(19)中华人民共和国国家知识产权局

(12)发明专利

(10)授权公告号 CN 104813321 B
(45)授权公告日 2018.02.06

(21)申请号 201380062151.3

(22)申请日 2013.02.27

(65)同一申请的已公布的文献号
申请公布号 CN 104813321 A

(43)申请公布日 2015.07.29

(85)PCT国际申请进入国家阶段日
2015.05.28

(86)PCT国际申请的申请数据
PCT/US2013/027870 2013.02.27

(87)PCT国际申请的公布数据
WO2014/133497 EN 2014.09.04

(73)专利权人 日立数据系统有限公司
地址 美国加利福尼亚州

(72)发明人 维塔利·佐罗茨基
凯文·斯科特·格里马迪
本杰明·伊舍伍德

(74)专利代理机构 北京银龙知识产权代理有限公司 11243
代理人 范胜杰 杨继平

(51)Int.Cl.
G06F 17/30(2006.01)
G06F 12/06(2006.01)

(56)对比文件
US 2012150930 A1,2012.06.14,说明书第2,8,52,53段,图1.
US 2011055178 A1,2011.03.03,说明书82-96,113,152-154段,图7-10,15.
US 2012173596 A1,2012.07.05,说明书64-66段,图2.
CN 102667748 A,2012.09.12,全文.
US 2007294310 A1,2007.12.20,说明书第10,55段,图3.

审查员 杨佳玉

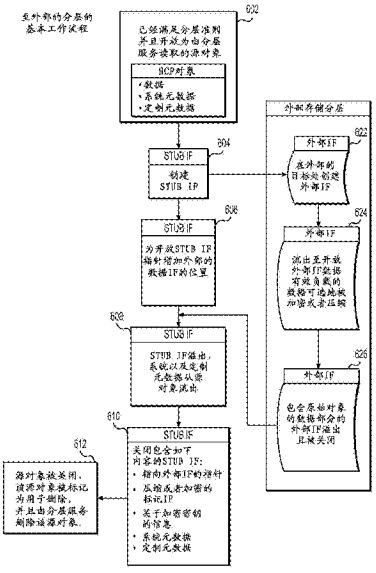
权利要求书2页 说明书16页 附图7页

(54)发明名称

在分布式对象存储生态系统中的去耦合的内容以及元数据

(57)摘要

一种存储系统包括:控制器;存储器;一个或者多个对象,每个对象具有内容数据以及包括系统元数据的元数据;以及策略,其管控一个或者多个对象的内容数据以及元数据从存储系统到外部存储的写入。所述策略包括可配置的准则和规则,所述可配置的准则用于去耦合给定对象的内容数据以及元数据,并且所述规则用于如果对象的内容数据和元数据是去耦合的,则以去耦合的方式在存储系统以及外部存储之间为内容数据和元数据确定存储位置。所述可配置的准则基于对象的元数据的属性。控制器能够操作为针对对象执行策略,并应用所述规则以便为去耦合对象的内容数据和元数据确定存储位置。



CN 104813321 B

1.一种存储系统,所述存储系统耦接至另一存储系统,所述存储系统包括:

存储第一对象的数据的一个或多个存储设备,所述第一对象具有第一内容数据和第一元数据;以及

处理器,其用于:

将所述第一对象复制至所述另一存储系统,并且

在验证了通过复制第一对象将被复制的第一对象的被复制的第一内容数据存储在所述另一存储系统,并且所述另一存储系统并没有设置来存储被复制的第一元数据而不存储被复制的第一内容数据之后,标记将从所述一个或多个存储设备中删除的第一对象的第一内容数据。

2.根据权利要求1所述的存储系统,其中,

所述处理器还用于通过标记第一内容数据来记录被复制的第一内容数据在另一存储系统中的位置。

3.根据权利要求1所述的存储系统,其中,

在获取第一对象至所述存储系统后经过第一量的时间之后,所述处理器还用于验证所复制的第一内容数据被存储在所述另一存储系统,并且验证所述另一存储系统并没有设置来存储被复制的第一元数据而不存储被复制的第一内容数据。

4.根据权利要求1所述的存储系统,其中,

所述处理器还用于从所述一个或多个存储设备中删除被标记的第一内容数据。

5.根据权利要求4所述的存储系统,其中,

所述处理器还用于在收到所述第一内容数据的读取请求时,从所述另一存储系统获取被复制的第一内容数据并保留所获取的被复制的第一内容数据长达第二量的时间。

6.根据权利要求5所述的存储系统,其中,

所述处理器还用于在已经经过所述第二量的时间之后,从所述一个或多个存储设备中删除所获取的被复制的第一内容数据。

7.根据权利要求4所述的存储系统,其中,

所述处理器还用于,在收到所述第一元数据的读取请求时,返回存储在所述一个或多个存储设备中的所述第一元数据而无需访问另一存储系统。

8.根据权利要求1所述的存储系统,其中,

所述一个或多个存储设备存储第二对象的数据,所述第二对象具有第二内容数据和第二元数据,并且

所述处理器还用于在验证了所述另一存储系统已经设置来存储被复制的第二元数据而不存储第二内容数据,并且所述存储系统并没有设置来存储第二元数据而不存储第二内容数据之后,将所述第二元数据发送至所述另一存储系统而不发送所述第二内容数据。

9.根据权利要求8所述的存储系统,其中,

所述处理器还用于发送第二元数据以及所述第二内容数据在所述存储系统中的位置信息。

10.根据权利要求8所述的存储系统,其中,

在获取第一对象至所述存储系统后经过第一量的时间之后,所述处理器还用于验证所述另一存储系统已经设置来存储被复制的第二元数据而不存储第二内容数据。

11.一种用于耦接至另一存储系统的存储系统的方法，所述方法包括：

管理第一对象的数据，所述第一对象具有第一内容数据和第一元数据；

将所述第一对象复制至另一存储系统；以及

在验证了通过复制第一对象将被复制的第一对象的被复制的第一内容数据存储在所述另一存储系统，并且所述另一存储系统并没有设置来存储被复制的第一元数据而不存储被复制的第一内容数据之后，标记将从一个或多个存储设备中删除的第一对象的第一内容数据。

12.根据权利要求11所述的方法，还包括：

通过标记第一内容数据，记录被复制的第一内容数据在另一存储系统中的位置。

13.根据权利要求11所述的方法，还包括：

在获取第一对象至所述存储系统后经过第一量的时间之后，验证所复制的第一内容数据被存储在所述另一存储系统，并且验证所述另一存储系统并没有设置来存储被复制的第一元数据而不存储被复制的第一内容数据。

14.根据权利要求11所述的方法，还包括：

从所述一个或多个存储设备中删除被标记的第一内容数据。

15.根据权利要求14所述的方法，还包括：

在收到所述第一内容数据的读取请求时，从所述另一存储系统获取被复制的第一内容数据并保留所获取的被复制的第一内容数据长达第二量的时间。

16.根据权利要求15所述的方法，还包括：

在已经经过所述第二量的时间之后，从所述一个或多个存储设备中删除所获取的被复制的第一内容数据。

17.根据权利要求14所述的方法，还包括：

在收到所述第一元数据的读取请求时，返回存储在所述一个或多个存储设备中的所述第一元数据而无需访问另一存储系统。

18.根据权利要求11所述的方法，还包括：

管理第二对象的数据，所述第二对象具有第二内容数据和第二元数据；以及

在验证了所述另一存储系统已经设置来存储被复制的第二元数据而不存储第二内容数据并且所述存储系统并没有设置来存储第二元数据而不存储第二内容数据之后，将所述第二元数据发送至所述另一存储系统而不发送所述第二内容数据。

19.根据权利要求18所述的方法，还包括：

发送第二元数据以及所述第二内容数据在所述存储系统中的位置信息。

20.根据权利要求18所述的方法，还包括：

在获取所述第一对象至所述存储系统后经过第一量的时间之后，验证所述另一存储系统已经设置来存储被复制的第二元数据而不存储第二内容数据。

在分布式对象存储生态系统中的去耦合的内容以及元数据

背景技术

[0001] 本发明总体上涉及存储系统，并且尤其涉及在分布式对象存储生态系统中用于存储的内容以及元数据的去耦合。

[0002] 随着无结构的非易变的数字内容的增长，越来越难以管理以及定位有关的数字内容。为了寻找有关的内容，关于数字内容的数据(即元数据)正变得比数字内容本身更加重要。对于传统的分布式环境中的对象存储系统(分布式对象存储生态系统)，数字内容以及元数据被一起存储在多个位置，以实现灾难恢复以及本地的引用(locality of reference)。这是通过利用复制技术(replication technology)来实现的，以确保复制件(拷贝)被分布至远程站点(remote sites)。传统系统的另一个缺点是它们对数据以及元数据应用相同的存储规则。例如，如果对象被存储于低延迟存储系统上的N个复制件中，则所有的N个复制件均包含数据以及元数据，即使在某些位置/应用中可能根本不需要数据部分。

[0003] 因为企业正变得越来越地理性地分散有许多单独的办公室甚至数据中心，为所有的数字内容提供本地的引用所需的复制技术变得更加复杂，并且数字内容的存储需要被增加了所需的本地引用的次数倍。当主要目标是仅仅为元数据具有本地的引用时，这意味着对于元数据的数字内容被不必要地存储在多个位置。数据复制也是非常消耗时间的过程，并且数据以及元数据的复制造成了大量的时间延迟、复制积压(backlog)以及不必要的带宽消耗。

发明内容

[0004] 本发明的示例性实施例提供了一种智能对象，其利用用户可定义的规则及其智能来识别在分布式对象存储系统中的哪个对象应当维持非易变的数字内容以及元数据的复制件，以及哪个对象应当仅仅包含元数据，同时仍然维持在整个环境的对数字内容的可访问性、数据保护以及灾难恢复能力。在数字内容被仅仅托管元数据的任意对象存储系统所需的情况下，它可以从其他的对象存储系统取回，就好像本地复制件总是可用一样。通过该方案，远程位置可以托管较小的容量对象存储系统，该对象存储系统将仅仅存储元数据，但提供对在托管在中央数据中心的较大对象存储系统中存储的数字内容以及元数据的集合的完整的访问。该方案用相同的数据访问提供更好的容量利用。内容获取(ingest)可以经由远程位置或者中央数据中心来完成，但当中央数据中心已经存储数字内容时，远程位置将仍然仅仅维持内容的仅元数据视图(view)。值得注意的是，本发明并不限制仅元数据的配置在哪里是被允许的。它可以在为之获取内容的系统或者在可能具有向其复制的内容的一些其它系统。本发明确保了在生态系统中的某处有至少一个复制件。

[0005] 用于从对象存储系统中的元数据分离数字内容的管理生命周期的方法也可以被用在非复制环境中。元数据将保持在由对象存储系统管理的存储中的活性地驻留，但是基于存储分层策略，数字内容或者内容的额外的复制件也可能被存储在另一个联网的存储设备上。存储在对象存储系统中的或者是在不同的联网的存储系统上的数字内容将仍然由包

含对象的元数据的任意对象存储系统通过将对象从其自身(如果驻留的话)或者从其它联网的存储设备取回来进行管理以及可访问。

[0006] 该机制创建智能内容分层,其有助于对象存储系统的数据保护层级、高效更新以及对象的定制元数据和系统的索引,同时还提供压缩和加密移动至网络存储设备的数据的选项。

[0007] 根据本发明的一个方面,一种存储系统包括:控制器;存储器;一个或者多个对象,每个对象具有内容数据以及包括系统元数据的元数据;以及策略,其管控一个或者多个对象的内容数据以及元数据从存储系统到外部存储的写入。所述策略包括可配置的准则和规则,所述可配置的准则用于去耦合给定对象的内容数据以及元数据,并且所述规则用于如果对象的内容数据和元数据是去耦合的,则以去耦合的方式在存储系统以及外部存储之间为内容数据和元数据确定存储位置。所述可配置的准则基于对象的元数据的属性。控制器能够操作为对存储系统的一个或者多个对象执行策略,包括基于所述可配置的准则将每个对象的元数据评估为将以去耦合的方式写入的候选对象;并且当对于去耦合对象,内容数据和元数据为去耦合时,应用所述规则以便为去耦合对象的内容数据和元数据确定存储位置。

[0008] 在有些实施例中,所述可配置的准则是基于对象的系统元数据的属性。执行策略包括基于所述可配置的准则将每个对象的系统元数据评估为将以去耦合的方式写入的候选对象。所述控制器能够操作为周期性地运行可调度的服务,以将来自存储系统的对象识别为将以去耦合的方式写入的候选对象,并且对识别的对象执行策略。

[0009] 在具体实施例中,所述控制器能够操作为:当策略的规则确定用于内容数据的存储位置是外部存储时,验证对象被写入外部存储,验证内容数据被存储于外部存储,并且当验证了二者时,从存储系统中移除所述内容数据并且在存储系统中记录所述外部存储是被写入的对象的内容数据的位置;并且当策略的规则确定用于内容数据的存储位置是存储系统时,验证对象将作为仅元数据被写入外部存储,验证对象的内容数据被存储于存储系统,并且当验证了二者时,将对象的元数据写入外部存储,并且将指针发送至外部存储,所述指针指向对象的内容数据在存储系统中的位置。

[0010] 在一些实施例中,当策略的规则确定用于内容数据的存储位置是外部存储时,所述控制器能够操作为:在由策略的规则所指定的第一预设量的时间之后,从存储系统移除所述内容数据,所述第一预设量的时间等于或大于零;并且在从外部存储取回所述内容数据之后,在存储系统中将取回的内容数据保留长达由策略的规则所指定的第二预设量的时间,所述第二预设量的时间等于或者大于零。

[0011] 在具体实施例中,所述外部存储位于另一个存储系统,并且所述存储系统和所述另一个存储系统是联网在一起的多个存储系统的一部分,以便在复制的环境中在存储系统之间复制对象。

[0012] 本发明的另一个方面提供了一种用于在存储系统之间管理对象的写入的装置,其中每个对象具有内容数据以及包括系统元数据的元数据。所述装置包括控制器和存储器。所述控制器能够操作为:对源存储系统的一个或多个对象执行策略,所述策略管控一个或者多个对象的内容数据以及元数据从源存储系统至目标存储系统的写入,所述策略包括可配置的准则和规则,所述可配置的准则用于去耦合给定对象的内容数据以及元数据,并且

所述规则用于如果对象的内容数据和元数据是去耦合的，则以去耦合的方式在源存储系统和目标存储系统之间为内容数据和元数据确定存储位置，其中所述可配置的准则基于对象的元数据的属性；其中，对一个或多个对象执行策略包括基于所述可配置的准则将每个对象的元数据评估为将以去耦合的方式写入的候选对象；并且当对于去耦合对象，内容数据和元数据为去耦合的时，应用所述规则以为去耦合对象的内容数据和元数据确定存储位置。

[0013] 本发明的另一个方面提供了一种管理对象的写入的方法，其中每个对象具有内容数据以及包括系统元数据的元数据。所述方法包括：对存储系统的一个或多个对象执行策略，所述策略管控一个或者多个对象的内容数据以及元数据从存储系统到外部存储的写入，所述策略包括可配置的准则和规则，所述可配置的准则用于去耦合给定对象的内容数据以及元数据，并且所述规则用于如果对象的内容数据和元数据是去耦合的，则以去耦合的方式在存储系统以及外部存储之间为内容数据和元数据确定存储位置，其中所述可配置的准则基于对象的元数据的属性；其中，对一个或者多个对象执行策略包括基于所述可配置的准则将每个对象的元数据评估为将以去耦合的方式写入的候选对象；并且当对于去耦合对象，内容数据和元数据为去耦合的时，应用所述规则以为去耦合对象的内容数据和元数据确定存储位置。

[0014] 考虑到下面的具体实施例的具体实施方式，本发明的这些以及其他特征和优点对本领域技术人员将变得明显。

附图说明

[0015] 图1是在其中可以应用本发明的方法以及装置的固定内容存储归档的简化的框图。

[0016] 图2是独立节点的冗余阵列的简化表示，其中每个独立节点都是对称的并且支持归档集群应用。

[0017] 图3是在给定的节点上执行的归档集群应用的各种组件的高层级表示。

[0018] 图4说明了在集群的给定的节点上的元数据管理系统的组件的示例。

[0019] 图5示出了说明用于使用情况(Use Case)1的分层服务处理的流程图的示例，其中，使用情况1具有为复制拓扑中的源系统上的对象定义的仅元数据特征(metadata only feature)。

[0020] 图6示出了说明用于外部分层以仅本地的存储元数据以及在外部卷上存储固定的数字内容的分层服务处理的流程图的示例。

[0021] 图7示出了用于实施分层服务的装置的示例。

具体实施方式

[0022] 在本发明接下来的详细的说明中，请参照附图，其中，附图形成本公开文本的一部分，并且在附图中所示的是说明性的而非限制性的，示例性实施例通过附图可以实施本发明。在附图中，在若干视图中，相似的附图标记描绘基本相似的组件。此外，值得注意的是，尽管具体实施方式部分提供了各种示例性实施例，正如下面所述的以及在图中所说明的，本发明并不限于本文所述以及所说明的实施例，而是可以扩展至本领域技术人员应当知道

的或者将会知道的其它实施例。在说明书中引用的“一个实施例”、“该实施例”或者“这些实施例”意味着与实施例连接的所描述的具体的特征、结构或者特性包括在至少一个本发明的实施例中，并且这些术语在说明书中各种地方的出现并不必然都指相同的实施例。此外，在接下来的具体实施方式中，阐述了许多具体细节以提供本发明的透彻的了解。然而，对于本领域技术人员而言很明显地，这些具体的细节可能并不全都是实践本发明所必须的。在其它的情况下，已知的结构、材料、电路、处理以及接口没有进行详细地描述，和/或可能以框图形式进行了说明，从而并非不必要地使得本发明不清楚。

[0023] 此外，接下来的一部分具体实施方式被呈现为在计算机内运行的符号表示和算法的形式。这些算法的描述以及符号表示是数据处理领域技术人员用于向本领域其他技术人员最有效地传达他们的创新的本质的手段。算法是导向理想的最终状态或结果的一系列定义的步骤。在本发明中，所执行的步骤需要用于实现有形的结果的有形的(tangible)量的物理操作。通常，虽然是非必要的，这些量采用能够被存储、转换、组合、比较以及其它方式的操纵的电的或者磁的信号或者指令的形式。将这些信号称为比特(bit)、值、元件，符号(symbol)、字符(character)、项(terms)、数字/号码(number)、指令等已被多次证明是方便的，主要是因为共同使用的原因。然而，应该牢记的是，所有这些以及相似的术语都与合适的物理量相关联，并且仅仅是应用于这些量的方便的标签。除非特别说明，否则从接下来的讨论很明显地，在整个描述中，使用术语，诸如“处理(processing)”、“计算(computing)”、“计算(calculating)”、“确定(determining)”、“显示(displaying)”等等的讨论，可以包括计算机系统或者其它信息处理设备的动作以及处理，其中，所述计算机系统或者其它信息处理设备将表示计算机系统的寄存器和存储器中的物理的(电子的)量的数据操纵和转换为类似地表示计算机系统的存储器或者寄存器或者其它信息存储、传输或者显示设备内的物理量的其它数据。

[0024] 本发明还涉及一种装置，用于执行本文中的操作。该装置可以根据所需的目的进行具体构造，或者它可以包括通过一个或者多个计算机程序选择性地激活的或者重新配置的一个或者多个通用的计算机。这样的计算机程序可以被存储在计算机可读存储介质中，该计算机可读存储介质包括非瞬态介质，诸如但是并不限于光盘、磁盘、只读存储器、随机存取存储器、固态设备以及驱动器，或者适于存储电子信息的任何其它类型的介质。本文所示的算法以及显示并不固有地与任意特定地计算机或者其它装置相关。可以使用具有根据本文所教示的程序以及模块的各种通用系统，或者可以证明构建更加定制的装置以执行所需的方法步骤是方便的。另外，并没有参考任何特定的编程语言来描述本发明。应当理解的是，各种编程语言可以被用于实施如本文所描述的本发明的教示。编程语言的指令可以通过一个或者多个处理设备例如中央处理器(CPU)、处理器或者控制器来执行。

[0025] 如下面将要更详细地描述的那样，本发明的示例性实施例提供用于去耦合在分布式对象存储生态系统中存储的内容以及元数据的装置、方法以及计算机程序。

[0026] I.固定内容分布式数据存储

[0027] 替代或者补充传统磁带以及光学存储方案的以高度可用、可靠以及持久的方式档案存储“固定内容”的需要已经得到发展。术语“固定内容”通常是指预计将保持不变而不因为引用或者其它目的而改变的任意类型的数字信息。这样的固定内容的例子其中包括电子邮件、文档、诊断图像、检查图像、录音、电影以及视频等。传统的独立节点冗余阵列(RAIN)

存储方法已经出现作为用于创建大型在线归档的选择的架构，该大型在线归档用于存储这样的固定内容信息资产。通过允许节点根据需要加入以及退出集群，RAIN架构将存储集群从一个或者多个节点的故障中隔绝。通过在多个节点上复制数据，RAIN类型的归档可以自动地补偿节点故障或者移除。通常，RAIN系统被较大地交付为从封闭系统内的相同组件设计的硬件装置。

[0028] 图1说明了一个这样的可扩展的基于磁盘的归档存储管理系统。节点可以包括不同的硬件并且因此可以被考虑为“不对称的/异构的(heterogeneous)”。节点通常具有对一个或者多个存储盘的访问，该盘可以是存储区域网络(SAN)中的实际的物理存储盘或者虚拟的存储盘。在各个节点上支持的归档集群应用(并且，可选地，在其上应用执行的底层操作系统)可以是相同的或者基本上相同的。在各个节点上的软件堆栈(其可以包括操作系统)是对称的，然而硬件可以是不对称的。使用该系统，如图1所示，企业可以为很多不同类型的固定内容信息，诸如文档、电子邮件、卫星图像、诊断图像、检查图像、录音，视频等创建永久存储。当然，这些类型仅是说明性的。高等级的可靠性是通过在独立服务器或者所谓的存储节点上复制数据来实现的。优选地，每个节点与其同伴是对称的。因此，由于优选地任意给定的节点可以执行所有的功能，所以任意一个节点的故障对归档的可用性具有很小的影响。

[0029] 如共同拥有的美国专利No.7,155,466所描述的，在基于RAIN的归档系统中合并在捕获(capture)、保存(preserve)、管理以及取回数字资产的每个节点上执行的分布式软件应用是已知的。图2说明了这样的一个系统。单独的归档的物理边界是指集群(或者系统)。通常，集群不是单一设备，而是设备的集合(collection)。设备可以是同质化的/对称的(homogeneous)或者异质化的/不对称的。典型的设备是计算机或者运行诸如Linux的操作系统的机器。在商用硬件上托管的基于Linux的系统的集群提供归档，该归档可以从一些存储节点服务器被调整至存储成千上万的百万兆字节的数据的很多节点。该架构确保存储容量总是能够跟上组织的增加的归档需要。

[0030] 在诸如上述的存储系统中，数据通常随机地分布于集群，从而使得归档总是受到保护以免于设备故障。如果盘或者节点故障，则集群自动地移至维持相同的数据的复制的、集群中的其它节点。尽管该方法出于数据保护的观点工作良好，但是计算出的对于集群的数据损失的平均时间(MTDL)可能还没有要求的那么高。具体地，MTDL通常表示在归档将要损失数据之前的时间的计算量。在数字归档中，任意数据损失都是不期望的，但是由于硬件以及软件组件的本性，这样的出现总是存在可能性(尽管很少)。由于对象及其复制件在归档集群内的随机分布，MTDL可能低于所需地结束，例如因为如果在给定的节点内的给定的盘(在其上存储镜像复制件)非预期地发生故障，则所需要的对象的复制件可能是不可用的。

[0031] 如图2所示，在其中实施了本发明的说明性的集群优选包括接下来的一般类别的组件：节点202，一对网络开关204，电源分配单元(PDU)206，以及不间断电源(UPS)208。节点202通常包括一个或者多个商用服务器并且包含CPU(例如，CPU，合适的随机存取存储器(RAM)，一个或者多个硬盘驱动器(例如，标准IDE/SATA，SCSI等等)，以及两个或者更多网络接口(NIC)卡。典型的节点是具有2.4GHz芯片、512MB RAM的以及六(6)个200GB硬盘驱动器的2U机架安装单元。然而，这并不是限制。网络开关204通常包括内部开关205，其允许节点

之间的点对点通信，以及外部开关207，其允许对每个节点的额外的集群访问。每个开关需要足够的端口以处理集群中的所有的潜在的节点。以太网或者GigE开关可以被用于该目的。PDU 206被用于为所有的节点和开关供电，而UPS 208被用于保护所有的节点和开关。虽然并未意味着限制，通常，集群可以连接至网络，诸如公共互联网，企业内部网，或者其它广域网或者局域网。在说明性的实施例中，集群被实施在企业环境内。它可以例如通过网站的企业域名系统(DNS)域名服务器导航来到达。因此，例如，集群的域可以是存在的域的新的子域。在有代表性的实施中，子域在公司DNS服务器中被委托至集群本身中的名称服务器。最终用户使用任意惯用的接口或者访问工具来访问集群。因此，例如对集群的访问可以通过任意基于IP的协议(HTTP、FTP、NFS、AFS、SMB、Web服务或者其它)经由API(应用程序接口)或者经由任意其它已知的或者后续开发的访问方法、服务、程序或者工具来执行。

[0032] 客户应用通过一个或者多个类型的外部的网关，诸如标准UNIX文件协议或者HTTP API来访问集群。归档优选地通过虚拟的文件系统来显露，该虚拟的文件系统可以可选地被安放于任意标准UNIX文件协议-导向的设施。这些包括NFS、FTP、SMB/CIFS或者其它。

[0033] 在一个实施例中，归档集群应用运行于(例如，经由以太网)联网在一起作为集群的独立节点的冗余阵列(H-RAIN)。给定的节点的硬件可以是异构的。然而，为了最大的可靠性，优选地，每个节点运行分布式应用的实例300(其可以是相同的实例，或者基本相同的实例)，其包括若干运行时间组件，现在如图3所示。因此，尽管硬件可能是异构的，但是节点上的软件堆栈(至少因为其与本发明相关)是相同的。这些软件组件包括网关协议层302、访问层304、文件业务(transaction)以及管理层306、以及内核组件层308。提供称呼“层”是为了说明的目的，本领域技术人员应当理解，功能可以被描述为其它有意义的方式。一个或者多个层(或者其中的组件)可以被集成或处于其他方式。一些组件可以跨层分享。

[0034] 网关协议层302中的网关协议向已有的应用提供透明性。具体地，网关提供本地(native)文件服务诸如NFS 310和SMB/CIFS 312以及Web服务API，以建立定制应用。还提供HTTP支持314。访问层304提供对归档的访问。具体地，根据本发明，固定内容文件系统(FCFS)316仿效本地文件系统，以提供对归档对象的完整的访问。FCFS赋予应用对归档内容的直接访问，就好像它们是通常的文件一样。优选地，归档的内容以其原始格式呈现，同时元数据显露为文件。FCFS 316提供目录以及许可以及例程(routine)文件级调用的常规的视图，从而使得管理者可以以他们熟悉的方式提供固定内容数据。文件访问调用优选地由用户-空间守护程序(daemon)拦截并且安排至(层308中)合适的内核组件，该合适的内核组件向调用应用动态地创建合适的视图。FCFS优选地调用由归档策略进行限制以便于自治的归档管理。因此，在一个实施例中，管理者或者应用不能够删除保存期限仍然有效(给定的策略)的归档对象。

[0035] 访问层304优选地还包括Web用户接口(UI)318以及SNMP网关320。Web用户接口318优选地被实施为管理者控制台(console)，该管理者控制台提供对文件业务以及管理层306中的管理引擎322的交互访问。管理控制台318优选地是密码保护的、基于Web的GUI，该GUI提供包括归档对象以及单独的节点的、归档的动态的视图。SNMP网关320为存储管理应用提供对管理引擎322的简易访问，使得他们能够安全地监视以及控制集群活动。管理引擎监控集群活动，集群活动包括系统以及策略事件。文件业务以及管理层306还包括请求管理处理324。请求管理器324协调(通过访问层304)来自外部世界的所有的请求，以及来自内核组件

层308中的策略管理器326的内部请求。

[0036] 除了策略管理器326之外，内核组件还包括元数据管理器328，以及存储管理器330的一个或者多个实例。元数据管理器328优选地安装于每个节点。共同地，集群中的元数据管理器用作为分布式数据库，管理所有的归档对象。在给定的节点，元数据管理器328管理归档对象的子集，其中优选地，每个对象映射在外部文件(“EF”，进入用于存储的归档的数据)和归档数据被物理定位于其中的一组内部文件(每个“IF”)之间。相同的元数据管理器328还管理从其它节点复制的一组归档对象。因此，每个外部文件的当前状态总是对若干节点上的多个元数据管理器可用。在节点故障的情况下，在其它节点上的元数据管理器继续提供对由故障节点先前管理的数据的访问。存储管理器330提供对分布式应用中的所有的其它组件可用的文件系统层。优选地，它在节点的本地文件系统中存储数据对象。在给定节点的每个驱动优选地具有它自己的存储管理器。这允许节点移除单独的驱动器并且优化总处理能力(throughput)。存储管理器330还提供系统信息、对数据的完整性检查以及直接跨越(traverse)本地结构的能力。

[0037] 仍然如图3所示，集群通过通信中间软件层332和DNS管理器334管理内部以及外部通信。基础设施332是能够在归档组件间通信的、有效且可靠的基于消息的中间软件层。在示例性实施例中，该层支持多点传送(multicast)以及点对点(point-to-point)通信。DNS管理器334运行将所有的节点连接至企业服务器的分布式名称服务。优选地，DNS管理器(单独地或者结合DNS服务)通过所有的节点加载平衡请求以确保最大化集群总处理能力以及可用性。

[0038] 在一示例性实施例中，应用实例在基础操作系统336，诸如Linux等上执行。通信中间软件是任意便利的分布式通信机制。其它组件可以包括FUSE(用户空间中的文件系统)，其可以被用于固定内容文件系统(FCFS)316。NFS网关310可以由标准nfsd Linux内核NFS驱动器来进行实施。可以实施每个节点的数据库，其为对象相关的数据库管理系统(ORDBMS)。节点可以包括Web服务器，诸如Jetty，其为Java HTTP服务器以及小服务程序容器(servlet container)。当然，上述机制仅是说明性的。

[0039] 在给定节点的存储管理器330负责管理物理存储设备。优选地，每个存储管理器实例负责单一的根目录，其中，所有的文件根据其放置算法被放置在该根目录下。多个存储管理器实例可以在相同的时间运行在一个节点，并且每个实例通常表示系统中的不同的物理盘。存储管理器从系统的剩余部分提取正在使用中的驱动器以及接口技术。当存储管理器实例被要求写文件时，该存储管理器实例为它将要负责的表示生成完整的路径以及文件名称。在有代表性的实施例中，将要在存储管理器上存储的每个对象被接收为将要被存储的原始数据，然后，当存储数据时存储管理器将其自己的元数据添加至文件以跟踪不同类型的信息。举例来说，该元数据包括:EF长度(以字节(byte)为单位的外部文件的长度)，IF段(Segment)大小(该块内部文件的大小)，EF保护表示(EF保护模式)，IF保护角色(role)(该内部文件的表示)，EF创建时间戳(外部文件时间戳)，签名(在写入(PUT)时内部文件的签名，包括签名类型)，以及EF文件名称(外部文件文件名称)。与内部文件数据一并存储该额外的元数据提供了额外层级的保护。具体地，清除(scavenging)可以从存储在内部文件中的元数据创建在数据库中的外部文件记录。其它策略可以使关于内部文件的内部文件哈希(hash)生效以使得内部文件保留完整无缺。

[0040] 内部文件可以是数据的“组块”，其表示在归档对象中的原始“文件”的一部分，并且它们可以被放置在不同的节点以实现条纹化(Striping)和保护块(block)。然而，该将外部文件分割为较小的组块的单元并不是一种需要；替代地，内部文件可以是外部文件的完整的复制件。通常，对于每个归档对象在元数据管理器中出现一个外部文件表项(entry)，而对于每个外部文件表项可以有很多内部文件表项。通常，内部文件布局取决于系统。在给定的实施中，盘上的该数据的实际物理格式被存储于一系列的可变的长度记录中。

[0041] 请求管理器324负责通过与系统中的其它组件互动来执行所需的一组操作以执行归档动作。请求管理器支持很多不同的类型的同时的动作，能够回滚(roll-back)任意故障业务，并支持能够花费很长时间来执行的业务。请求管理器还确保归档中的读取/写入操作被适当地处理，并且保证在所有的时间所有的请求都处于已知的状态。它还提供用于协调节点间多个读取/写入操作的业务控制以符合给定的客户请求。另外，请求管理器为最近使用的文件缓存元数据管理器表项，并且为会话(session)以及数据块提供缓冲。

[0042] 集群的主要责任是在盘上可靠地存储无限制的数量的文件。在其可能是不可到达或者否则因为任意原因不可用的情况下，给定的节点可能被认为是“不可靠的”。这样的潜在地不可靠的节点的集合合作创建了可靠以及高度可用的存储。一般地，存在两种类型的需要被存储的信息：文件本身以及关于文件的元数据。固定内容分布式数据存储的额外的细节可以参见美国专利公开No.2007/0189153以及美国专利No.7,657,581，它们通过引用合并于此。

[0043] II.元数据管理

[0044] 元数据管理系统负责组织和提供对给定的元数据诸如系统元数据的访问。该系统元数据包括关于放置在归档中的文件的信息，以及配置信息、显示在管理UI上的信息、度量(metric)、关于不能恢复的(irreparable)策略破坏(violation)的信息等等。虽然没有详细示出，其它类型的元数据(例如，与归档的文件相关联的用户元数据)也可以使用现在描述的元数据管理系统进行管理。

[0045] 在集群的有代表性的实施例中，元数据管理系统为一组元数据对象提供持久性(persistence)，该元数据对象可能包括一个或者多个接下来的对象类型(只是说明性的)：

[0046] ExternalFile(外部文件)：由归档的用户理解的文件；

[0047] InternalFile(内部文件)：由存储管理器存储的文件；典型地，在外部文件和内部文件之间可以存在一对多的关系。

[0048] ConfigObject(配置对象)：用于配置集群的名称/值对；

[0049] AdminLogEntry(管理日志表项)：将要显示在管理者UI上的消息；

[0050] MetricsObject(度量对象)：打时间戳的键/值(key/value)对，表示在时间上某点处的归档的某些测量(例如，文件的数量)；以及

[0051] PolicyState(策略状态)：某些策略的违反。

[0052] 每个元数据对象可能具有唯一的名称，该名称优选地永不改变。元数据对象被组织至区域(region)。区域包括可信的区域复制件以及“故障的可忍受的点”(TPOF)数量(一组零或者更多)备份区域复制件。对于零号复制件，元数据管理系统是可扩展的但是可能不是高度可用的。通过哈希一个或者多个对象属性(例如，对象的名称，诸如完整合格的路径名称，或者其部分)以及提取给定数量的比特的哈希值来选择区域。这些比特包括区域号

码。该选择的比特可能是低阶比特、高阶比特、中阶比特或者单独的比特的任意组合。在有代表性的实施例中，给定的比特是哈希值的低阶比特。可以使用任意便利的哈希函数来哈希对象的属性。这些函数包括但不限于基于Java的哈希函数，诸如java.lang.string.hashCode等等。优选地，包括区域号码的比特的数量由配置参数控制，这里是指regionMapLevel(区域映射等级)。例如，如果该配置参数被设定为6，则产生2^6=64个区域。当然，也允许更大数量的区域，并且可以使用名称空间分区方案自动地调整区域的数量。

[0053] 可以冗余地存储每个区域。如上面提到的，存在区域的一个可信的复制件，以及零个或者更多备份复制件。如已描述的那样，备份复制件的数量由元数据TPOF配置参数来控制。优选地，区域复制件分布于集群的所有的节点从而平衡每个节点的可信的区域复制件的数量，并且平衡每个节点的全部区域复制件的数量。

[0054] 元数据管理系统将元数据对象存储在运行在每个节点的数据库中。该数据库被用于支持区域映射。使用PostgreSQL来实施一示例性数据库，其作为开源可用。优选地，对于每个区域复制件都存在概要(schema)，并且在每个概要中为每个类型的元数据对象存在有表格。概要只是名称空间，该名称空间能够拥有表格、索引、程序以及其它数据库对象。每个区域优选地具有其自己的概要。每个概要具有一完整组的表格，每个表格对应每个元数据对象。这些表格中的一个表格的一行对应于一个单独的元数据对象。

[0055] 这里所用的名称空间是集群的逻辑分区，并且基本上用作为具体对于至少一个定义的应用的对象的集合。每个名称空间具有相对于其它名称空间的私有文件系统。此外，对一个名称空间访问并不授权用户对另一个名称空间访问。节点的集群/系统是物理归档实例。占有者(tenant)是名称空间和可能的其它子占有者的组合。顶级占有者是没有父占有者的占有者，例如，企业。子占有者是其父占有者是另一个占有者的占有者；例如企业的财务部门。缺省占有者是仅仅包含缺省名称空间的顶级占有者。集群/系统是物理归档实例。详见US2011/0106802，其全部通过引用合并于此。

[0056] 如图4所示，每个节点400具有一组处理或组件：一个或者多个区域管理器(RGM)402a-n，元数据管理器(MM)404，至少一个元数据管理器客户端(MMC)406，以及具有一个或者多个概要410a-n的数据库408。RGM、MM以及MMC组件用虚拟机412，诸如Java虚拟机执行。对于每个区域复制件有一个RGM。因此，存在对于可信的区域复制件的RGM、对于每个备份区域复制件的RGM、以及对于每个不完整区域复制件的RGM。还存在对于每个RGM 402的数据库概要410，其用来管理该概要。数据库还存储区域映射405。每个节点优选地具有相同的区域映射的全局视图，具有由同步方案执行的需求。区域管理器RGM 402负责对区域复制件(其可能是可信的、备份的或者不完整的，视情况而定)的操作，并且负责执行由元数据管理器客户端406以及由其它区域管理器402提交的请求。请求通过任意便利的方式，诸如通信中间软件或者图3所示的其它消息层被提供至给定的RGM。区域管理器例如通过提供至数据库的连接，来提供在其中执行这些请求的执行环境，其中，所述数据库被配置为对正由相应的RGM所管理的概要进行操作。每个区域管理器在数据库408存储其数据。元数据管理器404是负责节点上的元数据管理的顶级组件。它负责创建以及毁灭区域管理器(RGM)并且组织由RGM所需的资源，例如集群配置信息以及数据库连接池。优选地，(在给定节点的)给定的元数据管理器发挥领导(leader)作用并且负责确定(贯穿节点集合或者子集的)哪个元数据

管理器负责哪个区域复制件。领导选举算法，诸如bully算法或者其变种，可能被用于选择元数据管理器领导。优选地，每个节点具有单一的元数据管理器，虽然每个节点运行多个MM是可能的。一旦区域所有权已经由名称空间分区方案所建立(如下面将描述的那样)，每个元数据管理器负责相应地调整一个或者多个区域管理器的它的集合。系统组件(例如，管理引擎、策略管理器等等)与元数据管理器MM通过元数据管理器客户端进行交互。MMC负责(使用区域映射)定位RGM以执行给定的请求，负责发布请求至所选择的RGM，以及如果所选择的RGM不可用(因为例如节点已经发生故障)，则负责重试请求。在后者的情况下，当在节点处接收新的区域映射时，重试请求将会成功。

[0057] 如上面提到的，区域映射用于识别可负责每个区域的每个复制件的节点。虚拟机412(以及本文的每个RGM、MM以及MMC组件)具有对区域映射405的访问；区域映射的复制件420在其已经被复制到JVM之后，也示于图4中。因此，区域映射对给定节点的数据库以及JVM都可用。在该说明性的实施例中，每个元数据对象具有属性(例如，名称)，该属性被哈希以生成0x0和0x3fffffff(含)之间的整数，即，30比特值。这些值可以用带符号的32比特整数良好的表示而不会遇到溢出问题(例如，当对范围的最高端加1时)。30比特允许高达接近十亿个区域，这甚至对大集群都足够了。区域表示一组哈希值，而所有区域的集合覆盖了所有可能的哈希值。对于每个区域存在不同的比特位置，并且不同的比特位置优选地处于固定的顺序。因此，每个区域可以通过号码来识别，该号码优选地通过提取哈希值的RegionLevelMap(区域层级映射)比特来得到。当配置参数被设置为6时，允许64个区域，所生成的哈希值是0x0至0x3f的号码。

[0058] 如先前所述的，区域复制件处于三(3)个状态之一："可信的"、"备份的"以及"不完整的"。如果区域复制件是可信的，则对区域的所有请求都前进至该复制件，并且对于每个区域都存在一个可信的复制件。如果区域复制件是备份，则复制件(从可信的区域管理器处理)接收备份请求。如果元数据正在被加载但是复制件(通常，相对于其它备份复制件)尚未被同步，则区域复制件是不完整的。不完整的区域复制件直至同步完成才具有资格以提升至另一个状态，在此时复制件变为备份复制件。每个区域具有一个可信的复制件以及给定的号码(如由元数据TPOF配置参数所设定)个备份或者不完整的复制件。

[0059] 通过执行可信的区域复制件及其TPOF备份复制件之间的给定的协议(或者"合约")保持备份区域复制件与可信的区域复制件之间的同步。现在描述该协议。

[0060] 作为简要的背景，当在MMC接收更新请求时，MMC在本地区域映射进行搜寻以寻找可信的区域复制件的位置。MMC将更新请求发送至与可信的区域复制件相关联的RGM，RGM随后交付更新。更新还被(与可信的区域复制件相关联的RGM)发送至每个TPOF备份复制件的RGM。然而，为了指示成功，可信的RGM不需要等待与备份区域复制件相关联的每个RGM交付更新；而是当与备份区域复制件相关联的RGM接收更新时，它就立即(向可信的RGM)返回或者尝试返回确收。当接收备份请求时且执行前发出该确收。在没有故障出现的情况下，一旦可信的RGM接收所有确收，它通知MMC，MMC随后向调用者返回成功。然而，如果给定的故障事件出现，则协议确保受影响的RGM(无论是备份的还是可信的)将其自身(以及潜在地受影响的节点)从服务移除，并且由MM领导发布新的区域映射。优选地，RGM通过降低JVM来从将其自身从服务移除，虽然也可以使用任意便利的技术。新的映射为丢失的区域复制件的指定替代。在该方式下，每个备份区域复制件是可信的区域复制件的"热备份(hotstandby)"，并

且因此在如果需要和当需要(或者因为可信的RGM故障,为了负载平衡的目的,或者其他目的)时有资格提升至可信的。

[0061] 更新处理可能失败存在若干方式。因此,例如,可信的区域管理器(在等待确收的同时)可能遭遇例外,该例外指示备份管理器处理已经停止工作,或者备份管理器处理可能失败于本地的处理更新请求,即使它已经发出了确收,或者备份区域管理器处理在发布确收的同时可能遭遇例外,该例外指示可信的区域管理器处理已经停止工作等等。如上面提到的,如果给定的备份RGM不能处理更新,则它将自身从服务移除。此外,当备份RGM或者可信的RGM停止工作时,发布新的区域映射。

[0062] 元数据管理系统保持被同步的区域的复制件。对可信的区域复制件中的对象完成的更新被复制在备份区域复制件上。一旦由可信的RGM交付更新,则相同的更新被应用于所有的备份区域复制件。元数据管理系统确保任意这样的故障(无论在节点层级,区域管理器层级或者其它)导致故障节点上的区域复制件的再分配(reassignment);因此,保证了剩余区域复制件的完整性。如果包含可信的RGM的节点发生故障,则备份RGM或者处于同步(具有或者不具有当前执行的更新),或者它们仅仅与中断的更新不同步。在后者的情况下,重新同步是简易的。因为备份区域被保持与可信的区域同步,(从备份至可信的)提升是即时的。

[0063] 节点故障还可能丢失备份区域。通过在某些其它节点创建新的、不完整的区域来恢复备份区域。一旦创建了不完整的区域,它就开始记录更新并且开始从可信的区域复制数据。当复制完成时,应用累积的更新,以生成最新的(up-to-date)备份。然后,新的备份区域告知MM领导它已经最新,这将使得MM领导发出包括区域的提升(从不完整至备份)的映射。

[0064] 值得注意的是,并不要求区域的数量对应于节点的数量。更一般地,区域的数量与独立节点的阵列中的节点的数量无关。元数据管理的额外的细节可以在美国专利No.7,657,581中找到。

[0065] III.具有去耦合的内容和元数据的智能内容分层

[0066] 分离静态数字内容(即,固定内容数据)及其元数据的管理生命周期的方法是智能内容分层的该特性的核心。该方法使得分布式存储系统能够智能地决定哪个单独的存储系统应当包含静态数字内容及其元数据一起,以及哪个单独的存储系统应当仅包含数字内容的元数据。

[0067] 有两个主要的实施组成部分。第一,赋予系统的用户创建一组规则的能力,其中,所述规则将会为静态数字内容及其元数据管控与存储位置相关的对象的行为。该能力是通过创建“服务计划(Service Plan)”或者策略并且将其分配至对象的集合来提供的。第二,可调度(schedulable)服务(例如,存储分层服务)周期性地运行以识别候选对象,对所选择的候选集合执行配置的服务计划,并且应用用户所定义的规则。

[0068] 在服务计划或者策略中,用于为对象从元数据中去耦合内容的可配置的准则是基于该对象的元数据(更具体地,在具体实施例中为系统元数据)的某些属性。系统元数据是关于对象的核心信息,诸如创建日期、大小、复制件的数量、其是否已经被复制等等。准则的例子包括是否对象已经被复制至另一个系统,在给定的内容平台系统内是否存在对象的另一个复制件,内容已经在一定量的时间内没有被访问等等。因此,可以至少部分基于将要被复制的对象的系统元数据的属性来设定策略。为对候选对象执行策略,存储分层服务基于

可配置的准则评估候选对象的系统元数据。

[0069] 利用用于分离系统中使用的数据以及元数据生命周期的服务计划，使用该实施具有两种主要的使用情况：(1)包括多个存储系统的复制环境，以及(2)具有外部存储卷的单独的存储系统。

[0070] III.A复制环境

[0071] 为了利用多个系统提供数据保护，复制环境可以具有多个系统。复制环境包括关于数据内容的源系统及目标系统。在简单复制拓扑中，将有一个源系统，但是潜在的多个目标系统。对于更完整链式复制技术，什么是源以及什么是目标将取决于拓扑中的上下文环境(context)。例如，在串联构件系统的3-系统链式拓扑中，在链的末端的系统将在链的中间的系统考虑作为它的源。此外，在链的中间的系统将链中的第一个系统考虑作为它的源。

[0072] 在复制的环境中，本发明的一个目的是对于对象的静态数字内容以及元数据部分的智能生命周期管理：(a)最小化数据存储中的冗余；以及(b)最小化非必须的数据传输。可以在源系统或者目标系统定义服务计划以定义下面的行为：(a)在自从获取后已经经过T1量长的时间之后仅保持元数据；以及(b)在读取时，恢复(rehydrate)对象并且保持T2量长的时间。

[0073] 当进行向对象的请求时，服务用户读取向对象的请求包括：

[0074] 1.如果用户仅仅请求对象元数据的部分，则其被从本地取回并且提供服务至用户而无需前往远程系统。

[0075] 2.如果在本地不具有对象的固定数字内容的系统上需要对象的固定数字内容，则该数字内容将从源或者目标透明地取回，并返回至用户。

[0076] 3.如果值T2(例如，在恢复后保持T2天)被定义了且大于0，则不具有数字数据内容的系统将在本地存储该内容长达T2量的时间。这将使得用户能够本地的取回静态数字内容直至T2时间过期。已经过T2时间之后发生的服务运行将会移除本地静态数字内容。远程内容将根本不必改变。

[0077] 接下来是对于不同的复制拓扑环境的两种使用情况。

[0078] 在使用情况1，针对复制拓扑中的源系统上的对象定义了具有仅元数据特征的服务计划。在获取对象后已经经过T1时间之后在第一服务运行时，分层服务将会：

[0079] 1.验证对象被复制到复制拓扑中的目标系统。

[0080] 2.验证在目标系统上，对象的数据部分被安全地存储了，并且没有被请求在目标系统上被存储为仅元数据。

[0081] 3.如果两个情况都满足，则对象的数据部分被标记为移除，并且数据部分在目标系统的位置被记录在源系统上。

[0082] 在服务计划完成之后，源系统将会在本地存储仅元数据而在复制系统存储数据部分以及元数据。

[0083] 图5示出了说明用于具有针对复制拓扑中的源系统上的对象所定义的仅元数据特征的使用情况1的分层服务处理的流程图的示例。处理开始于源对象，该源对象已经满足分层准则并且开放为由分层服务读取(步骤502)。在步骤504，分层服务确定在复制拓扑中是否存在对象。如果不存在，则处理停止并且对象被关闭(步骤506)。如果存在，则在步骤508，对开放存根内部文件(Open Stub IF)指针增加当前拥有对象的复制拓扑中的集群的UUID

(通用唯一识别码),并且将标记(flag)增加至寻迹精简(track pruning)。在步骤510,Stub IF溢出(flushed),并且系统和定制元数据从源对象流出(streamed)。在步骤512,关闭包含下述内容的Stub IF:具有数据的集群的UUID的指针、具有精简信息的标记、系统元数据以及定制元数据。在步骤514,源对象被关闭,将该源对象标记为用于删除,并且由分层服务删除该源对象。

[0084] 在使用情况2,针对在复制拓扑中的目标系统上的对象定义具有仅元数据特征的服务计划。当在源系统的复制排队上正在处理对象时,复制服务将会:

[0085] 1.确定对象是否是在目标系统上的仅元数据的候选。

[0086] 2.验证在源系统上,对象的数据部分安全地存储且不需要进行仅元数据。

[0087] 3.如果两个情况全部满足,对象的仅元数据部分与指向源系统上的数据部分的位置的指针一起,替代复制完整数据,被发送至目标系统。

[0088] 在服务计划完成后,目标系统将使得仅元数据本地存储。数据部分将会存储在源系统上。

[0089] III.B具有外部存储卷的单独存储系统

[0090] 典型的存储系统包含以及管理仅在其系统内部的存储卷。而对于本发明,从主要存储系统上下文环境的可用的存储还包括通过其它网络技术(例如,NFS)暴露的存储卷。有必要构建策略并进行处理,以确定内容将于何时何地存储于外部的存储卷上。

[0091] 在具有外部存储卷的单独的存储系统的上下文环境下,对象的元数据部分以及静态数字内容的智能生命周期管理用于:(a)优化低延迟/高成本的内部数据存储的使用,以及(b)启用在外部的介质上的数据存储,但是仍然处于内容平台(例如,HCP)管理之下。服务计划可以定义下面的行为:(a)在自从获取已经经过T1时间之后将静态数字内容的一个复制件移动至外部的存储池,以及(b)在读取时,恢复静态数字内容,并且保存在本地,直至T2时间已经经过。服务计划定义将包括下面的配置:

[0092] 1.用于存储的指定网络协议规定(例如,由NFS服务器提供的NFS共享)的外部卷。

[0093] 2.内容将被存储于外部卷的时间量T1。

[0094] 3.当写至外部存储时可选的对数据的压缩。

[0095] 4.当写至外部存储时可选的对数据的加密。

[0096] 当分层服务基于服务计划定义确定对象是将写入至外部存储的候选时,处理将会包括下面的内容:

[0097] 1.验证对象是被复制的(如果需要)以及被索引的(如果需要)。

[0098] 2.如果被请求,压缩前往外部卷的数据。

[0099] 3.如果被请求,加密前往外部卷的数据。

[0100] 4.将对象的静态数字内容移至外部卷。

[0101] 5.标记对象的静态数字内容部分的本地复制件用于移除,并且记录数据部分在外部卷上的位置。

[0102] 在该运行之后,源系统将会具有存储在本地的仅元数据。固定数字内容部分将会存储在外部卷。

[0103] 对存储在外部卷上的对象的服务用户读取请求将会包括下面的内容:

[0104] 1.如果用户请求对象的仅元数据部分,则将其本地取回并返回至用户,而无需前

往外部卷。

[0105] 2.如果需要对象的静态数字内容部分且系统本地没有，则该数字内容将会从外部卷透明地取回、解压缩、解密并呈现给用户。

[0106] 3.如果定义了恢复时间值T2且其大于0，系统将会本地存储该内容长达T2量的时间。这将会使用户能够本地获取数据内容长达T2时间。经过T2时间之后发生的服务运行将会移除本地静态数字内容部分。而存储在外部卷的远程内容根本不发生改变。

[0107] 图6示出了说明了分层服务处理的流程图的示例，所述分层服务处理用于外部的分层以本地存储仅元数据以及在外部卷存储固定数字内容。处理开始于源对象，该源对象已经满足分层准则并且开放为由分层服务读取(步骤602)。在步骤604，创建Stub IF。在步骤606，为Open Stub IF指针增加外部的数据IF的位置。另外，在外部存储分层中，在外部的目标创建外部IF(步骤622)。流出至开放外部IF数据有效负载的数据可选地被加密或者压缩(步骤624)。包含原始对象的数据部分的外部IF被溢出且关闭(步骤626)。

[0108] 接下来，在步骤608，Stub IF溢出，而系统以及定制元数据从源对象流出。在步骤610，关闭包含如下内容的Stub IF：指向外部IF的指针，压缩或者加密的标记，关于加密密钥的信息，系统元数据以及定制元数据。在步骤612，关闭源对象，将该源对象标记为用于删除，并且由分层服务删除该源对象。

[0109] 具有去耦合内容以及元数据的该智能内容分层机制创建智能内容分层，该智能内容分层有助于对象存储系统的数据保护层级、有效的更新、和索引对象的系统及定制元数据、以及有助于提供压缩及加密移动至网络存储设备的数据的选项。

[0110] 图7示出了用于实施分层服务的装置的示例。装置700可能是存储对象(虚线所示的712)的对象系统的一部分(虚线所示的710)，或者是从用于存储对象722的系统720(包括对象系统和外部存储)分离的管理计算机的一部分，或者是如上所述(参见图1-4)的独立节点的冗余阵列中的系统的一部分等。装置700包括处理器或者控制器702以及存储器704，并且可操作用于在对象上执行分层服务操作。

[0111] 用户体验

[0112] 在实施这些内容分层特征的系统上的用户体验并无改变。用户在系统上存储对象，并且定义所有的熟悉的对象参数(盼望对象必须存活多久，需要多少对象的复制件等等)。在此之后，在对象生命周期期间内，它保持对用户可用。对象的额外的属性可以由系统管理者定义。这些属性包括两项：用于对象的数据部分的存储分层或者将数据部分存储于拓扑中的任意其它系统中的许可。

[0113] 情况1：在复制环境中具有数据共享的仅元数据对象

[0114] 在具有各种复制技术的环境中，会频繁地使用数据存储以及取回系统。该实施将会允许由用户为对象的数据部分所请求的数据保护层级(DPL)被维持在整个复制拓扑而非每个涉及的系统。这将会为每个系统提供相同的数据可用性以及安全性，但对于整个复制拓扑可以大量地节省存储使用。

[0115] 对象生命周期像往常一样开始。用户将会在系统上存储数据，并且创建任何需要的元数据且频繁地使用。系统将会根据为对象定义的规则为对象排队进行复制。由于对象对于在复制链的另一端需要哪个部分拥有新的智能，仅该部分将被发送至复制。如果在存储策略中复制系统被定义为仅元数据，则将仅发送对象的元数据部分。这将会潜在地节省

带宽以及目标系统的存储容量。复制系统上的用户将会具有针对对象的数据以及元数据部分的完整的访问，但是元数据将在复制的本地进行存储而数据将通过复制链接可以被访问。由于在多数情况下，元数据对于应用就足够了，将会服务用户的请求而无需任意额外的数据传输。

[0116] 情况2:存储层被定义为NFS共享

[0117] 当对象被存储于系统时对象生命周期开始。然后用户可以创建用户需要以及定期(regularly)使用的一些元数据。在创建元数据之后，由于用户仅关心元数据，对象数据变得冗余。

[0118] 如果系统管理者允许对象的数据部分迁移至NFS共享，则系统将会把数据部分从内容平台(例如，HCP)低延迟硬件移动至具有更高延迟的更廉价的NFS共享。这将会允许更好的使用昂贵且有价值的存储而不影响用户体验或者数据保护以及安全性。用户将会继续具有针对对象元数据相同速度的访问。如果用户需要对象数据部分，系统将会将其取回并且稍微延迟地为该请求提供服务，但是处理将会对用户完全透明。在整个对象生命周期中将会维持相同的数据保护层级(DPL)水平以及数据一致性。

[0119] 当然，如图1和图4所示的系统配置仅是可在其中实施本发明的包括内容平台或者复制的对象存储系统的系统的示例，而本发明并不限于具体的硬件配置。实施本发明的计算机以及存储系统可能还具有可以存储以及读取用于实施上述发明的模块、程序以及数据结构的已知的I/O设备(例如，CD以及DVD驱动器、软盘驱动器、硬盘驱动器等)。这些模块、程序以及数据结构可能被编码于这些计算机可读介质。例如，本发明的数据结构可能被存储在独立于驻留本发明中所使用的程序的一个或者多个计算机可读介质之外的计算机可读介质。系统的组件可以通过任意形式或者数字数据通信介质，例如通信网络，互联。通信网络的例子包括局域网，广域网，例如互联网、无线网、存储区域网络等等。

[0120] 在本说明书中，阐述了许多细节用于说明的目的，以提供对本发明的更深入的了解。然而，对于本领域技术人员显然的是，为实施本发明并不需要所有的这些特定的细节。还应注意的是，本发明可以被描述为过程，其通常被描绘成流程表、流程图、结构图或框图。虽然流程图将操作描述为时序的处理，但是很多操作可以并行或者同时的进行。另外，操作的顺序可以重排。

[0121] 本领域公知的是，上述的操作可以由硬件、软件或其组合来实施。本发明的实施例的一些方面可以使用电路以及逻辑设备(硬件)来实施，而其它一些方面可以使用存储在机器可读介质上的指令(软件)来实施，如果由处理器执行则其可以使得处理器执行方法以执行本发明的实施例。此外，本发明的一些实施例可以仅由硬件执行，而其他实施例可以仅由软件执行。另外，所述各种功能可以由单一的单元来进行，或者可以由若干组件以任一数量的方式分布执行。当由软件执行的时候，该方法可以由处理器诸如通用目的计算机基于存储在计算机可读介质上的指令来执行。如果需要，指令可以以压缩和/或加密的格式被存储在介质上。

[0122] 从上述内容可以看出，本发明提供方法、装置以及存储在计算机可读介质上的程序，用于提供内容类调用的机制，以定义将会针对对象的无结构内容及其元数据构建结构的蓝图，并且便于有效的索引和搜索。此外，尽管在本说明书中说明和描述了具体的实施例，但是对本领域技术人员而言，显然的是，对于公开的具体实施例，为实现相同目的所计

算出的任一安排均可以替换。本公开文本意在覆盖任一和所有本发明的适应性修改以及变化，而且应当理解的是，在下面权利要求中所使用的术语不应当被理解为限制本发明至在说明书中所公开的具体实施例。而应是，本发明的保护范围将全部由权利要求所确定，其将根据权利要求解释的建立的说法进行解释，而且具有与其等价的全部范围的权利。

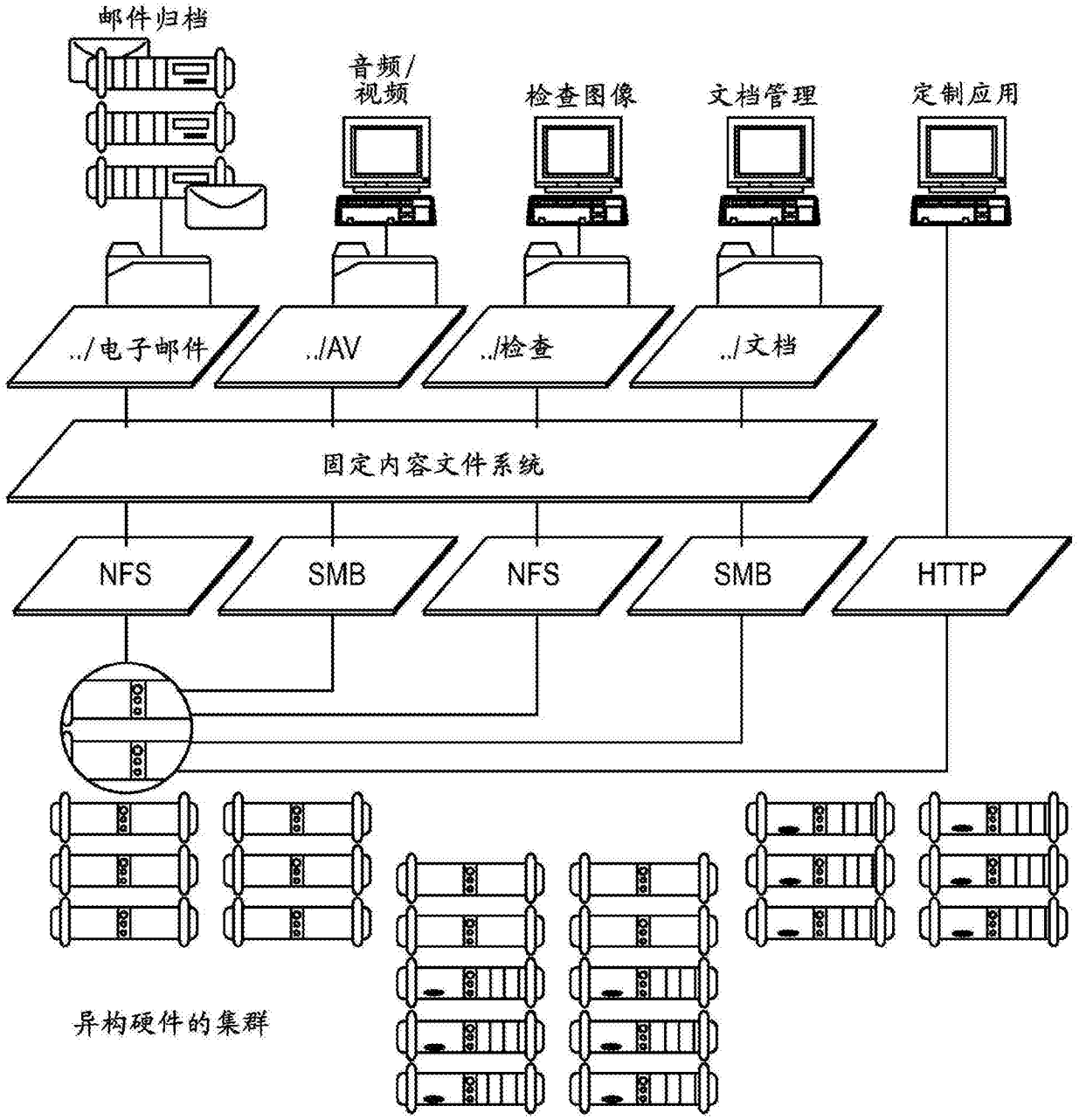

邮件归档
音频/
视频
检查图像
文档管理
定制应用
../电子邮件
../AV
../检查
../文档
固定内容文件系统
NFS
SMB
NFS
SMB
HTTP
异构硬件的集群

图1

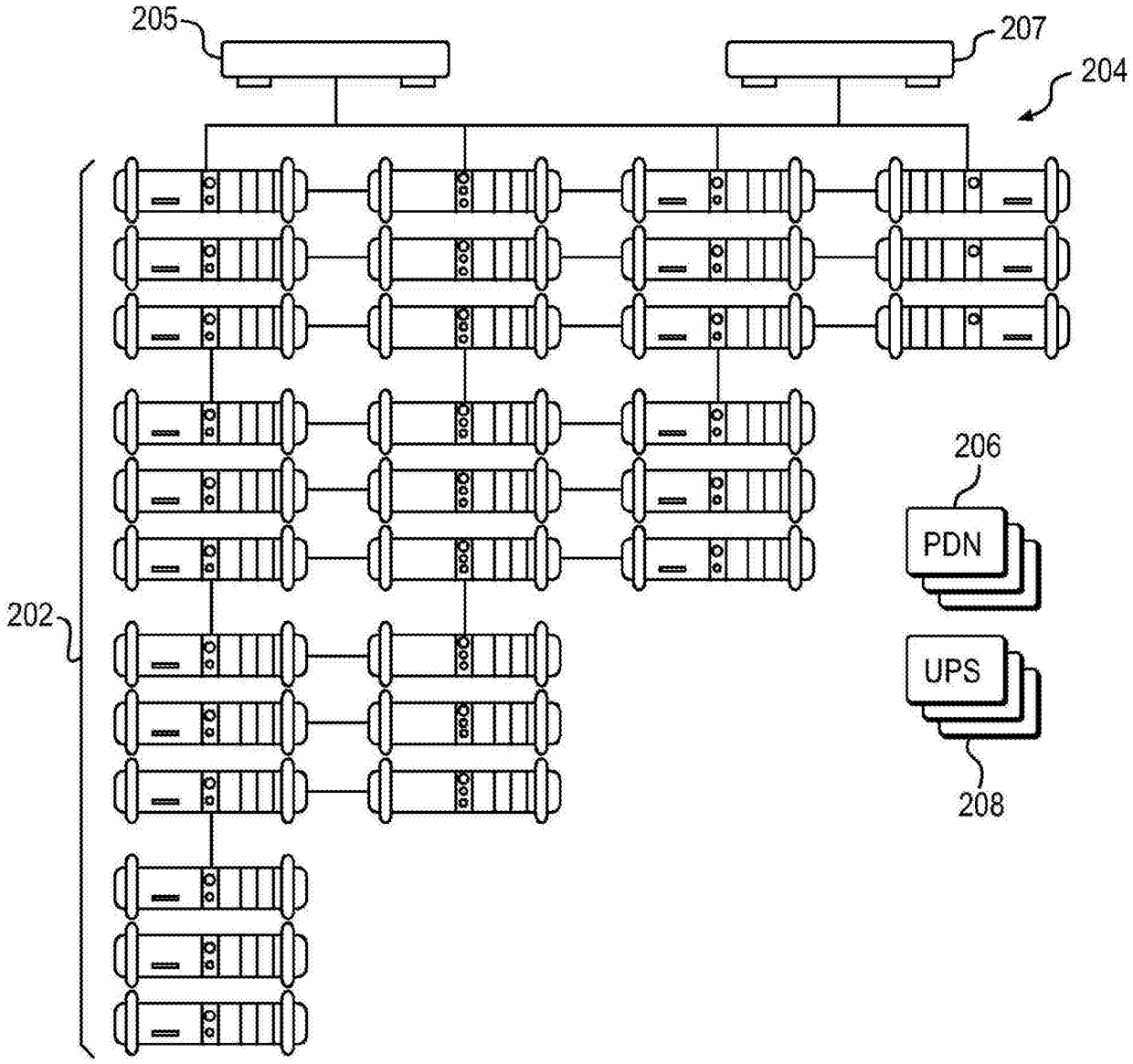


图2

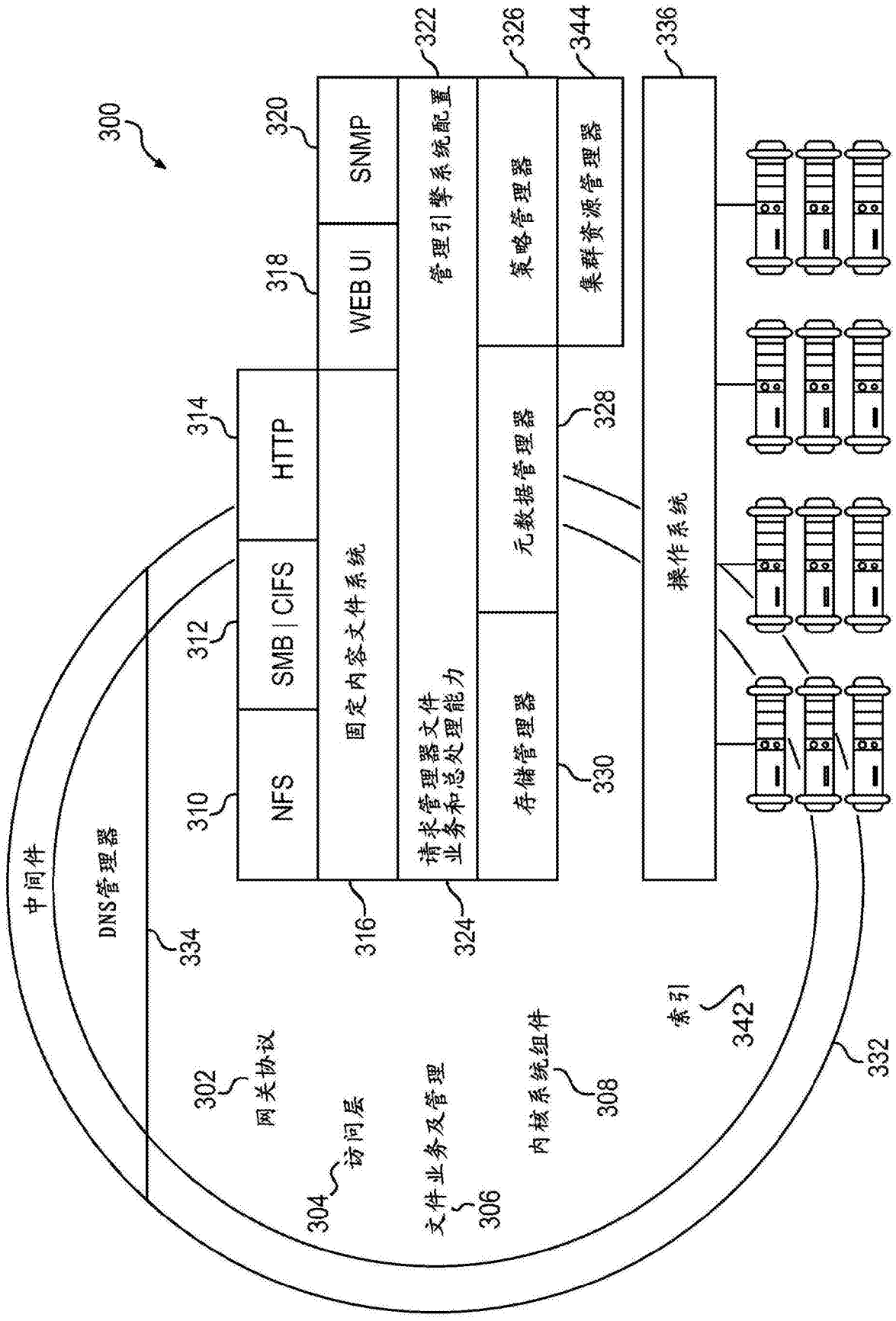

300
320
SNMP
318
WEB UI
322
管理引擎系统配置
326
策略管理器
344
集群资源管理器
336
操作系统
314
HTTP
328
元数据管理器
312
SMB | CIFS
固定内容文件系统
310
NFS
请求管理器文件
业务和总处理能力
存储管理器
330
316
324
334
中间件
DNS管理器
302
网关协议
访问层
304
文件业务及管理
306
内核系统组件
308
索引
342
332

图3

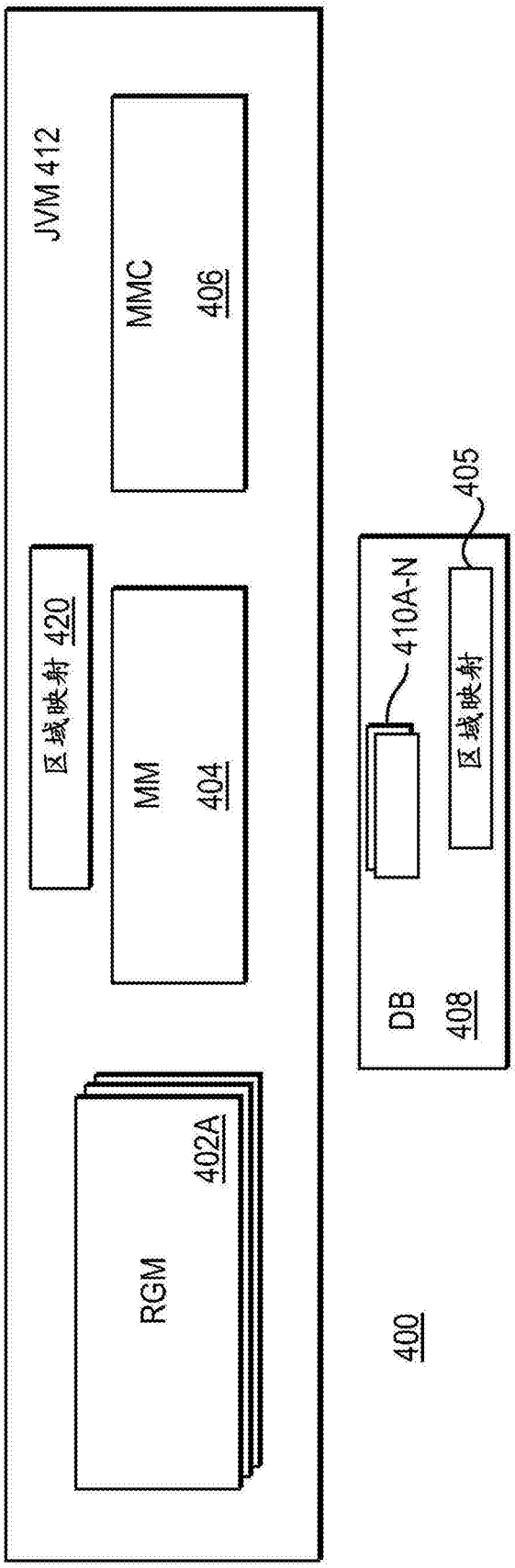

JVM 412
MMC 406
区域映射 420
MM 404
RGM 402A
405
410A-N
区域映射
DB 408
400

图4

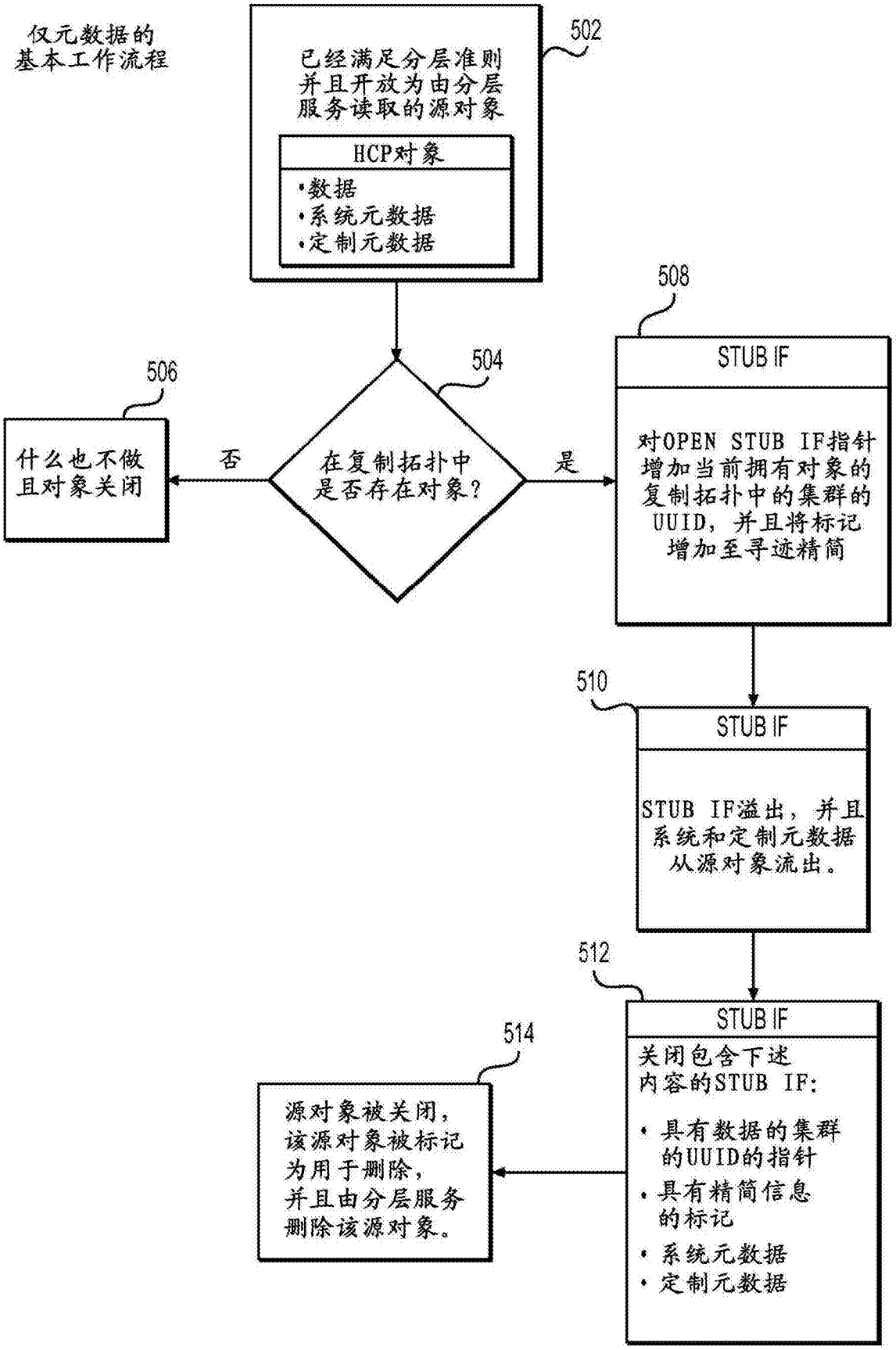

仅元数据的
基本工作流程
已经满足分层准则
并且开放为由分层
服务读取的源对象
502
HCP对象
•数据
•系统元数据
•定制元数据
504
在复制拓扑中
是否存在对象?
否
是
506
什么也不做
且对象关闭
508
STUB IF
对OPEN STUB IF指针
增加当前拥有对象的
复制拓扑中的集群的
UUID，并且将标记
增加至寻迹精简
510
STUB IF
STUB IF溢出，并且
系统和定制元数据
从源对象流出。
512
STUB IF
关闭包含下述
内容的STUB IF:
• 具有数据的集群
的UUID的指针
• 具有精简信息
的标记
• 系统元数据
• 定制元数据
514
源对象被关闭，
该源对象被标记
为用于删除，
并且由分层服务
删除该源对象。

图5

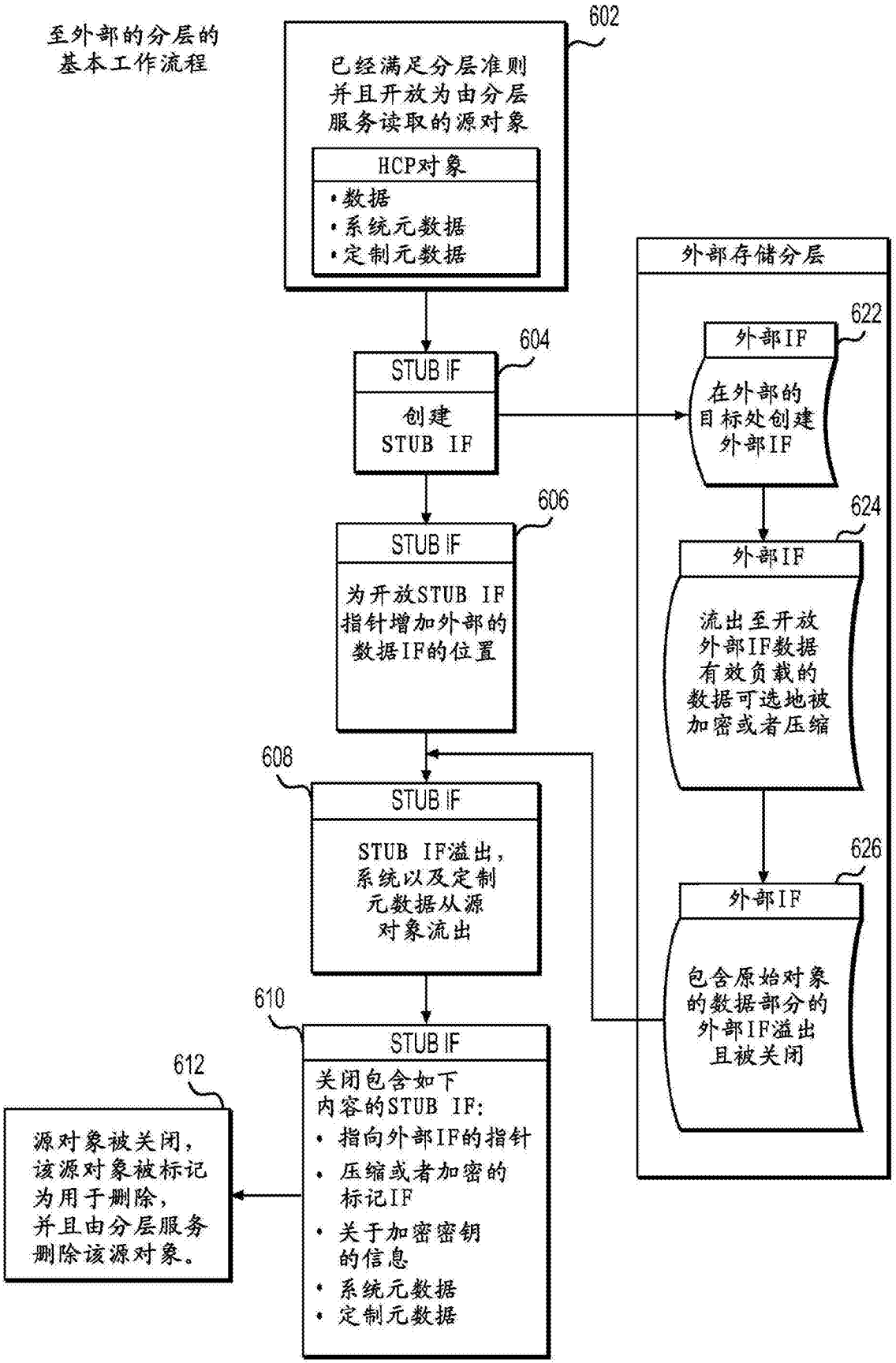

至外部的分层的
基本工作流程
602
已经满足分层准则
并且开放为由分层
服务读取的源对象
HCP对象
• 数据
• 系统元数据
• 定制元数据
外部存储分层
604
STUB IF
创建
STUB IF
622
外部IF
在外部的
目标处创建
外部IF
606
STUB IF
为开放STUB IF
指针增加外部的
数据IF的位置
624
外部IF
流出至开放
外部IF数据
有效负载的
数据可选地被
加密或者压缩
608
STUB IF
STUB IF溢出，
系统以及定制
元数据从源
对象流出
626
外部IF
包含原始对象
的数据部分的
外部IF溢出
且被关闭
610
STUB IF
关闭包含如下
内容的STUB IF:
• 指向外部IF的指针
• 压缩或者加密的
标记IF
• 关于加密密钥
的信息
• 系统元数据
• 定制元数据
612
源对象被关闭，
该源对象被标记
为用于删除，
并且由分层服务
删除该源对象。

图6

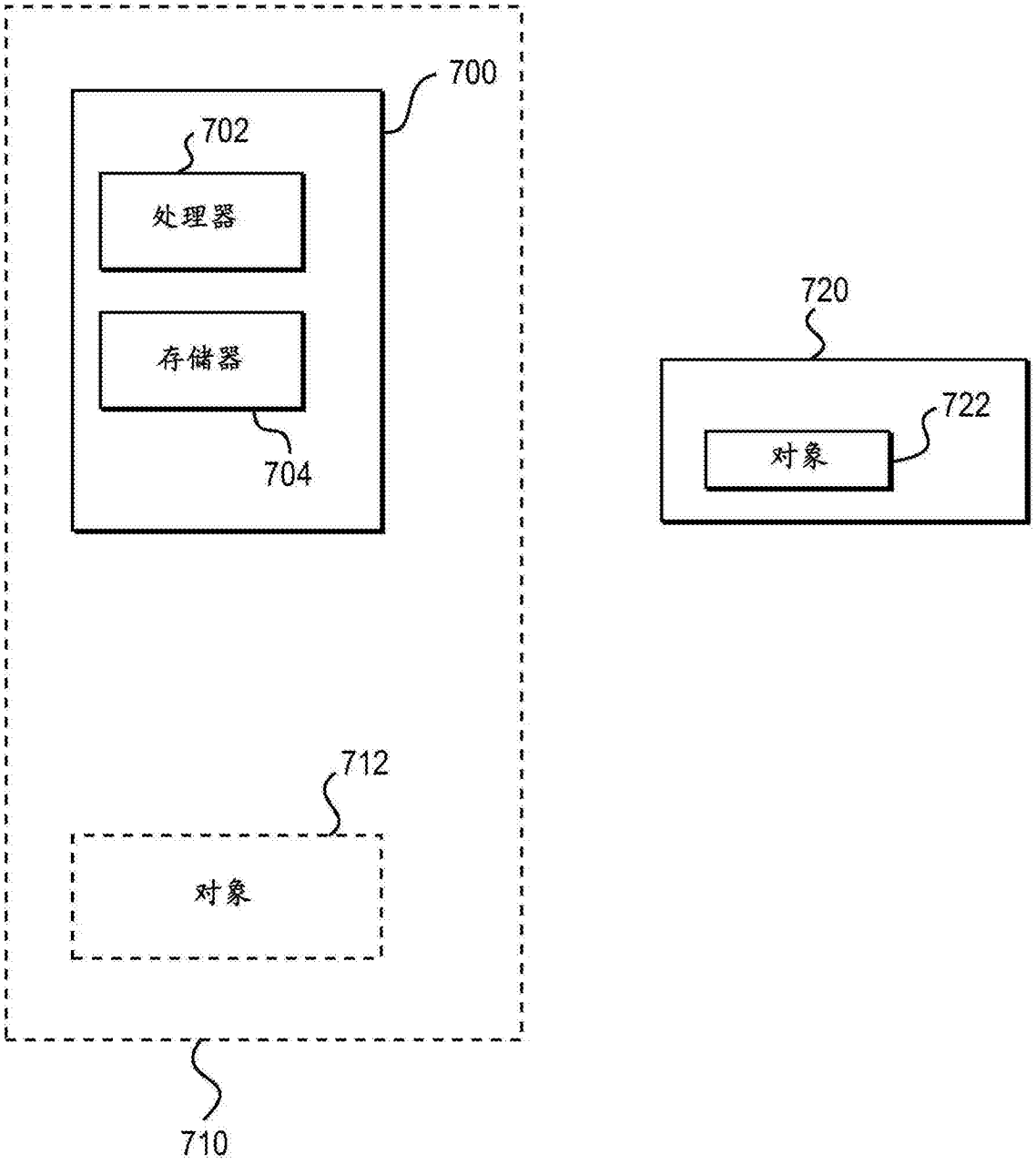


图7