给所述对象存储设备；所述对象存储设备用于：接收所述条带写请求，所述条带写请求中携带待写条带、待写条带版本号、待写条带偏移量、以及所述待写条带的对象 ID，其中，所述待写条带版本号与所述待写条带所属文件的最近一次快照的快照 ID 对应，所述待写条带偏移量描述所述待写条带在所属对象中的位置，所述待写条带的对象 ID 是所述待写条带所属对象的 ID；用于判断由所述待写条带版本号和所述对象 ID 确定的对象是否已备份：如果已备份，则将所述待写条带写入由所述对象 ID、所述待写条带版本号以及所述待写条带偏移量确定的存储位置；如果未备份，则使用所述待写条带建立一个拼接对象，然后把所述拼接对象写入由所述待写条带版本号和所述对象 ID 确定的存储位置。

第十六方面，本发明实施例提供一种数据存储系统，包括客户服务器和对象存储设备，所述客户服务器用于：接收卷写请求，所述卷写请求携带有待写数据、待写数据偏移量以及卷的编号 ID，所述待写数据是所述卷的一部分；根据所述卷的 ID 查询所述卷的元数据，获得所述卷的版本号，其中，所述卷的版本号与所述卷的最近一次快照的快照 ID 对应；按照所述待写数据偏移量以及所述待写数据的大小，把所述待写数据段拆分成包括所述待写条带的多个条带，确定所述待写条带所属对象的 ID，以及获得所述待写条带偏移量；创建所述条带写请求发送给所述对象存储设备；所述对象存储设备用于：接收所述条带写请求，所述条带写请求中携带待写条带、待写条带版本号、待写条带偏移量、以及所述待写条带的对象 ID，其中，所述待写条带版本号与所述待写条带所属卷的最近一次快照的快照 ID 对应，所述待写条带偏移量描述所述待写条带在所属对象中的位置，所述待写条带的对象 ID 是所述待写条带所属对象的 ID；用于判断由所述待写条带版本号和所述对象 ID 确定的对象是否已备份：如果已备份，则将所述待

写条带写入由所述对象 ID、所述待写条带版本号以及所述待写条带偏移量确定的存储位置；如果未备份，则使用所述待写条带建立一个拼接对象，然后把所述拼接对象写入由所述待写条带版本号和所述对象 ID 确定的存储位置。

第十七方面， 本发明实施例提供一种读数据系统，包括客户服务器和对象存储设备，所述客户服务器用于：接收文件读请求，所述文件读请求中携带文件的名称、待读数据大小、待读数据偏移量，所述待读数据是所述文件的一部分；根据所述文件的名称获得文件编号 FID，根据 FID 查询所述文件的元数据，获得所述文件的版本号，将所述文件的版本号作为所述待读条带版本号，其中，所述文件的版本号与所述待读条带所属文件的最近一次快照的快照 ID 对应；按照所述待读数据偏移量以及所述待读数据的大小，确定所述待读条带所属对象的 ID，以及获得所述待读条带偏移量；生成条带读请求并发送；所述对象存储设备用于：接收所述读条带请求，所述读条带请求中携带待读条带大小、待读条带偏移量、待读条带版本号以及待读条带的对象 ID，其中，所述待读条带版本号与所述待读条带所属文件的最近一次快照的快照 ID 对应，所述待读条带的对象 ID 是所述待读条带所属对象的 ID；判断由所述对象 ID、所述待读条带版本号以及所述待读条带偏移量所确定的条带是否已备份：如果已备份，则读取由所述对象 ID、所述待读条带版本号、所述待读条带偏移量以及所述待读条带大小所确定的数据，把读取的数据作为待读条带发送给所述客户服务器；如果未备份，则从对象 ID 和所述待读条带的对象 ID 相同、版本号和所述待读条带的版本号不同的对象中，按照对象的快照时间由晚到早的顺序，逐个对象进行查找，直至找到在所述待读条带偏移量的存储位置存储有有效数据的对象，把找到的有效数据作为待读条带发送给所述客户服务器，其中，所述对象的版本号和所述对象生成之前，所属文件或者卷的最近一次快照的快照 ID 对应。

第十八方面， 本发明实施例提供一种读数据系统，包括客户服务器和

对象存储设备, 所述客户服务器用于: 接收卷读请求, 所述卷读请求中携带卷 ID、待读数据大小、待读数据偏移量, 所述待读数据是所述卷的一部分; 根据卷 ID 查询所述卷的元数据, 获得所述卷的版本号, 将所述卷的版本号作为所述待读条带版本号, 其中, 所述卷的版本号与所述待读条带所属卷的最近一次快照的快照 ID 对应; 按照所述待读数据偏移量以及所述待读数据的大小, 确定所述待读条带所属对象的 ID, 以及获得所述待读条带偏移量; 生成条带读请求并发送; 所述对象存储设备用于: 接收所述读条带请求, 所述读条带请求中携带待读条带大小、待读条带偏移量、待读条带版本号以及待读条带的对象 ID, 其中, 所述待读条带版本号与所述待读条带所属文件或者卷的最近一次快照的快照 ID 对应, 所述待读条带的对象 ID 是所述待读条带所属对象的 ID; 判断由所述对象 ID、所述待读条带版本号以及所述待读条带偏移量所确定的条带是否已备份: 如果已备份, 则读取由所述对象 ID、所述待读条带版本号、所述待读条带偏移量以及所述待读条带大小所确定的数据, 把读取的数据作为待读条带发送给所述客户服务器; 如果未备份, 则从对象 ID 和所述待读条带的对象 ID 相同、版本号与所述待读条带的版本号不同的对象中, 按照对象的快照时间由晚到早的顺序, 逐个对象进行查找, 直至找到在所述待读条带偏移量的存储位置存储有有效数据的对象, 把找到的有效数据作为待读条带发送给所述客户服务器, 其中, 所述对象的版本号和所述对象生成之前, 所属文件或者卷的最近一次快照的快照 ID 对应。

应用本发明, 使用对象 ID 与版本号的组合替代现有技术中的对象 ID, 减少了对对象 ID 的数量, 从而降低了系统资源的损耗。

附图说明

为了更清楚地说明本发明实施例的技术方案, 下面将对实施例或现有技术描述中所需要使用的附图作简单地介绍, 下面描述中的附图仅仅是本发明的一些实施例, 还可以根据这些附图获得其他的附图。

图 1 是本发明实施例对象存储系统的架构图;

图 2 是本发明数据处理方法实施例流程图；

图 3A、图 3B 均是本发明实施例条带分布策略示意图；

图 4 是基于 ROW 的读条带方案实施例图；

图 5 是基于 COW 的读条带方案实施例图；

图 6 是本发明存储系统实施例结构示意图；

图 7 是本发明存储系统实施例组成示意图。

具体实施方式

下面将结合本发明实施例中的附图，对本发明的技术方案进行清楚、完整地描述，显然，所描述的实施例仅仅是本发明一部分实施例，而不是全部的实施例。基于本发明中的实施例所获得的所有其他实施例，都属于本发明保护的范围。

如图 1 所示是对象存储系统(Object-based Storage System)的架构图，可以包括客户服务器 11、对象存储设备 12。对象存储设备 12 可以为客户服务器 11 提供对象 (Object) 的存储服务。

基于对象的存储设备 (Object-based Storage Device, OSD) 可以称为对象存储设备。在对象存储技术中，基于对象存储设备构建存储系统，每个对象存储设备可以具有一定的智能，能够自动管理其上的数据分布。

对象是系统中数据存储的基本单位，以文件为例，一个对象实际上就是文件的一部分数据和这部分数据的属性信息的组合，属性信息又称为元数据 (Meta Data)，可以定义基于文件的廉价磁盘冗余阵列 (Redundant Arrays of Independent Disks, RAID) 参数、数据分布和服务质量等，而传统的存储系统中用文件或块作为基本的存储单位，在块存储系统中还需要始终追踪系统中每个块的属性，对象通过与存储系统通信维护自己的属性。在对象存储设备中，所有对象都有一个对象标识 (ID)，以便对对象进行访问。

OSD 具有一定的智能，它可以拥有 CPU、内存和存储介质。OSD 和块设备相比，提供的访问接口可以不同。在同一个存储系统中，可以有一个

或者多个 OSD，图 1 以 2 个 OSD 进行示例。目前国际上通常采用刀片式结构实现对象存储设备。OSD 可以提供三个功能：

(1) 数据存储。OSD 管理对象，并将它们存储在磁盘等存储介质中，OSD 不提供块接口访问方式，客户端请求数据时使用对象 ID、偏移量进行数据读写。

(2) 智能分布。OSD 用其自身的 CPU 和内存优化本地所存储的数据的分布，并支持数据的预取。由于 OSD 可以智能地支持对象的预取，从而可以优化数据读取速度。

(3) 每个对象元数据的管理。OSD 管理存储在其上对象的元数据，该元数据记录在一种称为索引节点 (index node, inode) 的数据结构中。元数据通常包括对象的大小、包含的条带数量等信息。在传统的网络附属存储 (Network Attached Storage, NAS) 系统中，这些元数据是由文件服务器维护的。对象存储架构可以将元数据有元数据服务器进行管理，也可以将系统中主要的元数据管理工作由 OSD 来完成，降低了客户端的开销。

当前一种存储方式是写时首次复制 (Copy On First Write, COFW)，有时简称为写时复制 (Copy On Write, COW)。即在新数据第一次写入到存储设备的某个存储位置时，首先将这个存储位置的原有数据读取出来，写到另一存储位置处 (这另一存储位置是为快照预留的存储位置，我们称为快照空间)，然后再将新数据写入到存储设备中。从 COW 的执行过程我们可以知道，这种实现方式需要执行一次读操作和两次写操作。

写时重定向写 (Redirect On First Write, ROW) 是另外一种存储新数据的方法。ROW 把新数据写入预留的存储位置，旧数据的存储位置保持不变。相对于 COW，它可以减少一次写操作。

在对象存储技术中，可以把大部分元数据管理工作分布到每个智能化的 OSD，每个 OSD 负责管理本地所存储的数据的分布和检索，90% 的元数据管理工作分布到智能的存储设备，仅 10% 的元数据管理工作有元数据服务器执行，从而提高了系统元数据管理的性能。此外，OSD 是与网络连接

的设备，它自身包含存储介质，如磁盘或磁带，并具有足够的智能可以管理本地存储的数据。存储服务器直接与 OSD 通信，访问它存储的数据，由于 OSD 具有智能，因此不需要文件服务器的介入。

对象是数据和数据属性的综合体。数据属性可以根据应用的需求进行设置，包括数据分布、服务质量等。客户服务器 11 可以是基于 NAS 协议的服务器或者存储区域网（Storage Area Network， SAN）协议的服务器存储区域网。也就是说，本发明的实施例既适用文件系统，也适用块系统。

对网络附属存储（Network Attached Storage， NAS）的数据而言，本发明实施例中的对象来自于文件，文件拆成多个分片，一个分片以及这个分片的属性、元数据等信息共同组成一个对象。类似的，对存储区域网（Storage Area Network， SAN）的数据而言，被拆成分片的是卷（Volume）。

现有技术使用对象 ID 确定对象，因此每个对象的 ID 是唯一的，当同一个文件多次更新后。会产生大量的对象 ID，记录对象 ID 需要耗费大量的存储空间。本发明各实施例中，使用对象 ID 和版本号的组合共同确定对象，当一个文件的数据发生了多次更新，如果更新的数据的偏移量范围不变，那么更新数据相应的对象 ID 可以保持不变，仅仅更新不同对象版本号即可，减少了系统维护的对象 ID 的总数。而且本发明实施例的方案中，对象版本号和快照 ID 有对应关系，在两次快照之间，不论文件的数据发生了多少次更新，同一个文件内的所有对象使用同一个版本号，因此版本号占用的存储空间很少。

现有技术中，文件或者卷的内容进行更新后，对修改涉及的对象的元数据，需要对存储在文件层（对块系统而言是卷语义层）的元数据进行更新，更新的数据量较多。此外，接入节点可以通过客户服务器访问 OSD，如果不同接入节点都可以访问修改所涉及的对象，那么节点间需要进行元数据同步，具体而言，当一个接入节点对对象的元数据进行更新后，会引发其他接入节点对修改的对象所在文件的所有对象 ID 进行整体更新，频繁的同步导致了元数据的严重膨胀。而本发明实施例提供的方案，对象 ID 不

需要改变，仅需要在 OSD 层对版本号进行更新，更新的数据量远远小于现有技术。此外，本发明实施例中对象 ID 是由偏移量计算获得的。

如图 2 所述，以文件请求为例，对本发明数据处理方法实施例流程图进行具体说明。如果把其中关于文件系统的各个术语，替换成块系统的相应术语，那么就是另外一种实施方式。例如：文件替换成卷，文件元数据替换成卷元数据，文件 ID 替换成卷 ID，文件版本号替换成卷版本号，文件 ID 替换成卷 ID。不同之处在于：（1）卷元数据有另外的存储位置，不是存储在 inode 中；（2）卷 ID 可以直接获得，不需要使用卷名转换获得。

步骤 20，创建快照，快照的目标是文件或者包括了文件的文件系统，也就是说快照的目标包括文件，为快照分配快照 ID。

创建快照包括 2 种方式，一种是对文件创建快照，快照的目标是单个文件。另外一种是对文件系统创建快照，快照的目标是整个文件系统，文件系统中包括多个文件。这两种方式下，文件元数据的保存位置不同。

在创建文件快照的方式中，选定文件创建快照，给文件设置快照名，如果快照名是没有使用过的，则为文件的快照分配快照 ID。并将文件快照 ID 作为文件的元数据保存在文件的 inode (index node) 中。需要说明的是，快照 ID 是快照的标记，例如使用创造快照的时间点作为快照 ID。或者按创建快照的时间点的先后顺序，使用递增的数字作为快照 ID。

在创建文件系统快照的方式中，选定文件系统进行快照，如果快照名是没有使用过的，则为文件系统的快照分配快照 ID。然后把分配的快照 ID 保存在文件系统的根 inode 中。在这种方式中，可以认为文件系统中各个文件的快照 ID 和文件系统的快照 ID 相同。和前一种方式所不同的是，文件的快照 ID 不是保存在文件的 inode 中，而是保存在文件系统的根 inode 中。

除了文件快照 ID，文件元数据中还包括文件编号(File Identification, FID)。文件元数据还可以包括文件大小 (Size) 以及写入时间等信息。

需要说明的是，步骤 20 是一个预置的步骤，和本方法实施例其他的步

骤有相对的独立性。本发明实施例主要描述在创建一次快照之后，以及创建下一次快照之前，客户服务器和 OSD 执行的操作。

步骤 21, 客户服务器接收文件写请求, 文件写请求中携带有待写数据、待写数据偏移量以及文件名。待写数据是文件的一部分。

具体而言, 本步骤可以由客户服务器的文件系统程序执行。文件写请求是能够被文件系统识别的写请求。文件写请求可以是创建某个文件, 或者使用待写数据对已经存在的文件进行更新, 待写数据是文件的一部分或者文件的全部。

所述文件写请求还可以携带待写数据大小, 以便后续步骤根据待写数据偏移量把待写数据拆分成条带。也可以不携带待写数据大小, 因为待写数据大小可以通过对待写数据进行测量获得。

待写数据偏移量描述了待写数据在文件内的相对位置。具体而言, 待写数据偏移量可以描述待写数据的起始位置相对于文件头的距离。如果待写数据偏移量是 0, 表示待写数据的起始位置是待写文件的起始位置。如果待写数据偏移量是 1KB, 表示待写数据的起始位置距离文件的起始位置 1KB 的数据大小。

可选的, 文件写请求还可以携带文件写请求的文件路径, 文件路径指示了文件以及映射关系表的存储位置。文件路径和文件名共同确定一个文件。例如文件路径和文件名的组合是 /root/mma/a1, 其中 /root/mma/ 是文件路径, a1 是文件名, /root/mma/ 这个路径下存储有文件以及映射关系表。

不同的文件可以有不同的文件名。同一个文件路径下的文件名不重复。

可选的, 写请求还可以携带映射关系表的存储位置, 映射关系表记录了文件名和 FID 的映射关系。

每创建一次快照会生成一个快照 ID, 每个快照 ID 有一个对应的文件版本号, 快照 ID 和文件版本号一一对应。而且相邻快照时间对应的快照 ID 的变化规律, 和相邻快照时间对应的文件版本号的变化规律相同。

在执行步骤 22 之前, 可以记录快照 ID 和文件版本号的映射关系,

包括以下两个步骤:

(1) 备份当前最新的文件元数据, 具体而言, 可以通过备份 inode 实现。文件级别的快照, 则备份文件的 inode。文件系统级别的快照, 则备份文件系统的 inode, 既包括文件的 inode 也包括文件的根 inode 中。

(2) 更新 inode 中的版本号。如果客户服务器中设置的写模式是 ROW, 更新的版本号保存在被备份的 inode 中。对于客户服务器中设置的写模式是 COW, 更新的版本号保存在备份生成的 inode 中; 可选的, 被备份的 inode 也可以记录更新的版本号。例如把 A inode 备份生成 B inode, 那么 A inode 是被备份的 inode; B inode 是备份生成的 inode。

步骤 20 中生成了快照 ID。文件版本号和快照 ID 存在对应关系, 而快照 ID 又和快照时间对应, 因此也可以认为文件版本号和快照时间有对应关系。对应关系是指, 每个文件版本号对应有一个唯一的快照 ID, 以及文件版本号的变化规律和快照 ID 相似。例如: 快照 ID 越大, 其文件版本号越大; 或者快照 ID 越大, 其文件版本号越小。在多个快照之间, 快照时间越晚的快照其 ID 也越大。

需要说明的是, 在例如 SAN 在内的基于块系统的写数据方法中。使用卷 ID 而不是文件名对卷进行标记。卷 ID 的作用和 FID 类似。此外卷没有和文件路径相类似的概念。因此, 在步骤 22 中不再需要查询映射关系表的步骤, 可以直接由卷 ID 查询卷元数据, 获得文件版本号。

步骤 22, 客户服务器使用文件名查询映射关系表, 获得待写数据所在文件的文件编号 (FID); 根据 FID 查询文件元数据, 获得文件版本号。

映射关系表记录了文件名和 FID 的映射关系, 文件名和 FID 一一对应。映射关系表的存储位置可以携带在文件写请求中, 由客户服务器从写请求中获得。映射关系表也可以由客户服务器预先存储在客户服务器中, 客户服务器根据文件路径找到映射关系表。映射关系表还可以存储在其他存储设备中。

还可以把获得的文件版本号更新到元数据中。更新后, 文件元数据中

记录了 FID 和文件版本号，使用 FID 即可从文件元数据中查询获得文件版本号。文件元数据可以保存在 inode 信息中。文件路径指示了 inode 的存储位置：由上文可知，对于 ROW，版本号保存在被备份的 inode 中，因此本步骤读取的是被备份的 inode。对于 COW，版本号保存在备份生成的 inode 中。因此本步骤读取的是备份生成的 inode。

文件版本号和文件快照 ID 有一一对应关系，客户服务器在生成快照 ID 后，生成与之一一对应的文件版本号。例如可以直接把快照 ID 作为文件版本号，也可以把快照 ID 运算后作为文件版本号。如果越晚创建的快照其快照 ID 越大，那么一种可选的方式是：越晚创建的快照，其快照 ID 的值越大；另外一种可选的方式是：越晚创建的快照，其快照 ID 的值越小。

在本发明各实施例中，有时候也采用待写条带版本号。待写条带版本号是待写条带所属文件的文件版本号。也就是说，来自同一文件的不同条带，其条带版本号相同。类似的，对象版本号（或者对象的版本号），是待写条带所属文件的文件版本号。也就是说，来自同一文件的不同对象，其对象版本号相同。

步骤 23，客户服务器将待写数据拆分成包括待写条带（strip）在内的多个条带。按照条带分布策略，获得待写条带偏移量以及待写条带所属对象的 ID，待写条带所属对象的 ID 也称为对象 ID。

客户服务器按照条带大小（Size）把数据拆分成一个或者多个条带。条带是一定大小的数据。其中，当待写数据小于或者等于单个条带的大小时，拆分成 1 个条带；否则拆分成多个条带。同一个文件拆分出的条带大小相同。条带大小（Size）可以保存在文件元数据中，在这种情况下，不同的文件可以使用不同的条带大小。条带大小也可以不保存在对象所属于的文件的元数据中，而是整个文件系统中的文件共用一个条带大小，在这种情况下，不同文件使用相同的条带大小，条带大小保存在文件系统的根 inode 中。对象可以看做一个容器，可以容纳条带。

举例：待写数据被拆分成若干个数据条带，则本步骤的条带是指被拆

分出的数据条带。或者待写数据在拆分成数据条带后，还生成若干个校验条带对数据条带进行数据保护，则本步骤的条带既包括数据条带也包括校验条带。

每个对象中拥有的条带总数可以保存在文件元数据中，在这种情况下，不同的文件的对象拥有的条带总数可以是不同的。每个对象中拥有的条带总数也可以不保存在对象所属于的文件的元数据中，在这种情况下，不同文件的对象拥有的条带总数是相同的。

需要说明的是，由待写数据偏移量可以知道待写数据在文件中的起始位置。由待写数据偏移量和待写数据大小可以知道待写数据在文件中的结束位置。如果待写数据的起始位置不是条带大小整数倍，或者结束位置的偏移量加 1 的值不是条带大小的整数倍，先按照条带大小对待写数据进行拆分，拆分的边界是条带大小整数倍。如果拆分后产生大小不足一个条带的数据（这种数据也可以称为条带的脏数据），将其补齐形成条带。由于本步骤的补齐操作，在没有特别说明的情况下，后续步骤中提到的条带、条带偏移量都是指补齐后的条带、条带偏移量。

例如：待写数据的偏移量范围是 4KB-300KB，条带的大小是 256KB。那么，以 0KB 和 256KB 作为边界拆分待写数据。形成 2 个数据块，这两个数据块在待写数据中的偏移量范围分别是 4KB-255KB 和 256KB-300KB。对这两个数据块进行补齐，形成 2 个大小为 256KB 的条带。其中，补齐前一个数据块的数据（大小是 $3\text{KB}-0\text{KB}=3\text{KB}$ ）来自前一个条带，补齐后一个数据块的数据（大小是 $511\text{KB}-300\text{KB}=21\text{KB}$ ）来自后一个条带。待写数据的偏移量是待写数据在文件内的相对位置。

另外一种补齐办法是：如果待写数据的起始位置不是条带大小整数倍，或者结束偏移量加 1 的值不是条带大小整数倍，可以对条带进行补齐操作。使得拆分后的条带大小一致，并且条带中不存在空白。可以把 OSD 中已经存储的数据读取出来作为补齐用的数据。

例如：待写数据的偏移量范围是 4KB-300KB，条带的大小是 256KB。

那么,可以把待写数据补齐后形成偏移量范围 0KB-511KB 的数据,然后再将其拆分成 0KB -255KB 和 256KB-511KB 共 2 个条带,使得每个条带的大小都是 256KB。

条带分布策略由客户服务器的文件系统提供。描述了条带所属于的对象,也就是条带和对象的对应关系。具体而言,可以是条带的偏移量和对象的对应关系。

对象 ID 唯一标记了一个对象,属于同一个文件的对象的 ID 不同,不同文件的对象的 ID 也不同。

可选的,对象 ID 和对象所属的文件的 FID 可以存在对应关系。也就是说例如,由对象 ID 可以知道这个对象 ID 所代表的对象来自的文件。

例如:一种可选的对象 ID 生成方式是,对象 ID 由 64 位二进制数组成,其中,前 32 位是对象所属文件的 ID,后 32 位由客户服务器赋予,后 32 位在文件内唯一,同一个文件不同对象的后 32 位不同,例如使用文件内的对象编号。在这种方式中,由对象 ID 的前 32 位即可获知对应的 FID。类似的,在块(block)系统中,也可以建立对象 ID 和卷 ID 的关系。

另外一种可选的对象 ID 生成方式是:对象 ID 由 48 位二进制数组成,前 16 位和文件对应,不同文件前 16 位不同;后 32 位由客户服务器赋予,后 32 位在文件内唯一,同一个文件不同对象的后 32 位不同。

在其他实施例中, ID 和对象所属的文件的 FID 存储存也可以不存在对应关系。

图 3A 与图 3B 示例了两种不同的条带分布策略。条带索引描述了条带在文件中的偏移量关系,条带索引是大于等于 0 的整数,最小的条带索引是 0,第二的条带索引是 1,第三的条带索引是 2,……,以此类推。索引数值相邻的 2 个条带,在文件中的偏移量也相邻。

一种可选的条带分布策略是,如图 3A: (1) 属于同一个文件的对象大小是固定的,由于同一个文件的条带大小是相同的,也就意味着不同的对象拥有的条带总数是相同的; (2) 条带按照索引顺序,先装满前一个对象

再装下一个对象，也就是说，按照条带在待写数据中的偏移量大小顺序，连续的若干个条带属于同一个对象中。如图 3A，每个对象固定由 4 个条带组成。样例：条带大小为 256KB，每个对象拥有 4 个条带，也就是说对象大小是 $256\text{KB} \times 4 = 1024\text{KB}$ 。那么第 1 个对象保存第 0~3 个条带，第 2 个对象保存 4~7 个条带，第 3 个对象保存第 8-11 个条带……相应的，第一个对象的 ID 是 0，第二个对象的 ID 是 1，第三个对象的 ID 是 3……

用条带偏移量描述条带在对象内的相对位置，具体而言，可以是条带的起始数据在对象内的相对位置。条带偏移量 = (条带索引 % 对象内的条带数量) $\times$ 条带大小。其中，% 是指取前项除以后项，然后取余数。因此，“条带索引 % 对象内的条带数量”的值是计算条带索引除以对象内的条带数量的余数。

另外一种可选的条带分布策略如图 3B：(1) 同一个文件中对象的大小不固定，也就是说，同一个文件的不同对象可以拥有不同的条带总数；(2) 对象总数固定，也就是说，不同文件拥有相同数量的对象，如图 3B，一共有 3 个对象。样例：条带大小为 256KB，对象总数固定为 3，则第 1 个条带（条带 0）位于第一个对象（对象 0）中，第 2 个条带（条带 1）位于第二个对象（对象 1）中，……，依次类推，第 4 个条带（条带 3）又位于第一个对象中，第 5 个条带（条带 4）又位于第二个对象中。条带索引是大于等于 0 的整数，描述条带之间在文件中的位置关系。同时可以确定各条带在所属对象内的偏移量，文件内的对象编号可以是条带索引除以文件中对象总数后得到的余数。具体计算公式可以是：文件内的对象编号 = 条带索引 % 文件中对象个数。条带偏移量 = (条带索引 / 对象个数) $\times$ 条带大小。

条带索引可以由待写数据的偏移量确定。例如：对整个文件而言，其拆分后起始数据位于第一个对象的条带（条带 0），而本次待写数据偏移量位于对象 1 的第 5 个条带（条带 4）。那么由待写数据拆分生成的条带中，第一个条带的索引就是 4，其余条带的索引依次类推。

以上介绍了两种计算条带所属对象的 ID 的方案，根据条带分布策略的

不同，还可以有其他实现方案，不同的分布策略使用的参数可以不同，而这些参数通常可以从客户服务器中查询获得。

由于各个条带的处理方式相同，因此下面仅以某一个“待写条带”为例进行介绍。

步骤 24，客户服务器选择用于存储待写条带的 OSD。

具体而言，本步骤可以由客户服务器的对象存储客户端执行。

一种可选的算法是根据待写条带的 FID 确定存储待写条带的 OSD。例如：FID 的哈希值除以 OSD 总数，余数作为存储待写条带的 OSD 的编号。也就是 FID 的哈希值对 OSD 总数取模。还可以有其他方案，例如由客户服务器选择任意一个 OSD 存储属于某个对象的待写条带。属于同一对象的条带可以存储到同一个 OSD 中。

此外，也可以根据待写条带的 FID 和对象 ID 共同确定存储条带的 OSD。实际上，算法可以任意选择，只要能选择一个 OSD 出来即可。

步骤 25，客户服务器发送条带写请求给 OSD，条带写请求携带待写条带、待写条带版本号、待写条带偏移量、待写条带所属对象 ID。可选的，还可以啊拨款待写条带大小。

可选的，OSD 既支持 ROW 也支持 COW 的情况下，还可以发送写模式，以便 OSD 按照客户服务器指定的写模式写入待写条带。写模式是 ROW 或者 COW。如果 OSD 仅支持一种写模式，则可以不用发送写模式给 OSD。

步骤 26，OSD 接收条带写请求，把待写条带写入 OSD 的存储介质。

当 OSD 只支持一种写模式时，OSD 可以不用确认写模式是 ROW 还是 COW，而直接以缺省写模式把待写条带写入存储介质。

OSD 把收到数据时先暂存在缓存中，本步骤可以把缓存中的待写数据存储到存储介质。

条带偏移量描述了条带在对象中的相对位置。具体而言，可以是条带的起始数据在对象中的相对位置；条带偏移量+条带大小=条带结束数据在对象中的相对位置。

数据的备份标记在 OSD 中，使用对象 ID 作为索引可以从 OSD 中查询数据的备份标记粒度。也可以缺省设置为 OSD 收到的所有条带都按照同样的备份标记粒度进行存储。属于同一个文件的条带使用同一种记录粒度。对于一个实际设备，可以只支持对象作为备份标记粒度或者仅支持条带作为备份标记粒度。在这种情况下，那么 OSD 可以不用查询备份标记粒度直接进行存储。

由于在 OSD 中，由对象 ID 和版本号这两个参数能够共同确定一个对象，因此本实施例中，把这两个参数组成的集合称作对象关键参数。由于在确定对象后，再借助条带偏移量，可以确定条带。也就是说，对象 ID、版本号以及条带偏移量这三个参数能够共同确定一个的条带，因此把这三个参数的组成的集合称为条带关键参数。

在 OSD 中，对象关键参数可以指向了一个用于存储对象的存储位置。具体而言，可以指向的是一个供对象使用的起始地址。可选的，也可以指向一个供对象使用的地址段。类似的，条带的关键参数也可以指向一个起始地址或者一个用于存储条带的地址段。起始地址、地址段可以是物理地址也可以是逻辑地址。

使用对象关键参数查找为对象参数所确定的对象的存储位置有多种可能情况。一种情况是，OSD 在收到条带写请求之前，已经记录有条带写请求中携带的对象关键参数，并为这组关键参数所代表的条带分配有存储位置。另外一种情况是，OSD 并没有记录这组关键参数，且并没有为这组关键参数所代表的条带分配存储位置，则 OSD 在收到条带写请之后，为这组对象关键参数分配存储位置。

对象集：对象 ID 相同、版本号不同的对象的集合。对象集中包括至少一个对象。对象集可以是一个逻辑概念，不用真实划分。

对象 ID：由对象内携带的数据在文件内的偏移量范围确定。如果同一个文件进行过多次快照，每次快照后把发生改变的数据存储到 OSD，那么发生改变的数据，如果偏移量相同的数据的对象 ID 相同。

OSD 中对会对对象或者条带是否被备份进行标记。备份标记粒度可以是条带或者对象，如果被标记的最小单元是条带，那么备份标记粒度就是条带，如果被标记的最小单元是对象，那么备份标记的粒度就是对象。

对象的备份标记：对象 ID+版本号确定的对象已经被备份。具体而言：创建版本号对应的快照后，对象 ID 对应的对象是否被备份。1 表示已经被备份，0 表示没有被备份。对象的备份标记是 0 具体有两种情况：一种情况是对象 ID+版本号确定的对象修改了，还没有执行备份操作；另外一种情况是没有对对象 ID+版本号确定的对象做修改。

条带的备份标记：对象 ID+版本号+条带偏移量确定的条带已经被备份。具体而言：创建版本号对应的快照后，对象 ID+条带偏移量对应的条带是否被备份。1 表示已经被备份，0 表示没有被备份。条带的备份标记是 0 具体有两种情况：一种情况是对象 ID+版本号+条带偏移量确定的条带修改了，还没有执行备份操作；另外一种情况是没有对对象 ID+版本号+条带偏移量确定的条带做修改。

通过比较对象版本号，可以确定同一个对象集中，不同对象快照时间的早晚。

OSD 写入待写条带的方式一共有四种可能性：（1）写模式是 ROW，备份标记粒度是条带；（2）写模式是 ROW，备份标记粒度是对象；（3）写模式是 COW，备份标记粒度是条带；（4）写模式是 COW，备份标记粒度是对象。对于一个 OSD，可以支持其中一种或者数种。下面对这四种可能性分别进行说明。

方式一：对于 ROW，OSD 中数据的备份标记粒度是条带。

按照条带请求中条带关键参数确定的存储位置，直接把待写条带写入 OSD 中。此外，写入完成后本步骤还可以把写入的条带所占用的存储位置（起始存储地址或者地址段）标记为“已写入有效数据”。条带存储在 OSD 的存储介质中所占用的存储位置也称为条带空间。

可以使用比特位标记对象内各个条带是否被备份。例如把这个条带的

存储位置的标识位设置为 1，1 表示已写入数据，0 表示无数据。可以用条带索引描述条带在对象中的先后次序，用标记位对对象内的各个条带进行标记。例如：一共有 4 个条带空间，那么 0000 表示 4 个条带空间都没有写入数据；0010 表示仅第 2 个条带空间写入了数据；0101 则表示第 1、3 个条带空间写入了数据，第 2、4 个条带空间没有写入数据。

需要说明的是，本实施例中所描述的第 N (N 是自然数) 个条带空间，是指条带空间在条带所属对象内的相对位置，并不是指条带索引。

至于对象内的条带编号的确定方法，例如可以使用条带的偏移量确定，偏移量的值越小其条带编号值越小，相邻条带的编号差值是 1，最小的条带编号是 0。如果条带分布策略是步骤图 3A 描述的策略，一种快捷的确定一个条带编号的算法是：条带编号=条带偏移量/条带大小。条带的偏移量是条带在对象内的偏移量。如果这个条带空间之前已经标记为“已备份”，则本步骤可以不重复标记，保持标记不变即可。

方式二：对于 ROW，OSD 中数据的备份标记粒度是对象。

和方式一相比，方式二判断备份标记的粒度不同，由判断条带的标识位变成判断对象的标识位。

使用条带写请求中携带对象关键参数在 OSD 的写入记录中进行查询，判断对象关键参数指向的存储位置是不是存储了有效数据。本实施例可以用判断标识位来确定一个存储位置是否存储了有效数据。例如，标识位是 1 表示存储了有效数据，标识位是 0 存储位置没有存储有效数据。通过判断对象关键参数指向的存储位置的标识位，可以确定本次收到的条带写请求，是不是在创建快照后的、对这个对象的第一次写入操作。例如，当标识位是 0 或者没有找到标识位，表明是快照后的首次写入；标识位是 1 表示不是快照后的首次写入。

如果不是对这个对象快照后的第一次写入，则直接把待写条带写入这个对象占用的存储位置，具体写入位置可以由条带关键参数确定。

如果条带写请求是对这个对象快照后，首次收到的对这个对象的写请

求。则用条带写请求中的待写条带以及从 OSD 中其他对象中获得的条带组合，拼接成一个称之为拼接对象的完整的对象。具体而言，余下部分来自的对象是：拥有有效数据的对象中，版本号最大（但是比条带请求中携带的版本号小）的对象。

也就是说从属于所述待写条带的对象 ID 的对象集、拥有有效数据的对象中，选择版本号最大的对象，从中获得与所述待写条带偏移量不同的条带，和所述待写条带共同组成拼接对象。所述 OSD 中存储的和所述待写条带的对象 ID 相同、且版本号不同的对象的集合称为所述待写条带的对象 ID 的对象集。写模式是 ROW 时，快照时间越晚则对应的对象版本号越大。待写条带的对象 ID 就是待写条带所属对象的 ID。

举例：每个对象由 32 个条带组成，而 OSD 收到的待写条带是其中的第 15 个，对于余下的 31 个条带，也就是第 1-第 14，以及第 16-第 32 个条带所来自的对象是：前一次快照后记录到 OSD 中的、拥有有效数据的相同对象 ID 的对象。

写入完成后，把这个对象的标识位记录为已经备份，例如把标识位设置为 1。这意味着已经完成了快照后的首次条带写入操作，也就是说，如果在下一次快照前再次对这个对象中的任意条带进行写入，将不再是快照后对这个对象的第一次写入，因此无需备份操作，直接写入条带即可。

由前面的介绍可知，同一个对象 ID 可以有多个对象，每个快照 ID 对应一个，这些对象写入 OSD 中的时间不同，写入时间相邻的对象的版本号相邻，写入时间越晚版本号越大。

在完成本次写操作后，本次新写入的对象成为对象集中新的成员。

方式三：写模式是 COW，数据的备份标记粒度是条带。

用条带写请求中的对象关键参数加上条带偏移量可以确定一个存储位置。先检测待写条带的关键参数所确定的存储位置是否已经存储有数据，如果判断结果为否或者没有找到记录，则意味着本次写请求是在创建快照后的首次写请求，需要先进行备份操作再写入待写条带。

通常情况下，在进行下一次快照之前，仅收到第一次条带写请求后，需要进行条带数据的备份，备份到待写条带的对象 ID、待写条带版本号以及待写条带偏移量共同确定的存储位置。因此，需要先把 OSD 已存储的最新条带备份到待写条带的关键参数指向的存储位置，再在备份出数据的存储位置中，写入本次收到的条带。OSD 已存储的最新条带是最近一次由客户服务器发送来的条带。本实施例中，它是 OSD 中已存储的条带中，拥有待写条带的对象 ID、版本号是 0、且偏移量和待写条带偏移量相同的条带。后续再收到条带写请求可以直接执行待写条带的写操作，不用再做备份。

在 COW 中，OSD 已存储的最新对象始终使用同一个版本号，例如使用 0 或者空（NULL）作为版本号，本实施例称之为初始版本号。在对象集的其余对象中，除了初始版本号之外的版本号中，版本号越小的对象，其对应的快照时间越晚。

在 ROW 或者 COW 中，文件第一次快照之前，写入数据到 OSD 时，使用的条带版本号就是初始版本号。初始版本号的值可以使用 0 或者空（NULL）。

在完成备份操作后，将条带写请求中携带的条带关键参数指向存储位置标记为已经存储有数据。在下一次快照之前，如果 OSD 再次收到对这个待写偏移量位置的 COW 写请求，可以不再迁移数据，以覆盖的方式把收到的条带写入版本号为 0 对象中、待写条带偏移量占用的存储位置。换句话说，把待写条带写入由待写条带对象 ID、初始版本号以及待写条带偏移量所确定的存储位置。

此外，本步骤还可以把写入待写条带的存储位置标记为已写入有效数据，具体标记方法可以参见方式一。

方式四，写模式是 COW，OSD 中数据的备份标记粒度是对象。

方式四和方式三的区别是：数据备份标记粒度由条带变成了对象，而且备份的粒度也由条带变成了对象。

用条带写请求中的对象关键参数可以确定一个存储位置。OSD 使用对

象关键参数在 OSD 的写入记录中进行查询，判断待写条带的对象关键参数指向的存储位置是否存储了有效数据。类似方式一中的描述，本实施例可以使用标识位对对象进行标记，例如用标识位 1 表示存储了有效数据，标识位是 0 或者在 OSD 的写入记录中没有查找到对象关键参数的标识位，表示没有存储有效数据。

通常情况下，在进行下一次快照之前，仅收到第一次条带写请求后，需要进行对象数据的备份。具体而言：如果存储有有效数据，意味着在创建快照后，已经对待写条带的对象 ID 以及待写条带版本号所共同确定的对象做过备份，不需要再次备份；如果没有存储有效数据或者在 OSD 中没有找到条带写请求中的对象关键参数的记录，意味着本步骤中需要先进行备份，然后才能写入本次收到的条带写请求中的待写条带。

如果对象关键参数所指向的存储位置已经存储有有效数据。则直接把待写条带写入待写对象 ID、版本号 0 以及待写条带偏移量共同确定的位置。

如果对象关键参数所指向的存储位置没有存储有有效数据，则先把 0 版本对象中的所有条带备份到条带写请求中对象关键参数指向的存储位置。备份完成后，把条带写请求中的对象关键数据指向的存储位置标记为 1。然后把待写条带写入 0 版本对象原本占用的存储位置中。写入位置由待写条带的对象 ID 待写条带版本号、初始版本号共同确定。

步骤 26 执行完成后，OSD 发送待写条带被存储成功的响应消息给客户服务器。

需要说明的是，步骤 26 在发生下一次快照之前执行。也就是说，步骤 21-26 都是第一次快照之后，下一次快照之前执行。步骤 21-26 是把待写条带写入 OSD 的流程。下面介绍如何把已经写入 OSD 的条带读出来，读、写过程是相对独立的两个方法。

步骤 27，客户服务器接收文件读请求，文件写请求中携带有文件名、待读数据大小、待读数据偏移量。

和文件写请求类似，文件读请求还可以携带文件读请求的文件路径，

文件路径记录了映射关系表的存储位置。由文件路径和文件名可以确定唯一确定一个文件。

具体而言，本步骤可以由客户服务器的文件系统程序执行。文件读请求是能够被文件系统识别的写请求。文件读请求请求读出的是一个完整的文件，或者文件的部分数据。

其中，待读数据偏移量述了待读数据在文件内的相对位置。具体而言，待读数据偏移量可以描述待读数据的起始位置相对于文件头的距离。如果待读数据偏移量是 0，表示待读数据的起始位置是待读文件的起始位置。如果待读数据偏移量是 2KB，表示待读数据的起始位置距离文件的起始位置 2KB 的数据大小。

可选的，文件读请求还可以携带文件路径，文件路径记录了映射关系表的存储位置。映射关系表的细节参见步骤 21 的介绍。

文件名可能是待读数据所在文件的文件名，也可能是待读数据所在文件的一个快照的文件名。如果是前者，说明文件读请求希望访问的是最新的待读数据；如果是后者，说明文件读请求希望访问的是某个快照的待读数据。

步骤 28，客户服务器使用文件名查询映射关系表，获得待读数据所在文件的 FID；根据 FID 查询文件元数据，获得文件版本号。

如果文件名是待读数据所在文件的的文件名，那么存储映射关系表的文件路径是待读数据所在文件的文件路径。根据文件对应的 FID 查询元数据获取到文件版本号。

如果文件名是快照的文件名，那么映射关系表的文件路径是快照文件所在路径。根据快照文件的 FID 查询元数据获取文件版本号。

映射关系表记录了文件名和 FID 的映射关系，文件名和 FID 一一对应。FID 的介绍以及 FID 和文件版本号的关系参见步骤 21 以及步骤 22。映射关系表的存储位置可以携带在文件读请求中，由客户服务器从写请求中获得。映射关系表也可以由客户服务器预先存储在客户服务器中，客户服务器根

据文件路径找到映射关系表。映射关系表还可以存储在其他存储设备中。

参见步骤 22，根据具体情况的不同。元数据可能存储在文件的 inode 中也可能存储在文件系统的根 inode 中。

快照 ID 和文件版本号存在对应一一关系，因此客户服务器根据快照 ID 可以获得文件版本号。这个对应关系可以存储在文件元数据中。

步骤 29，客户服务器将文件读请求处理转换成包括条带读请求在内的多个读请求。每个条带读请求用于请求读出一个条带，条带读请求用于向 OSD 请求读出待读条带。确定每个读请求对应的对象 ID。条带读请求中携带：待读条带版本号、待读条带偏移量、待读条带大小以及待读条带的对象 ID。

具体而言，根据待读数据大小、待读数据偏移量可以知道包括待读条带内的需要读出的每个条带的偏移量。

参见步骤 23 把生成条带的方法，按照条带大小，由待写数据偏移量和待写数据的长度可以把待写数据拆分成条带，获得待读条带的偏移量。依照同样的办法，本步骤由条带大小、待读数据偏移量以及待读数据长度同样可以获得每个需要读出的条带的偏移量。条带大小可以来自文件 inode，在这种情况下，不同文件可以使用不同的条带大小。也可以整个系统所有文件共用一个条带大小。

在获得待读条带的偏移量后，按照和步骤 23 相同的办法，可以获得待读条带所在对象的 ID。需要说明的是，不论文件名是待读数据所在文件的文件名还是快照的文件名，查询读请求对应的对象 ID 所使用的 FID 都是待读数据所在文件的 FID。

步骤 30，客户服务器选择用于发送条带读请求的 OSD。

具体而言，本步骤可以由客户服务器的对象存储客户端执行。

同一个条带的条带读请求和条带写请求必须对应到同一个 OSD。一种可行的办法是：使用和步骤 24 相同的 OSD 选择算法。

步骤 31，客户服务器发送条带读请求给步骤 30 选出的 OSD。

待读条带版本号实际上是待读条带所属文件的版本号。

可选的，还可以发送写模式给 OSD，写模式和步骤 25 中条带写请求中携带写模式保持一致。待读条带的对象 ID 就是待读条带所属对象的 ID。

步骤 32，OSD 接收条带读请求，查找待读条带的存储位置，把待读条带发送给客户服务器。

待读条带的存储位置可以是待读条带的起始地址，从起始地址开始读起，读出一个条带大小的数据，被读出的数据就是待读条带。

按照步骤 26 中，条带被写入的方式有多种可能。因此 OSD 可以使用相应的方式读出待读条带，下面同样分别进行说明。判断条带/对象是否已备份的方法可以使用步骤 26 介绍的标识位，例如如果标识位是 1 表示已备份，标识为 0 表示未备份。

对于 COW，可以有个特例，如果条带读请求中携带的版本号是初始版本号，其读出待读条带的方式不同于其他情况。相当于把初始版本号指定为最大版本号（即使初始版本号的值是 0）。因此，例如步骤 26 中介绍的版本号为 0 的情况，因为它已经是最大的版本号。则可以不用判断待读条带的关键参数确定的条带是否已备份，直接读出这个存储位置中的数据作为待读条带发送给客户服务器。在其他情况下，可以按如下两种方式读出待读条带。除了这个特例外，其他情况可以分为以下两种方式。

方式一，OSD 中数据的备份标记粒度是条带。

判断待读条带中携带的条带关键参数所确定条带是否已备份。换句话说，判断待读条带的对象 ID、待读条带以及待读条带偏移量确定的存储位置的条带是否已备份。本步骤中，可以把待读条带偏移量转换成待读条带在待读条带所属对象内的编号。转换方法参见步骤 26 的方式一。

如果已备份，则读出由待读条带的对象 ID、待读条带以及待读条带偏移量确定的条带，作为待读条带发送给客户服务器。

如果没有备份，则判断待读条带的对象 ID 的对象集中，前一个快照对象中的条带数据是否存在有效数据，直至找到有效条带数据返回。

具体而言，是从属于所述待读条带的对象 ID 的对象集、且快照时间比所述待读条带的快照时间早的对象中，使用待读条带偏移量，按照对象的快照时间由晚到早的顺序逐个对象进行查找，直至找到标记为已备份的条带，把找到的条带作为待读条带发送给所述客户服务器。其中，对象的快照时间是指生成这个对象前，文件或者包含这个文件的文件系统最晚一次快照的时间。

如果快照时间越晚快照版本号越大。那么按照对象的快照时间由晚到早进行查找，具体而言：对于 ROW，按照版本号从大到小的顺序逐个进行查找；对于 COW，从版本号从小到大顺序逐个进行查找。

当然，如果在条带被写入 OSD 时，使用的待读条带版本号是：快照时间点越晚则版本号越大。那么本步骤中查找的待读条带顺序相反。

方式二，OSD 中数据的备份标记粒度是对象。

本步骤和方式的区别在于备份标记粒度由条带变为了对象。

判断待读条带中携带的条带关键参数确定的存储位置是否存储有有效数据。换句话说，判断待读条带的对象 ID、待读条带版本号确定的存储位置（对象空间）是否存储有有效数据。

如果有有效数据，则读出由待读条带的对象 ID、待读条带版本号以及待读条带偏移量确定的有效数据，作为待读数据发送给客户服务器。

如果没有存储有效数据，则与本步骤方式一相似的方式，按照快照版本号由小到大的顺序，依次查找对象集中的对象，直至找到存储有有效的快照对象。按照所述待读条带偏移量从中读出待读条带发送给客户服务器。

图 4 是基于 ROW 的读条带方案，如图所示，文件 A 由对象 1、对象 2、对象 3 共同组成。在这些对象首次被存储到 OSD 后，他们的版本号是 0。在图 4 中，对象 1.0 表示对象 ID 是 1，版本号是 0 的对象。类似的，对象 3.2 表示对象 ID 是 3，版本号是 2 的对象。实线的对象表示这个对象有备份，虚线的对象的没有备份。

本例中，第一次快照（版本号是 1）后，对象 1 的数据没有更新，对象

2 以及对象 3 有备份, 对象 1 没有备份。第二次快照 (版本号是 2) 后, 对象 3 有备份, 对象 1 以及对象 2 有备份。第三次快照 (版本号是 3) 后, 对象 1 有备份, 对象 2、3 没有备份。

由对象集的概念可知, 在对象 1.0 所在的对象集中包括对象 1.0 和对象 1.3。在对象 2 所在的对象集中, 包括对象 2.0 和对象 2.1。在对象 3.0 所在的对象集中, 包括对象 3.0, 对象 3.1 以及对象 3.2。

图 4 中箭头的方向标记了对象的查找关系。如果条带读请求希望读出对象 1.2 中的条带, 由图可知, 这个对象没有备份, 而按照版本号由大到小的顺序, 对象 1.0 有备份, 因此读出读写 1.0 中的条带。基于同样的道理, 如果条带读请求希望读出对象 2.2 或者对象 2.3 中的待读条带, 那么实际读出的是对象 2.1 中的数据。当然, 如果条带读请求希望读出对象 1.3 或者对象 2.1 或者对象 3.2 中的数据, 由于这些对象已备份, 因此可以直接读出。

图 5 是基于 COW 的读条带方案, 和图 4 不同的是, 查找的顺序相反, 按照版本号由小到大的顺序进行查找。

如果备份标记粒度是条带, 其原理和图 4、图 5 相似, 区别在于备份标记所标记的目标物是对象中的条带而不是对象。

使用上述方式一或者方式二, 客户服务器收到条带读请求以及其他读请求的返还的数据, 把它们拼接起来即可生成待读数据。

如图 6, 是执行上述方法的硬件。客户服务器 41 的接口 413 和对象存储设备 42 的接口 423 连接。其中客户服务器 41 由处理器 411、存储介质 412、接口 413 组成, 处理器 411 和存储介质 412、接口 413 连接。存储介质 412 例如是内存, 其中存储有计算机程序。处理器 411 运行存储介质 412 中的程序, 执行上述方法中由客户服务器执行的步骤。接口 413 提供与 OSD 之间的接口, 例如发送读条带请求或者写条带请求给 OSD。客户服务器 41 可以不设置永久性存储器, 也就是说上述方法中涉及的需要客户服务器 41 记录的所有信息可以都记录在客户服务器 41 的易失性存储介质 412 中。

OSD42 包括处理器 421、存储介质 422、接口 423 以及硬盘 424 组成。处理器 421 和存储介质 422、接口 423 连接;硬盘 424 和存储介质 422 连接。存储介质 422 可以是易失性介质,例如是内存,其中存储有计算机程序。处理器 421 运行存储介质 422 中的程序,执行上述方法中由客户服务器执行的步骤。接口 423 提供与 OSD 之间的接口,例如发送读条带请求或者写条带请求给 OSD。硬盘 424 对条带提供持久化存储,例如为待写条带/对象提供物理存储空间,以及存储待读条带/对象,通常是非易失性存储介质。影片 424 可以用闪存或者可擦写光盘等其成介质代替。

参见图 7,是本发明实施例数据处理系统的结构图。

数据处理系统由客户服务装置 51 和对象存储装置 52 组成。其中客户服务装置 51 可以是物理设备例如服务器,也可以是由运行在服务器上的软件实现的虚拟模块;对象存储装置 52 可以是物理设备例如对象存储设备,也可以是由运行在对象存储设备上的软件实现的虚拟模块。客户服务装置 51 可以用于执行上述方法中由客户服务器执行的步骤;对象存储装置 52 可以用于执行上述方法中由对象存储设备执行的步骤,

客户服务装置 51 包括条带请求生成模块 511、和条带请求生成模块 511 连接的条带请求发送模块 512。可选的,还可以包括与条带请求生成模块 511 连接的快照模块 513。

对象存储装置 52 包括条带请求接收模块 521,以及和条带接收模块 521 连接的条带存储模块 522、条带读取模块 523。在实现存储条带的功能时,条带读取模块不是必须的。在实现读取条带的功能时,条带存储模块不是必须的。条带请求接收模块 521 和条带请求发送模块 512 连接。

下面对各模块功能继续具体说明。

快照模块 513,用于创建快照,快照的目标包括文件,为快照分配快照 ID。

创建快照包括 2 种方式,一种是对文件创建快照,快照的目标是单个文件。另外一种是对文件系统创建快照,快照的目标是整个文件系统,文

件系统中包括多个文件。这两种方式下，文件元数据的保存位置不同。

在创建文件快照的方式中，选定文件创建快照，给文件设置快照名，如果快照名是没有使用过的，为文件的快照分配快照 ID。并将文件快照 ID 作为文件的元数据保存在文件的 inode (index node) 中。需要说明的是，快照 ID 是快照的标记，例如使用创造快照的时间点作为快照 ID。或者按创建快照的时间点的先后顺序，使用递增的数字作为快照 ID。

在创建文件系统快照的方式中，选定文件系统进行快照，如果快照名是没有使用过的，则为文件系统的快照分配快照 ID。然后把分配的快照 ID 保存在文件系统的根 inode 中。在这种方式中，可以认为文件系统中各个文件的快照 ID 和文件系统的快照 ID 相同。和前一种方式所不同的是，文件的快照 ID 不是保存在文件的 inode 中，而是保存在文件系统的根 inode 中。

除了文件快照 ID，文件元数据中还包括文件编号 (FID, File Identification)。文件元数据还可以包括文件大小 (Size)、写入时间以及等信息。

需要说明的是，快照模块 513 是可选的。本发明实施例主要描述在创建一次快照之后，以及创建下一次快照之前，客户服务装置和对象存储装置的操作。

条带请求生成模块 511，用于接收文件写请求，文件写请求中携带有待写数据、待写数据偏移量以及文件名。待写数据是文件的一部分。

具体而言，条带请求生成模块 511 的功能可以是由客户服务器的文件系统程序执行。文件写请求是能够被文件系统识别的写请求。文件写请求可以是创建某个文件，或者使用待写数据对已经存在的文件进行更新，待写数据是文件的一部分或者文件的全部。

所述文件写请求还可以携带待写数据大小，以便后续根据待写数据偏移量把待写数据拆分成条带。也可以不携带待写数据大小，因为待写数据大小可以通过对待写数据进行测量获得。

待写数据偏移量描述了待写数据在文件内的相对位置。具体而言，待

写数据偏移量可以描述待写数据的起始位置相对于文件头的距离。如果待写数据偏移量是 0，表示待写数据的起始位置是待写文件的起始位置。如果待写数据偏移量是 1KB，表示待写数据的起始位置距离文件的起始位置 1KB 的数据大小。

可选的，文件写请求还可以携带文件写请求的文件路径，文件路径指示了文件以及映射关系表的存储位置。文件路径和文件名共同确定一个文件。例如文件路径和文件名的组合是 /root/mma/a1，其中 /root/mma/ 是文件路径，a1 是文件名，/root/mma/ 这个路径下存储有文件以及映射关系表。

不同的文件可以有不同的文件名。同一个文件路径下的文件名不重复。

可选的，写请求还可以携带映射关系表的存储位置，映射关系表记录了文件名和 FID 的映射关系。

在执行使用文件名查询映射关系表之前，可以记录快照 ID 和文件版本号的映射关系，可以执行以下两个操作。

(1) 备份当前最新的文件元数据，具体而言，可以通过备份 inode 实现。文件级别的快照，则备份文件的 inode。文件系统创建快照，则备份文件系统的 inode，既包括文件的 inode 也包括文件的根 inode 中。

(2) 更新 inode 中的版本号。如果客户服务器中设置的写模式是 ROW，更新的版本号保存在被备份的 inode 中。对于客户服务器中设置的写模式是 COW，更新的版本号保存在备份生成的 inode 中；可选的，被备份的 inode 也可以记录更新的版本号。例如把 A inode 备份生成 B inode，那么 A inode 是被备份的 inode；B inode 是备份生成的 inode。

文件版本号和快照 ID 存在对应关系，而快照 ID 又和快照时间对应，因此也可以认为文件版本号和快照时间有对应关系。对应关系是指，每个文件版本号对应有一个唯一的快照 ID。以及文件版本号的变化规律和快照 ID 相似。例如：快照 ID 越大，其文件版本号越大。或者快照 ID 越大，其文件版本号越小。在多个快照之间，快照时间越晚的快照其 ID 也越大。

需要说明的是，在例如 SAN 在内的基于块系统的写数据技术中。使用

卷 ID 而不是文件名对卷进行标记。卷 ID 的作用和 FID 类似。此外卷没有和文件路径相类似的概念。因此，不再需要查询映射关系表，可以直接由卷 ID 查询卷元数据，获得文件版本号。

条带请求生成模块 511，还用于使用文件名查询映射关系表，获得待写数据所在文件的文件编号（FID）；根据 FID 查询文件元数据，获得文件版本号。

映射关系表记录了文件名和 FID 的映射关系，文件名和 FID 一一对应。映射关系表的存储位置可以携带在文件写请求中，由客户服务器从写请求中获得。映射关系表也可以由客户服务器预先存储在客户服务器中，客户服务器根据文件路径找到映射关系表。映射关系表还可以存储在其他存储设备中。

条带请求生成模块 511 还可以把获得的文件版本号更新到元数据中。更新后，文件元数据中记录了 FID 和文件版本号，使用 FID 即可从文件元数据中查询获得文件版本号。文件元数据可以保存在 inode 信息中。文件路径指示了 inode 的存储位置：由上文可知，对于 ROW，版本号保存在被备份的 inode 中，因此条带请求生成模块 511 读取的是被备份的 inode。对于 COW，版本号保存在备份生成的 inode 中。因此条带请求生成模块 512 读取的是备份生成的 inode。

文件版本号和文件快照 ID 有一一对应关系，客户服务器在生成快照 ID 后，生成与之一一对应的文件版本号。例如可以直接把快照 ID 作为文件版本号，也可以把快照 ID 运算后作为文件版本号。如果越晚创建的快照其快照 ID 越大，那么一种可选的方式是：越晚创建的快照，其快照 ID 的值越大；另外一种可选的方式是：越晚创建的快照，其快照 ID 的值越小。

条带请求生成模块 511，还用于将待写数据拆分成包括待写条带（strip）在内的多个条带。按照条带分布策略，获得待写条带偏移量以及待写条带所属于的对象的 ID，这个 ID 称为对象 ID。

客户服务器按照条带大小（Size）把数据拆分成一个或者多个条带。条

带是一定大小的数据。其中，当待写数据小于或者等于单个条带的大小时，拆分成 1 个条带；否则拆分成多个条带。同一个文件拆分出的条带大小相同。条带大小（Size）可以保存在文件元数据中，在这种情况下，不同的文件可以使用不同的条带大小。条带大小也可以不保存在对象所属于的文件的元数据中，而是整个文件系统中的文件共用一个条带大小，在这种情况下，不同文件使用相同的条带大小，条带大小保存在文件系统根的 inode 中。对象可以看做一个容器，可以容纳条带。

举例：待写数据被拆分成若干个数据条带，则拆分生成的条带是指被拆分出的数据条带。或者待写数据在拆分成数据条带后，还生成若干个校验条带对数据条带进行数据保护，则拆分生成的条带既包括数据条带也包括校验条带。

每个对象中拥有的条带总数可以保存在文件元数据中，在这种情况下，不同的文件的对象拥有的条带总数可以是不同的。每个对象中拥有的条带总数也可以不保存在对象所属于的文件的元数据中，在这种情况下，不同文件的对象拥有的条带总数是相同的。

需要说明的是，由待写数据偏移量可以知道待写数据在文件中的起始位置。由待写数据偏移量和待写数据大小可以知道待写数据在文件中的结束位置。如果待写数据的起始位置不是条带大小整数倍，或者结束位置的偏移量加 1 的值不是条带大小的整数倍，先按照条带大小对待写数据进行拆分，拆分的边界是条带大小整数倍。如果拆分后产生大小不足一个条带的的数据（这种数据也可以称为条带的脏数据），将其补齐形成条带。由于条带请求生成模块 511 执行的补齐操作，在没有特别说明的情况下，后续提到的条带、条带偏移量都是指补齐后的条带、条带偏移量。

例如：待写数据的偏移量范围是 4KB-300KB，条带的大小是 256KB。那么，以 0KB 和 256KB 作为边界拆分待写数据。形成 2 个数据块，这两个数据块在待写数据中的偏移量范围分别是 4KB-255KB 和 256KB-300KB。对这两个数据块进行补齐，形成 2 个大小为 256KB 的条带。其中，补齐前

一个数据块的数据（大小是 $3\text{KB}-0\text{KB}=3\text{KB}$ ）来自前一个条带，补齐后一个数据块的数据（大小是 $511\text{KB}-300\text{KB}=21\text{KB}$ ）来自后一个条带。待写数据的偏移量是待写数据在文件内的相对位置。

另外一种补齐办法是：如果待写数据的起始位置不是条带大小整数倍，或者结束偏移量加 1 的值不是条带大小整数倍，可以对条带进行补齐操作。使得拆分后的条带大小一致，并且条带中不存在空白。可以把 OSD 中已经存储的数据读取出来作为补齐用的数据。

例如：待写数据的偏移量范围是 $4\text{KB}-300\text{KB}$ ，条带的大小是 256KB 。那么，可以把待写数据补齐后形成偏移量范围 $0\text{KB}-511\text{KB}$ 的数据，然后再将其拆分成 $0\text{KB}-255\text{KB}$ 和 $256\text{KB}-511\text{KB}$ 共 2 个条带，使得每个条带的大小都是 256KB 。

条带分布策略由客户服务器的文件系统提供。描述了条带所属于的对象，也就是条带和对象的对应关系。具体而言，可以是条带的偏移量和对象的对应关系。

对象 ID 唯一标记了一个对象，属于同一个文件的对象的 ID 不同，不同文件的对象的 ID 也不同。

可选的，对象 ID 和对象所属的文件的 FID 可以存在对应关系。也就是说例如，由对象 ID 可以知道这个对象 ID 所代表的对象所来自的文件。

例如：一种可选的对象 ID 生成方式是，对象 ID 由 64 位二进制数组成，其中，前 32 位是对象所属的文件的 ID，后 32 位由客户服务器赋予，后 32 位在文件内唯一，同一个文件不同对象的后 32 位不同，例如使用文件内的对象编号。在这种方式中，由对象 ID 的前 32 位即可获知对应的 FID。类似的，在块（block）系统中，也可以建立对象 ID 和卷 ID 的关系。

另外一种可选的对象 ID 生成方式是：对象 ID 由 48 位二进制数组成，前 16 位和文件对应，不同文件前 16 位不同；后 32 位由客户服务器赋予，后 32 位在文件内唯一，同一个文件不同对象的后 32 位不同。

在其他实施例中，ID 和对象所属的文件的 FID 存储也可以不存在对

应关系。

图 3A 与图 3B 示例了两种不同的条带分布策略。条带索引描述了条带在文件中的偏移量关系，条带索引是大于等于 0 的整数，最小的条带索引是 0，第二小的条带索引是 1，第三小的条带索引是 2，...，以此类推。索引数值相邻的 2 个条带，在文件中的偏移量也相邻。

一种可选的条带分布策略是，如图 3A：（1）属于同一个文件的对象大小是固定的，由于同一个文件的条带大小是相同的，也就意味着不同的对象拥有的条带总数是相同的；（2）条带按照索引顺序，先装满前一个对象再装下一个对象，也就是说，按照条带在待写数据中的偏移量大小顺序，连续的若干个条带属于同一个对象中。如图 3A，每个对象固定由 4 个条带组成。样例：条带大小为 256KB，每个对象拥有 4 个条带，也就是说对象大小是 $256\text{KB} \times 4 = 1024\text{KB}$ 。那么第 1 个对象保存第 0~3 个条带，第 2 个对象保存 4~7 个条带，第 3 个对象保存第 8-11 个条带...相应的，第一个对象的 ID 是 0，第二个对象的 ID 是 1，第三个对象的 ID 是 3....

用条带偏移量描述条带在对象内的相对位置，具体而言，可以是条带的起始数据在对象内的相对位置。条带偏移量 = (条带索引 % 对象内的条带数量) × 条带大小。其中，条带索引 % 对象内的条带数量是计算条带索引除以对象内的条带数量的余数的意思。

另外一种可选的条带分布策略如图 3B：（1）同一个文件中对象的大小不固定，也就是说，同一个文件的不同对象可以拥有不同的条带总数；（2）对象总数固定，也就是说，不同文件拥有相同数量的对象，如图 3B，一共有 3 个对象。样例：条带大小为 256KB，对象总数固定为 3，则第 1 个条带（条带 0）位于第一个对象（对象 0）中，第 2 个条带（条带 1）位于第二个对象（对象 1）中，...，依次类推，第 4 个条带（条带 3）又位于第一个对象中，第 5 个条带（条带 4）又位于第二个对象中。条带索引是大于等于 0 的整数，描述条带之间在文件中的位置关系。同时可以确定各条带在所属对象内的偏移量，文件内的对象编号可以是条带索引除以文件中对象

总数取模后得到的余数。具体计算公式可以是：文件内的对象编号=条带索引%文件中对象个数。条带偏移量=(条带索引/对象个数)×条带大小。

条带索引可以由待写数据的偏移量确定。例如：对整个文件而言，其拆分后起始数据位于第一个对象的条带（条带 0），而本次待写数据偏移量位于对象 1 的第 5 个条带（条带 4）。那么由待写数据拆分生成的条带中，第一个条带的索引就是 4，其余条带的索引依次类推。

以上介绍了两种计算条带所属对象的 ID 的方案，根据条带分布策略的不同，还可以有其他实现方案，不同的分布策略使用的参数可以不同，而这些参数通常可以从客户服务器中查询获得。

由于各个条带的处理方式相同，因此下面仅以“待写条带”作为代表进行介绍。

条带请求发送模块 512，用于选择用于存储待写条带的 OSD。

一种可选的算法是根据待写条带的 FID 确定存储待写条带的 OSD。例如：FID 的哈希值除以 OSD 总数，余数作为存储待写条带的 OSD 的编号。也就是 FID 的哈希值对 OSD 总数取模。还可以有其他方案，例如由客户服务器选择任意一个 OSD 存储属于某个对象的待写条带。属于同一对象的条带可以存储到同一个 OSD 中。

此外，也可以根据待写条带的 FID 和对象 ID 共同确定存储条带的 OSD。实际上，算法可以任意选择，只要能选择一个 OSD 出来即可。

条带请求发送模块 512，还用于发送条带写请求给 OSD，条带写请求携带待写条带、待写条带版本号、待写条带大小、待写条带偏移量、待写条带所属对象 ID。

可选的，OSD 既支持 ROW 也支持 COW 的情况下，还可以发送写模式，以便 OSD 按照客户服务器指定的写模式写入待写条带。写模式是 ROW 或者 COW。如果 OSD 仅支持一种写模式，则可以不用发送写模式给 OSD。

条带请求接收模块 521，用于接收条带写请求，把待写条带写入 OSD 的存储介质。

条带请求接收模块 521 可以执行步骤 26 的方法。例如可以使用四种方式中的一种或者多种实现待写条带的写入。

条带请求生成模块 511, 还可以用于接收文件读请求, 文件写请求中携带有文件名、待读数据大小、待读数据偏移量。

和文件写请求类似, 文件读请求还可以携带文件读请求的文件路径, 文件路径记录了映射关系表的存储位置。由文件路径和文件名可以确定唯一确定一个文件。

具体而言, 本步骤可以由客户服务器的文件系统程序执行。文件读请求是能够被文件系统识别的写请求。文件读请求请求读出的是一个完整的文件, 或者文件的部分数据。

其中, 待读数据偏移量述了待读数据在文件内的相对位置。具体而言, 待读数据偏移量可以描述待读数据的起始位置相对于文件头的距离。如果待读数据偏移量是 0, 表示待读数据的起始位置是待读文件的起始位置。如果待读数据偏移量是 2KB, 表示待读数据的起始位置距离文件的起始位置 2KB 的数据大小。

可选的, 文件读请求还可以携带文件路径, 文件路径记录了映射关系表的存储位置。映射关系表的细节参见步骤 21 的介绍。

文件名可能是待读数据所在文件的文件名, 也可能是待读数据所在文件的一个快照的文件名。如果是前者, 说明文件读请求希望访问的是最新的待读数据; 如果是后者, 说明文件读请求希望访问的是某个快照的待读数据。条带存储模块 522, 用于使用文件名查询映射关系表, 获得待读数据所在文件的 FID; 根据 FID 查询文件元数据, 获得文件版本号。

如果文件名是待读数据所在文件的的文件名, 那么存储映射关系表的文件路径是待读数据所在文件的文件路径。根据文件对应的 FID 查询元数据获取到文件版本号。

如果文件名是快照的文件名, 那么映射关系表的文件路径是快照文件所在路径。根据快照文件的 FID 查询元数据获取文件版本号。

映射关系表记录了文件名和FID的映射关系,文件名和FID一一对应。FID的介绍以及FID和文件版本号的关系参见步骤21以及步骤22。映射关系表的存储位置可以携带在文件读请求中,由客户服务器从写请求中获得。映射关系表也可以由客户服务器预先存储在客户服务器中,客户服务器根据文件路径找到映射关系表。映射关系表还可以存储在其他存储设备中。

参见步骤22,根据具体情况的不同。元数据可能存储在文件的inode中也可能存储在文件系统的根inode中。

快照ID和文件版本号存在对应一一关系,因此客户服务器根据快照ID可以获得文件版本号。这个对应关系可以存储在文件元数据中。

条带请求生成模块512,还可以用于:将文件读请求处理转换成包括条带读请求在内的多个读请求。每个条带读请求用于请求读出一个条带,条带读请求用于向OSD请求读出待读条带。确定每个读请求对应的对象ID。条带读请求中携带:待读条带版本号、待读条带偏移量、待读条带大小以及待读条带的对象ID。

具体而言,根据待读数据大小、待读数据偏移量可以知道包括待读条带内的需要读出的每个条带的偏移量。

参见步骤23把生成条带的方法,按照条带大小,由待写数据偏移量和待写数据的长度可以把待写数据拆分成条带,获得待读条带的偏移量。依照同样的办法,本步骤由条带大小、待读数据偏移量以及待读数据长度同样可以获得每个需要读出的条带的偏移量。条带大小可以来自文件inode,在这种情况下,不同文件可以使用不同的条带大小。也可以整个系统所有文件共用一个条带大小。

在获得待读条带的偏移量后,按照和步骤23相同的办法,可以获得待读条带所在对象的ID。需要说明的是,不论文件名是待读数据所在文件的文件名还是快照的文件名,查询读请求对应的对象ID所使用的FID都是待读数据所在文件的FID。

条带请求发送模块512,还可以用于:选择用于发送条带读请求的OSD。

具体而言，本步骤可以由客户服务器的对象存储客户端执行。

同一个条带的条带读请求和条带写请求必须对应到同一个 OSD。一种可行的办法是：使用和步骤 24 相同的 OSD 选择算法。

条带请求发送模块 512，还可以用于：发送条带读请求给选出的 OSD。

待读条带版本号实际上是待读条带所属文件的版本号。

可选的，还可以发送写模式给 OSD，写模式和步骤 25 中条带写请求中携带写模式保持一致。待读条带的对象 ID 就是待读条带所属对象的 ID。

条带请求接收模块 521，还可以用于：接收条带读请求，查找待读条带的存储位置，把待读条带发送给客户服务装置。

条带请求接收模块 521 可以实现步骤 32 的功能，例如使用步骤 32 提及方式一或者方式二读出待读条带。因此条带请求接收模块 521 的具体功能，可以参见步骤 32。

本发明的各个方面、或各个方面的可能实现方式可以被具体实施为系统、方法或者计算机程序产品。因此，本发明的各方面、或各个方面的可能实现方式可以采用完全硬件实施例、完全软件实施例（包括固件、驻留软件等等），或者组合软件和硬件方面的实施例的形式，在这里都统称为“电路”、“模块”或者“系统”。此外，本发明的各方面、或各个方面的可能实现方式可以采用计算机程序产品的形式，计算机程序产品是指存储在计算机可读介质中的计算机可读程序代码。

计算机可读介质可以是计算机可读信号介质或者计算机可读存储介质。计算机可读存储介质包含但不限于电子、磁性、光学、电磁、红外或半导体系统、设备或者装置，或者前述的任意适当组合，如随机存取存储器 (RAM)、只读存储器 (ROM)、可擦除可编程只读存储器 (EPROM 或者快闪存储器)、光纤、便携式只读存储器 (CD-ROM)。

计算机中的处理器读取存储在计算机可读介质中的计算机可读程序代码，使得处理器能够执行在流程图中每个步骤、或各步骤的组合中规定的

功能动作；生成实施在框图的每一块、或各块的组合中规定的功能动作的装置。

权 利 要 求 书

1、一种数据存储方法，其特征在于，所述方法包括：

对象存储设备 OSD 接收客户服务器发送的条带写请求，所述条带写请求中携带待写条带、待写条带版本号、待写条带偏移量、以及所述待写条带的对象 ID，其中，所述待写条带版本号与所述待写条带所属文件或者卷的最近一次快照的快照 ID 对应，所述待写条带偏移量描述所述待写条带在所属对象中的位置，所述待写条带的对象 ID 是所述待写条带所属对象的 ID；

所述 OSD 将所述待写条带写入由所述对象 ID、所述待写条带版本号以及所述待写条带偏移量确定的存储位置。

2、根据权利要求 1 所述的方法，其特征在于，在接收所述客户服务器发送的条带写请求之前，所述方法还包括：

所述客户服务器对所述待写条带所属文件或者卷进行快照，生成所述最近一次快照的快照 ID；

根据所述最近一次快照的快照 ID 生成所述待写条带版本号。

3、根据权利要求 2 所述的方法，其特征在于，所述方法还包括：

所述客户服务器把所述待写条带版本号更新到所述文件或者卷的元数据中。

4、根据权利要求 1、2 或 3 所述的方法，在所述 OSD 接收所述条带写请求之前，所述方法进一步包括：

所述客户服务器接收文件写请求，所述文件写请求携带待写数据、待写数据偏移量、以及文件的名称，所述待写数据是所述文件的一部分；

所述客户服务器根据所述文件的名称获得文件编号 FID，根据 FID 查询所述文件的元数据，获得所述文件的版本号，将所述文件的版本号作为所述待写条带版本号，其中，所述文件的版本号与所述待写条带所属文件的最近一次快照的快照 ID 对应；

所述客户服务器按照所述待写数据偏移量以及所述待写数据的大小，把所述待写数据拆分成包括所述待写条带的多个条带，确定所述待写条带所属对象

的 ID，以及获得所述待写条带偏移量；

创建所述条带写请求。

5、根据权利要求 1、2 或 3 所述的方法，在所述 OSD 接收所述条带写请求之前，所述方法进一步包括：

所述客户服务器接收卷写请求，所述卷写请求携带有待写数据、待写数据偏移量以及卷的编号 ID，所述待写数据是所述卷的一部分；

所述客户服务器根据所述卷的 ID 查询所述卷的元数据，获得所述卷的版本号，将所述卷的版本号作为所述待写条带版本号，其中，所述卷的版本号与所述待写条带所属卷的最近一次快照的快照 ID 对应；

所述客户服务器按照所述待写数据偏移量以及所述待写数据的大小，把所述待写数据段拆分成包括所述待写条带的多个条带，确定所述待写条带所属对象的 ID，以及获得所述待写条带偏移量；

创建所述条带写请求。

6、一种数据存储方法，其特征在于，所述方法包括：

对象存储设备 OSD 接收客户服务器发送的条带写请求，所述条带写请求中携带待写条带、待写条带版本号、待写条带偏移量、以及所述待写条带的对象 ID，其中，所述待写条带版本号与所述待写条带所属文件或者卷的最近一次快照的快照 ID 对应，所述待写条带偏移量描述所述待写条带在所属对象中的位置，所述待写条带的对象 ID 是所述待写条带所属对象的 ID；

所述 OSD 判断由所述待写条带版本号和所述对象 ID 确定的对象是否已备份：

如果已备份，则所述 OSD 将所述待写条带写入由所述对象 ID、所述待写条带版本号以及所述待写条带偏移量确定的存储位置；

如果未备份，则所述 OSD 使用所述待写条带建立一个拼接对象，然后把所述拼接对象写入由所述待写条带版本号和所述对象 ID 确定的存储位置。

7、根据权利要求 6 所述的方法，则所述 OSD 使用所述待写条带建立一个

拼接对象，具体包括：

所述 OSD 从属于所述待写条带的对象 ID 的对象集中的已备份对象中，选择快照时间最晚的对象，从中获得与所述待写条带偏移量不同的条带，使用与所述待写条带偏移量不同的条带以及所述待写条带共同组成所述拼接对象；

其中，所述 OSD 中存储的和所述待写条带的对象 ID 相同、且与所述待写条带版本号不同的对象的集合称为所述待写条带的对象 ID 的对象集。

8、根据权利要求 6 或 7 所述的方法，其特征在于，在接收所述客户服务器发送的条带写请求之前，所述方法还包括：

所述客户服务器对所述待写条带所属文件或者卷进行快照，生成所述最近一次快照的快照 ID；

根据所述最近一次快照的快照 ID 生成所述待写条带版本号；

所述客户服务器把所述待写条带版本号更新到所述文件或者卷的元数据中。

9、一种数据处理方法，其特征在于，该方法包括：

对象存储设备 OSD 接收客户服务器发送的条带写请求，所述条带写请求中携带待写条带、待写条带版本号、待写条带偏移量、以及所述待写条带的对象 ID，其中，所述待写条带版本号与所述待写条带所属文件或者卷的最近一次快照的快照 ID 对应，所述待写条带偏移量描述所述待写条带在所属对象中的位置，所述待写条带的对象 ID 是待写条带所属对象的 ID；

所述 OSD 判断由所述待写条带版本号、所述待写条带的对象 ID 以及所述待写条带偏移量确定的条带是否已备份；

如果已备份，则将所述待写条带写入由所述待写条带版本号、所述待写条带的对象 ID 以及所述待写条带偏移量确定的存储位置；

如果未备份，则将所述 OSD 中初始版本对象中位于所述待写条带偏移量、大小是所述待写条带大小的数据备份到由所述待写条带版本号、所述待写条带偏移量以及所述待写条带的对象 ID 确定的存储位置，其中，所述初始版本对象的对象 ID 和所述待写条带的对象 ID 相同，所述初始版本对象的版本号是初始

版本号；把所述待写条带写入由所述待写条带的对象 ID、所述初始版本号以及所述待写条带偏移量确定的存储位置。

10、根据权利要求 9 所述的方法，其特征在于，在接收所述客户服务器发送的条带写请求之前，所述方法还包括：

所述客户服务器对所述待写条带所属文件或者卷进行快照，生成所述最近一次快照的快照 ID；

根据所述最近一次快照的快照 ID 生成所述待写条带版本号。

11、根据权利要求 10 所述的方法，其特征在于，所述方法还包括：

所述客户服务器把所述待写条带版本号更新到所述文件或者卷的元数据中。

12、根据权利要求 9、10 或 11 所述的方法，在所述 OSD 接收所述条带写请求之前，所述方法进一步包括：

所述客户服务器接收文件写请求，所述文件写请求携带待写数据、待写数据偏移量、以及文件的名称，所述待写数据是所述文件的一部分；

所述客户服务器根据所述文件的名称获得文件编号 FID，根据 FID 查询所述文件的元数据，获得所述文件的版本号，将所述文件的版本号作为所述待写条带版本号，其中，所述文件的版本号与所述待写条带所属文件的最近一次快照的快照 ID 对应；

所述客户服务器按照所述待写数据偏移量以及所述待写数据的大小，把所述待写数据拆分成包括所述待写条带的多个条带，确定所述待写条带所属对象的 ID，以及获得所述待写条带偏移量；

创建所述条带写请求。

13、根据权利要求 9、10 或 11 所述的方法，在所述 OSD 接收所述条带写请求之前，所述方法进一步包括：

所述客户服务器接收卷写请求，所述卷写请求携带有待写数据、待写数据偏移量以及卷的编号 ID，所述待写数据是所述卷的一部分；

所述客户服务器根据所述卷的 ID 查询所述卷的元数据，获得所述卷的版本

号, 将所述卷的版本号作为所述待写条带版本号, 其中, 所述卷的版本号与所述待写条带所属卷的最近一次快照的快照 ID 对应;

所述客户服务器按照所述待写数据偏移量以及所述待写数据的大小, 把所述待写数据段拆分成包括所述待写条带的多个条带, 确定所述待写条带所属对象的 ID, 以及获得所述待写条带偏移量;

创建所述条带写请求。

14、一种数据处理方法, 其特征在于, 所述方法包括:

对象存储设备 OSD 接收客户服务器发送的条带写请求, 所述条带写请求中携带待写条带、待写条带版本号、待写条带偏移量、以及所述待写条带的对象 ID, 其中, 所述待写条带版本号与所述待写条带所属文件或者卷的最近一次快照的快照 ID 对应, 所述待写条带偏移量描述所述待写条带在所属对象中的位置, 所述待写条带的对象 ID 是所述待写条带所属对象的 ID;

所述 OSD 判断由所述待写条带版本号和所述对象 ID 确定的对象是否已备份:

如果已备份, 则所述 OSD 将所述待写条带写入由所述对象 ID、所述对象的版本号以及所述待写条带偏移量确定的存储位置;

如果未备份, 则将所述 OSD 中初始版本对象中的数据备份到由所述待写条带版本号以及所述对象 ID 确定的存储位置, 其中, 所述初始版本对象的对象 ID 和所述待写条带的对象 ID 相同, 所述初始版本对象的版本号是初始版本号;

所述 OSD 将所述待写条带写入由所述对象 ID、所述初始版本号以及所述待写条带偏移量确定的存储位置。

15、一种读数据方法, 其特征在于, 所述方法包括:

对象存储设备 OSD 接收客户服务器发送的读条带请求, 所述读条带请求中携带待读条带大小、待读条带偏移量、待读条带版本号以及待读条带的对象 ID, 其中, 所述待读条带版本号与所述待读条带所属文件或者卷的最近一次快照的快照 ID 对应, 所述待读条带的对象 ID 是所述待读条带所属对象的 ID;

所述 OSD 判断由所述对象 ID、所述待读条带版本号以及所述待读条带偏移量所确定的条带是否已备份:

如果已备份,则读取由所述对象 ID、所述待读条带版本号、所述待读条带偏移量以及所述待读条带大小所确定的数据,把读取的数据作为待读条带发送给所述客户服务器;

如果未备份,则从对象 ID 和所述待读条带的对象 ID 相同、版本号和所述待读条带的版本号不同的对象中,按照对象的快照时间由晚到早的顺序,逐个对象进行查找,直至找到在所述待读条带偏移量的存储位置存储有有效数据的对象,把找到的有效数据作为待读条带发送给所述客户服务器,其中,所述对象的版本号和所述对象生成之前,所属文件或者卷的最近一次快照的快照 ID 对应。

16、根据权利要求 15 所述的方法,该方法进一步包括:

如果所述 OSD 的写模式是 ROW,则对象的快照时间由晚到早的顺序是版本号由大到小的顺序;或者

如果所述 OSD 的写模式是 COW,则对象的快照时间由晚到早的顺序是版本号由小到大的顺序。

17、根据权利要求 15 或 16 的方法,在所述 OSD 接收所述客户服务器发送的条带写请求之前,所述方法还包括:

所述客户服务器接收文件读请求,所述文件读请求中携带文件的名称、待读数据大小、待读数据偏移量,所述待读数据是所述文件的一部分;

所述客户服务器根据所述文件的名称获得文件编号 FID,根据 FID 查询所述文件的元数据,获得所述文件的版本号,将所述文件的版本号作为所述待读条带版本号,其中,所述文件的版本号与所述待读条带所属文件的最近一次快照的快照 ID 对应;

所述客户服务器,按照所述待读数据偏移量以及所述待读数据的大小,确定所述待读条带所属对象的 ID,以及获得所述待读条带偏移量;

生成所述条带读请求。

18、根据权利要求 15 或 16 的方法，在所述 OSD 接收所述客户服务器发送的条带写请求之前，所述方法还包括：

所述客户服务器接收卷读请求，所述卷读请求中携带卷 ID、待读数据大小、待读数据偏移量，所述待读数据是所述卷的一部分；

所述客户服务器根据卷 ID 查询所述卷的元数据，获得所述卷的版本号，将所述卷的版本号作为所述待读条带版本号，其中，所述卷的版本号与所述待读条带所属卷的最近一次快照的快照 ID 对应；

所述客户服务器，按照所述待读数据偏移量以及所述待读数据的大小，确定所述待读条带所属对象的 ID，以及获得所述待读条带偏移量；

生成所述条带读请求。

19、一种读数据方法，应用于对象存储系统中，所述对象存储系统包括基于对象的存储设备 OSD 和客户服务器，其特征在于，所述方法包括：

所述 OSD 接收所述客户服务器发送的读条带请求，所述读条带请求中携带待读条带大小、待读条带偏移量、待读条带版本号以及待读条带的对象 ID，其中，所述待读条带版本号与所述待读条带所属文件或者卷的最近一次快照的快照 ID 对应，所述待读条带的对象 ID 是所述待读条带所属对象的 ID；

所述 OSD 判断由所述对象 ID、所述待读条带版本号确定的对象是否已备份：

如果已备份，则读取由所述对象 ID、所述待读条带版本号、所述待读条带偏移量以及所述待读条带大小所确定的数据，把读取的数据作为待读条带发送给所述客户服务器；

如果未备份，则从对象 ID 和所述待读条带的对象 ID 相同、版本号和所述待读条带的版本号不同的对象中，按照对象的快照时间由晚到早的顺序，逐个对象进行查找，直至找到在所述待读条带偏移量的存储位置存储有有效数据的对象，把找到的有效数据作为待读条带发送给所述客户服务器，其中，所述对象的版本号和所述对象生成之前，所属文件或者卷的最近一次快照的快照 ID

对应。

20、一种数据存储装置，其特征在于，所述装置包括：

条带请求接收模块，用于接收客户服务器发送的条带写请求，所述条带写请求中携带待写条带、待写条带版本号、待写条带偏移量、以及所述待写条带的对象 ID，其中，所述待写条带版本号与所述待写条带所属文件或者卷的最近一次快照的快照 ID 对应，所述待写条带偏移量描述所述待写条带在所属对象中的位置，所述待写条带的对象 ID 是所述待写条带所属对象的 ID；

条带存储模块，用于将所述待写条带写入由所述对象 ID、所述待写条带版本号以及所述待写条带偏移量确定的存储位置。

21、一种数据存储装置，其特征在于，所述装置包括：

条带请求接收模块，用于接收客户服务器发送的条带写请求，所述条带写请求中携带待写条带、待写条带版本号、待写条带偏移量、以及所述待写条带的对象 ID，其中，所述待写条带版本号与所述待写条带所属文件或者卷的最近一次快照的快照 ID 对应，所述待写条带偏移量描述所述待写条带在所属对象中的位置，所述待写条带的对象 ID 是所述待写条带所属对象的 ID；

条带存储模块，用于判断由所述待写条带版本号和所述对象 ID 确定的对象是否已备份：

如果已备份，则所述条带存储模块还用于将所述待写条带写入由所述对象 ID、所述待写条带版本号以及所述待写条带偏移量确定的存储位置；

如果未备份，则所述条带存储模块还用于使用所述待写条带建立一个拼接对象，然后把所述拼接对象写入由所述待写条带版本号和所述对象 ID 确定的存储位置。

22、根据权利要求 21 所述的装置，则所述条带存储模块还用于使用所述待写条带建立一个拼接对象，具体包括：

所述条带存储模块，用于从属于所述待写条带的对象 ID 的对象集中的已备份对象中，选择快照时间最晚的对象，从中获得与所述待写条带偏移量不同的

条带，使用与所述待写条带偏移量不同的条带以及所述待写条带共同组成所述拼接对象；

其中，所述数据存储装置中存储的和所述待写条带的对象 ID 相同、且与所述待写条带版本号不同的对象的集合称为所述待写条带的对象 ID 的对象集。

23、一种数据处理装置，其特征在于，所述装置包括：

条带请求接收模块，用于接收客户服务器发送的条带写请求，所述条带写请求中携带待写条带、待写条带版本号、待写条带偏移量、以及所述待写条带的对象 ID，其中，所述待写条带版本号与所述待写条带所属文件或者卷的最近一次快照的快照 ID 对应，所述待写条带偏移量描述所述待写条带在所属对象中的位置，所述待写条带的对象 ID 是待写条带所属对象的 ID；

条带存储模块，用于判断由所述待写条带版本号、所述待写条带的对象 ID 以及所述待写条带偏移量确定的条带是否已备份；

如果已备份，则将所述待写条带写入由所述待写条带版本号、所述待写条带的对象 ID 以及所述待写条带偏移量确定的存储位置；

如果未备份，则将所述数据存储装置中初始版本对象中位于所述待写条带偏移量、大小是所述待写条带大小的数据备份到由所述待写条带版本号、所述待写条带偏移量以及所述待写条带的对象 ID 确定的存储位置，其中，所述初始版本对象的对象 ID 和所述待写条带的对象 ID 相同，所述初始版本对象的版本号是初始版本号；把所述待写条带写入由所述待写条带的对象 ID、所述初始版本号以及所述待写条带偏移量确定的存储位置。

24、一种数据处理装置，其特征在于，所述装置包括：

条带请求接收模块，用于接收客户服务器发送的条带写请求，所述条带写请求中携带待写条带、待写条带版本号、待写条带偏移量、以及所述待写条带的对象 ID，其中，所述待写条带版本号与所述待写条带所属文件或者卷的最近一次快照的快照 ID 对应，所述待写条带偏移量描述所述待写条带在所属对象中的位置，所述待写条带的对象 ID 是所述待写条带所属对象的 ID；

条带存储模块,用于判断由所述待写条带版本号和所述对象 ID 确定的对象是否已备份:

如果已备份,则将所述待写条带写入由所述对象 ID、所述对象的版本号以及所述待写条带偏移量确定的存储位置;

如果未备份,则将初始版本对象中的数据备份到由所述待写条带版本号以及所述对象 ID 确定的存储位置,其中,所述初始版本对象的对象 ID 和所述待写条带的对象 ID 相同,所述初始版本对象的版本号是初始版本号;

所述条带存储模块,还用于将所述待写条带写入由所述对象 ID、所述初始版本号以及所述待写条带偏移量确定的存储位置。

25、一种读数据装置,其特征在于,所述装置包括:

条带请求接收模块,用于接收客户服务器发送的读条带请求,所述读条带请求中携带待读条带大小、待读条带偏移量、待读条带版本号以及待读条带的对象 ID,其中,所述待读条带版本号与所述待读条带所属文件或者卷的最近一次快照的快照 ID 对应,所述待读条带的对象 ID 是所述待读条带所属对象的 ID;

条带读取模块,用于判断由所述对象 ID、所述待读条带版本号以及所述待读条带偏移量所确定的条带是否已备份:

如果已备份,则读取由所述对象 ID、所述待读条带版本号、所述待读条带偏移量以及所述待读条带大小所确定的数据,把读取的数据作为待读条带发送给所述客户服务器;

如果未备份,则从对象 ID 和所述待读条带的对象 ID 相同、版本号和所述待读条带的版本号不同的对象中,按照对象的快照时间由晚到早的顺序,逐个对象进行查找,直至找到在所述待读条带偏移量的存储位置存储有有效数据的对象,把找到的有效数据作为待读条带发送给所述客户服务器,其中,所述对象的版本号和所述对象生成之前,所属文件或者卷的最近一次快照的快照 ID 对应。

26、根据权利要求 25 所述的装置,所述对象的快照时间由晚到早的顺序,

具体包括:

如果所述数据存储装置的写模式是 ROW, 则对象的快照时间由晚到早的顺序是版本号由大到小的顺序; 或者

如果所述数据存储装置的写模式是 COW, 则对象的快照时间由晚到早的顺序是版本号由小到大的顺序。

27、一种读数据装置, 其特征在于, 所述装置包括:

条带请求接收模块, 用于接收客户服务器发送的读条带请求, 所述读条带请求中携带待读条带大小、待读条带偏移量、待读条带版本号以及待读条带的对象 ID, 其中, 所述待读条带版本号与所述待读条带所属文件或者卷的最近一次快照的快照 ID 对应, 所述待读条带的对象 ID 是所述待读条带所属对象的 ID;

条带读取模块, 用于判断由所述对象 ID、所述待读条带版本号确定的对象是否已备份:

如果已备份, 则读取由所述对象 ID、所述待读条带版本号、所述待读条带偏移量以及所述待读条带大小所确定的数据, 把读取的数据作为待读条带发送给所述客户服务器;

如果未备份, 则从对象 ID 和所述待读条带的对象 ID 相同、版本号和所述待读条带的版本号不同的对象中, 按照对象的快照时间由晚到早的顺序, 逐个对象进行查找, 直至找到在所述待读条带偏移量的存储位置存储有有效数据的对象, 把找到的有效数据作为待读条带发送给所述客户服务器, 其中, 所述对象的版本号和所述对象生成之前, 所属文件或者卷的最近一次快照的快照 ID 对应。

28、一种数据存储系统, 包括客户服务器和对象存储设备, 其特征在于:

所述客户服务器用于:

接收文件写请求, 所述文件写请求携带待写数据、待写数据偏移量、以及文件的名称, 所述待写数据是所述文件的一部分;

所述客户服务器根据所述文件的名称获得文件编号 FID, 根据 FID 查询所

述文件的元数据，获得所述文件的版本号，其中，所述文件的版本号与所述文件的最近一次快照的快照 ID 对应；

所述客户服务器按照所述待写数据偏移量以及所述待写数据的大小，把所述待写数据拆分成包括待写条带的多个条带，确定所述待写条带所属对象的 ID，以及获得待写条带偏移量；

创建条带写请求发送给所述对象存储设备；

所述对象存储设备用于：

接收所述条带写请求，所述条带写请求中携带所述待写条带、待写条带版本号、所述待写条带偏移量、以及所述待写条带的对象 ID，其中，所述文件的版本号为所述待写条带版本号，所述待写条带偏移量描述所述待写条带在所属对象中的位置，所述待写条带的对象 ID 是所述待写条带所属对象的 ID；

所述 OSD 将所述待写条带写入由所述对象 ID、所述待写条带版本号以及所述待写条带偏移量确定的存储位置。

29、根据权利要求 28 所述的系统，其特征在于，所述客户服务器还用于：
在接收所述客户服务器发送的条带写请求之前，对所述待写条带所属文件进行快照，生成所述最近一次快照的快照 ID；

根据所述最近一次快照的快照 ID 生成所述待写条带版本号；

把所述待写条带版本号更新到所述文件的元数据中。

30、一种数据存储系统，包括客户服务器和对象存储设备，其特征在于：

所述客户服务器用于：

接收卷写请求，所述卷写请求携带有待写数据、待写数据偏移量以及卷的编号 ID，所述待写数据是所述卷的一部分；

根据所述卷的 ID 查询所述卷的元数据，获得所述卷的版本号，其中，所述卷的版本号与所述卷的最近一次快照的快照 ID 对应；

按照所述待写数据偏移量以及所述待写数据的大小，把所述待写数据段拆分成包括待写条带的多个条带，确定所述待写条带所属对象的 ID，以及获得待

写条带偏移量;

创建条带写请求发送给所述对象存储设备;

所述对象存储设备用于:

接收所述条带写请求, 所述条带写请求中携带所述待写条带、待写条带版本号、所述待写条带偏移量、以及所述待写条带的对象 ID, 其中, 所述卷的版本号为所述待写条带版本号, 所述待写条带偏移量描述所述待写条带在所属对象中的位置, 所述待写条带的对象 ID 是所述待写条带所属对象的 ID;

所述 OSD 将所述待写条带写入由所述对象 ID、所述待写条带版本号以及所述待写条带偏移量确定的存储位置。

31、根据权利要求 30 所述的系统, 其特征在于, 所述客户服务器还用于:
在接收所述客户服务器发送的条带写请求之前, 对所述待写条带所属卷进行快照, 生成所述最近一次快照的快照 ID;

根据所述最近一次快照的快照 ID 生成所述待写条带版本号;

把所述待写条带版本号更新到所述者卷的元数据中。

32、一种数据存储系统, 包括客户服务器和对象存储设备, 其特征在于:

所述客户服务器用于:

接收文件写请求, 所述文件写请求携带待写数据、待写数据偏移量、以及文件的名称, 所述待写数据是所述文件的一部分;

所述客户服务装置根据所述文件的名称获得文件编号 FID, 根据 FID 查询所述文件的元数据, 获得所述文件的版本号, 其中, 所述文件的版本号与所述文件的最近一次快照的快照 ID 对应;

所述客户服务器按照所述待写数据偏移量以及所述待写数据的大小, 把所述待写数据拆分成包括所述待写条带的多个条带, 确定所述待写条带所属对象的 ID, 以及获得所述待写条带偏移量;

创建所述条带写请求发送给所述对象存储设备;

所述对象存储设备用于:

接收所述条带写请求,所述条带写请求中携带待写条带、待写条带版本号、待写条带偏移量、以及所述待写条带的对象 ID,其中,所述待写条带版本号与所述待写条带所属文件的最近一次快照的快照 ID 对应,所述待写条带偏移量描述所述待写条带在所属对象中的位置,所述待写条带的对象 ID 是所述待写条带所属对象的 ID;

用于判断由所述待写条带版本号和所述对象 ID 确定的对象是否已备份:

如果已备份,则将所述待写条带写入由所述对象 ID、所述待写条带版本号以及所述待写条带偏移量确定的存储位置;

如果未备份,则使用所述待写条带建立一个拼接对象,然后把所述拼接对象写入由所述待写条带版本号和所述对象 ID 确定的存储位置。

33、根据权利要求 32 所述的系统,其特征在于,所述使用所述待写条带建立一个拼接对象,具体包括:

从属于所述待写条带的对象 ID 的对象集中的已备份对象中,选择快照时间最晚的对象,从中获得与所述待写条带偏移量不同的条带,使用与所述待写条带偏移量不同的条带以及所述待写条带共同组成所述拼接对象;

其中,所述数据存储设备中存储的和所述待写条带的对象 ID 相同、且与所述待写条带版本号不同的对象的集合称为所述待写条带的对象 ID 的对象集。

34、一种数据存储系统,包括客户服务器和对象存储设备,其特征在于:

所述客户服务器用于:

接收卷写请求,所述卷写请求携带有待写数据、待写数据偏移量以及卷的编号 ID,所述待写数据是所述卷的一部分;

根据所述卷的 ID 查询所述卷的元数据,获得所述卷的版本号,其中,所述卷的版本号与所述卷的最近一次快照的快照 ID 对应;

按照所述待写数据偏移量以及所述待写数据的大小,把所述待写数据段拆分成包括所述待写条带的多个条带,确定所述待写条带所属对象的 ID,以及获得所述待写条带偏移量;

创建所述条带写请求发送给所述对象存储装置；

所述对象存储装置用于：

接收所述条带写请求，所述条带写请求中携带待写条带、待写条带版本号、待写条带偏移量、以及所述待写条带的对象 ID，其中，所述待写条带版本号与所述待写条带所属卷的最近一次快照的快照 ID 对应，所述待写条带偏移量描述所述待写条带在所属对象中的位置，所述待写条带的对象 ID 是所述待写条带所属对象的 ID；

用于判断由所述待写条带版本号和所述对象 ID 确定的对象是否已备份：

如果已备份，则将所述待写条带写入由所述对象 ID、所述待写条带版本号以及所述待写条带偏移量确定的存储位置；

如果未备份，则使用所述待写条带建立一个拼接对象，然后把所述拼接对象写入由所述待写条带版本号和所述对象 ID 确定的存储位置。

35、根据权利要求 34 所述的系统，其特征在于，所述使用所述待写条带建立一个拼接对象，具体包括：

从属于所述待写条带的对象 ID 的对象集中的已备份对象中，选择快照时间最晚的对象，从中获得与所述待写条带偏移量不同的条带，使用与所述待写条带偏移量不同的条带以及所述待写条带共同组成所述拼接对象；

其中，所述数据存储设备中存储的和所述待写条带的对象 ID 相同、且与所述待写条带版本号不同的对象的集合称为所述待写条带的对象 ID 的对象集。

36、一种读数据系统，包括客户服务器和对象存储设备，其特征在于：

所述客户服务器用于：

接收文件读请求，所述文件读请求中携带文件的名称、待读数据大小、待读数据偏移量，所述待读数据是所述文件的一部分；

根据所述文件的名称获得文件编号 FID，根据 FID 查询所述文件的元数据，获得所述文件的版本号，其中，所述文件的版本号与所述文件的最近一次快照的快照 ID 对应；

按照所述待读数据偏移量以及所述待读数据的大小，确定所述待读条带所属对象的 ID，以及获得所述待读条带偏移量；

生成条带读请求并发送；

所述对象存储设备用于：

接收所述读条带请求，所述读条带请求中携带待读条带大小、待读条带偏移量、待读条带版本号以及待读条带的对象 ID，其中，所述待读条带版本号与所述待读条带所属文件的最近一次快照的快照 ID 对应，所述待读条带的对象 ID 是所述待读条带所属对象的 ID；

判断由所述对象 ID、所述待读条带版本号以及所述待读条带偏移量所确定的条带是否已备份；

如果已备份，则读取由所述对象 ID、所述待读条带版本号、所述待读条带偏移量以及所述待读条带大小所确定的数据，把读取的数据作为待读条带发送给所述客户服务器；

如果未备份，则从对象 ID 和所述待读条带的对象 ID 相同、版本号和所述待读条带的版本号不同的对象中，按照对象的快照时间由晚到早的顺序，逐个对象进行查找，直至找到在所述待读条带偏移量的存储位置存储有有效数据的对象，把找到的有效数据作为待读条带发送给所述客户服务器，其中，所述对象的版本号和所述对象生成之前，所属文件或者卷的最近一次快照的快照 ID 对应。

37、一种数据读系统，包括客户服务装置和对象存储设备，其特征在于：

所述客户服务器用于：

接收卷读请求，所述卷读请求中携带卷 ID、待读数据大小、待读数据偏移量，所述待读数据是所述卷的一部分；

根据卷 ID 查询所述卷的元数据，获得所述卷的版本号，其中，所述卷的版本号与所述卷的最近一次快照的快照 ID 对应；

按照所述待读数据偏移量以及所述待读数据的大小，确定所述待读条带所

属对象的 ID，以及获得所述待读条带偏移量；

生成条带读请求并发送；

所述对象存储设备用于：

接收所述读条带请求，所述读条带请求中携带待读条带大小、待读条带偏移量、待读条带版本号以及待读条带的对象 ID，其中，所述待读条带版本号与所述待读条带所属文件或者卷的最近一次快照的快照 ID 对应，所述待读条带的对象 ID 是所述待读条带所属对象的 ID；

判断由所述对象 ID、所述待读条带版本号以及所述待读条带偏移量所确定的条带是否已备份；

如果已备份，则读取由所述对象 ID、所述待读条带版本号、所述待读条带偏移量以及所述待读条带大小所确定的数据，把读取的数据作为待读条带发送给所述客户服务器；

如果未备份，则从对象 ID 和所述待读条带的对象 ID 相同、版本号和所述待读条带的版本号不同的对象中，按照对象的快照时间由晚到早的顺序，逐个对象进行查找，直至找到在所述待读条带偏移量的存储位置存储有有效数据的对象，把找到的有效数据作为待读条带发送给所述客户服务器，其中，所述对象的版本号和所述对象生成之前，所属文件或者卷的最近一次快照的快照 ID 对应。

38、一种对象存储设备，包括处理器、和所述处理器连接的存储介质和接口；

所述接口，用于接收客户服务器发送的条带写请求，所述条带写请求中携带待写条带、待写条带版本号、待写条带偏移量、以及所述待写条带的对象 ID，其中，所述待写条带版本号与所述待写条带所属文件或者卷的最近一次快照的快照 ID 对应，所述待写条带偏移量描述所述待写条带在所属对象中的位置，所述待写条带的对象 ID 是所述待写条带所属对象的 ID；

所述存储介质存储有计算机程序；

所述处理器，通过运行所述计算机程序，执行步骤：

将所述待写条带写入由所述对象 ID、所述待写条带版本号以及所述待写条带偏移量确定的存储位置。

39、一种对象存储设备，包括处理器、和所述处理器连接的存储介质和接口：

所述接口，用于接收所述客户服务器发送的条带写请求，所述条带写请求中携带待写条带、待写条带版本号、待写条带偏移量、以及所述待写条带的对象 ID，其中，所述待写条带版本号与所述待写条带所属文件或者卷的最近一次快照的快照 ID 对应，所述待写条带偏移量描述所述待写条带在所属对象中的位置，所述待写条带的对象 ID 是所述待写条带所属对象的 ID；

所述存储介质存储有计算机程序；

所述处理器，通过运行所述计算机程序，执行步骤：

判断由所述待写条带版本号和所述对象 ID 确定的对象是否已备份：

如果已备份，则所述条带存储模块还用于将所述待写条带写入由所述对象 ID、所述待写条带版本号以及所述待写条带偏移量确定的存储位置；

如果未备份，则所述条带存储模块还用于使用所述待写条带建立一个拼接对象，然后把所述拼接对象写入由所述待写条带版本号和所述对象 ID 确定的存储位置。

40、根据权利要求 23 所述的对象存储设备，所述处理器使用所述待写条带建立一个拼接对象，具体包括：

所述处理器，用于从属于所述待写条带的对象 ID 的对象集中的已备份对象中，选择快照时间最晚的对象，从中获得与所述待写条带偏移量不同的条带，使用与所述待写条带偏移量不同的条带以及所述待写条带共同组成所述拼接对象；

其中，所述数据存储设备中存储的和所述待写条带的对象 ID 相同、且与所述待写条带版本号不同的对象的集合称为所述待写条带的对象 ID 的对象集。

41、一种对象存储设备，包括处理器、和所述处理器连接的存储介质和接口：

所述接口，用于接收客户服务器发送的条带写请求，所述条带写请求中携带待写条带、待写条带版本号、待写条带偏移量、以及所述待写条带的对象 ID，其中，所述待写条带版本号与所述待写条带所属文件或者卷的最近一次快照的快照 ID 对应，所述待写条带偏移量描述所述待写条带在所属对象中的位置，所述待写条带的对象 ID 是待写条带所属对象的 ID；

所述存储介质存储有计算机程序；

所述处理器，通过运行所述计算机程序，执行步骤：

判断由所述待写条带版本号、所述待写条带的对象 ID 以及所述待写条带偏移量确定的条带是否已备份；

如果已备份，则将所述待写条带写入由所述待写条带版本号、所述待写条带的对象 ID 以及所述待写条带偏移量确定的存储位置；

如果未备份，则将所述数据存储设备中初始版本对象中位于所述待写条带偏移量、大小是所述待写条带大小的数据备份到由所述待写条带版本号、所述待写条带偏移量以及所述待写条带的对象 ID 确定的存储位置，其中，所述初始版本对象的对象 ID 和所述待写条带的对象 ID 相同，所述初始版本对象的版本号是初始版本号；把所述待写条带写入由所述待写条带的对象 ID、所述初始版本号以及所述待写条带偏移量确定的存储位置。

42、一种对象存储设备，包括处理器、和所述处理器连接的存储介质和接口：

所述接口，用于接收客户服务器发送的条带写请求，所述条带写请求中携带待写条带、待写条带版本号、待写条带偏移量、以及所述待写条带的对象 ID，其中，所述待写条带版本号与所述待写条带所属文件或者卷的最近一次快照的快照 ID 对应，所述待写条带偏移量描述所述待写条带在所属对象中的位置，所述待写条带的对象 ID 是所述待写条带所属对象的 ID；

所述存储介质存储有计算机程序;

所述处理器,通过运行所述计算机程序,执行步骤:

判断由所述待写条带版本号和所述对象 ID 确定的对象是否已备份:

如果已备份,则所述数据存储设备将所述待写条带写入由所述对象 ID、所述对象的版本号以及所述待写条带偏移量确定的存储位置;

如果未备份,则将所述数据存储设备中初始版本对象中的数据备份到由所述待写条带版本号以及所述对象 ID 确定的存储位置,其中,所述初始版本对象的对象 ID 和所述待写条带的对象 ID 相同,所述初始版本对象的版本号是初始版本号;

所述数据存储设备将所述待写条带写入由所述对象 ID、所述初始版本号以及所述待写条带偏移量确定的存储位置。

43、一种对象存储设备,包括处理器、和所述处理器连接的存储介质和接口:

所述接口,用于接收客户服务器发送的读条带请求,所述读条带请求中携带待读条带大小、待读条带偏移量、待读条带版本号以及待读条带的对象 ID,其中,所述待读条带版本号与所述待读条带所属文件或者卷的最近一次快照的快照 ID 对应,所述待读条带的对象 ID 是所述待读条带所属对象的 ID;

所述存储介质存储有计算机程序;

所述处理器,通过运行所述计算机程序,执行步骤:

判断由所述对象 ID、所述待读条带版本号以及所述待读条带偏移量所确定的条带是否已备份:

如果已备份,则读取由所述对象 ID、所述待读条带版本号、所述待读条带偏移量以及所述待读条带大小所确定的数据,把读取的数据作为待读条带发送给所述客户服务器;

如果未备份,则从对象 ID 和所述待读条带的对象 ID 相同、版本号和所述

待读条带的版本号不同的对象中，按照对象的快照时间由晚到早的顺序，逐个对象进行查找，直至找到在所述待读条带偏移量的存储位置存储有有效数据的对象，把找到的有效数据作为待读条带发送给所述客户服务器，其中，所述对象的版本号和所述对象生成之前，所属文件或者卷的最近一次快照的快照 ID 对应。

44、根据权利要求 43 所述的对象存储设备，所述对象的快照时间由晚到早的顺序，具体包括：

如果所述数据存储设备的写模式是 ROW，则对象的快照时间由晚到早的顺序是版本号由大到小的顺序；或者

如果所述数据存储设备的写模式是 COW，则对象的快照时间由晚到早的顺序是版本号由小到大的顺序。

45、一种对象存储设备，包括处理器、和所述处理器连接的存储介质和接口：

所述接口，用于接收客户服务器发送的读条带请求，所述读条带请求中携带待读条带大小、待读条带偏移量、待读条带版本号以及待读条带的对象 ID，其中，所述待读条带版本号与所述待读条带所属文件或者卷的最近一次快照的快照 ID 对应，所述待读条带的对象 ID 是所述待读条带所属对象的 ID；

所述存储介质存储有计算机程序；

所述处理器，通过运行所述计算机程序，执行步骤：

判断由所述对象 ID、所述待读条带版本号确定的对象是否已备份：

如果已备份，则读取由所述对象 ID、所述待读条带版本号、所述待读条带偏移量以及所述待读条带大小所确定的数据，把读取的数据作为待读条带发送给所述客户服务器；

如果未备份，则从对象 ID 和所述待读条带的对象 ID 相同、版本号和所述待读条带的版本号不同的对象中，按照对象的快照时间由晚到早的顺序，逐个对象进行查找，直至找到在所述待读条带偏移量的存储位置存储有有效数据

的对象，把找到的有效数据作为待读条带发送给所述客户服务器，其中，所述对象的版本号和所述对象生成之前，所属文件或者卷的最近一次快照的快照 ID 对应。