

一种创建表空间的方法、装置、存储介质及电子设备

技术领域

[01]本说明书涉及计算机技术领域，尤其涉及一种创建表空间的方法、装置、存储介质及电子设备。

背景技术

[02]随着信息科技的发展，数据库得到了广泛的应用。随着数据库广泛应用，数据库中表、索引等对应的数据文件也越来越多。为了便于管理数据文件，可以为数据库创建表空间。表空间是一个逻辑结构，是用于管理数据文件的存储空间。相对于物理结构，逻辑结构能方便服务器进行管理数据文件。例如，服务器可以使用表空间在数据库之间转移数据，不需要先下载一个数据库的数据，再将数据插入到另一个数据库中，服务器可以通过复制数据库的表空间的所有数据从而实现将一个数据库的数据转移到另一个数据库中。

[03]目前，采用新型无共享(share-nothing)架构的分布式数据库应用的比较广泛，然而，share-nothing 架构的分布式数据库是由各个作为节点的服务器共同组成的数据库，share-nothing 架构的分布式数据库的数据存储在不同节点中，并且各个节点的存储空间是不共享的。

[04]因此，如何在 share-nothing 架构的分布式数据库上创建表空间是一个亟待解决的问题。

发明内容

[05]本说明书提供一种创建表空间的方法、装置、存储介质及电子设备，以部分的解决上述问题。

[06]本说明书采用下述技术方案：本说明书提供一种创建表空间的方法，所述方法应用于分布式数据库，包括：接收表空间的创建指令，其中，所述创建指令中携带所需创建的表空间的容量，以及构成所述容量的存储空间所在的服务器的标识；在所述分布式数据库中，确定各标识分别对应的服务器，作为指定服务器；根据所述容量，在各指定服务器的剩余存储空间中，确定出总量为所述容量的存储空间；基于确定出的存储空间，

创建所述容量的虚拟文件；创建包含所述虚拟文件的表空间，所述表空间用于管理所述表空间自身包含的虚拟文件，所述虚拟文件用于写入数据。

[07]可选地，创建所述容量的虚拟文件之前，所述方法包括：确定每个指定服务器的剩余存储空间均不小于预设的单位容量。

[08]可选地，基于确定出的存储空间，创建所述容量的虚拟文件，具体包括：根据所述容量以及预设的单位容量，确定指定数量的大小均为预设的单位容量的各子文件；将各子文件轮循写入所述各指定服务器的剩余存储空间中，直至将所述指定数量的子文件全部写入为止；创建由各子文件构成的虚拟文件。

[09]可选地，将各子文件轮循写入所述各指定服务器的剩余存储空间中，具体包括：将所述各指定服务器进行排序，得到指定服务器序列；依次针对每个指定服务器，将该指定服务器作为待写入服务器；判断所述待写入服务器的剩余存储空间是否不小于所述单位容量；若是，将子文件写入所述待写入服务器的剩余存储空间中；否则，根据所述指定服务器序列，将该指定服务器的下一个指定服务器重新作为待写入服务器，直至写入子文件为止。

[10]可选地，所述方法还包括：针对每个子文件，确定该子文件在所述虚拟文件中的逻辑地址，并将该子文件所在的指定服务器作为目标服务器，确定该子文件在所述目标服务器中的物理地址，作为该子文件对应的物理地址；建立该子文件的所述逻辑地址与该子文件对应的所述物理地址的映射关系。

[11]可选地，所述方法还包括：接收删除指令，其中，所述删除指令中携带所需删除的服务器的标识；确定所述删除指令携带的标识对应的服务器，作为待删除服务器；将基于所述待删除服务器的存储空间创建的各虚拟文件，作为待删除虚拟文件；针对各待删除虚拟文件，将该待删除虚拟文件中的数据移动到不包含任一待删除虚拟文件的表空间中，将该待删除虚拟文件删除；将所述待删除服务器的标识从所述分布式数据库的服务器的标识列表中删除，所述服务器标识列表包含构成所述分布式数据库的所有服务器的标识。

[12]本说明书提供了一种创建表空间的装置，所述装置应用于分布式数据库，包括：接收模块，用于接收表空间的创建指令，其中，所述创建指令中携带所需创建的表空间的容量，以及构成所述容量的存储空间所在的服务器的标识；第一确定模块，用于在所述分布式数据库中，确定各标识分别对应的服务器，作为指定服务器；第二确定模块，用

于根据所述容量，在各指定服务器的剩余存储空间中，确定出总量为所述容量的存储空间；第一创建模块，用于基于确定出的存储空间，创建所述容量的虚拟文件；第二创建模块，用于创建包含所述虚拟文件的表空间，所述表空间用于管理所述表空间自身包含的虚拟文件，所述虚拟文件用于写入数据。

[13]可选地，所述第一创建模块，在创建所述容量的虚拟文件之前，还用于，确定每个指定服务器的剩余存储空间均不小于预设的单位容量。

[14]可选地，所述第一创建模块具体用于，根据所述容量以及预设的单位容量，确定指定数量的大小均为预设的单位容量的各子文件；将各子文件轮循写入所述各指定服务器的剩余存储空间中，直至将所述指定数量的子文件全部写入为止；创建由各子文件构成的虚拟文件。

[15]可选地，所述第一创建模块具体用于，将所述各指定服务器进行排序，得到指定服务器序列；依次针对每个指定服务器，将该指定服务器作为待写入服务器；判断所述待写入服务器的剩余存储空间是否不小于所述单位容量；若是，将子文件写入所述待写入服务器的剩余存储空间中；否则，根据所述指定服务器序列，将该指定服务器的下一个指定服务器重新作为待写入服务器，直至写入子文件为止。

[16]可选地，所述装置还包括：映射模块，用于针对每个子文件，确定该子文件在所述虚拟文件中的逻辑地址，并将该子文件所在的指定服务器作为目标服务器，确定该子文件在所述目标服务器中的物理地址，作为该子文件对应的物理地址；建立该子文件的所述逻辑地址与该子文件对应的所述物理地址的映射关系。

[17]可选地，所述装置还包括：删除模块，用于接收删除指令，其中，所述删除指令中携带所需删除的服务器的标识；确定所述删除指令携带的标识对应的服务器，作为待删除服务器；将基于所述待删除服务器的存储空间创建的各虚拟文件，作为待删除虚拟文件；针对各待删除虚拟文件，将该待删除虚拟文件中的数据移动到不包含任一待删除虚拟文件的表空间中，将该待删除虚拟文件删除；将所述待删除服务器的标识从所述分布式数据库的服务器标识列表中删除，所述服务器标识列表包含构成所述分布式数据库的所有服务器的标识。

[18]本说明书提供了一种计算机可读存储介质，所述存储介质存储有计算机程序，所述计算机程序被处理器执行时实现上述创建表空间的方法。

[19]本说明书提供了一种电子设备，包括存储器、处理器及存储在存储器上并可在处理

器上运行的计算机程序，所述处理器执行所述程序时实现上述创建表空间的方法。

[20]本说明书采用的上述至少一个技术方案能够达到以下有益效果：本说明书提供的创建表空间的方法中，接收表空间的创建指令，根据创建指令中携带的服务器的标识，在分布式数据库中，确定标识对应的服务器作为指定服务器，根据创建指令中携带的创建表空间的容量和指定服务器的剩余存储空间，确定出总量为容量的存储空间，基于确定出的存储空间创建虚拟文件，创建包含虚拟文件的表空间。

[21]从上述方法中可以看出，本方法中根据分布式数据库中的指定服务器的剩余存储空间和所需创建的表空间的容量，确定出总量为该容量的存储空间，基于确定出的存储空间创建虚拟文件，可以使原本分散的各指定服务器的剩余存储空间通过虚拟文件统一起来，从而可以创建包含虚拟文件的表空间，以在 share-nothing 架构下实现跨服务器的表空间的功能，可以利用表空间管理表空间包含的虚拟文件，可以向虚拟文件写入数据。

附图说明

[22]此处所说明的附图用来提供对本说明书的进一步理解，构成本说明书的一部分，本说明书的示意性实施例及其说明用于解释本说明书，并不构成对本说明书的不当限定。

[23]图 1 为本说明书中一种创建表空间的方法的流程示意图。

[24]图 2 为本说明书中一种创建虚拟文件的过程示意图。

[25]图 3 为本说明书中一种一种表空间扩容的流程示意图。

[26]图 4 为本说明书中一种创建扩容虚拟文件的过程示意图。

[27]图 5 为本说明书中一种对分布式数据库中的节点进行删除的流程示意图。

[28]图 6 为本说明书提供的一种创建表空间的装置的示意图。

[29]图 7 为本说明书提供的对应于图 1 的电子设备示意图。

具体实施方式

[30]为使本说明书的目的、技术方案和优点更加清楚，下面将结合本说明书具体实施例及相应的附图对本说明书技术方案进行清楚、完整地描述。显然，所描述的实施例仅是本说明书一部分实施例，而不是全部的实施例。基于本说明书中的实施例，本领域普通技术人员在没有做出创造性劳动前提下所获得的所有其他实施例，都属于本说明书保护

的范围。

[31]以下结合附图，详细说明本说明书各实施例提供的技术方案。

[32]图 1 为本说明书中一种创建表空间的方法的流程示意图，包括步骤 S100 至步骤 S108。

[33] S100: 接收表空间的创建指令，其中，所述创建指令中携带所需创建的表空间的容量，以及构成所述容量的存储空间所在的服务器的标识。

[34]share-nothing 架构的分布式数据库的数据存储在作为节点的多个不同服务器中，并且各个服务器的存储空间是不共享的，所以无法直接创建表空间。

[35]基于此，在本说明书中，创建表空间的设备接收表空间的创建指令，其中，创建指令中携带所需创建的表空间的容量，以及构成该容量的存储空间所在的服务器的标识。上述创建表空间的设备可以是分布式数据库中的作为普通节点的服务器，也可以是分布式数据库中的作为管理节点的服务器，为了方便说明，下面以作为普通节点的服务器为执行主体进行说明。创建指令可以是客户端向服务器发送的，以使服务器根据创建指令创建表空间，创建指令中携带的构成该容量的存储空间所在的服务器的标识实际上就是在 share-nothing 架构下创建的表空间所希望覆盖到的服务器的标识，后续可以根据该标识在分布式数据库中确定该标识对应的服务器作为创建表空间的指定服务器。

[36]S102: 在所述分布式数据库中，确定各标识分别对应的服务器，作为指定服务器。

[37]在分布式数据库中，将创建指令中携带的各标识分别对应的服务器，作为指定服务器，其中，指定服务器可以是创建表空间的服务器自身，也可以是在分布式数据库中的其他作为普通节点的服务器。

[38]例如，服务器 1 接收到的创建指令为：CREATE VIRTUAL FILE '/data1.ob' ON '/data/1/data1@server1; /data/1/data1@server2; /data/1/data1@server3;' SIZE 3G; CREATE TABLESPACE tablespacel DATAFILE '/data1.ob'; 则创建指令中携带的服务器的标识是 server1、server2 和 server3，服务器 1 为创建表空间的设备，则服务器 1 在分布式数据库中，确定各标识分别对应的服务器作为指定服务器，server1 对应的服务器是服务器 1，server 2 对应的服务器是服务器 2，server 3 对应的服务器是服务器 3，也就是将服务器 1、服务器 2 和服务器 3 作为指定服务器。

[39]S104: 根据所述容量，在各指定服务器的剩余存储空间中，确定出总量为所述容量的存储空间。

[40]服务器根据该容量，在各指定服务器的剩余存储空间中，确定出总量为该容量的存储空间。具体的，当该容量不大于各指定服务器的剩余存储空间的总量时，在各指定服务器的剩余存储空间中确定出总量为该容量的存储空间。

[41]继续沿用上例，服务器1接收到的创建指令中携带的所需创建的表空间的容量是3G，假设服务器1、服务器2和服务器3的剩余存储空间分别是1G、2G和4G，当该容量不大于各指定服务器的剩余存储空间的总量时，在各指定服务器的剩余存储空间中确定出总量为该容量的存储空间，也就是服务器1、服务器2和服务器3的剩余存储空间的总量为7G，不小于该容量3G，在各指定服务器的剩余存储空间中确定出总量为该容量的存储空间。从服务器1、服务器2和服务器3的剩余存储空间中确定出3G的存储空间，每个指定服务器的剩余存储空间占3G的比例可以是相同的也可以是不同的，可以将3G均分，服务器1、服务器2和服务器3只需从剩余存储空间中提供1G的存储空间，对于本例，服务器1的剩余存储空间只有1G，故服务器1最多可以提供1G的存储空间，若服务器1提供1G的存储空间，剩下的2G由服务器2和服务器3的剩余存储空间随机提供，只要能从服务器1、服务器2和服务器3的剩余存储空间中确定出3G的存储空间即可。

[42]S106：基于确定出的存储空间，创建所述容量的虚拟文件。

[43]基于确定出的存储空间，服务器创建容量为创建指令中携带的容量的虚拟文件。具体的，为了在share-nothing架构下创建的表空间所希望覆盖到的服务器（即，指定服务器）需要都可以被覆盖到，这就需要每个指定服务器均存在足够的剩余存储空间，服务器可先判断每个指定服务器的剩余存储空间是否均不小于预设的单位容量，若否，则说明存在至少一个指定服务器没有足够的存储空间，服务器无法在这种没有足够存储空间的指定服务器上创建虚拟文件中的任何一个单位，该指定服务器也就不能被后续创建的表空间覆盖到。若是，基于确定出的存储空间，根据该容量以及预设的单位容量，将该容量按照预设的单位容量进行划分，将该容量与预设的单位容量的商上取整得到的值作为指定数量，确定指定数量的大小均为预设的单位容量的各子文件，将各子文件轮循写入各指定服务器的剩余存储空间中，直至将指定数量的子文件全部写入为止，服务器创建由各子文件构成的虚拟文件。

[44]其中，在将各子文件轮循写入各指定服务器的剩余存储空间中时，服务器将各指定服务器进行排序，得到指定服务器序列，依次针对每个指定服务器，将该指定服务器作为待写入服务器，判断待写入服务器的剩余存储空间是否不小于上述预设的单位容量，

若是，将子文件写入待写入服务器的剩余存储空间中。否则，根据指定服务器序列，将该指定服务器的下一个指定服务器重新作为待写入服务器，直至写入子文件为止。

[45]由于在上述步骤 S104 中在各指定服务器的剩余存储空间中确定出总量为该容量的存储空间时，需要在该容量不大于各指定服务器的剩余存储空间的总量时，才能在各指定服务器的剩余存储空间中确定出总量为该容量的存储空间，但是创建表空间的设备可能仅仅可以获知各指定服务器的剩余存储空间的总量不小于该容量，但并不能获知每个指定服务器（除创建表空间的设备自身以外）具体的剩余存储空间，而在 share-nothing 架构下创建的表空间所希望覆盖到的服务器需要都可以被覆盖到，也就是创建指令携带的服务器的标识对应的各指定服务器都可以被覆盖到，这就需要各指定服务器都存在足够的剩余存储空间。创建表空间的设备不知道各指定服务器是否都存在足够的剩余存储空间，也不知道各指定服务器是否都可以被覆盖到。因此，可能会有指定服务器不存在足够的剩余存储空间，指定服务器没有被覆盖到的情况，若有这种情况，创建表空间的设备无法创建该容量的虚拟文件。

[46]故在步骤 S106 中基于确定出的存储空间，创建该容量的虚拟文件之前，为了确定每个指定服务器都有足够的剩余存储空间，服务器需要确定每个指定服务器的剩余存储空间均不小于预设的单位容量。因此，可以将该容量按照预设的单位容量进行划分，确定指定数量的大小均为预设的单位容量的各子文件，将各子文件轮循写入各指定服务器的剩余存储空间中，直至将指定数量的子文件全部写入为止。通过将各子文件轮循写入各指定服务器的剩余存储的方式可以使各指定服务器的剩余存储空间中都存在写入的至少一个子文件，再创建由各子文件构成的虚拟文件。在将一个子文件写入一个指定服务器时，若该指定服务器的剩余存储空间不足，则说明该指定服务器的剩余存储空间小于预设的单位容量，不能向该指定服务器的剩余存储空间中写入该子文件，该指定服务器的剩余存储空间已经写满，需要按照服务器序列，向该指定服务器的下一个指定服务器的剩余存储空间中写入该子文件，而针对一个指定服务器，若未成功向该指定服务器写入任何一个子文件，则说明该指令服务器的剩余存储空间小于预设的单位容量，该指定服务器不存在足够的剩余存储空间，该指定服务器没有被覆盖到。

[47]针对每个子文件，将该子文件所在的指定服务器作为目标服务器，确定该子文件在目标服务器中的物理地址，作为该子文件对应的物理地址，建立该子文件在虚拟文件中的逻辑地址与该子文件对应的物理地址的映射关系。

[48]继续沿用上例，结合图 2，图 2 为本说明书中一种创建虚拟文件的过程示意图，假

设服务器 1 预设的单位容量为 2MB，服务器 1、服务器 2 和服务器 3 的剩余存储空间分别是 1G、2G 和 4G，由于服务器 1 仅仅可以知道自身的剩余存储空间为 1G，不可以获知其他指定服务器的剩余存储空间，即服务器 2 的剩余存储空间为 2G 和服务器 3 的剩余存储空间为 4G，但每个指定服务器（即，服务器 1、服务器 2 和服务器 3）需要都可以被覆盖到，也就是每个指定服务器需要存在足够的剩余存储空间，故服务器 1 可以通过向每个指定服务器的剩余存储空间中写入预设的单位容量的子文件的方式，试探每个指定服务器的剩余存储空间是否均不小于预设的单位容量。服务器 1 可以将 3G 的容量按照单位容量 2MB 划分，将 3G 与 2MB 的商上取整得到的值 1536 作为指定数量，确定 1536 个大小均为 2MB 的各子文件。服务器 1 将各指定服务器（即，服务器 1、服务器 2 和服务器 3）进行排序，假设排好序得到的指定服务器序列为服务器 1，服务器 2，服务器 3。根据指定服务器序列，将子文件 1~子文件 1536 轮循写入服务器 1、服务器 2 和服务器 3 的剩余存储空间中，直至将 1536 个子文件（即，子文件 1~子文件 1536）全部写入为止。比如将服务器 1 作为待写入服务器，将子文件 1 写入到服务器 1 的剩余存储空间时，服务器 1 的剩余存储空间为 1G，不小于预设的单位容量 2MB，将子文件 1 写入到待写入服务器（即，服务器 1）的剩余存储空间中。但在将一个子文件写入一个指定服务器时，比如，将子文件 100 写入到服务器 1 的剩余存储空间时，若服务器 1 的剩余存储空间不足，则说明服务器 1 的剩余存储空间小于预设的单位容量，不能向服务器 1 的剩余存储空间中写入子文件 100，服务器 1 的剩余存储空间已经写满，需要按照服务器序列，向服务器 1 的下一个指定服务器（即，服务器 2）的剩余存储空间中写入子文件 100。在将 1536 个子文件全部写入后，服务器 1 创建由子文件 1~子文件 1536 构成的虚拟文件，即虚拟文件 1，也就是创建指令中的 data1.ob。

[49]再假设服务器 1、服务器 2 和服务器 3 的剩余存储空间分别是 1G、1MB 和 4G，服务器 1 根据指定服务器序列，将子文件 1~子文件 1536 轮循写入服务器 1、服务器 2 和服务器 3 的剩余存储空间中，直至将 1536 个子文件（即，子文件 1~子文件 1536）全部写入为止。假设将子文件 2 写入到服务器 2 的剩余存储空间中，服务器 2 的剩余存储空间为 1MB，小于 2MB，无法将子文件 2 写入到服务器 2 的剩余存储空间中，可以按照指定服务器序列，将子文件 2 写入到服务器 2 的下一个服务器中，即服务器 3。

[50]但针对一个指定服务器，比如服务器 2，在向服务器 2 写入子文件失败时，可判断之前是否已经成功向服务器 2 写入过子文件，若未成功向服务器 2 写入任何一个子文件，则说明服务器 2 的剩余存储空间小于预设的单位容量 2MB，服务器 2 不存在足够的剩余存储空间，服务器 2 没有被覆盖到，故服务器 1 无法创建容量为 3G 的虚拟文件。

[51]针对每个子文件，将该子文件所在的指定服务器作为目标服务器，确定该子文件在目标服务器中的物理地址，作为该子文件对应的物理地址，建立该子文件在虚拟文件中的逻辑地址与该子文件对应的物理地址的映射关系，也就是假设子文件 1~子文件 1536 在虚拟文件中的逻辑地址分别是 1, 2, 3.....1535, 1536。针对子文件 1~子文件 1536，确定子文件 1 在虚拟文件中的逻辑地址为 1，将子文件 1 所在的服务器，即服务器 1，作为目标服务器，确定子文件 1 在服务器 1 中的物理地址，建立子文件 1 的逻辑地址 1 与子文件 1 在服务器 1 中的物理地址的映射关系（图 2 中只示出了子文件的逻辑地址，并未示出各子文件存储在服务器中的物理地址）。其他子文件 2~子文件 1536 均通过上述方式建立各子文件的逻辑地址与各子文件对应的物理地址的映射关系。

[52]S108：创建包含所述虚拟文件的表空间，所述表空间用于管理所述表空间自身包含的虚拟文件，所述虚拟文件用于写入数据。

[53]服务器创建包含上述虚拟文件的表空间，通过表空间管理表空间自身包含的虚拟文件，可以向虚拟文件中写入数据。继续沿用上例，服务器 1 创建包含虚拟文件 1 的表空间，即表空间 1，也就是创建指令中的 tablespace1。

[54]从上述方法中可以看出，本方法中，根据分布式数据库中的指定服务器的剩余存储空间和所需创建的表空间的容量，确定出总量为该容量的存储空间。由于在 share-nothing 架构下各指定服务器的存储空间是不共享的，只有指定服务器自身知道自己的剩余存储空间，而在 share-nothing 架构下创建的表空间所希望覆盖到的服务器需要都被覆盖到，需要创建指令中的各指定服务器都存在足够的剩余存储空间。故基于确定出的存储空间，确定各指定服务器的剩余存储空间均不小于预设的单位容量，将容量按照预设的单位容量划分成指定数量的各子文件，将各子文件轮循写入到各指定服务器的存储空间中，直至将指定数量的子文件全部写入为止，创建由各子文件构成的虚拟文件，各指定服务器的剩余存储空间中均存在子文件，通过由各子文件构成的虚拟文件将分散的各指定服务器的剩余存储空间统一起来，从而可以创建包含虚拟文件的表空间，以在 share-nothing 架构下实现跨服务器的表空间的功能，可以利用表空间管理表空间包含的虚拟文件，进而管理向虚拟文件写入的数据。

[55]在本说明书实施例中，在对分布式数据库中的表空间进行扩容时，需要基于分布式数据库中作为节点的服务器的剩余存储空间创建虚拟文件，将虚拟文件加入到需要扩容的表空间中，如图 3 所示。图 3 为本说明书中一种表空间扩容的流程示意图，包括步骤 S200 至步骤 S206。

[56]S200: 接收所述表空间的扩容指令, 其中, 所述扩容指令中携带所需扩容容量, 以及构成所述扩容容量的存储空间所在的服务器的标识。

[57]S202: 在分布式数据库中, 确定所述构成所述扩容容量的存储空间所在的服务器的标识分别对应的服务器, 作为扩容服务器。

[58]服务器接收表空间的扩容指令, 在分布式数据库中, 服务器确定构成扩容容量的存储空间所在的服务器的标识分别对应的服务器, 作为扩容服务器, 其中, 扩容服务器可以是分布式数据库中已经存在的作为普通节点的服务器, 也可以是新加入分布式数据库中作为节点的服务器。

[59]例如, 服务器 1 接收到的表空间的扩容指令为: CREATE VIRTUAL FILE '/data2.ob' ON'/data/1/data2@server3; /data/1/data2@server4;' SIZE 1G; ALTER TABLESPACE tablespacel ADD DATAFILE '/data2.ob';则扩容指令中携带的服务器的标识是 server3 和 server4, 服务器 1 在分布式数据库中, 确定构成扩容容量的存储空间所在的服务器的标识分别对应的服务器, 作为扩容服务器, server3 对应的服务器是服务器 3, server 4 对应的服务器是服务器 4, 也就是将服务器 3 和服务器 4 作为扩容服务器。其中, 服务器 3 是分布式数据库中已经存在的服务器, 服务器 4 是新加入分布式数据库的服务器。

[60]S204: 根据所述扩容容量, 在所述扩容服务器的剩余存储空间中, 创建存储空间为所述扩容容量的扩容虚拟文件。

[61]服务器根据扩容容量, 在扩容服务器的剩余存储空间中, 创建存储空间为扩容容量的扩容虚拟文件。扩容虚拟文件的具体创建过程与上述图 1 所示的方法中步骤 S104 和步骤 S106 基本一致, 在此就不再赘述。

[62]继续沿用上例, 结合上述图 1 所示的方法中步骤 S104 和步骤 S106 中的例子, 如图 4 所示。图 4 为本说明书中一种创建扩容虚拟文件的过程示意图, 图 4 中只示出了子文件的逻辑地址, 并未示出各子文件存储在服务器中的物理地址, 假设服务器 3 和服务器 4 的剩余存储空间均是 1G, 预设的单位容量为 2MB, 按照上述图 1 所述的方法中步骤 S104 和步骤 S106 中的例子, 服务器 1 根据扩容容量 1G, 在扩容服务器 (即, 服务器 3 和服务器 4) 的剩余存储空间中, 创建存储空间为 1G 的扩容虚拟文件, 也就是虚拟文件 2, 即扩容指令中的 data2.ob。

[63]S206: 将所述扩容虚拟文件添加到所述表空间中。

[64]服务器将扩容虚拟文件添加到表空间中, 继续沿用上例, 服务器 1 将虚拟文件 2 添

加到上述图 1 所示的方法创建的表空间 1 中，表空间 1 中有虚拟文件 1 和虚拟文件 2，表空间 1 的容量从原来的 3G（即，虚拟文件 1 的容量）变成 4G（即，虚拟文件 1 的 3G 容量和虚拟文件 2 的 1G 容量之和）。

[65]在本说明书实施例中，在对分布式数据库进行缩容时，也就是从分布式数据库的服务器标识列表中删除服务器的标识，需要先转移基于待删除服务器的存储空间创建的各虚拟文件，再将待删除服务器的标识从分布式数据库的服务器标识列表中删除，如图 5 所示。图 5 为本说明书中一种对分布式数据库中的节点进行删除的流程示意图，包括步骤 S300 至步骤 S308。

[66]S300：接收删除指令，其中，所述删除指令中携带所需删除的服务器的标识。

[67]S302：确定所述删除指令携带的标识对应的服务器，作为待删除服务器。

[68]S304：将基于所述待删除服务器的存储空间创建的各虚拟文件，作为待删除虚拟文件。

[69]服务器根据删除指令中携带的所需删除的服务器的标识确定对应的服务器，作为待删除服务器，将基于待删除服务器的存储空间创建的各虚拟文件，作为待删除虚拟文件。继续沿用上例，假设服务器 3 的存储空间中有虚拟文件 1 的子文件（即，上述图 1 所示的方法中由子文件 1~子文件 1536 构成的虚拟文件）和虚拟文件 2（即，上述图 3 所示的方法中由扩容子文件 1~扩容子文件 512 构成的虚拟文件）的子文件（即，上述图 3 所示的方法中的扩容子文件），删除指令中携带的标识是 3，标识 3 对应的服务器是服务器 3，将服务器 3 作为待删除服务器，将基于服务器 3 的存储空间创建的各虚拟文件，作为待删除虚拟文件，也就是将虚拟文件 1 和虚拟文件 2 作为待删除虚拟文件。

[70]S306：针对各待删除虚拟文件，将该待删除虚拟文件中的数据移动到不包含任一待删除虚拟文件的表空间中，将该待删除虚拟文件删除。

[71]服务器针对各待删除虚拟文件，将该待删除虚拟文件中的数据移动到不包含任一待删除虚拟文件的表空间中，将该待删除虚拟文件删除。其中，不包含任一待删除虚拟文件的表空间可以是存在分布式数据库中不包含任一待删除虚拟文件的表空间，也可以是重新创建的新表空间。继续沿用上例，假设在分布式数据库中有包含虚拟文件 1 和虚拟文件 2 的表空间 1，不包含虚拟文件 1 和虚拟文件 2 的表空间 2，针对各待删除虚拟文件，将该待删除虚拟文件中的数据移动到不包含任一待删除虚拟文件的表空间中，将该待删除虚拟文件删除，也就是服务器 1 将虚拟文件 1 和虚拟文件 2 中的数据移动到不包

含任一待删除虚拟文件（即，虚拟文件 1 和虚拟文件 2）的表空间中，即表空间 2，将虚拟文件 1 和虚拟文件 2 删除。

[72]S308：将所述待删除服务器的标识从所述分布式数据库的服务器标识列表中删除，所述服务器标识列表包含构成所述分布式数据库的所有服务器的标识。

[73]服务器将待删除服务器的标识从分布式数据库的服务器标识列表中删除，服务器标识列表包含构成分布式数据库的所有服务器的标识。继续沿用上例，假设构成分布式数据库的服务器有服务器 1、服务器 2、服务器 3、服务器 4 和服务器 5，服务器标识列表为 1、2、3、4 和 5，服务器 1 将上例中待删除服务器（即，服务器 3）对应的标识 3 从服务器标识列表中删除，删除后的服务器标识列表为 1、2、4 和 5，构成分布式数据库的服务器变成服务器 1、服务器 2、服务器 4 和服务器 5。

[74]在本说明书实施例中，针对在分布式数据库中作为节点的各服务器，当该服务器宕机时，该服务器的存储空间中的各子文件不可用，其他服务器的存储空间中的各子文件不受影响，当该服务器重新恢复后，可以继续使用该服务器的存储空间中的各子文件。

[75]以上为本说明书的一个或多个实施例提供的创建表空间的方法，基于同样的思路，本说明书还提供了相应的创建表空间的装置，如图 6 所示。

[76]图 6 为本说明书提供一种创建表空间的装置示意图，包括：接收模块 400，用于接收表空间的创建指令，其中，所述创建指令中携带所需创建的表空间的容量，以及构成所述容量的存储空间所在的服务器的标识；第一确定模块 402，用于在所述分布式数据库中，确定各标识分别对应的服务器，作为指定服务器；第二确定模块 404，用于根据所述容量，在各指定服务器的剩余存储空间中，确定出总量为所述容量的存储空间；第一创建模块 406，用于基于确定出的存储空间，创建所述容量的虚拟文件；第二创建模块 408，用于创建包含所述虚拟文件的表空间，所述表空间用于管理所述表空间自身包含的虚拟文件，所述虚拟文件用于写入数据。

[77]可选地，所述第一创建模块 406，在创建所述容量的虚拟文件之前，还用于，确定每个指定服务器的剩余存储空间均不小于预设的单位容量。

[78]可选地，所述第一创建模块 406 具体用于，根据所述容量以及预设的单位容量，确定指定数量的大小均为预设的单位容量的各子文件；将各子文件轮循写入所述各指定服务器的剩余存储空间中，直至将所述指定数量的子文件全部写入为止；创建由各子文件构成的虚拟文件。

[79]可选地,所述第一创建模块 406 具体用于,将所述各指定服务器进行排序,得到指定服务器序列;依次针对每个指定服务器,将该指定服务器作为待写入服务器;判断所述待写入服务器的剩余存储空间是否不小于所述单位容量;若是,将子文件写入所述待写入服务器的剩余存储空间中;否则,根据所述指定服务器序列,将该指定服务器的下一个指定服务器重新作为待写入服务器,直至写入子文件为止。

[80]可选地,所述装置还包括:映射模块 410,用于针对每个子文件,确定该子文件在所述虚拟文件中的逻辑地址,并将该子文件所在的指定服务器作为目标服务器,确定该子文件在所述目标服务器中的物理地址,作为该子文件对应的物理地址;建立该子文件的所述逻辑地址与该子文件对应的所述物理地址的映射关系。

[81]可选地,所述装置还包括:删除模块 412,用于接收删除指令,其中,所述删除指令中携带所需删除的服务器的标识;确定所述删除指令携带的标识对应的服务器,作为待删除服务器;将基于所述待删除服务器的存储空间创建的各虚拟文件,作为待删除虚拟文件;针对各待删除虚拟文件,将该待删除虚拟文件中的数据移动到不包含任一待删除虚拟文件的表空间中,将该待删除虚拟文件删除;将所述待删除服务器的标识从所述分布式数据库的服务器标识列表中删除,所述服务器标识列表包含构成所述分布式数据库的所有服务器的标识。

[82]本说明书还提供了一种计算机可读存储介质,该存储介质存储有计算机程序,计算机程序可用于执行上述图 1 所示的创建表空间的方法。

[83]本说明书还提供了图 7 所示的电子设备的示意结构图。如图 7 所述,在硬件层面,该电子设备包括处理器、内部总线、网络接口、内存以及非易失性存储器,当然还可能包括其他业务所需要的硬件。处理器从非易失性存储器中读取对应的计算机程序到内存中然后运行,以实现上述图 1 所示的创建表空间的方法。当然,除了软件实现方式之外,本说明书并不排除其他实现方式,比如逻辑器件抑或软硬件结合的方式等等,也就是说以下处理流程的执行主体并不限于各个逻辑单元,也可以是硬件或逻辑器件。

[84]在 20 世纪 90 年代,对于一个技术的改进可以很明显地区分是硬件上的改进(例如,对二极管、晶体管、开关等电路结构的改进)还是软件上的改进(对于方法流程的改进)。然而,随着技术的发展,当今的很多方法流程的改进已经可以视为硬件电路结构的直接改进。设计人员几乎都通过将改进的方法流程编程到硬件电路中来得到相应的硬件电路结构。因此,不能说一个方法流程的改进就不能用硬件实体模块来实现。例如,可编程逻辑器件(Programmable Logic Device, PLD)(例如现场可编程门阵列(Field

Programmable Gate Array, FPGA))就是这样一种集成电路,其逻辑功能由用户对器件编程来确定。由设计人员自行编程来把一个数字系统“集成”在一片PLD上,而不需要请芯片制造厂商来设计和制作专用的集成电路芯片。而且,如今,取代手工地制作集成电路芯片,这种编程也多半改用“逻辑编译器(logic compiler)”软件来实现,它与程序开发撰写时所用的软件编译器相类似,而要编译之前的原始代码也得用特定的编程语言来撰写,此称之为硬件描述语言(Hardware Description Language, HDL),而HDL也并非仅有一种,而是有许多种,如ABEL(Advanced Boolean Expression Language)、AHDL(Altera Hardware Description Language)、Confluence、CUPL(Cornell University Programming Language)、HDCal、JHDL(Java Hardware Description Language)、Lava、Lola、MyHDL、PALASM、RHDL(Ruby Hardware Description Language)等,目前最普遍使用的是VHDL(Very-High-Speed Integrated Circuit Hardware Description Language)与Verilog。本领域技术人员也应该清楚,只需要将方法流程用上述几种硬件描述语言稍作逻辑编程并编程到集成电路中,就可以很容易得到实现该逻辑方法流程的硬件电路。

[85]控制器可以按任何适当的方式实现,例如,控制器可以采取例如微处理器或处理器以及存储可由该(微)处理器执行的计算机可读程序代码(例如软件或固件)的计算机可读介质、逻辑门、开关、专用集成电路(Application Specific Integrated Circuit, ASIC)、可编程逻辑控制器和嵌入微控制器的形式,控制器的例子包括但不限于以下微控制器:ARC 625D、Atmel AT91SAM、Microchip PIC18F26K20 以及Silicone Labs C8051F320,存储器控制器还可以被实现为存储器的控制逻辑的一部分。本领域技术人员也知道,除了以纯计算机可读程序代码方式实现控制器以外,完全可以通过将方法步骤进行逻辑编程来使得控制器以逻辑门、开关、专用集成电路、可编程逻辑控制器和嵌入微控制器等的形式来实现相同功能。因此这种控制器可以被认为是一种硬件部件,而对其内包括的用于实现各种功能的装置也可以视为硬件部件内的结构。或者甚至,可以将用于实现各种功能的装置视为既可以是实现方法的软件模块又可以是硬件部件内的结构。

[86]上述实施例阐明的系统、装置、模块或单元,具体可以由计算机芯片或实体实现,或者由具有某种功能的产品来实现。一种典型的实现设备为计算机。具体的,计算机例如可以为个人计算机、膝上型计算机、蜂窝电话、相机电话、智能电话、个人数字助理、媒体播放器、导航设备、电子邮件设备、游戏控制台、平板计算机、可穿戴设备或者这些设备中的任何设备的组合。

[87]为了描述的方便,描述以上装置时以功能分为各种单元分别描述。当然,在实施本

说明书时可以把各单元的功能在同一个或多个软件和/或硬件中实现。

[88]本领域内的技术人员应明白，本发明的实施例可提供为方法、系统、或计算机程序产品。因此，本发明可采用完全硬件实施例、完全软件实施例、或结合软件和硬件方面的实施例的形式。而且，本发明可采用在一个或多个其中包含有计算机可用程序代码的计算机可用存储介质（包括但不限于磁盘存储器、CD-ROM、光学存储器等）上实施的计算机程序产品的形式。

[89]本发明是参照根据本发明实施例的方法、设备（系统）、和计算机程序产品的流程图和/或方框图来描述的。应理解可由计算机程序指令实现流程图和/或方框图中的每一流程和/或方框、以及流程图和/或方框图中的流程和/或方框的结合。可提供这些计算机程序指令到通用计算机、专用计算机、嵌入式处理机或其他可编程数据处理设备的处理器以产生一个机器，使得通过计算机或其他可编程数据处理设备的处理器执行的指令产生用于实现在流程图一个流程或多个流程和/或方框图一个方框或多个方框中指定的功能的装置。

[90]这些计算机程序指令也可存储在能引导计算机或其他可编程数据处理设备以特定方式工作的计算机可读存储器中，使得存储在该计算机可读存储器中的指令产生包括指令装置的制造品，该指令装置实现在流程图一个流程或多个流程和/或方框图一个方框或多个方框中指定的功能。

[91]这些计算机程序指令也可装载到计算机或其他可编程数据处理设备上，使得在计算机或其他可编程设备上执行一系列操作步骤以产生计算机实现的处理，从而在计算机或其他可编程设备上执行的指令提供用于实现在流程图一个流程或多个流程和/或方框图一个方框或多个方框中指定的功能的步骤。

[92]在一个典型的配置中，计算设备包括一个或多个处理器（CPU）、输入/输出接口、网络接口和内存。

[93]内存可能包括计算机可读介质中的非永久性存储器，随机存取存储器（RAM）和/或非易失性内存等形式，如只读存储器（ROM）或闪存（flash RAM）。内存是计算机可读介质的示例。

[94]计算机可读介质包括永久性和非永久性、可移动和非可移动媒体可以由任何方法或技术来实现信息存储。信息可以是计算机可读指令、数据结构、程序的模块或其他数据。计算机的存储介质的例子包括，但不限于相变内存（PRAM）、静态随机存取存储器

(SRAM)、动态随机存取存储器(DRAM)、其他类型的随机存取存储器(RAM)、只读存储器(ROM)、电可擦除可编程只读存储器(EEPROM)、快闪记忆体或其他内存技术、只读光盘只读存储器(CD-ROM)、数字多功能光盘(DVD)或其他光学存储、磁盒式磁带、磁带磁磁盘存储或其他磁性存储设备或任何其他非传输介质,可用于存储可以被计算设备访问的信息。按照本文中的界定,计算机可读介质不包括暂存电脑可读媒体(transitory media),如调制的数据信号和载波。

[95]还需要说明的是,术语“包括”、“包含”或者其任何其他变体意在涵盖非排他性的包含,从而使得包括一系列要素的过程、方法、商品或者设备不仅包括那些要素,而且还包括没有明确列出的其他要素,或者是还包括为这种过程、方法、商品或者设备所固有的要素。在没有更多限制的情况下,由语句“包括一个……”限定的要素,并不排除在包括所述要素的过程、方法、商品或者设备中还存在另外的相同要素。

[96]本领域技术人员应明白,本说明书的实施例可提供为方法、系统或计算机程序产品。因此,本说明书可采用完全硬件实施例、完全软件实施例或结合软件和硬件方面的实施例的形式。而且,本说明书可采用在一个或多个其中包含有计算机可用程序代码的计算机可用存储介质(包括但不限于磁盘存储器、CD-ROM、光学存储器等)上实施的计算机程序产品的形式。

[97]本说明书可以在由计算机执行的计算机可执行指令的一般上下文中描述,例如程序模块。一般地,程序模块包括执行特定任务或实现特定抽象数据类型的例程、程序、对象、组件、数据结构等等。也可以在分布式计算环境中实践本说明书,在这些分布式计算环境中,由通过通信网络而被连接的远程处理设备来执行任务。在分布式计算环境中,程序模块可以位于包括存储设备在内的本地和远程计算机存储介质中。

[98]本说明书中的各个实施例均采用递进的方式描述,各个实施例之间相同相似的部分互相参见即可,每个实施例重点说明的都是与其他实施例的不同之处。尤其,对于系统实施例而言,由于其基本相似于方法实施例,所以描述的比较简单,相关之处参见方法实施例的部分说明即可。

[99]以上所述为本说明书的实施例,并不用于限制本说明书。对于本领域技术人员来说,本说明书可以有各种更改和变化。凡在本说明书的精神和原理之内所作的任何修改、等同替换、改进等,均应包含在本说明书的权利要求范围之内。

权利要求书

1、一种创建表空间的方法，所述方法应用于分布式数据库，所述方法包括：

接收表空间的创建指令，其中，所述创建指令中携带所需创建的表空间的容量，以及构成所述容量的存储空间所在的服务器的标识；

在所述分布式数据库中，确定各标识分别对应的服务器，作为指定服务器；

根据所述容量，在各指定服务器的剩余存储空间中，确定出总量为所述容量的存储空间；

基于确定出的存储空间，创建所述容量的虚拟文件；

创建包含所述虚拟文件的表空间，所述表空间用于管理所述表空间自身包含的虚拟文件，所述虚拟文件用于写入数据。

2、如权利要求1所述的方法，创建所述容量的虚拟文件之前，所述方法包括：

确定每个指定服务器的剩余存储空间均不小于预设的单位容量。

3、如权利要求1所述的方法，基于确定出的存储空间，创建所述容量的虚拟文件，包括：

根据所述容量以及预设的单位容量，确定指定数量的大小均为预设的单位容量的各子文件；

将各子文件轮循写入所述各指定服务器的剩余存储空间中，直至将所述指定数量的子文件全部写入为止；

创建由各子文件构成的虚拟文件。

4、如权利要求3所述的方法，将各子文件轮循写入所述各指定服务器的剩余存储空间中，具体包括：

将所述各指定服务器进行排序，得到指定服务器序列；

依次针对每个指定服务器，将该指定服务器作为待写入服务器；

判断所述待写入服务器的剩余存储空间是否不小于所述单位容量；

若是，将子文件写入所述待写入服务器的剩余存储空间中；

否则，根据所述指定服务器序列，将该指定服务器的下一个指定服务器重新作为待写入服务器，直至写入子文件为止。

5、如权利要求3所述的方法，所述方法还包括：

针对每个子文件，确定该子文件在所述虚拟文件中的逻辑地址，并将该子文件所在的指定服务器作为目标服务器，确定该子文件在所述目标服务器中的物理地址，作为该子文件对应的物理地址；

建立该子文件的所述逻辑地址与该子文件对应的所述物理地址的映射关系。

6、如权利要求 1 所述的方法，所述方法还包括：

接收删除指令，其中，所述删除指令中携带所需删除的服务器的标识；

确定所述删除指令携带的标识对应的服务器，作为待删除服务器；

将基于所述待删除服务器的存储空间创建的各虚拟文件，作为待删除虚拟文件；

针对各待删除虚拟文件，将该待删除虚拟文件中的数据移动到不包含任一待删除虚拟文件的表空间中，将该待删除虚拟文件删除；

将所述待删除服务器的标识从所述分布式数据库的服务器标识列表中删除，所述服务器标识列表包含构成所述分布式数据库的所有服务器的标识。

7、一种创建表空间的装置，所述装置应用于分布式数据库，所述装置包括：

接收模块，用于接收表空间的创建指令，其中，所述创建指令中携带所需创建的表空间的容量，以及构成所述容量的存储空间所在的服务器的标识；

第一确定模块，用于在所述分布式数据库中，确定各标识分别对应的服务器，作为指定服务器；

第二确定模块，用于根据所述容量，在各指定服务器的剩余存储空间中，确定出总量为所述容量的存储空间；

第一创建模块，用于基于确定出的存储空间，创建所述容量的虚拟文件；

第二创建模块，用于创建包含所述虚拟文件的表空间，所述表空间用于管理所述表空间自身包含的虚拟文件，所述虚拟文件用于写入数据。

8、如权利要求 7 所述的装置，所述第一创建模块，在创建所述容量的虚拟文件之前，还用于，确定每个指定服务器的剩余存储空间均不小于预设的单位容量。

9、如权利要求 7 所述的装置，所述第一创建模块具体用于，根据所述容量以及预设的单位容量，确定指定数量的大小均为预设的单位容量的各子文件；将各子文件轮循写入所述各指定服务器的剩余存储空间中，直至将所述指定数量的子文件全部写入为止；创建由各子文件构成的虚拟文件。

10、如权利要求 9 所述的装置，所述第一创建模块用于，将所述各指定服务器进行排序，得到指定服务器序列；依次针对每个指定服务器，将该指定服务器作为待写入服务器；判断所述待写入服务器的剩余存储空间是否不小于所述单位容量；若是，将子文件写入所述待写入服务器的剩余存储空间中；否则，根据所述指定服务器序列，将该指定服务器的下一个指定服务器重新作为待写入服务器，直至写入子文件为止。

11、如权利要求 9 所述的装置，所述装置还包括：

映射模块，用于针对每个子文件，确定该子文件在所述虚拟文件中的逻辑地址，并将该子文件所在的指定服务器作为目标服务器，确定该子文件在所述目标服务器中的物理地址，作为该子文件对应的物理地址；建立该子文件的所述逻辑地址与该子文件对应的所述物理地址的映射关系。

12、如权利要求 7 所述的装置，所述装置还包括：

删除模块，用于接收删除指令，其中，所述删除指令中携带所需删除的服务器的标识；确定所述删除指令携带的标识对应的服务器，作为待删除服务器；将基于所述待删除服务器的存储空间创建的各虚拟文件，作为待删除虚拟文件；针对各待删除虚拟文件，将该待删除虚拟文件中的数据移动到不包含任一待删除虚拟文件的表空间中，将该待删除虚拟文件删除；将所述待删除服务器的标识从所述分布式数据库的服务器标识列表中删除，所述服务器标识列表包含构成所述分布式数据库的所有服务器的标识。

13、一种计算机可读存储介质，所述存储介质存储有计算机程序，所述计算机程序被处理器执行时实现上述权利要求 1~6 任一项所述的方法。

14、一种电子设备，包括存储器、处理器及存储在存储器上并可在处理器上运行的计算机程序，所述处理器执行所述程序时实现上述权利要求 1~6 任一项所述的方法。