

(12) 发明专利

(10) 授权公告号 CN 101751390 B

(45) 授权公告日 2012. 07. 04

(21) 申请号 200810178007. 7

审查员 欧阳琦

(22) 申请日 2008. 12. 08

(73) 专利权人 财团法人工业技术研究院
地址 中国台湾新竹县

(72) 发明人 林旭政

(74) 专利代理机构 北京市柳沈律师事务所
11105

代理人 蒲迈文

(51) Int. Cl.

G06F 17/30 (2006. 01)

G06F 3/06 (2006. 01)

(56) 对比文件

US 2007/0118676 A1, 2007. 05. 24, 全文.

CN 1267379 A, 2000. 09. 20, 全文.

CN 1506844 A, 2004. 06. 23, 全文.

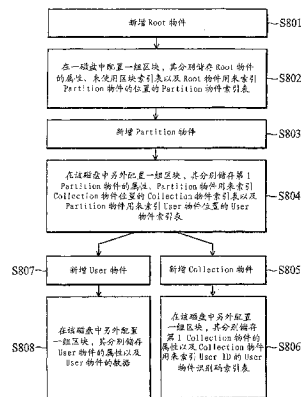
权利要求书 5 页 说明书 11 页 附图 17 页

(54) 发明名称

物件导向储存装置的磁盘配置方法

(57) 摘要

一种物件导向储存装置的磁盘配置方法。初始化一磁盘, 并且新增一个根 (Root) 物件, 同时在该磁盘中配置一组区块以分别储存该 Root 物件的属性、一未使用区块索引表以及一分割区 (Partition) 物件索引表。当新增一 Partition 物件时, 在该磁盘中配置一组区块以分别储存该 Partition 物件的属性、一收集 (Collection) 物件索引表以及一使用者 (User) 物件索引表, 其中该 Partition 物件根据该 Collection 物件索引表来索引 Collection 物件的位置以及根据该 User 物件索引表来索引一 User 物件的位置。当新增一 Collection 物件或一 User 物件时, 在该磁盘中配置二组区块以分别储存对应的数据。



1. 一种物件导向储存装置的磁盘配置方法,其适用于一磁盘,包括下列步骤:
初始化该磁盘,且新增一根 (Root) 物件;
当新增该根物件时,在该磁盘中配置一或多组区块,其分别储存该根物件的属性、一未使用区块索引表以及一分割区 (Partition) 物件索引表,其中该根物件根据该分割区物件索引表来索引一分割区物件的位置;
新增一分割区物件;
当新增该分割区物件时,在该磁盘中配置一或多组区块,其分别储存该分割区物件的属性、一收集 (Collection) 物件索引表以及一使用者 (User) 物件索引表,其中该分割区物件根据该收集物件索引表来索引收集物件的位置以及根据该使用者物件索引表来索引一使用者物件的位置;
新增一使用者物件;以及
当新增该使用者物件时,在该磁盘中配置一组或多组区块,其分别储存该使用者物件的属性以及该使用者物件的数据。
2. 如权利要求 1 所述的物件导向储存装置的磁盘配置方法,其还包括新增一收集物件,并且在该磁盘中配置一组或多组区块,其分别储存该收集物件的属性以及一使用者物件识别码索引表,其中该收集物件根据该使用者物件识别码索引表来索引一使用者识别码。
3. 如权利要求 1 所述的物件导向储存装置的磁盘配置方法,其中,根据该根物件的该分割区物件索引表取得上述分割区物件。
4. 如权利要求 1 所述的物件导向储存装置的磁盘配置方法,其中,根据该分割区物件的该收集物件索引表取得上述收集物件。
5. 如权利要求 1 所述的物件导向储存装置的磁盘配置方法,其中,根据该分割区物件的该使用者物件索引表取得上述使用者物件。
6. 如权利要求 1 所述的物件导向储存装置的磁盘配置方法,其中,该分割区物件索引表、该收集物件索引表以及该使用者物件索引表为一或多组区块所组成并且用以分别存放分割区物件识别码、收集物件识别码与使用者物件识别码,以及分别存放上述分割区物件、收集物件、与使用者物件的起始位置。
7. 如权利要求 1 所述的物件导向储存装置的磁盘配置方法,其中,该未使用区块索引表内所存放的每一笔记录代表一个连续的未使用的空间,而每一笔记录的格式表示为位置加上连续区块长度。
8. 如权利要求 2 所述的物件导向储存装置的磁盘配置方法,其中,该收集物件的该使用者物件识别码索引表可直接索引该使用者物件的识别码。
9. 如权利要求 8 所述的物件导向储存装置的磁盘配置方法,其中,当该使用者物件识别码在该使用者物件写入该磁盘且该使用者物件属于该收集物件时,则将该使用者物件识别码加入该收集物件的该使用者物件识别码索引表。
10. 如权利要求 1 所述的物件导向储存装置的磁盘配置方法,其中,该根物件的属性可包括记录该磁盘中的所有区块使用的信息、该属性使用的区块数量、与各物件的索引表相关的记录、一定位区块的长度、一物件识别码的长度以及相关于该物件识别码的计数器。
11. 如权利要求 1 所述的物件导向储存装置的磁盘配置方法,其中,该分割区物件的属

性可包括记录与该收集物件索引表以及该使用者物件索引表相关的记录,包括起始位置及每次配置表格的大小。

12. 如权利要求 1 所述的物件导向储存装置的磁盘配置方法,其中,该收集物件的属性可包括记录该使用者物件索引表的大小及位置。

13. 如权利要求 1 所述的物件导向储存装置的磁盘配置方法,其中,该使用者物件的数据可当成属性内数据 (In-property Data) 来储存。

14. 如权利要求 13 所述的物件导向储存装置的磁盘配置方法,其中,若该数据不是以属性内数据的形式存在,则该使用者物件记录该数据是存在于一额外的区块串列中,其中该区块串列由多个连续记录组成,每一个记录存放数据区块的位置及其长度。

15. 如权利要求 1 所述的物件导向储存装置的磁盘配置方法,其中,每一物件索引表使用预设数量的区块,该预设数量记载在根物件的属性中。

16. 如权利要求 1 所述的物件导向储存装置的磁盘配置方法,其中,当区块定位使用一定量字节,则每一组区块的第一区块的前面保留两个定量的字节,其中第一个定量字节为前一组区块的起始位置 (SP),用以记录前一组区块的位置,而第二个定量字节为下一组区块的起始位置 (SN),用以记录下一组区块的位置。

17. 如权利要求 16 所述的物件导向储存装置的磁盘配置方法,其中,当只有一组区块时,则该组区块的前一组区块的起始位置及下一组区块的起始位置都表示为 NULL,而当有多组区块,则第一组区块的前一组区块的起始位置表示为 NULL,而最后一组区块的下一组区块的起始位置表示为 NULL。

18. 如权利要求 16 所述的物件导向储存装置的磁盘配置方法,其中,每一该物件索引表内部所存放的每一笔记录代表一个物件索引,而该物件索引的格式为物件识别码加上位置,则一组区块的物件索引表可以记载的索引数量表示为:

$$\text{一组区块可记录索引的数量} = \frac{\text{一组区块大小} - SP - SN}{\text{索引长度}},$$

其中,SP 表示前一组区块的起始位置,SN 表示下一组区块的起始位置。

19. 如权利要求 1 所述的物件导向储存装置的磁盘配置方法,其中,可对上述每一物件执行新增、移除与改变等操作。

20. 如权利要求 19 所述的物件导向储存装置的磁盘配置方法,其中,新增任一分割区物件,其还包括下列步骤:

在该磁盘中的配置一组区块,其分别储存该分割区物件的属性、该分割区物件用来索引一收集物件位置的一收集物件索引表以及该分割区物件用来索引一使用者物件位置的一使用者物件索引表;以及

修改该根物件属性以及该根物件的该未使用区块索引表。

21. 如权利要求 20 所述的物件导向储存装置的磁盘配置方法,删除任一分割区物件,其还包括下列步骤:

判断该分割区物件是否含有至少一收集物件或一使用者物件;

若该分割区物件含有至少一收集物件或一使用者物件,则删除所有收集物件及使用者物件;以及

若该分割区物件未含有任何收集物件或使用者物件,则修改该根物件属性以及该根物

件的该未使用区块索引表。

22. 如权利要求 21 所述的物件导向储存装置的磁盘配置方法,其中,新增任一收集物件,其还包括下列步骤:

在该磁盘中的配置一组区块,其分别储存该收集物件的属性以及该收集物件用来索引一使用者物件识别码的一使用者物件识别码索引表;以及

修改该分割区物件的属性、该分割区物件的收集物件索引表、该根物件属性以及该根物件的该未使用区块索引表。

23. 如权利要求 22 所述的物件导向储存装置的磁盘配置方法,其中,删除任一收集物件,其还包括下列步骤:

删除该收集物件以及该使用者识别码索引表;以及

修改该分割区物件属性、该分割区物件的该收集物件索引表、修改根物件属性以及该根物件的该未使用区块索引表。

24. 如权利要求 20 所述的物件导向储存装置的磁盘配置方法,其中,新增任一使用者物件,其还包括下列步骤:

在该磁盘中的配置一组区块,其分别储存该使用者物件的属性与该使用者物件储存数据的空间;以及

修改该分割区物件属性、该分割区物件的该使用者物件索引表、该根物件属性以及该根物件的该未使用区块索引表。

25. 如权利要求 24 所述的物件导向储存装置的磁盘配置方法,其中,删除任一使用者物件,其还包括下列步骤:

判断该使用者物件是否属于该收集物件;

若该使用者物件属于该收集物件,则删除所有收集物件中含有使用者物件识别码的索引;

若该使用者物件不属于该收集物件,则修改该分割区物件属性与该分割区物件的该使用者物件索引表、该根物件属性以及该根物件的该未使用区块索引表。

26. 一种物件导向储存装置的磁盘配置装置,其适用于一磁盘,该装置包括:

初始化该磁盘,并且新增一个根 (Root) 物件的模块;

当新增该根物件时,在该磁盘中配置第一组区块,其分别储存该根物件的属性、一未使用区块索引表以及一分割区 (Partition) 物件索引表的模块,其中该根物件根据该分割区物件索引表来索引一分割区物件的位置;

新增一分割区物件;

当新增该分割区物件时,在该磁盘中配置第二组区块,其分别储存该分割区物件的属性、一收集 (Collection) 物件索引表以及一使用者 (User) 物件索引表的模块,其中该分割区物件根据该收集物件索引表来索引收集物件的位置以及根据该使用者物件索引表来索引一使用者物件的位置;

新增一使用者物件的模块;以及

当新增该使用者物件时,在该磁盘中配置一组或多组区块,其分别储存该使用者物件的属性以及该使用者物件的数据的模块。

27. 如权利要求 26 所述的物件导向储存装置的磁盘配置装置,其还包括新增一收集物

件,并且在该磁盘中配置一组或多组区块,其分别储存该收集物件的属性以及一使用者物件识别码索引表的模块,其中该收集物件根据该使用者物件识别码索引表来索引一使用者识别码。

28. 如权利要求 26 所述的物件导向储存装置的磁盘配置装置,还包括根据该根物件的该分割区物件索引表取得上述分割区物件的模块。

29. 如权利要求 26 所述的物件导向储存装置的磁盘配置装置,还包括根据该分割区物件的该收集物件索引表取得上述收集物件的模块。

30. 如权利要求 26 所述的物件导向储存装置的磁盘配置装置,还包括根据该分割区物件的该使用者物件索引表取得上述使用者物件的模块。

31. 如权利要求 26 所述的物件导向储存装置的磁盘配置装置,其中,所述物件索引表为一或多组区块所组成并且用以分别存放分割区物件识别码、收集物件识别码与使用者物件识别码,以及分别存放上述分割区物件、收集物件、与使用者物件的起始位置。

32. 如权利要求 26 所述的物件导向储存装置的磁盘配置装置,其中,该未使用区块索引表内所存放的每一笔记录代表一个连续的未使用的空间,而每一笔记录的格式表示为位置加上连续区块长度。

33. 如权利要求 26 所述的物件导向储存装置的磁盘配置装置,其中,该收集物件的该使用者物件索引表可直接索引该使用者物件的识别码。

34. 如权利要求 33 所述的物件导向储存装置的磁盘配置装置,还包括当该使用者物件识别码在该使用者物件写入该磁盘且该使用者物件属于该收集物件时,则将该使用者物件识别码加入该收集物件的该使用者物件索引表的模块。

35. 如权利要求 26 所述的物件导向储存装置的磁盘配置装置,其中,该根物件的属性记录该磁盘中的所有区块使用的信息、该属性使用的区块数量、与各物件的索引表相关的记录、一定位区块的长度、一物件识别码的长度以及相关于该物件识别码的计数器。

36. 如权利要求 26 所述的物件导向储存装置的磁盘配置装置,其中,该分割区物件的属性记录与该收集物件索引表以及该使用者物件索引表相关的记录,包括起始位置及每次配置表格的大小。

37. 如权利要求 26 所述的物件导向储存装置的磁盘配置装置,其中,该收集物件的属性记录该使用者物件索引表的大小及位置。

38. 如权利要求 26 所述的物件导向储存装置的磁盘配置装置,其中,该使用者物件的数据是当成属性内数据来储存。

39. 如权利要求 38 所述的物件导向储存装置的磁盘配置装置,其中,若该数据不是以属性内数据(In-property Data)的形式存在,则该使用者物件记录该数据是存在于一额外的区块串列中,其中该区块串列由多个连续记录组成,每一个记录存放数据区块的位置及其长度。

40. 如权利要求 26 所述的物件导向储存装置的磁盘配置装置,其中,每一物件索引表使用预设数量的区块,该预设数量记载在根物件的属性中。

41. 如权利要求 26 所述的物件导向储存装置的磁盘配置装置,还包括当区块定位使用一定量字节,则每一组区块的第一区块的前面保留两个定量的字节的模块,其中第一个定量字节为前一组区块的起始位置(SP),用以记录前一组区块的位置,而第二个定量字节为

下一组区块的起始位置 (SN),用以记录下一组区块的位置。

42. 如权利要求 26 所述的物件导向储存装置的磁盘配置装置,其中,当只有一组区块时,则该组区块的前一组区块的起始位置及下一组区块的起始位置都表示为 NULL,而当有多组区块,则第一组区块的前一组区块的起始位置表示为 NULL,而最后一组区块的下一组区块的起始位置表示为 NULL。

43. 如权利要求 26 所述的物件导向储存装置的磁盘配置装置,其中,每一物件索引表内部所存放的每一笔记录代表一个物件索引,而该物件索引的格式为物件识别码加上位置,则一组区块的物件索引表可以记载的索引数量表示为:

$$\text{一组区块可记录索引的数量} = \frac{\text{一组区块大小} - SP - SN}{\text{索引长度}},$$

其中, SP 表示前一组区块的起始位置, SN 表示下一组区块的起始位置。

44. 如权利要求 26 所述的物件导向储存装置的磁盘配置装置,其中,可对上述每一物件执行新增、移除与改变等操作。

45. 一种物件导向储存装置的磁盘配置方法,包括下列步骤:

对一磁盘进行磁盘配置,使得该磁盘具有:

一根 (Root) 物件,其更提供根物件属性、一未使用区块索引表以及该根物件用来索引一分割区 (Partition) 物件位置的一分割区物件索引表;

至少一分割区物件,其更提供分割区物件属性、该分割区物件用来索引一收集 (Collection) 物件位置的一收集物件索引表以及该分割区物件用来索引一使用者 (User) 物件位置的一使用者物件索引表;以及

至少一使用者物件,其中该使用者物件更提供使用者物件属性以及使用者物件数据。

46. 如权利要求 45 所述的物件导向储存装置的磁盘配置方法,其还包括对所述磁盘进行磁盘配置,使得该磁盘还具有至少一收集物件,其中该收集物件更提供收集物件属性以及该收集物件用来索引一使用者识别码的一使用者物件识别码索引表。

物件导向储存装置的磁盘配置方法

技术领域

[0001] 本发明涉及一种物件导向储存装置 (Object-Based Storage Device, OSD) 的磁盘配置 (Disk Layout) 方法。

背景技术

[0002] 现行使用的网络储存架构的下有网络磁盘机 (Network-attached Storage, NAS) 及储存区域网络 (Storage Area Network, SAN), NAS 的问题在于元数据 (metadata) 都放在文档伺服器上, 当大量的使用者存取时, 会造成伺服器本身的效能低落, 因此在 SAN 所提供了比 NAS 更多的优点的同时, 却又产生了网络安全性上的问题。因此 OSD 被提出来做为网络储存上效能及安全性一个解决方案。

[0003] OSD 为一种在网络储存架构下能够提供自我管理、分享及安全性储存的一种储存方法, 其将部份较低阶的功能如空间管理由文档系统移至磁盘装置本身, 装置的存取则为标准物件界面。此外, 在此 OSD 架构下, 又定义了物件存取的内容及方法, 其特殊的架构及规格也必须要有一相对应的磁盘输出方法搭配。目前针对 OSD 架构的实作方式有两类, 第一类为利用传统文档系统模拟, 如 Linux 下的第二延伸文档系统 (Second Extended File System, Ext2); 第二类为关于 OSD 的磁盘配置新方法的研究, 例如, 物件导向文档系统 (Object-Based File System, OBFS) 及其衍生的物件导向储存装置文档系统 (Object-based Storage Device File System, OSDFS), 其架构 Ext2 大致相同, 不同的部份只在其设计为可变区块的大小及索引方式, 然而其设计有其限制所在。以下针对此二类方法作说明。

[0004] 该磁盘配置非针对 OSD 的物件特性而设计。

[0005] OSD 是以物件为基础并以唯一的识别码 (ID) 做为识别物件的根据。物件又分成根 (Root)、分区 (Partition)、收集 (Collection) 及使用者 (User) 四种类型, 而这四种物件类型又分别有不同的属性配置。因此, 为了要针对其特殊规格而设计, 必须要有一特别的物件描述方式储存在实体装置中。

[0006] 图 1 是显示 Ext2 的磁盘配置的示意图。如图 1 所示, 在一磁盘分割 (Partition) 中, 除了记录文档系统信息的超级区块 (Superblock) 之外, 依照分割区的大小划分多个区块群组 (Block Group)。在 Superblock 中记录了区块的总数量、已经使用及未使用的区块、inode 数量... 等等。每一区块群组包括一群组信息, 其记录对应区块群组的已使用区块、未使用区块、inode 以及真正存放数据的数据区块。文档与目录的相关信息皆存放在 inode 中, 其中目录被视为一种特殊形式的文档, 可据以建立阶层关系。

[0007] 图 2 是显示 OBFS 的磁盘配置的示意图。参考图 2, OSDFS (以 OBFS 为基础) 的整体架构与 Ext2 大致相同, 其将整个磁盘分割成同样大小的大范围的区域 (Region), 例如, 256MB。区域的大小视需要来分配, 此外, 可以依文档大小需求, 再将每一区域划分成全部都是大区块或小区块。每一区域中会有记录已使用或未使用的区块信息, 以及真正存放数据的数据区块。每一区域的元数据 (Metadata) 存放在 onode 中, 其中元数据是用来存放每一

个物件的状态,其利用区域识别码 (Region ID) 及物件识别码来进行索引。

[0008] 目前大部份的 OSD 文档系统技术是利用现行传统的文档系统 (例如, ext2、ext3... 等) 来模拟。传统上 Linux 操作系统的磁盘配置是以内节点 (inode) 为基础,因为其 metadata (如建立时间等) 没办法完整的描述一个物件,只能以模拟物件的方式进行,与 OSD 所欲表达的精神与特性实有不同。此外,一些新的 OSD 的磁盘配置方法 (例如,OBFS 或 OSDFS) 是结合现行的文档系统及 OSD 的部份特性而设计,并非完全专门针对 OSD 的规格中的特性而开发,其未提供完整的属性的储存,所以其描述物件的能力也相对不足。

[0009] 此外,在网络储存架构下的物件可携性问题、硬盘空间使用无弹性且无法充份利用、无法配合现行操作系统架构、无法有效索引物件以及无整体性等问题。因此,现今文档系统的存取方式无法完整有效的将 OSD 的特性显露出来,此外,尚需通过其它的辅助的方式才能展现物件的附属关系。

发明内容

[0010] 本发明实施例揭示了一种物件导向储存装置的磁盘配置方法,其适用于一磁盘。该方法包括下列步骤:初始化该磁盘,并且新增一个 Root 物件。当新增该 Root 物件时,在该磁盘中第一个可使用的区块开始配置一组区块,其分别储存该 Root 物件属性 (此后物件属性包括规格书上定义的属性 (attributes) 加上前述的 metadata)、一未使用区块索引表以及一 Partition 物件索引表,其中该 Root 根据该 Partition 物件索引表来索引一 Partition 物件的位置。当新增一 Partition 物件时,在该磁盘中配置一组区块,其分别储存该 Partition 物件的属性、一 Collection 物件索引表以及一 User 物件索引表,其中该 Partition 物件根据该 Collection 物件索引表来索引 Collection 物件的位置以及根据该 User 物件索引表来索引一 User 物件的位置。当新增一 Collection 物件时,在该磁盘中配置一组区块,其分别储存该 Collection 物件的属性以及一 User 物件识别码索引表,其中该 Collection 物件根据该 User 物件识别码索引表来索引一 User 物件识别码。当新增一 User 物件时,在该磁盘中配置一组区块,其分别储存该 User 物件的属性以及该 User 物件的数据。

[0011] 本发明实施例还揭示了一种物件导向储存装置的磁盘配置方法,其对一磁盘进行磁盘配置,使得该磁盘具有一 Root 物件、至少一 Partition 物件以及至少一 User 物件。该 Root 物件还提供 Root 物件属性、一未使用区块索引表以及该 Root 物件用来索引一 Partition 物件位置的一 Partition 物件索引表。该 Partition 物件还提供 Partition 物件属性、该 Partition 物件用来索引一 Collection 物件位置的一 Collection 物件索引表以及该 Partition 物件用来索引一 User 物件位置的一 User 物件索引表。该 User 物件还提供 User 物件属性以及 ser 物件数据。

附图说明

[0012] 图 1 是显示 Ext2 的磁盘配置的示意图。

[0013] 图 2 是显示 OBFS 的磁盘配置的示意图。

[0014] 图 3 是显示本发明实施例的磁盘配置的示意图。

[0015] 图 4 是显示本发明实施例的 User 物件的数据区块的排列方式的示意图。

- [0016] 图 5 是显示本发明实施例的物件索引表中的区块组关联的示意图。
- [0017] 图 6 是显示本发明实施例的 Partition 物件中的配置示意图。
- [0018] 图 7 是显示本发明实施例的 User 物件索引表的内容存取的示意图
- [0019] 图 8A ~ 8D 是显示本发明实施例的物件导向储存装置的磁盘配置方法的步骤流程图。
- [0020] 图 9 ~ 14 是显示本发明实施例的物件导向储存装置的磁盘配置的流程圖。
- [0021] 附图符号说明
- [0022] SN ~ 下一组区块的起始位置
- [0023] SP ~ 前一组区块的起始位置
- [0024] S801.. S808 ~ 流程步骤

具体实施方式

[0025] 为了使本发明的目的、特征、及优点能更明显易懂，下文特举较佳实施例，并结合附图图 3 至图 14，做详细的说明。本发明说明书提供不同的实施例来说明本发明不同实施方式的技术特征。其中，实施例中的各元件的配置是为说明之用，并非用以限制本发明。且实施例中的附图标号的部分重复，是为了简化说明，并非意指不同实施例之间的关联性。

[0026] 本发明实施例揭示了一种物件导向储存装置的磁盘配置方法。

[0027] 磁盘配置是储存永久数据的基本方式。若传统文档系统要快速、稳定的存取硬盘，并呈现相关内容给终端使用者 (end-user)，通常使用阶层形式来储存文档系统 (例如，Ex2、Ext3、NTFS... 等等)，这些文档系统都有其特定的磁盘配置方式。然而，以物件形式存在的数据储存方式有其特殊需求，包括特定的物件类别及每一物件类别的属性，必须根据识别码来进行存取。目前仅有 OBFS 及 OSDFS 制定出其磁盘配置方式，但是在数据存取上仍被受到限制。

[0028] 基于 OSD 规格书的定义，物件的使用数量是没有限制的。本发明支持 OSD 规格书的定义，且提供非必要性的支持。图 3 是显示本发明实施例的磁盘配置的示意图。参考图 3，本实施例的磁盘配置包含 Root 物件属性，Partition 物件的属性，User 物件的属性，以及 User 物件的数据区。磁盘的使用上不预设任何的分割，而以磁盘能支持的为极限，其中 Root 物件 (即，相等于 Ext2 的 Superblock) 包含属性以及可使用区块或已使用 Partition 的物件索引。此外，本发明可以无限制切割 Root 物件成为多个 Partition 物件。除了属性区块外，Partition 物件更包含可使用区块、已使用 Collection 物件索引及 User 物件索引。另外，Collection 物件也记录了其下的 User 物件识别码 (User Object ID)。每一种物件都有索引可供直接存取到物件属性及数据所在的区块。

[0029] 以下先说明物件导向储存装置 (OSD) 所要表达的储存概念。

[0030] OSD 是一个经标准实作后用来储存成为物件的数据的装置，其中每一个物件依照数据的顺序来储存，且并赋予每一物件一识别码，使得存取数据都要通过物件识别码。对外而言，数据是指一个物件，而非区块组成的文档，并且利用一安全机制将物件存放在储存媒体中。物件种类包括 Root 物件、Partition 物件、Collection 物件以及 User 物件。每一物件包括各自的属性，分别说明如下：

[0031] Root 物件用来描述逻辑单元 (Logic Unit) 的储存空间以及用来管理 Partition

物件。Partition 物件用来描述给 Collection 物件及 User 物件使用的储存空间,以及描述 Collection 物件及 User 物件的安全性、空间限制... 等原则。Collection 物件为用来分类与索引 User 物件的管理性物件。User 物件用来存放使用者数据。

[0032] 下文将分为数个部分来说明本发明的 OSD 的磁盘配置。

[0033] 磁盘配置概念

[0034] 在建立了磁盘逻辑单元 (Logic Unit) 之后,请参考图 3 的实施例说明以本发明进行磁盘配置的储存结构。假设预设存放的基本单位是以一个区块 (Block) 大小为 4K 字节,或者使用者可以自订区块大小。但基于属性数据存放的便利性、空间使用上的完整性、以及配合操作系统存储器管理机制,以 4K 大小的区块最适合。

[0035] Root 物件即为描述 OSD 本身,亦即当使用者欲以 OSD 存放物件,即可将 OSD 视为唯一的 Root 物件。Root 物件有其属性,因此第一个区块即存放 Root 物件的属性数据,第一个区块是在位移 1024 字节之处。如果整个硬盘是给 OSD 使用,则必须保留磁区给开机管理程序 (Boot Loader),而接在属性区块之后的是未使用区块索引表。在未使用区块索引表之后是存放 Partition 物件索引表。因为 Partition 物件可以无限制的产生,因此需要可以存放大量 Partition 物件的表格,表格的配置及使用方式会在之后说明。

[0036] Partition 物件是一个用来储存 Collection 物件与 User 物件的容器 (Container)。当建立一个新的 Partition 物件时,会建立 Partition 物件的属性区块,接在之后的是已使用的 Collection 物件索引表及 User 物件索引表。

[0037] 基本属性数据

[0038] 除了规格书定义的属性及 metadata 之外,本发明仍可自订多个属性以搭配操作系统及磁盘配置的需求。因此,除规格书定义的属性,以下每一物件种类说明可能产生的额外属性,但并非用以限定本发明。

[0039] 「Root 物件可能的额外属性」:其类似 Ext2 的 Superblock 的功能,其记录所有区块使用的信息、属性使用的区块数量、与各物件的索引表相关的记录、定位区块的长度 (32 或 64 位)、物件识别码的长度 (32 或 64 位)、物件识别码的计数器等,或其它所需的属性。

[0040] 「Partition 物件的可能额外属性」:其记录与 Collection 物件及 User 物件的索引表相关的记录,包括起始位置及每次配置表格的大小等,或其它所需的属性。

[0041] 「Collection 物件可能的额外属性」:其记录 User 物件索引表的大小及位置等。因为 User 物件和 Collection 物件是多对多的关系,所以必须要相互索引以快速寻找。

[0042] 「User 物件可能的额外属性」:由于 User 物件负责存放数据,因此若数据量很小,也就是属性加数据小于一个区块 (如:4K bytes) 大小时,可以将该数据当成属性储存,就是在属性表内会有一个为属性内数据 (In-property Data) 的字段,数据存放在此字段中。或其它所需的属性。

[0043] 如果 User 物件的数据不是以 In-property Data 的形式存在,则代表该数据是存在于额外的区块中,因此 User 物件必须要记录使用的区块,在其属性表中有一个属性,其称为区块串列 (Block List)。

[0044] 区块串列由多个连续记录组成,每一个记录存放数据区块的位置及其长度,即先定义 1 或 2 个字节记载后面衔接的数据区块所记录的数量,接下来以位置及长度为一组记载数据区块的记录,其中每一记录的格式为记录内容是分区配置 (Extent-based) 的方式

存在,也就是以位置 (Location)+连续区块长度 (Length),以 32 位的配置来说,其每一记录的使用空间为位置 (32 位)+连续区块长度 (32 位) = 8 字节。定位用的位置是指第 n 个区块的位置,而非实际地址,其中实际地址必须利用公式计算而得,该公式表示为 $n \times$ 区块大小 (例如,4K、8K、...)。以一个 4K 的区块为例,当位置 (Location) 为 100 时,其实际地址为 $100 \times 4K$ 。以下说明公式中的变数。

[0045] 图 4 是说明 User 物件区块串列属性的数据区块的排列方式,其中第 1 个字段为 1 个字节,用来记录已储存数据区块的数量,因此最多可存放 255 个不连续区块。

[0046] 位置 (例如,长度为 32 位):可以使用 2^{32} 个区块,定址到 $2^{32} \times 4K = 17592G$ 字节 = 16T 字节的地址空间。

[0047] 长度 (例如,长度为 32 位):使用连续区块的数量,单一个连续区块可以使用 $2^{32} \times 4K = 17592G$ 字节 = 16T 字节的硬盘空间。

[0048] 举例来说,若属性只用一个区块,一个区块大小为 4K,假设基本属性的部份占了 3K 的空间,剩于 1K 的空间给区块串列使用,则最多可以记录 $(1024-2)/8 = 127$ 个不连续的数据空间;或者给属性内数据存放 1K 的数据。若从未使用区块索引表取得可以存放的空间,但是记录的数量超过该可以存放的空间,表示空间使用将近完毕而造成磁区使用过于分散,故需要进行磁盘重组。

[0049] 索引及表格的使用

[0050] 由于索引表在磁盘配置扮演重要的数据定位及寻找的功能,以下将细部说明各种索引表分类及其使用方式。每一种索引表将从其最基本的表格结构说明起,之后会以例子说明其用法。为了因应无限制物件使用的需求,在本发明实施例中设计几种特殊的表格,包括物件索引表、未使用区块索引表以及 Collection 物件的 User 物件识别码索引表。

[0051] 物件索引表

[0052] 在 OSD 中是以物件为存取为核心,如何快速寻找及大量储存物件变成此磁盘配置的核心工作,因此必须设计符合需求的物件索引。

[0053] Table 结构及其连结

[0054] 物件索引表使用预设一次配置所需数量的区块,其预设数量会记载在 Root 物件的属性中。假设一组区块包括数个区块不够使用时,亦即超过上限临界值 (Upper Threshold),可以再动态分配一组新的未使用的区块,其是根据未使用区块索引表来取得分配。当将多组区块连结使用时,假设有 m 组区块,如果记录因删除导致降低到第 m-1 组的下限临界值 (LowerThreshold),则将第 m 组区块释放移除。

[0055] 区块组之间彼此有连结的关系,假设区块定位使用 32 位 = 8 字节,每一组区块的第一个区块的前面保留 16 字节,前 8 个字节为前一组区块的起始位置 (Start location of previous set of blocks (SP)),其记录前一组区块的位置,而后 8 个字节为下一组区块的起始位置 (Start location of next set of blocks (SN)),其记录下一组区块的位置。如果只有一组区块,则 SP 及 SN 都表示为 NULL。如果有多组区块,则第一组区块的 SP 表示为 NULL,而最后一组区块的 SN 表示为 NULL。如果其中一组区块的数据将近额满,亦即超过临界值 (其中该临界值属于 Root 的额外属性),则必须再增加一组新的区块,如图 5 所示。

[0056] 存放于物件索引表的记录

[0057] 物件索引表可包括属于 Root 物件的 Partition 物件索引表 (PartitionObject

Table)、属于 Partition 物件的 Collection 物件索引表 (Collection ObjectTable) 以及 User 物件索引表 (User Object Table), 每一物件索引表内部所存放的每一笔记录代表一个物件索引, 而物件索引的格式为物件识别码 (Object ID) 加上位置 (Location), 则一物件索引表可以记载的索引数量公式为:

[0058]

$$\text{一组区块可记录索引的数量} = \frac{\text{一组区块大小} - SP - SN}{\text{索引长度}}。$$

[0059] 如果物件识别码 (包括 Partition 识别码 (Partition ID)、收集识别码 (Collection ID) 以及使用者识别码 (User Object ID)) 的长度为 32 位 (4 字节), 区块位置 (Block Location) 固定为 32 位时, 则一笔记录的长度为 Object ID+Location = 8 字节。因此, 1 个 4K 大小的区块如果以 8 字节做为记录单位, 则可储存 $4096 - 8 - 8 / 8 = 510$ 个记录。

[0060] 另外, 如果物件识别码的长度为 64 位 (8 字节), 区块位置 (BlockLocation) 固定为 32 位时, 则一笔记录的长度为 Object ID+Location = 12 字节。因此, 1 个 4K 大小的区块如果以 12 字节为记录单位, 扣除 SP 及 SN 各占 8 字节, 则可存放 $(4096 - 8 - 8) / 12 = 340$ 个记录。

[0061] 由于整个物件索引表是以连结多组区块所形成, 以一次一组 16 个区块来看, 共占用 $16 \times 4K = 64K$ 的数据空间。若物件识别码的长度为 32 位, 则可存放 $(16 \times 4096 - 8 - 8) / 8 = 8190$ 个物件索引。若物件识别码的长度为 64 位, 则以单一组区块可存放 $(16 \times 4096 - 8 - 8) / 12 = 5460$ 个物件索引。

[0062] 未使用区块索引表 (Free Block Table)

[0063] 未使用区块索引表可定义在 Root 物件中。若要取得空间, 必须要跟 Root 物件中的未使用区块请求适合的大小。未使用区块索引表内部所存放的每一笔记录代表一个连续的未使用的空间, 未使用的区块记录如果是多组分散的, 则形成一个多笔记录, 而每一笔记录的格式也是利用分区配置的方式记录, 位置 (Location)+ 区块长度 (Length)。

[0064] 以一次一组 8 个区块来看, 共占用 $8 \times 4K = 32K$ 的数据空间, 可记录 $510 \times 8 = 4080$ 笔不连续未使用的储存空间。

[0065] 记录内容会因执行新增、寻找、删除及修改 (因释放区块所产生的合并, 需要修改大小) 等操作而变动。在对磁盘进行格式化时, 会忽略损毁的区块, 只在未使用区块索引表中记录正常的区块。因此, 格式化磁盘时如果没有损坏的区块, 则在格式化完成后仅会在未使用区块索引表中产生一笔记录。相对的, 格式化磁盘时如果有损坏的区块, 则在格式化完成后会在未使用区块索引表中产生对应损坏区块的多笔记录。

[0066] Collection 物件的 User 物件识别码索引表

[0067] 在 Collection 物件中, 除了属性本身之外, 也包括储存 User 物件识别码的记录, 所述记录是存放在 User 物件识别码索引表中。User 物件识别码索引表 (User Object ID Table) 可视为 Collection 物件的数据, 且其记录了 User 物件识别码。识别码索引表的增加及减少也是利用上述阈值规则。以一次一组 4 个区块来看, 共占用 $4 \times 4K = 16K$ 的数据空间。

[0068] 若物件识别码的长度为 32 位, 则可存放 $(4096/4) \times 4 = 4096$ 个 User 物件识别码。若物件识别码的长度为 64 位, 则可存放 $(4096/8) \times 4 = 2048$ 个 User 物件识别码。

[0069] 各种物件存取方法的说明

[0070] Root 物件

[0071] 一开始先存放 Root 物件的属性区块,接下来利用一组区块存放整个磁盘逻辑单元的未使用区块索引表,再利用一组区块存放 Partition 物件索引表,以下说明 Partition 物件存取的一些操作。

[0072] 以新增 Partition 物件为例,从未使用区块索引表取得一笔记录可以有适当数量的区块给 Partition 物件,其中这些区块包括 Partition 物件的属性、预设使用区块数量的 Collection 物件索引表以及预设使用区块数量的 User 物件索引表,再加入一笔 Partition 物件索引到 Partition 物件索引表中,并且修改 Root 物件的相关属性值。完成 Partition 物件配置后便可将此记录的位置往后移,移到已使用的区块位置之后。

[0073] 以移除 Partition 物件为例,移除 Partition 物件包括强制性移除与非强制性移除。强制性移除必须将 Partition 物件里所有的 Collection 物件与 User 物件移除,才能移除 Partition 物件。非强制性移除则与移除一个清空的 Partition 物件一样,将 Partition 物件及其以下的 Collection 物件及 User 物件使用的区块记录归还未使用区块索引表,然后从 Partition 物件索引表将该 Partition 物件的索引移除,并且修改 Root 物件的相关属性值。

[0074] 就改变 Partition 大小来说,只需修改 Partition 物件的相关属性值。

[0075] Partition 物件

[0076] 先存放 Partition 物件的属性区块,利用一组区块存放 Collection 物件索引表与 User 物件索引表。Collection 物件索引表为选择性的,如果系统选择不支持使用 Collection 物件,则不放置 Collection 物件索引表。有关 Partition 物件及其内容的存取如图 6 所示。

[0077] 改变 Partition 大小

[0078] 只须修改 Partition 物件的相关属性值。

[0079] Collection 物件存取

[0080] Collection 物件存取是选择性的,在格式化时决定要不要支持 Collection 物件,其存取方式包括新增与移除 Collection 物件。新增 Collection 物件是从未使用区块索引表取得适当大小的区块,以存放 Collection 物件属性及 User 物件识别码索引表 (User Object ID Table),之后加入一笔 Collection 物件索引到 Partition 物件的 Collection 物件索引表中,并修改 Partition 物件的相关属性值。

[0081] 移除 Collection 物件是将 Collection 物件索引从 Collection 物件索引表中移除,再将 Collection 物件属性及 User 物件识别码索引表使用的区块释放回 Root 物件的未使用区块索引表中,并修改 Partition 物件的相关属性值。

[0082] User 物件的存取

[0083] User 物件的存取包括新增 User 物件、移除 User 物件以及改变 User 物件的数据区块。

[0084] 新增 User 物件是先从 Root 物件的未使用区块索引表取得区块,包括给予属性及数据使用的区块,再加入一笔 User 物件索引到 Partition 物件的 User 物件索引表中,并修改 Partition 物件的相关属性值。

[0085] 移除 User 物件是将 User 物件属性及数据的区块释放回 Root 物件的未使用区块索引表中,再从 Partition 物件的 User 物件索引表中将 User 物件索引移除,并修改 Partition 物件的相关属性值。

[0086] User 物件的数据区块是将数据使用的区块记录在属性值中,因此当数据大小有改变时会连带牵动属性及未使用区块索引表的记录。

[0087] 需注意到,Partition 物件不是根据 Partition 物件索引表产生而是程序建立产生。在产生 Partition 物件之后,会产生一个 Partition 的 ID 索引,这个 ID 索引会放到 Partition 物件索引表里。

[0088] Collection 物件 (选择性 (Optional))

[0089] Collection 物件为选择性的支持,因此在磁盘格式化过程中可以选择要不要加入此物件。当有支持此物件时,可以新增及删除 User 物件的 ID 索引。因为 User 物件的 ID 索引是放在 Collection 物件的 User 物件识别码索引表中,所以索引表的增减也适用上述阈值规则。

[0090] 需注意到,Collection 物件不是根据 Collection 物件索引表产生而是程序建立产生。在产生 Collection 物件之后,会产生一个 Collection 的 ID 索引,这个 ID 索引会放到 Collection 物件索引表里。

[0091] User 物件

[0092] User 物件为真正存放数据的物件类别,如图 7 所示,当前一组存放物件索引的表格不足时,就会用到一组新的表格。

[0093] 存取 User 物件的操作包括新增 User 物件、删除 User 物件以及改变物件数据。

[0094] 新增 User 物件是先从 Root 物件的未使用区块索引表中取得适当大小足够的空间,以取得一次性的足够储存空间为主。如果不能从单一笔记录取得的足够的连续空间,则必须使用多笔记录的不连续的空间,计算取得不连续空间的大小到足够写入此物件数据。接着,修改 User 物件属性,然后加入一笔数据到 Partition 物件的 User 物件索引表中。

[0095] 删除 User 物件是将 ID 索引从 Partition 物件的 User 物件索引表中移除,并将属性中使用到的区块归还到 Root 物件的未使用区块索引表。从 User 物件的属性中可以得知,如果物件属于某些 Collection 物件,则必须从 Partition 物件的 Collection 物件索引表中去找到对映的 Collection 物件,并从这些 Collection 物件的 User 物件识别码索引表里移除里面的 User 物件的 ID 记录。

[0096] 数据会因存取而造成空间使用上的改变。如果预先配置的储存空间足够,则直接修改 User 物件的数据长度的属性即可。如果储存空间不足,则必须从 Root 物件的未使用区块索引表中请求区块来使用。如果因为删除数据而产生多余的区块,则归还到 Root 物件的未使用区块索引表中。

[0097] 使用 In-property data 时,如果是数据缩减,只会修改 User 物件的数据长度属性内容。如果数据长度增加而不足以存放在属性区块中时,则必须向 Root 物件的未使用区块索引表中要求分配未使用的区块,并将原有的 In-property data 数据移至被分配的区块中。是否支持此属性可以由格式化时预设,或者当设定属性时改变。

[0098] 以上简单说明了本发明实施例的磁盘配置与各物件的功能与使用方式,下文将说明物件导向储存装置的磁盘配置方法的实施流程。

[0099] 需注意到, User 物件不是根据 User 物件索引表产生而是程序建立产生。在产生 User 物件之后,会产生一个 User 的 ID 索引,这个 ID 索引会放到 User 物件索引表里。

[0100] 图 8A 是显示本发明实施例的物件导向储存装置的磁盘配置方法的步骤流程图。

[0101] 首先,在初始执行磁盘配置时,新增一个 Root 物件(步骤 S801),即在一磁盘中配置一组区块,其分别储存 Root 物件的属性、未使用区块索引表(Free Block Table)以及 Root 物件用来索引 Partition 物件的位置的 Partition 物件索引表(步骤 S802),如图 9 所示。

[0102] 需注意到,在配置给该 Root 物件的该组区块中,包括给 Partition 物件属性使用的一个区块,给未使用区块索引表的一或一个以上的区块,给 Partition 物件索引表使用的一或一个以上的区块。上述区块是根据 Root 物件属性值而定,而 Root 物件的属性是在磁盘初始格式化之后就决定,也就是利用格式化参数或者预设值来决定上述区块。

[0103] 当新增一个 Partition 物件时(步骤 S803),在该磁盘中另外配置一组区块,其分别储存第 1 Partition 物件的属性、Partition 物件用来索引 Collection 物件位置的 Collection 物件索引表以及 Partition 物件用来索引 User 物件位置的 User 物件索引表(步骤 S804),例如图 10 所示。此时,Partition 物件索引表的内容如图 11 所示,其中第 1 分割区在第 10 个区块位置。

[0104] 需注意到,在配置给该 Partition 物件的该组区块中,包括给 Partition 物件属性使用的一个区块,给 Collection 物件索引表使用的一个或一个以上的区块,给 User 物件索引表使用的一个或一个以上的区块。上述区块是根据 Root 物件属性值而定,而 Root 物件的属性是在磁盘初始格式化之后就决定,也就是利用格式化参数或者预设值来决定上述区块。

[0105] 当新增一个 Collection 物件时(步骤 S805),在该磁盘中另外配置一组区块,其分别储存第 1 Collection 物件的属性以及 Collection 物件用来索引 User ID 的 User 物件识别码索引表(步骤 S806),如图 12 所示,其中 User 物件识别码索引表是用以得知 Collection 物件包括哪些 User 物件。

[0106] 需注意到,Collection 物件存取是选择性的。在配置给该 Collection 物件的该组区块中,包括给 Collection 物件属性使用的一个区块,区块给 User 物件识别码索引表使用的一或一个以上的区块。上述区块是根据 Root 物件属性值而定,而 Root 物件的属性是在磁盘初始格式化之后就决定,也就是利用格式化参数或者预设值来决定上述区块。

[0107] 若再新增一个 Collection 物件,即在该磁盘中另外配置一组区块,其分别储存第 2 Collection 物件的属性以及 Collection 物件用来索引 User ID 的 User 物件识别码索引表,如图 13 所示,其中 User 物件识别码索引表是用以得知 Collection 物件包括哪些 User 物件。

[0108] 当新增一个 User 物件时(步骤 S807),在该磁盘中另外配置一组区块,其分别储存 User 物件的属性以及 User 物件的数据(Data)(步骤 S808),如图 14 所示。需注意到,在配置给该 User 物件的该组区块中,包括给 User 物件属性使用的一个区块,以及给 User 物件的数据区块使用的零或零个以上的区块。属性区块是固定配置,数据区块则根据应用程序的需要可事先配置(可不必储存数据)或者动态延伸。数据区块的位置记录在 User 物件属性中。

[0109] 需注意到,Root 物件可以产生不限数量的 Partition 物件,Partition 物件可以产生不限数量的 Collection 物件及 User 物件。

[0110] 以下进一步说明 Partition 物件、Collection 物件与 User 物件的新增与删除操作流程实施例。

[0111] 图 8B 是显示本发明实施例的新增与删除 Partition 物件的步骤流程图。

[0112] 首先,判断新增或删除 Partition 物件(步骤 S811)。若欲新增 Partition 物件,则在一磁盘中的配置一组区块,其分别储存 Partition 物件的属性、Partition 物件用来索引 Collection 物件位置的 Collection 物件索引表以及 Partition 物件用来索引 User 物件的位置的 User 物件索引表,并且修改 Root 物件属性以及修改 Root 物件的未使用区块索引表(步骤 S812)。若欲删除 Partition 物件,则判断 Partition 物件是否含有 Collection 或 User 物件(步骤 S813)。若 Partition 物件含有 Collection 或 User 物件,则删除所有 Collection 物件及 User 物件(步骤 S814)。若 Partition 物件未含有 Collection 或 User 物件,则修改 Root 物件属性以及修改 Root 物件的未使用区块索引表(步骤 S815)。

[0113] 图 8C 是显示本发明实施例的新增与删除 Collection 物件的步骤流程图。

[0114] 首先,判断新增或删除 Collection 物件(步骤 S821)。若欲新增 Collection 物件,则在一磁盘中的配置一组区块,其分别储存 Collection 物件的属性以及 Collection 物件用来索引 User 物件识别码的 User 物件识别码索引表,并且修改 Partition 物件属性及 Partition 物件的 Collection 物件索引表,以及修改 Root 物件属性及 Root 物件的未使用区块索引表(步骤 S822)。若欲删除 Collection 物件,则删除 Collection 物件及 User 识别码索引表(步骤 S823),并且修改 Partition 物件属性及 Partition 物件的 Collection 物件索引表,修改 Root 物件属性,以及修改 Root 物件的未使用区块索引表(步骤 S824)。

[0115] 图 8D 是显示本发明实施例的新增与删除 User 物件的步骤流程图。

[0116] 首先,判断新增或删除 User 物件(步骤 S831)。若欲新增 User 物件,则在一磁盘中的配置一组区块,其分别储存 User 物件的属性、User 物件储存数据的空间,并且修改 Partition 物件属性及 Partition 物件的 User 物件索引表,修改 Root 物件属性,以及修改 Root 物件的未使用区块索引表(步骤 S832)。若欲删除 User 物件,则判断 User 物件是否属于 Collection 物件(步骤 S833)。若 User 物件属于 Collection 物件,则删除所有 Collection 物件中含有 User 物件识别码的索引(步骤 S834)。若 User 物件不属于 Collection 物件,则修改 Partition 物件属性与 Partition 物件的 User 物件索引表,修改 Root 物件属性,以及修改 Root 物件的未使用区块索引表(步骤 S835)。

[0117] 如上所述,藉由本发明实施例的物件导向储存装置的磁盘配置方法,当物件管理装置无法使用或找不到物件时,可以从储存装置中直接存取,而当储存装置无法随着物件管理装置移植时,可以不需要物件管理装置,而可以从储存装置中直接索引与存取数据。

[0118] 由于索引表是用于存取物件,因此可以利用数据快取的方式来存取藉由本发明的磁盘配置所实作而成的系统。索引表所占的记体空间不大,当读取索引表中的数据时,以一次读取一整组区块到存储器中,并预先扫描并分析这组区块,在扫描时同时记录在索引表中未使用记录的位置。在索引表中产生未使用记录的位置的原因在于索引表在存取记录时,会有删除记录的操作,因而在索引表中产生记录缺口(Record Holes)(即,未使用的记录)。

[0119] 若发现在快取的索引表中有未使用记录的位置,则先以填入这些记录缺口为优先(即,新增操作),若没有时就附加在索引表中的最后记录之后面。当找到欲删除的记录时,将记录的内容全设为 0(表示为 NULL),则该记录即被删除。在索引表中寻找记录的速度是影响磁盘配置的关键,可利用循序搜寻(Sequential Search)或其它搜寻方式来实现。

[0120] 排序的数据可加快记录的寻找,而记录排序的方式是依照各个索引表的功能性,并且根据不同的关键(Key)值来排序。。

[0121] 以从头开始分配连续的区块为主,如遇到文档变更而需要额外的储存区块,但是无法分配后面的连续区块,则新分配的区块以最接近的区块为原则,并且以不连续空间方式储存。当遇到大型的新文档时,则以分配连续的空间为优先,或者以能取得的最多的连续空间为主。对未使用区块可执行新增、列表及删除操作。

[0122] 寻找索引表的操作可利用相关的演算法来执行。每一索引表所分配的一组区块数量皆可以参数化方式来订定。可以提供工具将表格最佳化处理,以缩短寻找时间,例如将表格排序,并且以不产生更多不连续空间为原则。

[0123] 综上所述,本发明的实施例提供:

[0124] 在硬盘起始规画方面,包括 Root 物件的 metadata,传统文档系统配置称为超级区块(Superblock)、Partition 物件索引表以及未使用区块索引表。在连续空间取得方面,由单一的未使用区块索引表取得,其为 extent-based。在物件属性存取方面,只要取得物件位置,即可直接存取属性与物件索引表,直接索引分类各物件。在数据存取方面,由属性内取得数据或由区块串列的记录读取。在现有实作的支持方面,实作完成 Linux 上 OSD 模拟文档系统的核心层模块。

[0125] 在物件索引方面,以集中的记录形成的表格,可一次性的读取所有的索引数据形成快取。并辅以搜寻演算法。在物件属性方面,metadata 合并物件属性连同数据部份可形成一连续记录。还可自订属性。在物件的阶层关系方面,清楚的阶层式配置。支持所有的物件。在操作系统的支持方面,完全搭配系统缓冲管理的需求做成核心模块(Kernel Module),以文档方式展现或另成一独立存取模式。在硬体整合方面,利用简单的配置设计,可单独设计 IC 晶片存取物件,以达成 OSD 所要求的空间管理等功能一并整合至储存装置中。因表格皆可快取,数据直接定位。

[0126] 本发明的方法,或特定型态或其部份,可以以程序码的型态存在。程序码可以包含于实体媒体,如软盘、光盘片、硬盘、或是任何其他机器可读取(如计算机可读取)储存媒体,其中,当程序码被机器,如计算机载入且执行时,此机器变成用以参与本发明的装置。程序码也可以通过一些传送媒体,如电线或电缆、光纤、或是任何传输型态进行传送,其中,当程序码被机器,如计算机接收、载入且执行时,此机器变成用以参与本发明的装置。当在一般用途处理单元实作时,程序码结合处理单元提供一操作类似于应用特定逻辑电路的独特装置。

[0127] 虽然本发明已以较佳实施例揭示如上,然其并非用以限定本发明,本领域的技术人员在不脱离本发明的精神和范围的前提下可作各种的更动与润饰,因此本发明的保护范围以本发明的权利要求为准。

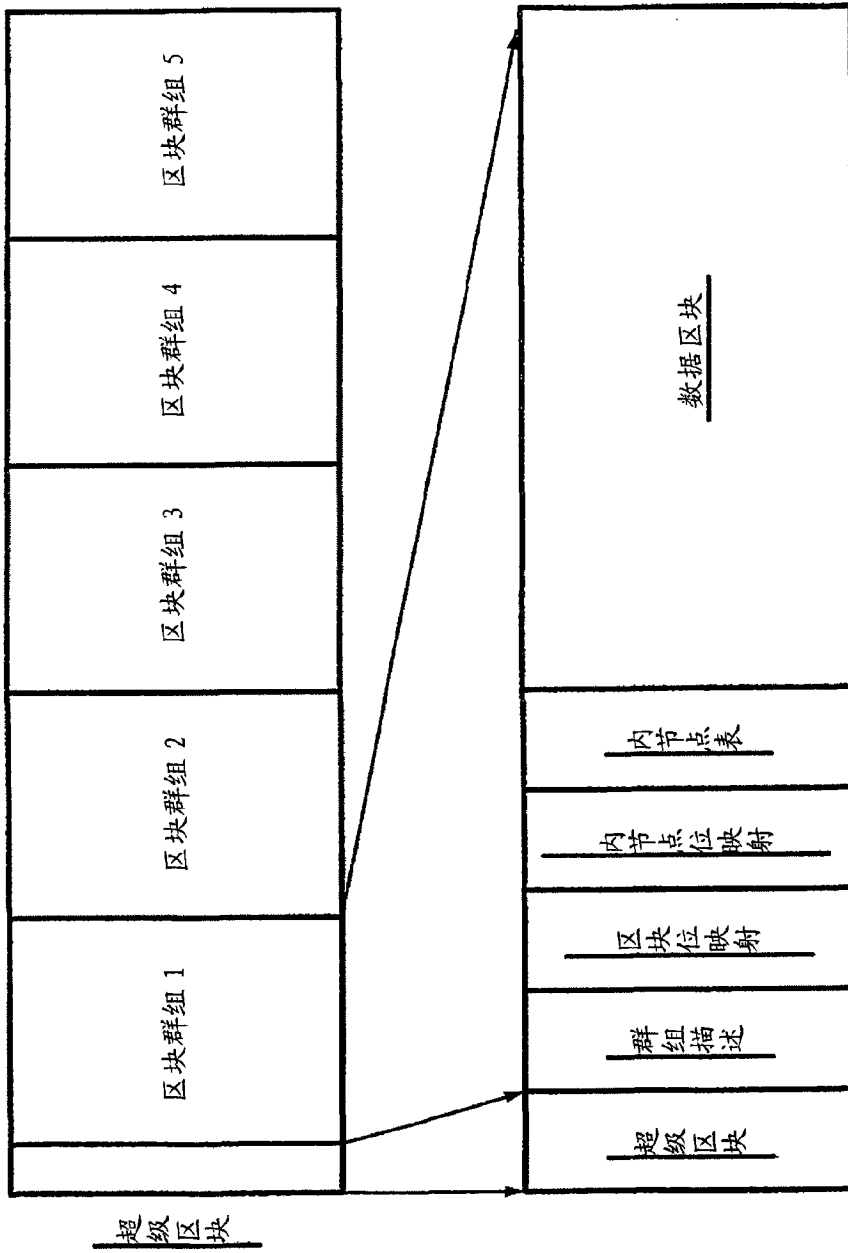


图 1

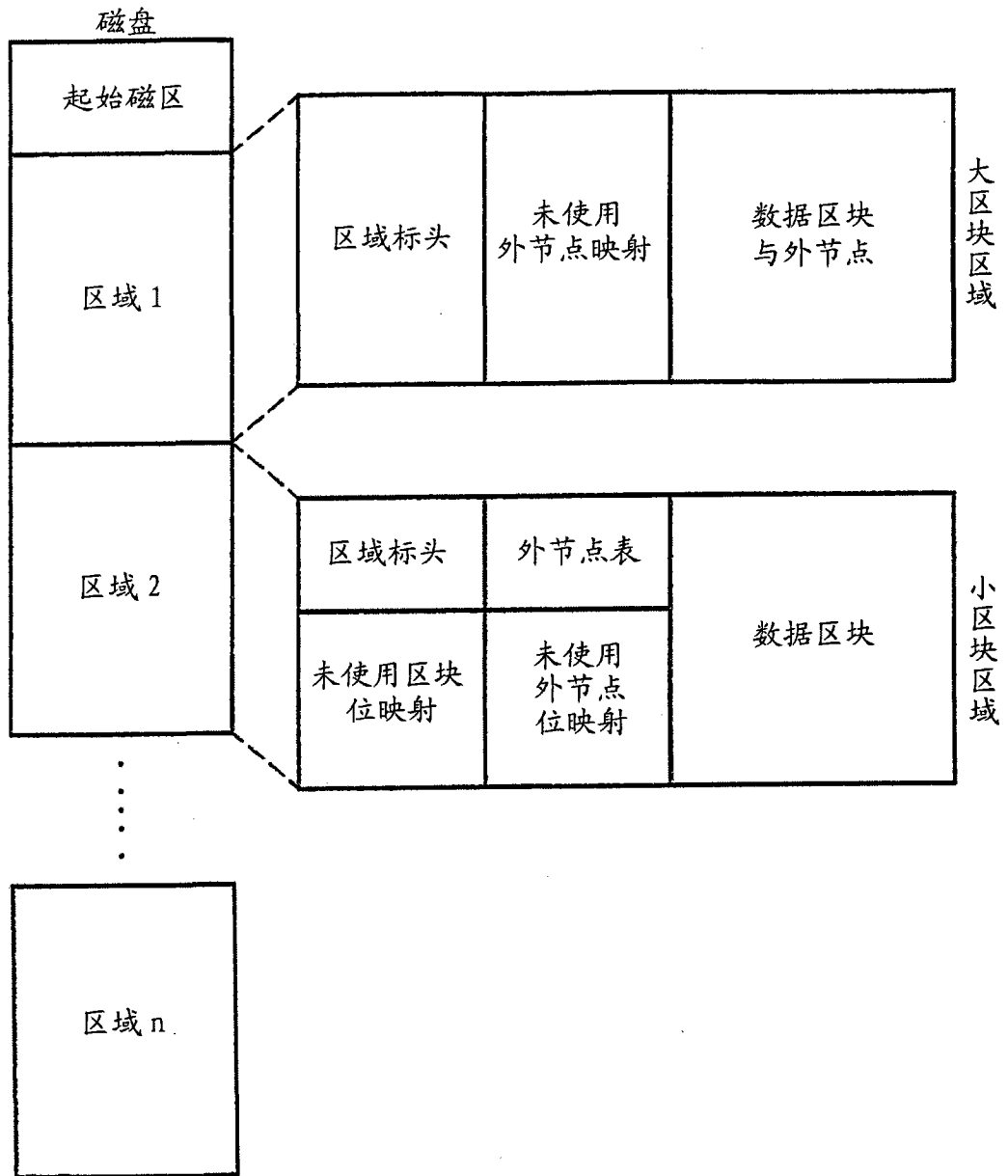


图 2

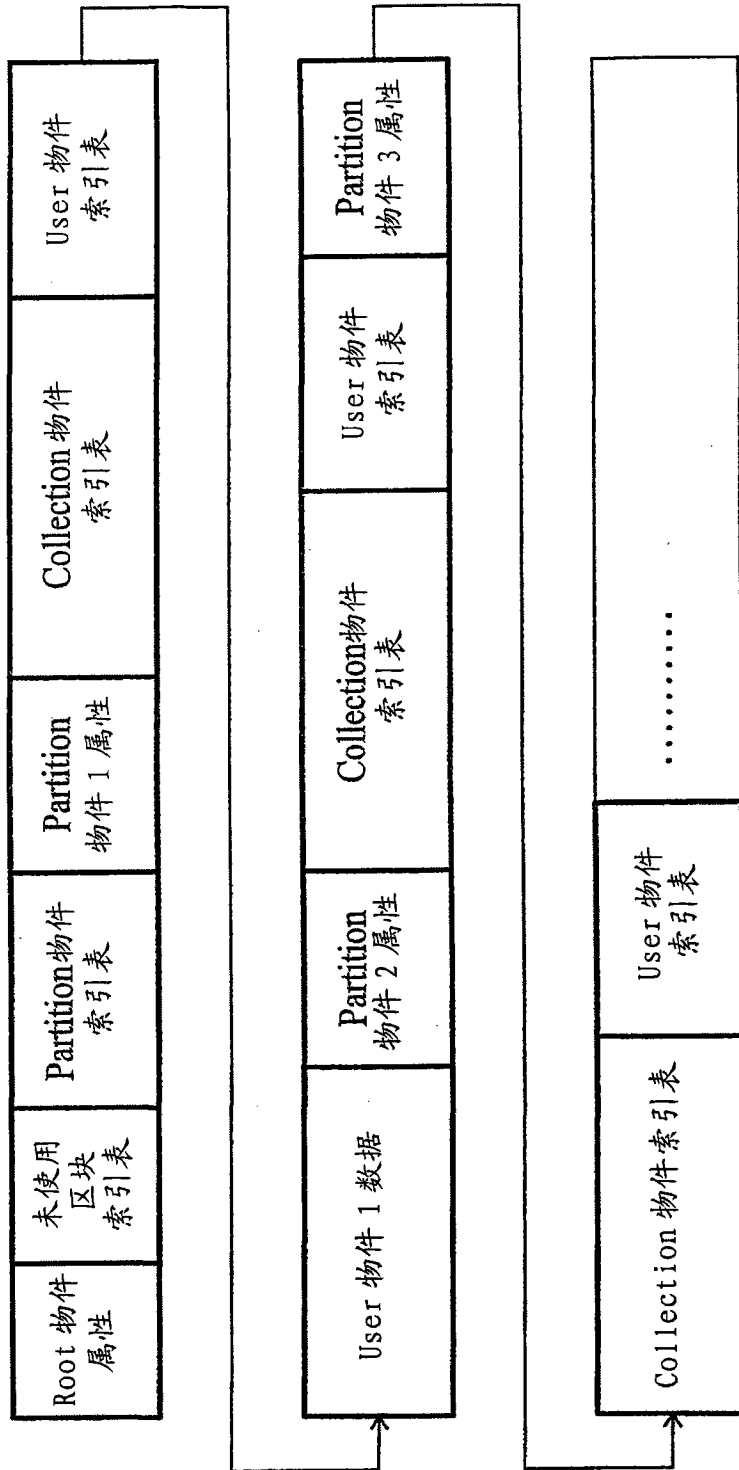


图 3

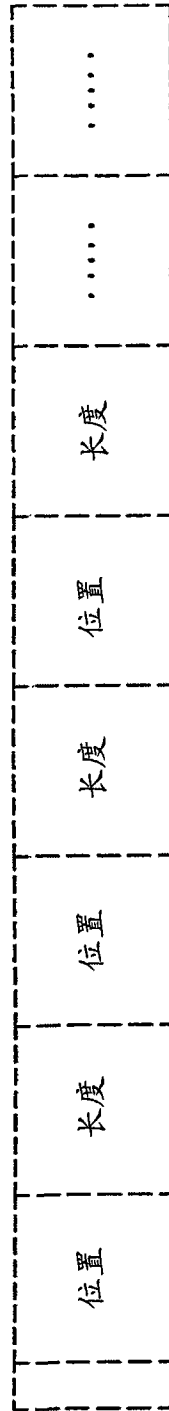


图 4

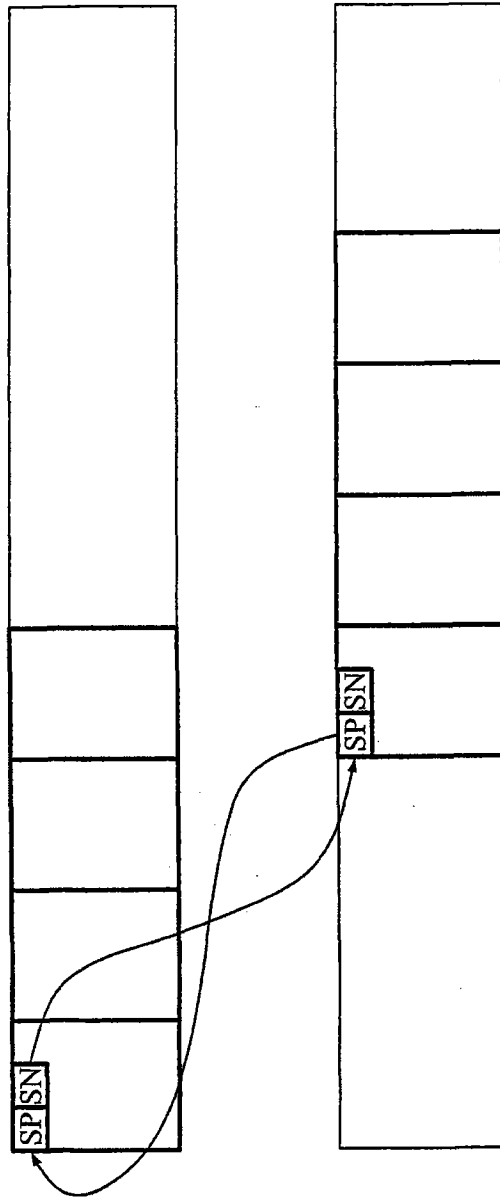


图 5

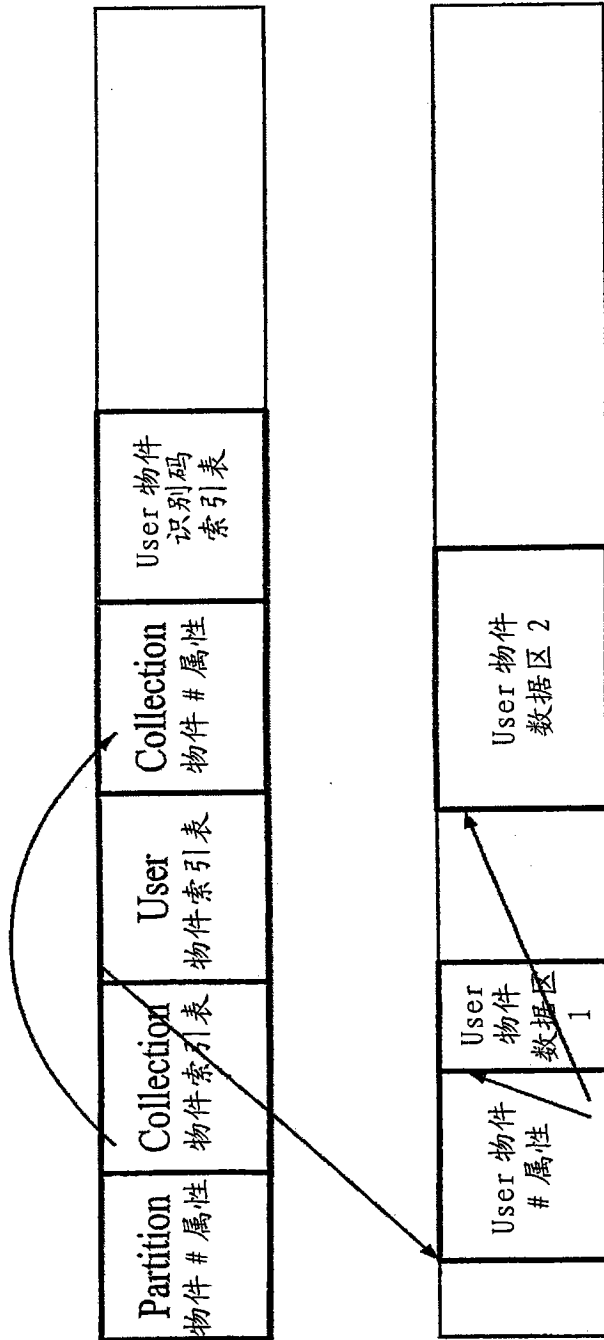


图 6

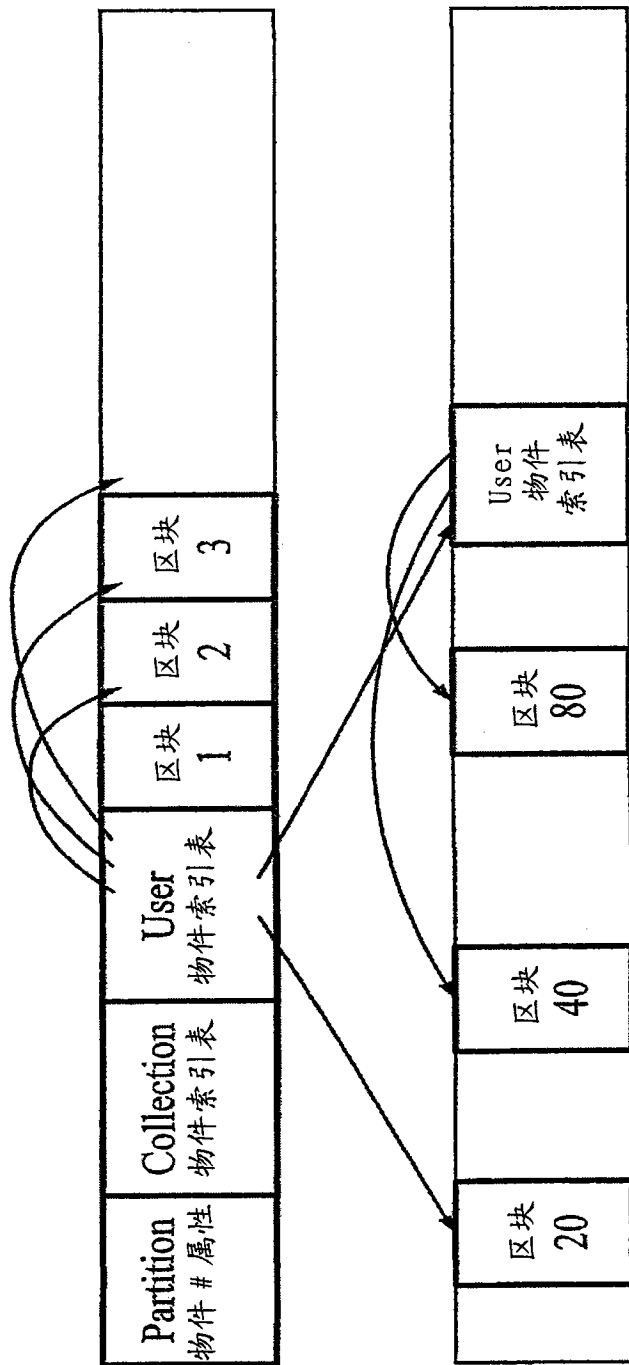


图 7

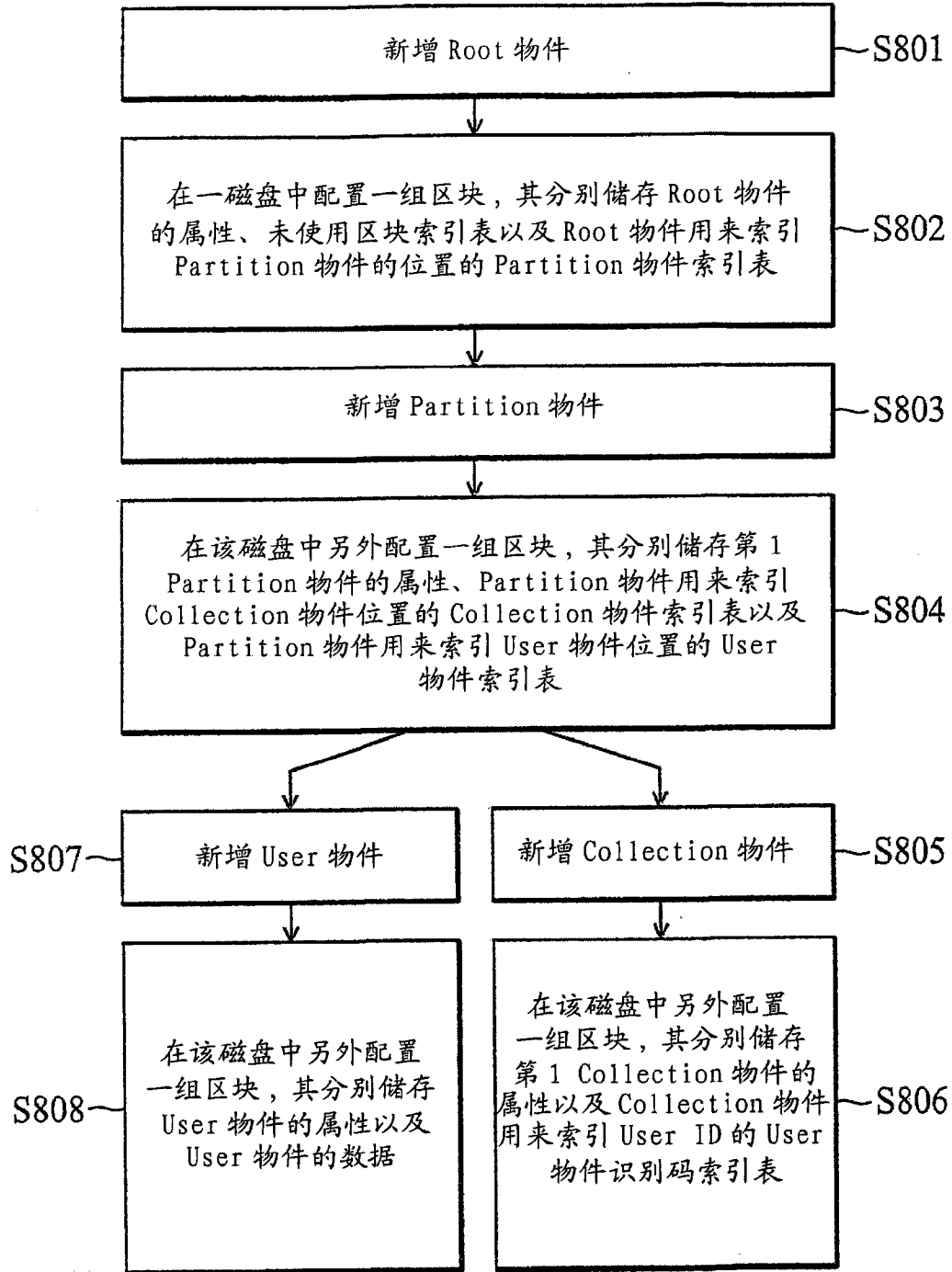


图 8A

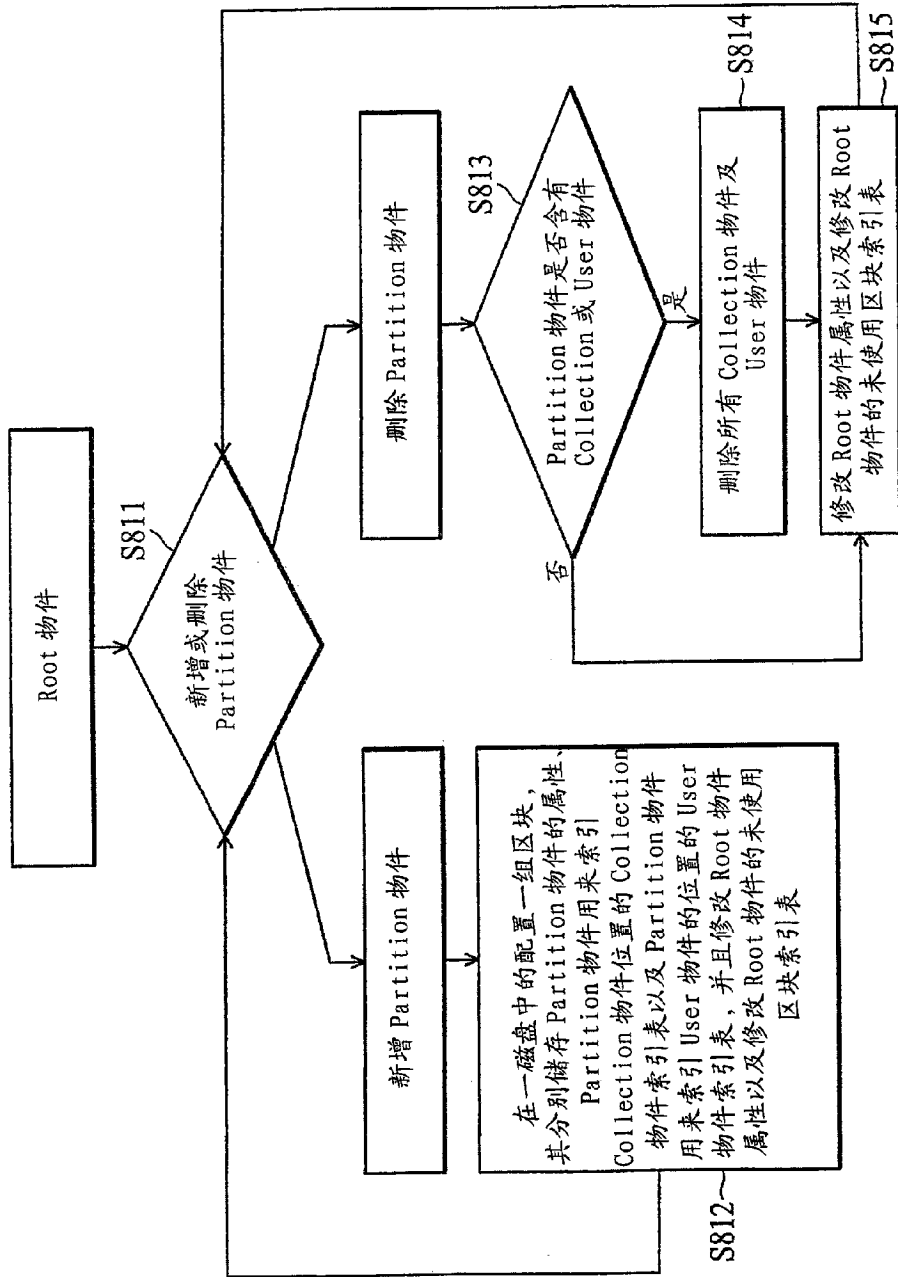


图 8B

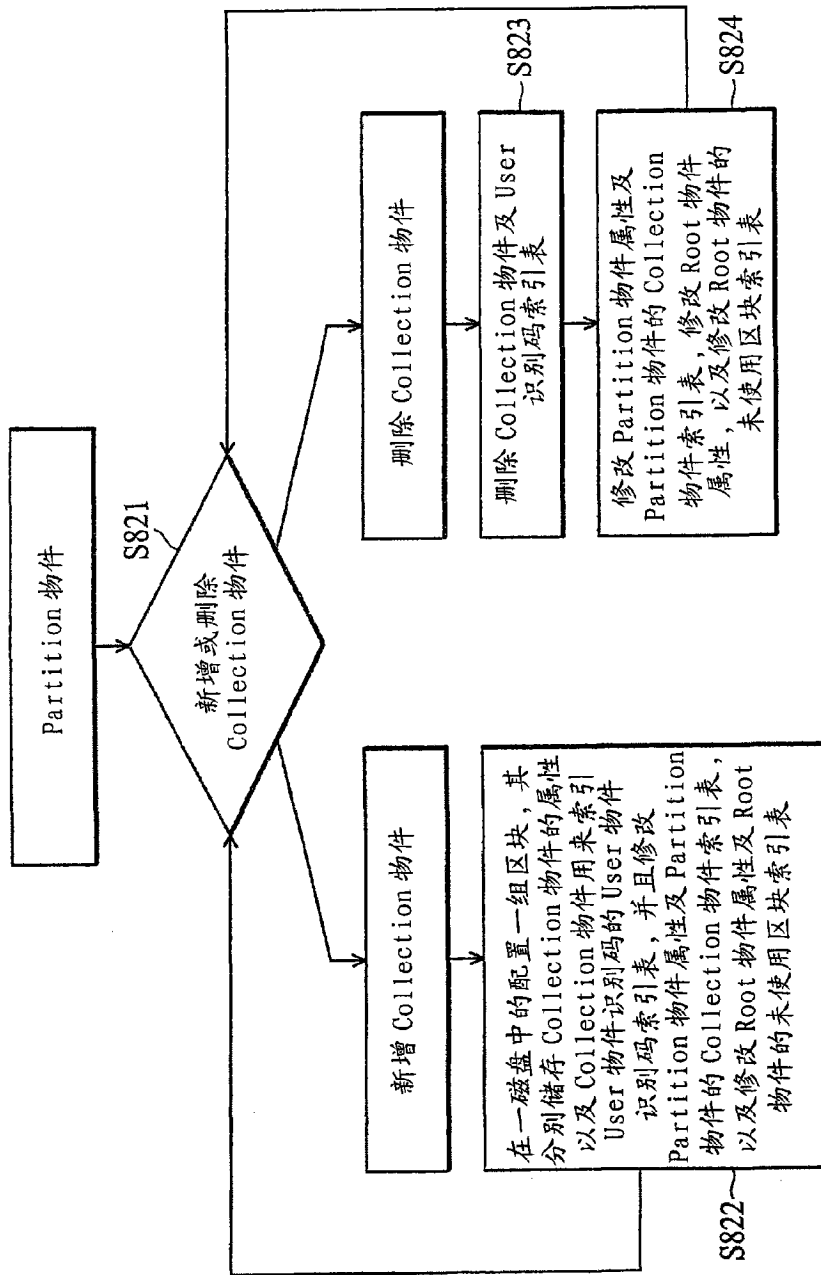


图 8C

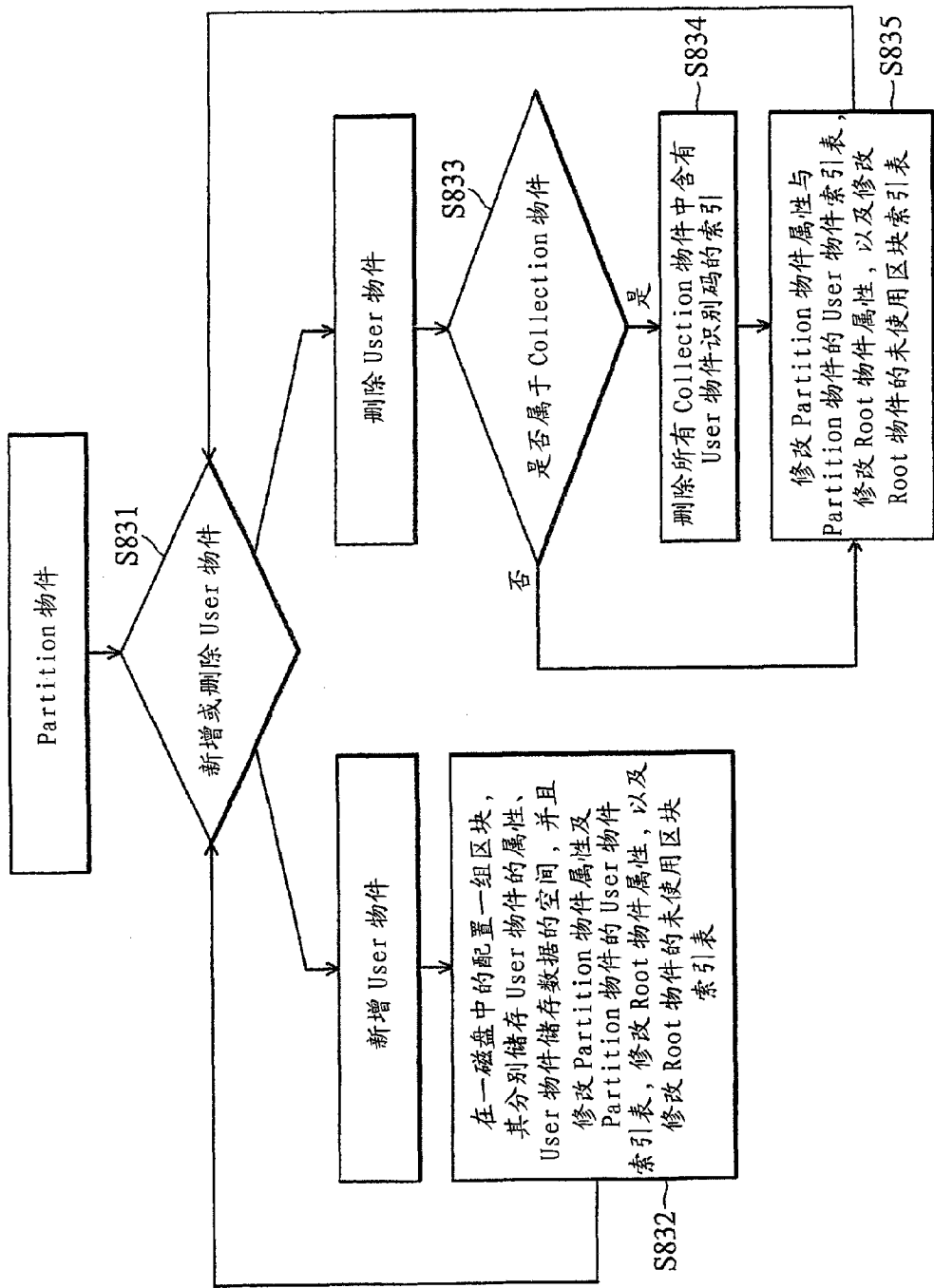


图 8D

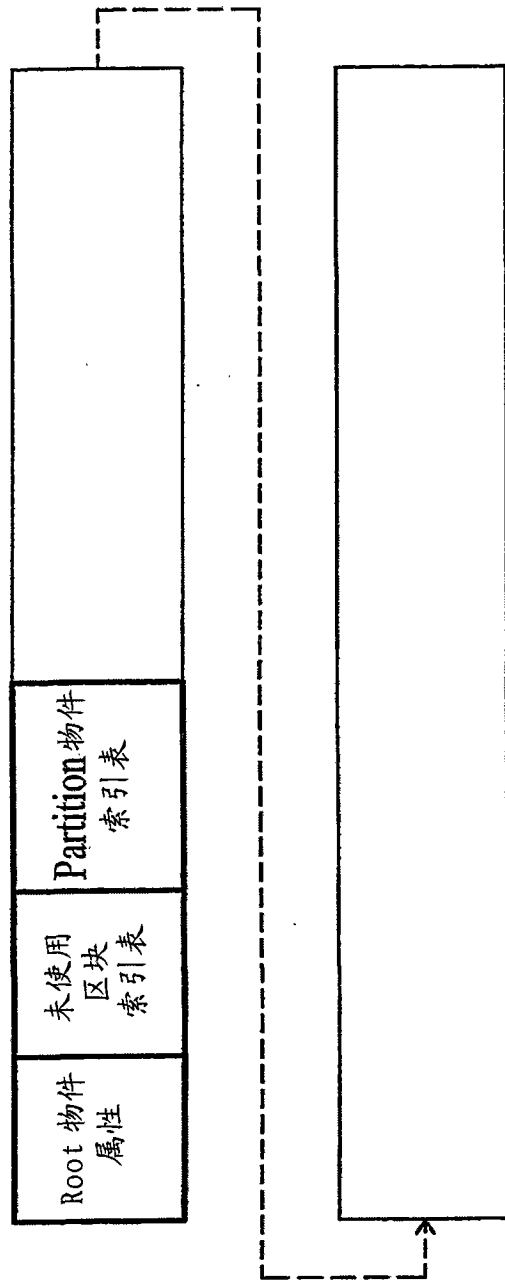


图 9

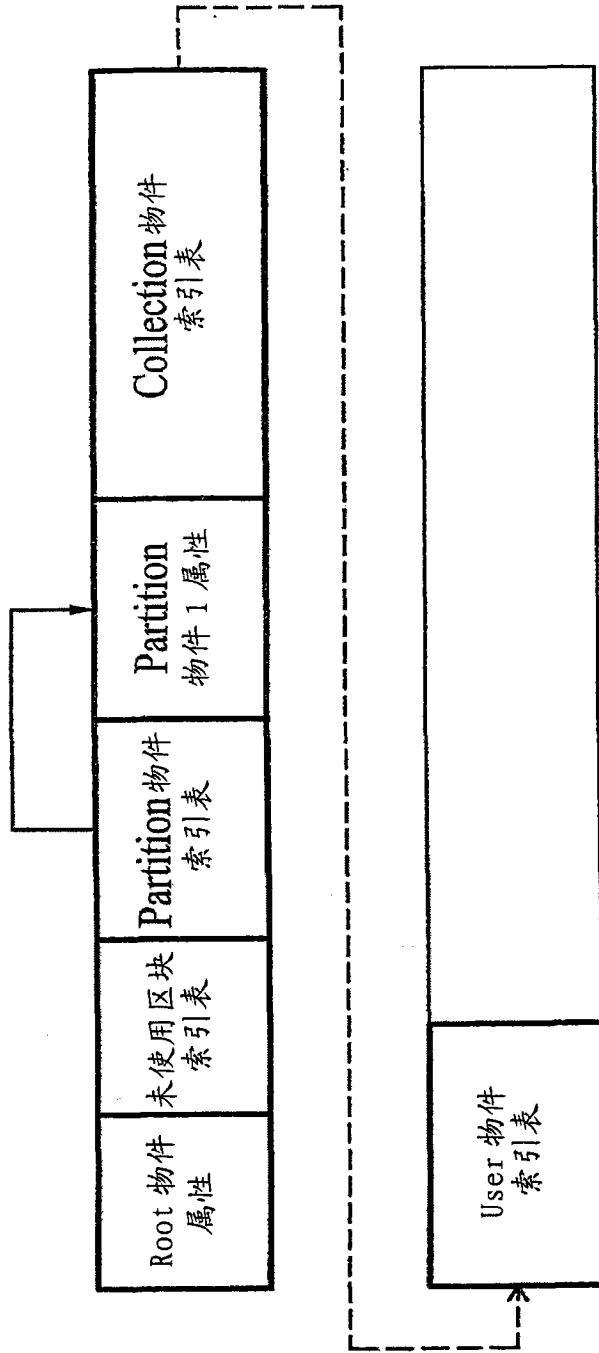


图 10

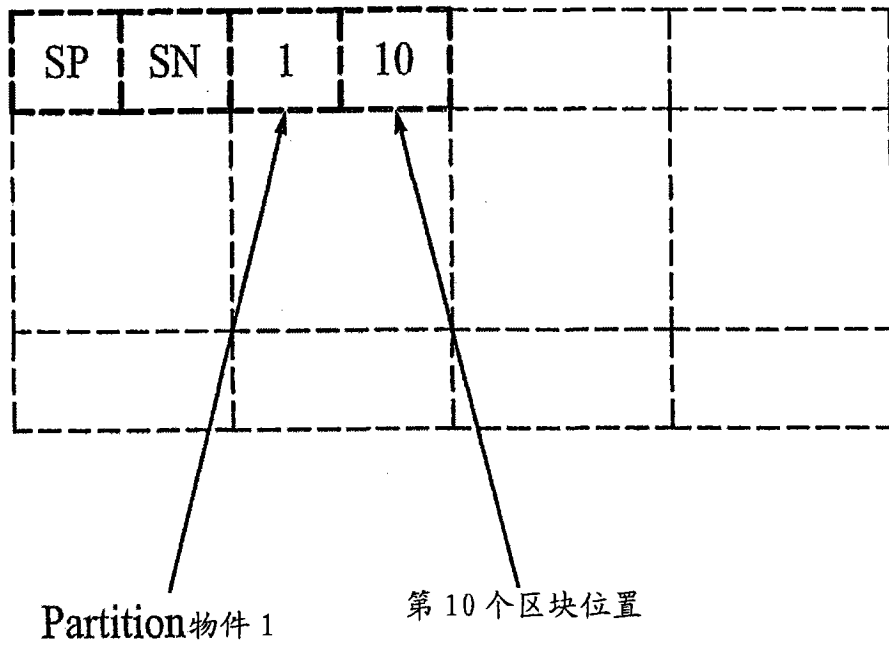


图 11

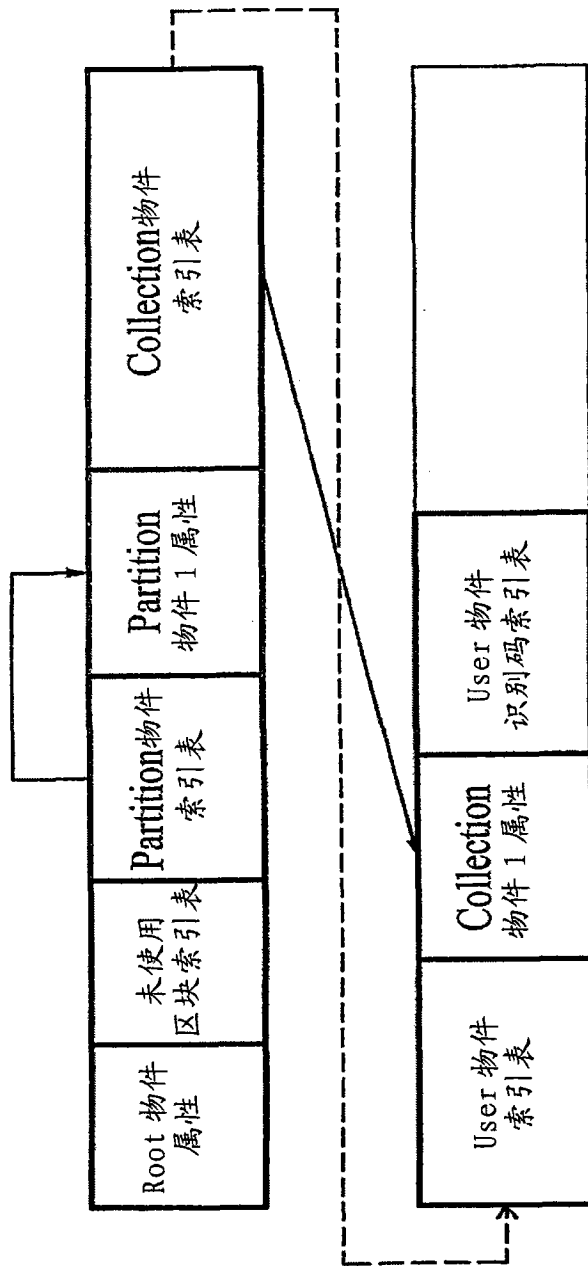


图 12

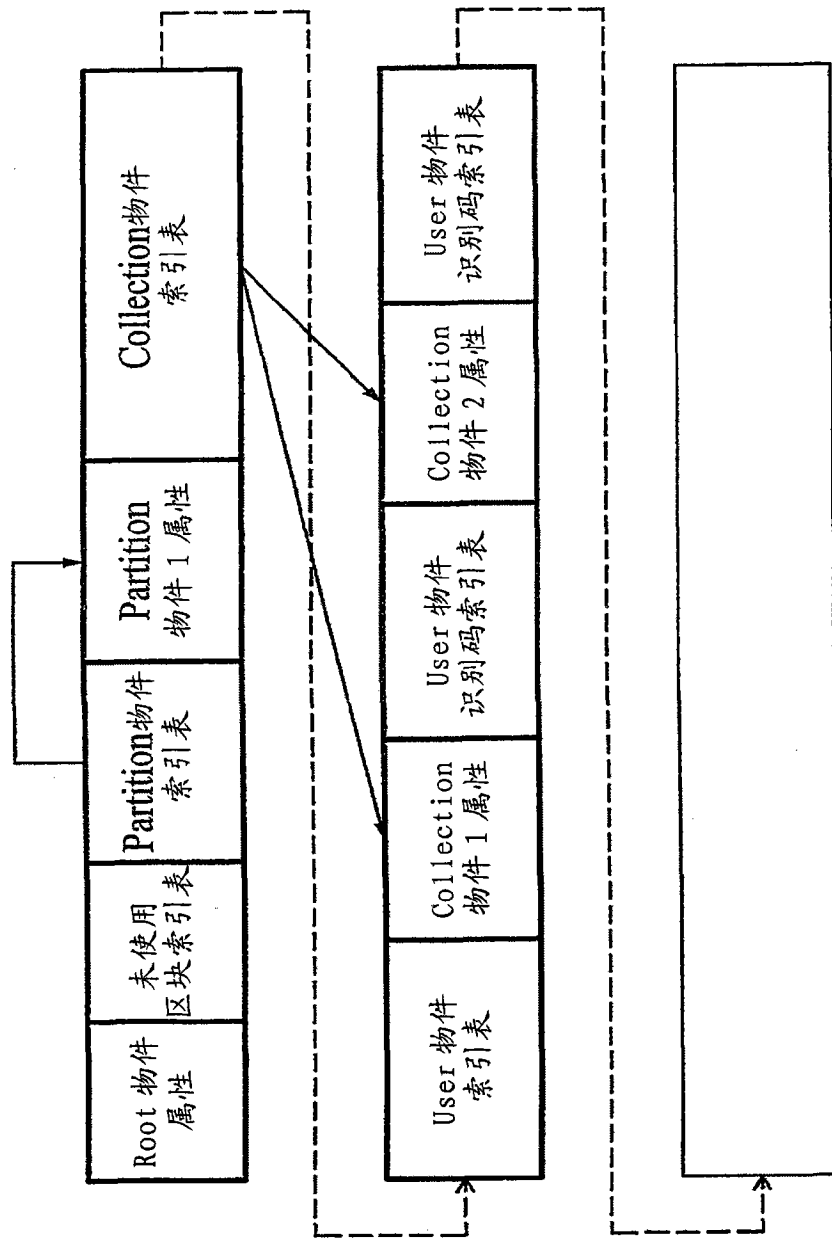


图 13

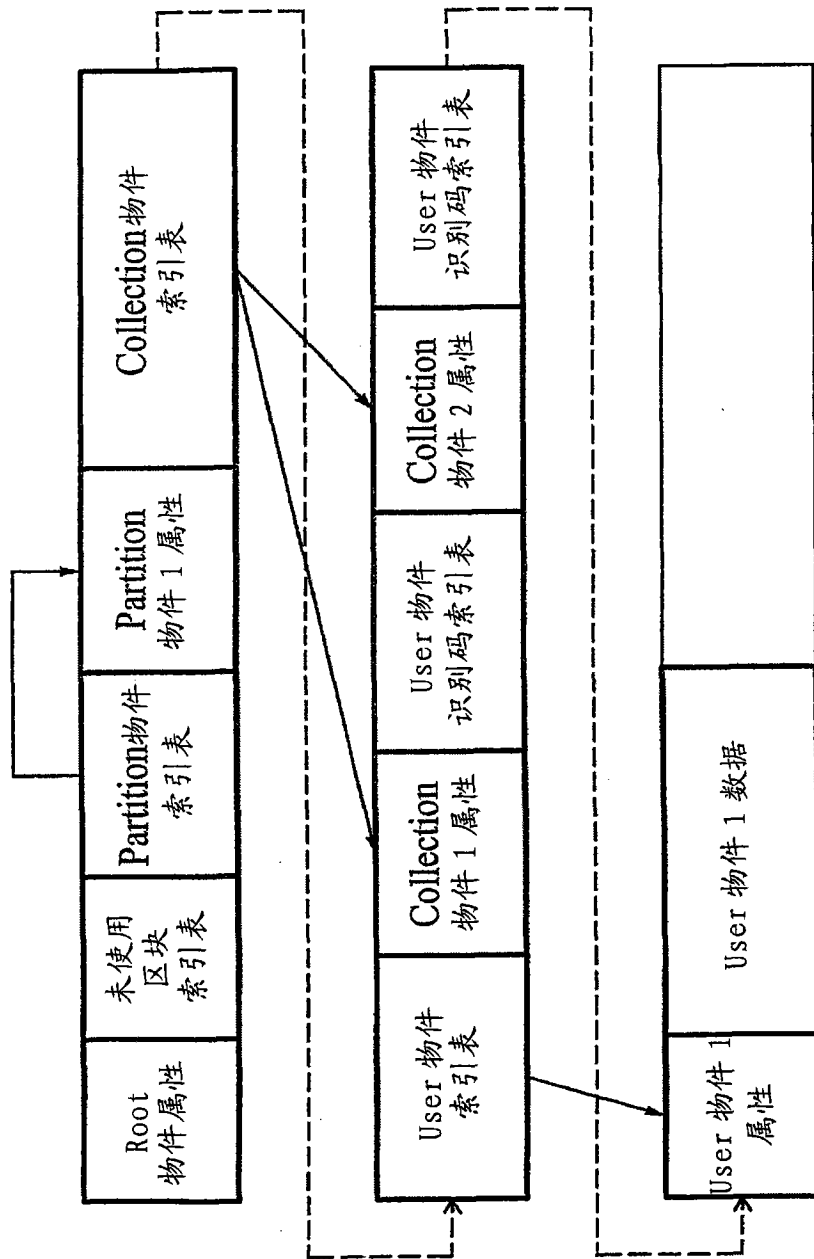


图 14