



## (12) 发明专利

(10) 授权公告号 CN 101473309 B

(45) 授权公告日 2014. 10. 01

(21) 申请号 200780009902. X

(51) Int. Cl.

(22) 申请日 2007. 03. 20

G06F 12/00 (2006. 01)

## (30) 优先权数据

60/784,022 2006.03.20 US

11/688,203 2007.03.19 US

## (85) PCT国际申请进入国家阶段日

2008.09.19

## (56) 对比文件

WO 2006/017584 A2, 2006.02.16,

WO 99/12098 A1, 1999.03.11,

US 005764972 A, 1998.06.09,

US 20050160243 A1, 2005.07.21,

审查员 武晓冬

## (86) PCT国际申请的申请数据

PCT/US2007/064440 2007.03.20

## (87) PCT国际申请的公布数据

W02007/109685 EN 2007.09.27

## (73) 专利权人 伊姆西公司

地址 美国马萨诸塞州

## (72) 发明人 岳睿

(74) 专利代理机构 隆天国际知识产权代理有限公司 72003

代理人 郑小军

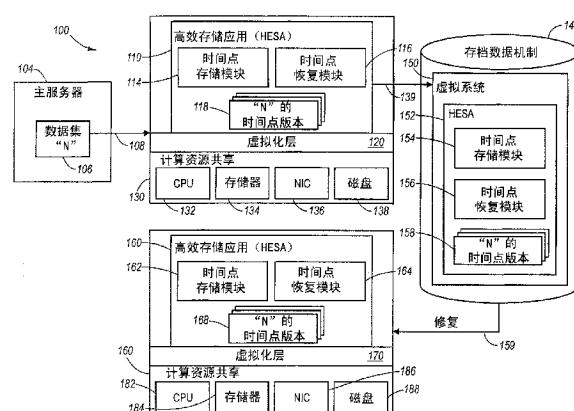
权利要求书3页 说明书7页 附图3页

## (54) 发明名称

高效可传递存档器

## (57) 摘要

一种高效可传递存档器，执行在虚拟化层上运行的存储系统，以存档原始数据集的时间点版本，并且该存储系统本身被作为存档介质上的虚拟系统。该高效可传递存档器能够在各种计算机结构中执行。该虚拟化层将基于计算机结构的组件的一组硬件的规范化代表项呈现给该存储系统，从而将该存储系统相对该计算机结构的实际硬件组件而屏蔽。该存储系统和该原始数据集的时间点版本能够被恢复至支持虚拟系统的任何硬件子系统。



1. 一种提供数据保护的方法,包括 :

在计算机结构上提供第一虚拟化层,所述计算机结构包括存储原始数据集的存储装置、以及存档存储机构和第一硬件子系统;

在所述第一虚拟化层上运行存储应用,所述存储应用包括存储模块和恢复模块,其中所述第一虚拟化层将基于所述计算机结构的组件的一组硬件的代表项呈现给所述存储应用;

操作所述存储应用,以生成在所述存储装置中存储的所述原始数据集的时间点版本;

在所述存档存储机构中存储虚拟系统,所述虚拟系统包括所述存储应用,其中所述存储应用具有所述原始数据集的时间点版本;以及

响应于由所述恢复模块接收的来自用户的恢复请求,使用所述恢复模块,基于在所述存档存储机构中存储的所述原始数据集的时间点版本来在所述第一硬件子系统上生成所述原始数据集的恢复版本。

2. 如权利要求 1 所述的方法,其中所述原始数据集具有尺寸,且其中所述原始数据集的时间点版本的尺寸小于所述原始数据集的尺寸乘以存储于所述存档存储机构中的版本数。

3. 如权利要求 1 所述的方法,还包括将所述存储应用传递到第二硬件子系统。

4. 如权利要求 1 所述的方法,其中所述第一硬件子系统至少部分不同于所述计算机结构的硬件组件,所述计算机结构的硬件组件被用于对所述存储应用的操作中,以在所述存档存储机构中存储所述原始数据集的时间点版本。

5. 如权利要求 1 所述的方法,其中所述时间点版本中的每一个与存在于所述原始数据集中不同时间点处的数据相关。

6. 如权利要求 1 所述的方法,其中所述第一硬件子系统、所述计算机结构的硬件组件或二者包括下列中的一个或多个:

服务器;

网络界面控制器;

中央处理器;

存储器;

存储装置;以及

网络。

7. 如权利要求 1 所述的方法,其中在所述存档存储机构中存储的所述存储应用是所述存储应用的一个存档版本,所述方法还包括:

响应于所述来自用户的恢复请求,提供包括基于在所述存档存储机构中存储的所述存储应用的所述存档版本而生成的存储应用的不同的虚拟化层上的新的虚拟机。

8. 如权利要求 1 所述的方法,其中所述第一虚拟化层由下列中的一个或多个提供:

VMware ESX;

VMware GSX;以及

XEN3.0。

9. 如权利要求 1 所述的方法,其中所述第一虚拟化层将所述计算机结构虚拟化成以可控方式分配给虚拟机的统一的共享,并且将所述统一的共享的硬件与一个或多个虚拟机分

离。

10. 如权利要求 9 所述的方法,还包括:

将所述虚拟系统传递到新的硬件子系统,其中所述新的硬件子系统与所述统一的共享的硬件子系统不相匹配。

11. 如权利要求 1 所述的方法,其中所述原始数据集的时间点版本的尺寸远小于被备份或复制的原始数据集的尺寸,并且其中生成所述原始数据集的时间点版本包括使用快照、文件区别技术、内容寻址存储系统和连续数据保护系统中的一个或多个。

12. 如权利要求 1 所述的方法,还包括:

将所述存储应用打包压缩成文件,以使得包括充分配置的应用、操作系统、BIOS 和虚拟硬件的全部系统能够从一个物理服务器被移动到另一个物理服务器。

13. 一种高效可传递存档器,所述高效可传递存档器包括:

用于在计算机结构上提供第一虚拟化层的装置,所述计算机结构包括存储原始数据集的存储装置、以及存档存储机构和第一硬件子系统;

用于在所述第一虚拟化层上运行存储应用的装置,所述存储应用包括存储模块和恢复模块,其中所述第一虚拟化层将基于所述计算机结构的组件的一组硬件的代表项呈现给所述存储应用;

用于操作所述存储应用,以生成在所述存储装置中存储的所述原始数据集的时间点版本的装置;

用于在所述存档存储机构中存储虚拟系统的装置,所述虚拟系统包括所述存储应用,其中所述存储应用具有所述原始数据集的时间点版本;以及

用于响应于由所述恢复模块接收的来自用户的恢复请求,使用所述恢复模块,基于在所述存档存储机构中存储的所述原始数据集的时间点版本来在所述第一硬件子系统上生成所述原始数据集的恢复版本的装置。

14. 如权利要求 13 所述的存档器,其中所述原始数据集具有尺寸,且其中所述原始数据集的时间点版本的尺寸小于所述原始数据集的尺寸乘以在所述存档存储机构中存储的版本数。

15. 如权利要求 14 所述的存档器,其中所述存储应用使用下列技术中的一个或多个来减小所述时间点版本的尺寸:

快照;

文件区别;

内容寻址存储;

连续数据保护;以及

公共因子分解。

16. 如权利要求 13 所述的存档器,其中所述计算机结构包括 x86 计算机结构。

17. 一种用于存档数据的系统,该系统包括:

用于在计算机结构上提供第一虚拟化层的装置,所述计算机结构包括存储原始数据集的存储装置、以及存档存储机构和第一硬件子系统;

用于在所述第一虚拟化层上运行存储应用的装置,所述存储应用包括存储模块和恢复模块,其中所述第一虚拟化层将基于所述计算机结构的组件的一组硬件的代表项呈现给所

述存储应用；

用于操作所述存储应用，以生成在所述存储装置中存储的所述原始数据集的时间点版本的装置；

用于在所述存档存储机构中存储虚拟系统的装置，所述虚拟系统包括所述存储应用，其中所述存储应用具有所述原始数据集的时间点版本；以及

用于响应于由所述恢复模块接收的来自用户的恢复请求，使用所述恢复模块，基于在所述存档存储机构中存储的所述原始数据集的时间点版本来在所述第一硬件子系统上生成所述原始数据集的恢复版本的装置。

18. 如权利要求 17 所述的系统，其中所述存储应用包括多个存储应用。

19. 如权利要求 17 所述的系统，其中所述第一虚拟化层呈现所述计算机结构的规范化的代表项，并将所述存储装置相对所述计算机结构而屏蔽。

20. 如权利要求 17 所述的系统，其中所述存储应用包括下列中的一个或多个：

存储模块，用于生成所述原始数据集的尺寸减小的版本；

恢复模块，用于生成所述原始数据集的恢复版本。

## 高效可传递存档器

### 技术领域

[0001] 本发明涉及一种用于存档数据和恢复数据的数据存储 (storage) 和备份方法。更具体地，本发明的实施例涉及一种以如下方式提供数据保护的软件、硬件、系统和方法，所述方式允许用户或客户获取自选择的时间点起的信息或所存数据的副本，并提供高效的数据存档和数据可传递性。

### 背景技术

[0002] 信息的可靠备份和存档是现今所需要的。对于专用于提供信息备份和存档的信息系统 (IS) 资源，企业花了大量的时间和金钱，其中所述信息存在于产生和依赖于数字信息的企业机构内的计算机和服务器中。数据存储行业的客户不但频繁要求其数据被正确备份，而且要求在需要时（例如导致数据丢失的系统故障之后）可在特定时间点 (point in time) 取得所备份的数据。换句话说，对接近连续数据保护的需求日益增加，所述连续数据保护能够使数据恢复到其在特定时刻的状态，该特定时刻最通常为刚好计算机或数据存储系统损毁或损失 (lost) 之前的时间点。虽然对于时间点数据保护的需求也必须权衡低数据存储成本的需要，但是典型地，高速光学存储系统和磁盘存储系统用作存档存储器要比磁带式数据存储系统昂贵。

[0003] 在时间点存档需求和数据存储发展的驱动下，已经出现了利用高效技术在磁盘存储器上存储主数据 (primary data) 的多个版本或时间点的新技术。这些技术允许以如下方式存储数据的多个副本，即具有来自不同时间点的特定尺寸的数据集“N”（例如，数据集 N1、N2、N3，其中数字 1、2、3 表示可对数据集 N 进行改变的不同时间点），其中所述方式比简单地以数据的自然状态存储数据会消耗更少的磁盘或光学数据存储器容量 (capacity)。例如，高效磁盘存储系统可以比原始数据集 N 的总尺寸小的存储容量来存储数据集 N、N1、N2 和 N3，或者至少使用比数据集 N+N1+N2+N3 的尺寸之和小的存储容量来存储数据集 N、N1、N2 和 N3。

[0004] 现今，为了提供用于备份、恢复和灾难恢复的时间点数据副本，会用到多种以高效方式存储数据副本的软件方式或应用方式的技术途径。这些技术包括但不限于快照 (snapshot)、文件区别技术 (file differencing technique)、内容寻址存储系统以及消除冗余数据分量 (component) 且其尺寸可变的系统。当提供更高效的存档数据的方法时，所有这些系统均使用磁盘存储器作为它们的主要存储机构，而非较便宜的磁带介质或磁带存储系统。

[0005] 此外，尽管存在这些高效存储技术，企业仍需要经常将数据存储或移动到其它存档系统上，所述其它存档 (archive) 系统可使用较便宜或具有不同管理属性的可移动磁带介质、光学存储器或其它磁盘存储系统。在一些情况下，需要这些存档以满足调控的 (regulatory) 需求或其它需求。这种存档的问题在于其效率经常很低，例如，为了存档目的，存档包括将数据扩展回其原始状态 (N+N1+N2+N3)。这种存档的另一问题在于，由于所实施的高效存储方法，数据被这样存储，其将来就不能容易地传递、恢复或管理。例如，使用基

于块的快照将主存储系统的所有卷复制到磁带将产生高效的数据集,但是在没有将数据恢复到与原始硬件平台的物理特性相匹配的系统的情况下,不能独立地读出或使用数据集。

[0006] 结果,现有的备份和存档技术不能满足数据存储客户的需求,因此一直需要提供连续或接近连续数据保护的先进技术。优选地,能够利用现有数据存储硬件以除了具有先进可传递性外的高效方式实现这种技术。

## 发明内容

[0007] 本发明的实施例旨在提供用于数据存储和数据存档的方法和系统。在虚拟化层上运行的存储应用能够生成在存储装置中存储的原始数据集的时间点版本。所述存储应用可压缩 (compress)、最小化或以别的方式缩小所述原始数据集的时间点版本。所述存储应用和所述原始数据集的版本可被存档到存档介质上以作为虚拟系统。所述存储应用和所述原始数据集的版本可被恢复至支持该虚拟系统的任何硬件子系统。

[0008] 提供上述概要以通过简单的方式介绍概念选择,在下面的具体实施方式部分会进一步描述该概要。该概要的目的不是识别所要求保护的主题的关键特征或必要特征,其目的也不是用于帮助确定所要保护的主题的范围。

## 附图说明

[0009] 为了进一步阐明本发明的上述和其它优点和特征,通过参照在附图中示出的本发明的具体实施例来给出本发明的更具体的描述。应理解,这些附图仅描述了本发明的典型实施例,因此不认为其限制了本发明的范围。通过使用附图就附加特征和细节对本发明进行描述和说明,其中:

[0010] 图 1 以方框图的形式示出根据本发明一个实施例的高效可传递存档 (HEPA) 系统;

[0011] 图 2 示出根据本发明另一实施例的 HEPA 系统,其使用虚拟化层将存储应用软件和相关的操作系统从计算机结构 (architecture) 中分开;

[0012] 图 3 是本发明的另一 HEPA 系统,除了主机结构 (hosted architecture) 之外,该系统与图 2 的系统相似;

[0013] 图 4 是本发明又一 HEPA 系统,其示出使用虚拟骨干网或层允许多个虚拟机访问的计算机资源共享 (pool) (包括用于提供数据集存档的装置);以及

[0014] 图 5 描述根据本发明实施例提供数据保护的方法。

## 具体实施方式

[0015] 现在参照附图来描述本发明示例性实施例的各种方案。应理解附图是这些示例性实施例的图解式或示意性表示,因此其不限制本发明的范围,而且也没有必要按比例绘制附图。

[0016] 本发明针对于用于提供高效可传递存档 (即存档器数据保护) 的方法及系统。高效可传递存档 (high efficiency portable archive, HEPA) 系统的一个实施例包括或使用高效存储系统。第二,其可包括虚拟化层以相对实际硬件装置而屏蔽 (shield) 高效存储系统。第三,HEPA 系统可使用或包括外部存档机构,例如磁带式数据存储系统。

[0017] 通过在虚拟化层上运行高效存储系统，整个高效存储系统作为虚拟系统被存档到存档介质上，例如磁带存储器。然后，该存档可被恢复 (restore) 至支持虚拟系统的任何硬件子系统，并且从高效存储系统本身可以恢复或访问高效存储系统中存储的数据的所有版本。

[0018] 使用上面提供的术语，本发明的 HEPA 系统由高效存储系统构成，该高效存储系统能够以小于数据集的尺寸之和（即  $N+N_1+N_2+N_3$  的尺寸之和）的空间存储数据集  $N$ （即  $N$ 、 $N_1$ 、 $N_2$ 、 $N_3$  等）的多个时间点版本。该高效存储系统可运行在虚拟化层上，例如设置在标准计算机结构（例如 x86 结构等）中的虚拟机 (VM) 中。虚拟化层对高效存储系统的硬件装置进行虚拟化。高效存储系统作为虚拟系统被存档到 HEPA 系统的存档介质上，例如磁带、光学或磁盘存储器，所以存档介质中的高效存档可被传递 (port) 到支持虚拟系统的任何硬件子系统上。这允许根据原始高效存储系统的恢复方法学 (methodology) 来恢复所存储的数据集 ( $N+N_1+N_2+N_3$ )。本发明的实际例子将使用 EMC 的 Avamar 软件，该 Avamar 软件在 VMware ESX 上运行并将 Avamar 的 VMguest 情况的副本存档到磁带存档器 (archive)。

[0019] 为了实现本发明，计算机和网络装置可以是用于提供上述功能的任何装置，包括公知的数据处理、存储和通信装置及系统，例如通常用作用户系统中的主机的计算机装置和被设置为在通信网络上保存然后发送数字数据的服务器装置，其中该用户系统具有处理组件 (component)、存储器组件和输入 / 输出组件。通常，数据以遵循标准通信和传输协议的数字格式进行通信。数据存储资源通常被描述为磁盘装置、光学装置和磁带装置，这些装置实现 RAID 和其它存储技术，并可使用 SCSI 和其它 I/O、数据传输 (transfer) 和存储协议，但是本发明的目的在于不局限于这些示例性实施例或具体硬件和存储机构，这是因为本发明几乎对其中生成和保存有数字数据存档（例如数据卷宗 (data volume)）的任何数据存储装置均是有用的。

[0020] 图 1 以简化的方框形式示出根据本发明的示例性高效可传递性存档 (HEPA) 系统 100。系统 100 包括一个或多个主服务器 104，其存储希望提供存档备份或数据保护的数据集。如所示出的，HEPA 系统 100 保护高效存储应用（或系统）110，其通常是否基于软件或软件实现的技术，所述技术用于处理数据，即原始数据集，从而创建尺寸减小的数据集以用于存储或备份。如所示出的，存储应用 (application) 110 包括时间点存储功能或模块 114，其用于生成数据集的压缩的或其它更有效的存储版本，例如尺寸远小于被备份或复制的原始集尺寸的数据集的时间点版本。此外，存储应用包括数据恢复方法学或模块 116，其允许使用存储应用 110（例如，模块 114 等）以有效方式存储一组时间点数据集后，使该组时间点的数据集被恢复或恢复。例如，时间点存储模块 114 或其它未示出的模块中的存储应用 110 可使用如下一种或多种技术：快照 (snapshot)、文件区别技术 (file differencing technique)、内容寻址存储系统、连续数据保护 (CDP) 及消除尺寸可变的冗余数据分量的系统。在一个实施例中，存储应用 110 执行公共因子分解技术 (commonality factoring technology)，该技术在美国专利申请 6810398 中被详细描述，美国专利申请 6810398 的整个内容通过参考援引于此。

[0021] 明显地，HEPA 系统 100 包括计算资源共享 130，其包括系统 100 的所有计算资源（例如处理器 /CPU、存储器、网络装置（包括 NIC）、服务器和存储器 / 存储装置）。共享 130 中的计算资源的具体排布不限于具有存储应用 110 的本发明，存储应用 110 使用这些资源

来存储处理过的数据（例如，压缩的或其它尺寸减小的数据集）、恢复的或复原的数据集，以及由存储应用 110（例如由时间点数据存储模块 114 和恢复模块 116）以有效存储和 / 或恢复存档数据的方式使用的信息 / 数据。为此，共享 130 包括 CPU132、存储器 134、NIC136 和磁盘 138。存储器 134 和 / 或磁盘 138 可用于存储用于存储应用 110 的处理信息。

[0022] 如 108 处所示，高效存储应用 110 起到对数据集 106 进行处理以生成 N 的高效时间点版本 118。数据集 N106 可以是存储应用 110 的主机或客户机 (client) (未示出) 数据，并且主机 / 客户机可利用高效存储应用 110 来存储或存档原数据 106 以符合外部数据机构或系统 140 的内部 (或外部) 调控行为 (regulatory agency)，外部数据机构或系统 140 通常包括数据集 N 的时间点版本 (如图 1 所示)。处理过的数据 118 可采用多种形式来实现本发明，但是当包括时间点版本的虚拟系统被较晚地存储在存档器 140 时，优选地以相对于版本数尺寸减小的方式存储处理过的数据 118 (例如，存档数据不简单地是版本数乘以数据集 N106 的尺寸，而是一些更小的尺寸，这是因为时间点版本 118 通常是更小的或更有效的版本)，同时版本 118 的尺寸和形式随着存储应用 110 的具体实施而变化。

[0023] 为了提供各种时间点处的数据集 N 的存档或保护版本，如 139 处所示，包括 N 的时间点版本 118 的高效存储应用 110 被存储在存档数据机构 140 中，如虚拟系统 150 所示。虚拟系统 150 包括高效存储应用 152，高效存储应用 152 包括存储模块 154 和恢复模块 156 以及 N 的时间点版本 158。通过在存档机构 140 (其通常是相对于共享 130 的外部装置或系统) 中存储整个虚拟机 150，如 159 处所示，通过在不同的虚拟化层 170 (其包括基于存档版本 152 生成的高效存储应用 160) 上提供新的虚拟机，能够对在特定时间点恢复或复原数据集 106 的请求作出响应。

[0024] 存储应用 160 通常包括时间点存储模块 162 和数据恢复模块 164。恢复模块 164 可用于对恢复或复原针对特定时间点的原始数据集 N106 (或对恢复其它数据丢失、数据损毁等) 的主机 / 客户机请求进行响应。为此，调整系统 100 以使存储应用 160 能传递到存档数据机构 140 以访问 N 的时间点版本 158，从而在共享 180 的硬件子系统上恢复数据 168，其中共享 180 包括 CPU182、存储器 184、NIC186 和磁盘 188 或其它装置。使用存储应用 160 的数据恢复工具或模块 164 执行这种恢复，并且能够在支持应用程序 110 的虚拟化的任何硬件系统，例如共享 180，上执行这种恢复，即使共享 180 的组件可能与共享 130 不同。

[0025] 更具体地，存储应用 110、160 使用虚拟化层 120、170 将高效存储应用 110、160 相对共享 130、180 中的实际硬件装置而屏蔽。使用虚拟化层 120、170 来运行存储应用 110、160 (例如，如同具有其自己的操作系统或具有主机操作系统的客户机或虚拟机) 能够使虚拟系统 150 被存档到存档数据机构 140 上。然后，能够将高效存档器 150 传递到新硬件子系统 (其可以是能够支持虚拟系统 150 的共享 170 中的一个或多个硬件装置) 上，并且根据原始高效存储系统 110 的恢复方法学 164 能够恢复所存储的数据集，如 168 处所示 (例如， $N+N_1+N_2+\dots+N_x$ )。系统 170 不必须与用于创建存档数据 158 的硬件平台 / 组件的物理特性相匹配 (与前面的存档系统的情况相同)。

[0026] 虚拟化层 120、170 可采用多种形式以实现本发明。通常，选择虚拟化层 120、170 以使其担当抽象层 (abstraction layer)，抽象层使物理硬件与存储应用 110、160 相关的操作系统分离，以产生更大的 IS 资源利用和适应性。虚拟化层 120、170 允许多个具有不同种类操作系统的虚拟机或客户机 (例如可用于运行应用程序 110、160)，从而分离地而非并

排 (side-by-side) 地在相同的或不同的物理机 (图 1 中未示出) 上运行。每个虚拟机具有其自己的虚拟硬件组 (例如共享 130、180 中的 RAM、CPU、NIC 等), 在虚拟硬件组上加载操作系统和包括存储应用 110、160 的应用程序。操作系统查看稳定的、规范化的硬件组而不考虑实际的物理硬件组件。

[0027] 通常, 虚拟机被打包压缩 (encapsulated) 成文件, 从而能够快速保存、复制和设置虚拟机。为了实现零停工维护和连续的工作量整理, 全部系统 (全部设置的应用程序、操作系统、BIOS 和虚拟硬件) 能够在数秒钟内从一个物理服务器移动到另一个。具有存储应用 110、160 的可传递或移动的虚拟机能够使用恢复模块 164 从存档数据机构 140 恢复时间点数据集 N168。虚拟化层 120、170 可提供分割 (partitioning), 以能够在单个物理系统中支持多个应用程序 (例如存储应用 110、160 的多个版本或实例) 和操作系统。共享 130、180 中的服务器可被整理成任一放大 (scale-up) 或缩小 (scale-out) 结构上的虚拟机, 以及计算资源被视为以可控方式分配给虚拟机的统一的 (uniform) 共享 130、180, 例如在虚拟化层 120、170 上运行存储应用 110、160 的客户机 (未示出)。

[0028] 虚拟化层 120、170 也提供隔离 (isolation)。虚拟机被完全与主机和其它虚拟机隔离。如果虚拟机崩溃, 所有其它部分都不受影响。数据不会通过虚拟机泄漏, 并且应用程序可以仅在所配置的网络连接上进行通信。优选地, 虚拟化层 120、170 还被配置或选择为提供打包压缩 (encapsulation)。在这些实施例中, 完整的虚拟机环境例如具有存储应用 110 或 160 的客户机, 被存作单个文件, 如 139 处所示, 其容易在系统 100 中备份、移动和复制到存档数据机构 140。虚拟化层 120、170 向应用 110、160 提供标准化且虚拟化的硬件, 应用程序 110、160 保证经由存储应用 160 恢复的应用程序 110 和存档 168 的兼容性或可传递性。

[0029] 图 2 示出根据本发明的 HEPA 系统 200。如所示出的, 计算机平台 210 (例如 x86 结构等的标准结构) 包括多种资源, 例如 CPU、存储器、网络装置、磁盘和磁带机构。提供虚拟化层 220 以管理对结构 210 的访问或将结构 210 虚拟化成统一的共享, 并将这个共享的硬件与一个或多个虚拟机分离 (即系统 200 示出的在虚拟化层 220 上运行的两个这种虚拟机或客户机, 当然也可运行更多的这种虚拟机)。虚拟化层 220 代表对每个虚拟机或客户机的操作系统 232、233 的虚拟化代表项 (representation) 236、237。操作系统 232、233 可以改变并包括例如 Windows、Linux、Novell、Solaris 和其它诸如 FreeBSD 等的操作系统。高效存储应用 230、231 被提供并与操作系统 232、233 的情形相关, 高效存储应用 230、231 使用虚拟系统 236、237 处理原数据以创建更大空间的有效版本 (例如与原始数据集的尺寸相比的压缩版本), 这些有效版本被存储到结构 210 (例如磁带装置或磁盘或光学装置) 中的存档机构。

[0030] 图 3 示出根据本发明的 HEPA 系统 300 的另一结构。如同系统 200 的情况, 系统 300 包括一组由虚拟化层 330 共享 (pool) 并相对高效存储应用 338 而屏蔽的计算机硬件或特定结构。然而, 在这种情况下, 虚拟机包括虚拟化层 330 并被安装和运行为例如与应用程序 322 相似的应用程序。虚拟化层 330 依赖于主机 OS320, 主机 OS320 用于结构 310 的装置支持 (device support) 和物理资源管理, 并将客户机操作系统 334 提供给存储应用 338。

[0031] 图 4 示出根据本发明的又一 HEPA 系统 400。HEPA 系统 400 包括计算机资源共享或计算机结构 / 平台 410, 计算机结构 / 平台 410 包括系统 400 中的硬件和相关可用的软件。

共享 410 包括存储器 411、418，其可包括用于数字数据的几乎任何类型的存储装置，例如磁带式存储装置和系统以及磁盘和光学装置。共享 410 还包括多网络 / 网络装置 412、414 和多个服务器或其它计算装置 413、415、416（其在存储器 411、418 用于存档或备份存储的某些情况下也可用作数据存储器）。

[0032] 在平台 410 上方，设置虚拟基础结构 420 用于将共享 410 中的资源连接至用户（或连接至企业 / 商业机构）435、445、455。虚拟基础结构 420 提供共享 410 中的资源到一个或多个虚拟机 430、440、450 的动态映射（dynamicmapping）。虚拟机 430、440、450 中的每一个都运行操作系统 432、442、452 和高效存储应用 434、444、454。存储应用 434、444、454 可以是相同的（例如 Avamar 公司的 Avamar）或不同的应用程序，每个存储应用 434、444、454 可被传递到已根据高效存储方法存档的数据中，并使用其恢复功能在共享 410 中的硬件子系统上恢复或恢复这种数据，所述共享由虚拟基础结构提供，能够支持虚拟机 430、440、450 及其相关的虚拟存储系统。例如，在存储器 411 或 418 中存储的数据集的时间点版本可通过一个或多个服务器 413、415、416（和 / 或共享 410 中的其它资源）由虚拟机 430、440、450 访问，以基于例如来自用户 435、445、455 的恢复请求恢复所需的数据集（例如 N+N1+Nx 等）。

[0033] 现在参照图 5，用于提供数据保护的方法 500 被示出。其中可实施方法 500 的普通计算机结构包括其上保存原始数据集的存储装置、存档存储机构以及一个或多个硬件组件。通过在计算机结构上提供虚拟化层的步骤 502 和在虚拟化层上运行高效存储应用的步骤 504 开始处理 500。虚拟化层基于计算机结构的组件，将硬件组的规范化的代表项呈现给存储应用。

[0034] 存储应用的一个目的是在原始数据集丢失、损坏或其它破坏的情况下生成并存储原始数据集的时间点版本。由此在步骤 506 操作存储应用，如上面已描述的，从而生成并存储原始数据集的版本。

[0035] 在步骤 508，通过存储应用，原始数据集的时间点版本可被移动或复制到存档存储机构。原始数据集的时间点版本可随同存储应用本身被转移到存档存储机构，作为完整的虚拟机环境。随后，整个虚拟机环境可被转移或复制到硬件子系统中，在该硬件子系统中，存储应用的时间点恢复模块可被用于生成硬件子系统中的原始数据集的恢复版本，如步骤 510。可响应于由恢复模块从用户接收到的恢复请求而开始恢复处理。

[0036] 如上所述，虚拟化层或基础结构（例如图 1- 图 4 中所示）可采用多种形式以实现本发明。然而，在一个实施例中，在诸如 VMware ESX 或 GSX（或类似产品）的 VMware 虚拟化产品上运行高效存储应用。诸如 VMware ESX 的虚拟化产品相对是公知的并被所属领域的技术人员了解，此处对于描述本发明，全面的讨论是不必要的。在其它实施例中，由来自 XenSource 的虚拟化产品 Xen3.0（或其它版本）提供虚拟化层，并且在其它实施例中，来自微软公司和 / 或 Intel 公司的虚拟化产品被用于提供此处所述的虚拟化层。

[0037] 如上所述，图 1- 图 4 的 HEPA 系统的高效存储应用可进行变化以实现本发明。在一个实例中，存储应用是 Avamar 产品或由 EMC 公司（以前的 Avamar 技术公司）提供的套件（suite），或者存储应用是基于在这种软件套件中使用的高效存储和数据恢复方法学。例如，美国专利 No. 6810398 中描述的数据存储处理，该专利由 EMC 公司拥有，通过参考援引该申请。在其它实施例中，CDP 或其它高效产品可用作高效存储应用，例如：ArcWare、InfoStage、Continuum 或由 EVault 公司配销的其它存储应用；来自 Iron

Mountain Digital 的 LiveVault 和其它存储应用；以及来自美国网域存储技术有限公司 (NetworkAppliance, Inc.) 的高效存储应用，但不限于这些程序。

[0038] 另外，下面的描述是对用作本发明的高效存储应用的部分的 Avamar 技术的纵览。Avamar 专注于企业数据保护的需求，并计划更有效地专注于数据备份和恢复的挑战。Avamar 使用取得专利的公共因子分解技术解决了在来源处的数据压缩和备份数据冗余的挑战，该技术降低甚至消除了冗余数据。Avamar 代理商在系统上及在每个客户机上随时识别和过滤在单个系统的文件中存储的重复数据序列，从而在企业中每个独特的数据仅被备份一次，由此大大减少了必须在网络上传输并被存储在 Avamar 系统中的整体数据。作为该方法的结果，复制或编辑的文件、共享的应用程序、嵌入的附件以及甚至每天变化的数据库仅生成用于 Avamar 系统的少量增量式存储量。

[0039] Avamar 的公共因子分解技术允许变化的文件被有效分析并被分成平均尺寸为 12KB 的尺寸可变块，然后基于信息的内容为这些块分配 20 字节的内容地址。之后，Avamar 快速评测在 Avamar 系统中是否已经存储信息（例如在过去的某一点，信息是否被备份到该系统或其它系统上），如果是，不再备份数据，而仅存储指向现有信息的内容地址。通过在客户机上执行这种分析，Avamar 为已存储在 Avamar 内的数据有效地释放 600-1 减少量（放出 20 字节代表 12KB 的数据块），并且平均来说，当执行每天的全部备份时，在备份数据中提供了大于 100-1 的减少量。

[0040] 如 Avamar 产品和技术的这种描述所暗示的，其对了解 Avamar 以全面理解 Avamar 是如何起到寻找变化的文件然后将这些文件分成块或微粒 (atomic) 的作用（例如 Avamar 如何选择数据序列中的断点 (breakpoint) 以更好地确定改变了什么以及什么需要备份）是有用的。这种技术可被描述如下：以子文件级 (sub-file level) 执行冗余数据缩减的一个关键因素在于，在分析大量数据时有效确定数据的共有序列的方法。用于评测子文件级变化的大多数技术都使用固定块边界（其尺寸通常为 512 字节）等简单的方法来分析文件。然而，在许多情况下这些技术执行效果较差。如果在文件或文档 (documents) 的起始处出现轻微的变化，整个文件的内容被改变，从而导致用于检测整个文件的固定块尺寸分析随之改变。Avamar 已经开发了一种用于分析数据序列和划分数据的专利方法，该方法解决了固定块  $\delta$  (delta) 分析的缺点，使得在多个相关和不相关的计算机系统中可找到公共元素，而不需要在计算机之间进行通信并且不用考虑文件的数据内容。这种方法确定数据序列中的逻辑断点或“粘性字节”断点，以有效且实用地将数据集分成多份，从而产生最佳共性。如果文档或文件被编辑，使得新信息被放置在文件的开始处且使得从文件的中部删除现有数据，则由 Avamar 执行的文件的“粘性字节”因子就确保了整个文件的小百分量比被实际检测为已被改变。当用这种技术执行大数据库转储文件 (dump file) 的  $\delta$  分析时，该技术也是非常有效的，从而整个文件中仅有小百分量被检测为被改变（并随后在网络上被备份），而不管整个数据库的表格变化。

[0041] 本发明可以其它具体形式实现，而不脱离其精神或实质特征。在所有方面所描述的实施例均被认为是例证性的而非限制性的。因此本发明的范围由所附的权利要求而非由前面的描述所表示。所有落入权利要求的内涵和等效范围的变化均被包括在其范围内。

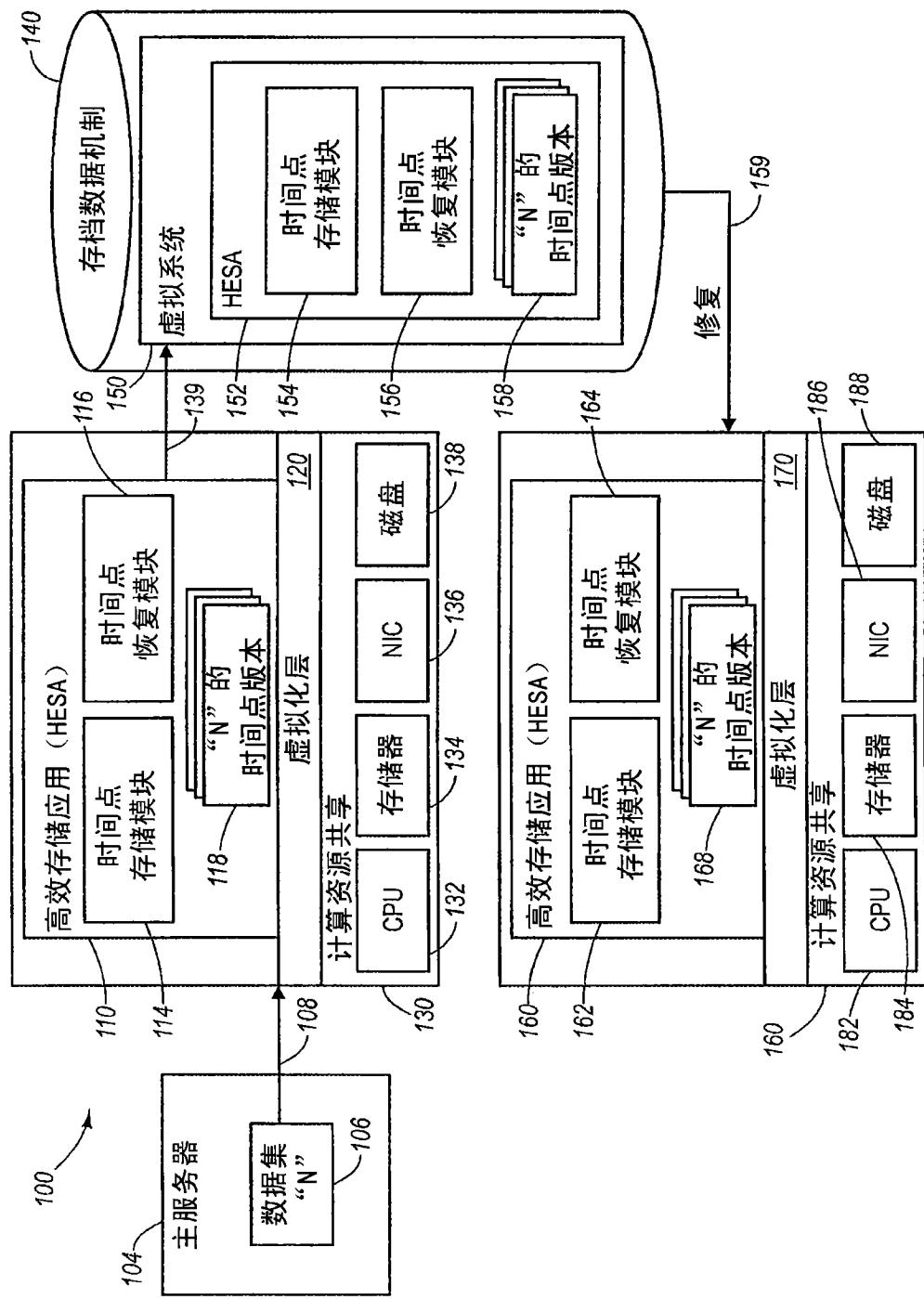


图1

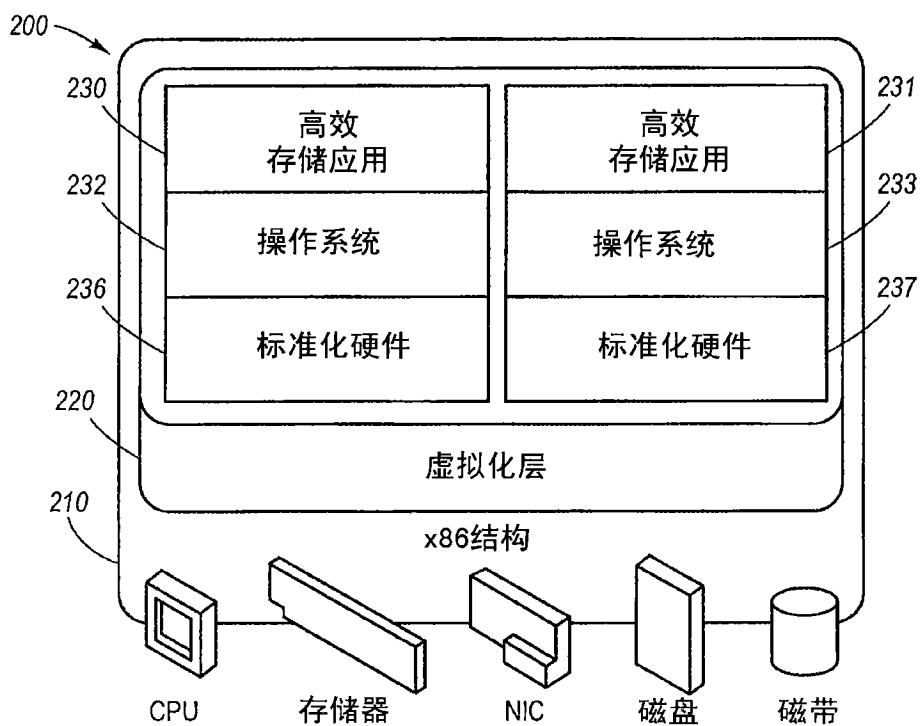


图 2

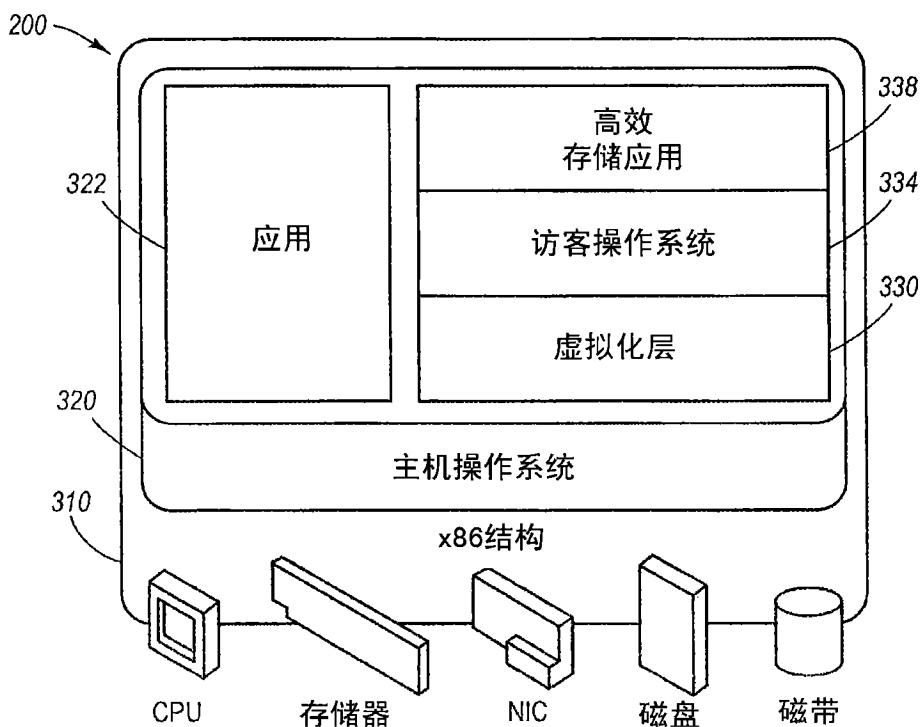


图 3

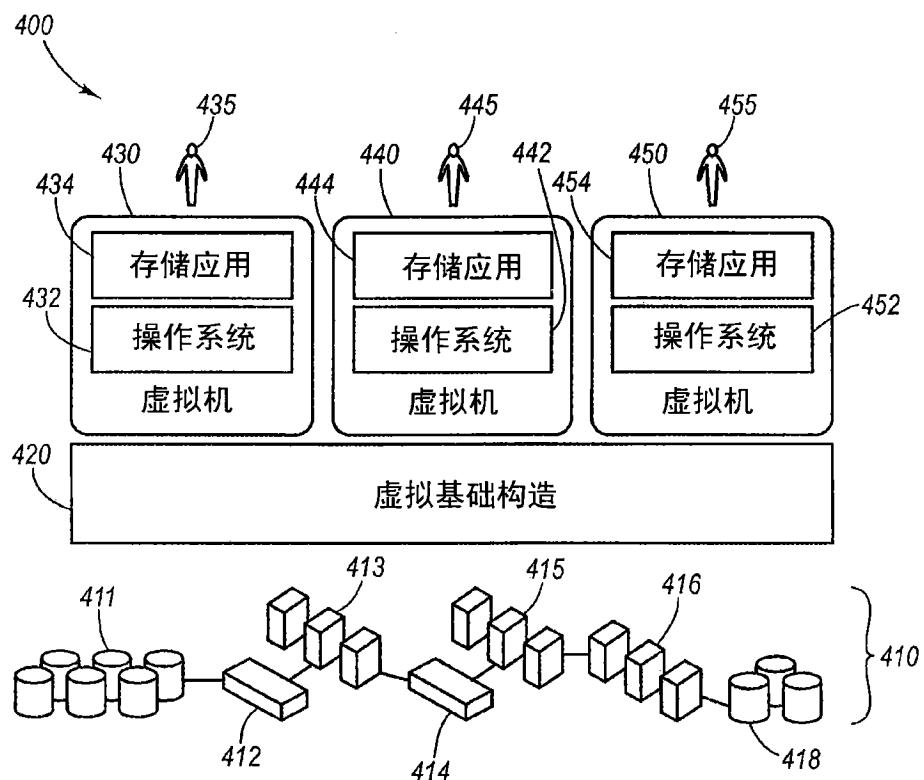


图 4

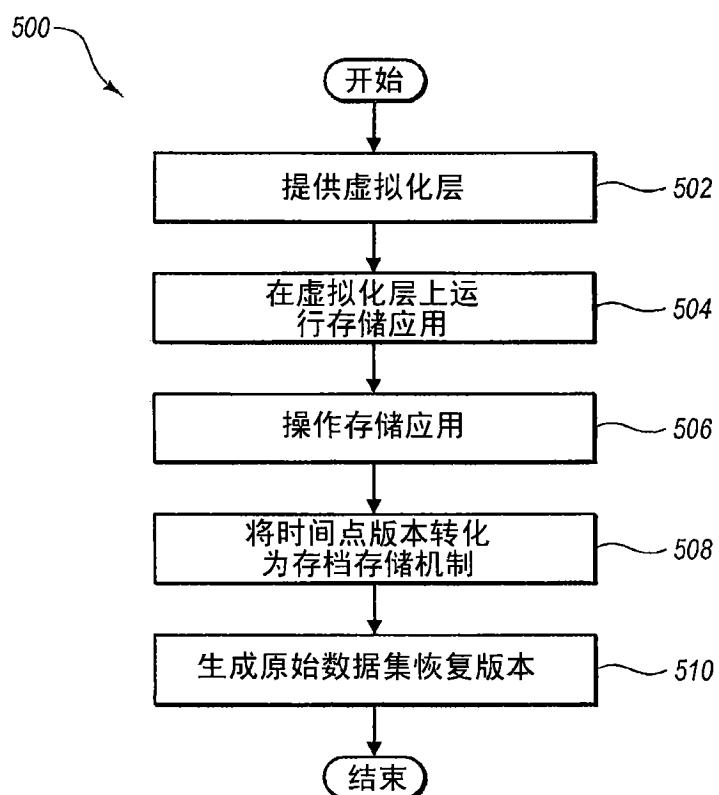


图 5