

## (12) 发明专利

(10) 授权公告号 CN 1906613 B

(45) 授权公告日 2010.10.20

(21) 申请号 200480040568.0

(56) 对比文件

(22) 申请日 2004.11.19

JP 特开 2000-163300 A, 2000.06.16, 全文.

## (30) 优先权数据

WO 03/060774 A1, 2003.07.24, 说明书第1  
页第16行到第7页第16行.

10/723,704 2003.11.26 US

CN 1447939 A, 2003.10.08, 全文.

10/862,505 2004.06.07 US

US 6363386 B1, 2002.03.26, 全文.

## (85) PCT申请进入国家阶段日

Dourish P et al. Presto:an experimental  
architecture for fluid interactive document  
spaces. ACM transactions on computer-human  
interaction ACM USA6 2.1999, 6(2), 133-  
161.

2006.07.17

审查员 王伟

## (86) PCT申请的申请数据

PCT/US2004/039038 2004.11.19

## (87) PCT申请的公布数据

W02005/055093 EN 2005.06.16

## (73) 专利权人 维瑞泰斯操作公司

地址 美国加利福尼亚

## (72) 发明人 杜拉巴加尤蒂·波萨库尔

纳尔·普莱默 卓塞弗·帕斯夸

## (74) 专利代理机构 中国国际贸易促进委员会专

利商标事务所 11038

代理人 董莘

## (51) Int. Cl.

G06F 17/30 (2006.01)

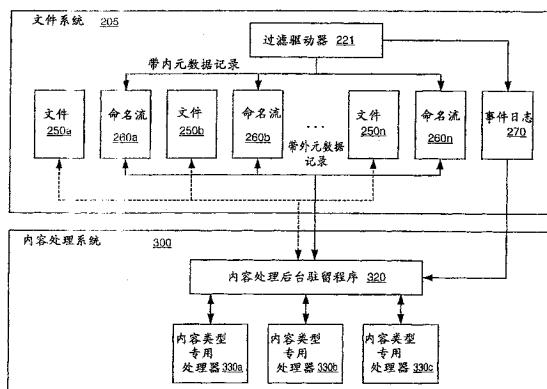
权利要求书 3 页 说明书 19 页 附图 8 页

## (54) 发明名称

产生可扩展的文件系统元数据和文件系统内  
容处理的系统和方法

## (57) 摘要

一种产生可扩展的文件系统元数据的系统和方法。在一个实施例中，系统可包括配置成保存数据的存储装置，和配置成管理对于存储装置的访问，以及保存文件系统内容的文件系统。文件系统还可被配置成检测文件系统内容访问事件，并响应检测文件系统内容访问事件，产生元数据记录，按照可扩展的自描述数据格式保存元数据记录。



1. 一种基于文件的存储系统,该存储系统中的文件系统将文件系统内容保存在与该存储系统相连接的存储装置中,所述文件系统内容包括多个文件,并且所述文件系统被配置成管理对于所述存储装置的应用访问,所述存储系统包括:

用于检测文件系统内容访问事件的装置;

用于响应于检测所述文件系统内容访问事件,产生指示所述文件系统内容访问事件的元数据记录的装置,其中按照可扩展的自描述数据格式,生成所述元数据记录,其中所述元数据记录中包含的各个数据元素被一个或多个对应的标记字段定界,其中所述元数据记录中包含的给定的各个数据元素具有如下的通用语法:

<descriptive\_tag>data element</descriptive\_tag>

其中“descriptive\_tag”定界符描述“data element”字段的某一方面;以及

用于将所述元数据记录保存在所述存储装置的文件系统内容中的装置。

2. 按照权利要求1所述的系统,其中检测所述文件系统内容访问事件在带内执行,其中所述带内检测与所述检测的文件系统内容访问事件的发生同步。

3. 按照权利要求1所述的系统,其中所述可扩展的自描述数据格式符合可扩展置标语言 XML 格式的版本。

4. 按照权利要求1所述的系统,其中所述元数据记录被保存在与给定文件相对应的命名流中。

5. 按照权利要求1所述的系统,其中所述元数据记录被保存在与所述文件系统相对应的事件日志中。

6. 按照权利要求1所述的系统,其中所述文件系统内容包括一个或多个元数据记录以及保存在所述多个文件的一个或多个中的文件数据。

7. 按照权利要求1所述的系统,还包括内容处理器和应用,其中所述内容处理器被配置成响应于带外检测所述文件系统内容访问事件,调用所述应用,其中所述带外检测发生在所述文件系统内容访问事件已经完成之后。

8. 按照权利要求1所述的系统,还包括查询引擎,应用借助于所述查询引擎来查询所述文件系统内容。

9. 一种基于文件的存储方法,其中文件系统将文件系统内容保存在存储装置中,所述文件系统内容包括多个文件,并且所述文件系统被配置成管理对于所述存储装置的应用访问,所述方法包括:

检测文件系统内容访问事件;

响应于检测所述文件系统内容访问事件,所述文件系统产生指示所述文件系统内容访问事件的元数据记录,其中按照可扩展的自描述数据格式,生成所述元数据记录,其中所述元数据记录中包含的各个数据元素被一个或多个对应的标记字段定界,其中所述元数据记录中包含的给定的各个数据元素具有如下的通用语法:

<descriptive-tag>data element</descriptive\_tag>

其中“descriptive\_tag”定界符描述“data element”字段的某一方面;以及

所述文件系统将所述元数据记录保存在所述存储装置的文件系统内容中。

10. 按照权利要求9所述的方法,其中检测所述文件系统内容访问事件在带内执行,其中所述带内检测与所述检测的文件系统内容访问事件的发生同步。

11. 按照权利要求 9 所述的方法, 其中所述可扩展的自描述数据格式符合可扩展置标语言 XML 格式的版本。

12. 按照权利要求 9 所述的方法, 其中所述元数据记录被保存在与给定文件相对应的命名流中。

13. 按照权利要求 9 所述的方法, 其中所述元数据记录被保存在事件日志中。

14. 按照权利要求 9 所述的方法, 其中所述文件系统内容包括一个或多个元数据记录和保存在多个文件的一个或多个中的文件数据。

15. 按照权利要求 9 所述的方法, 还包括 :

响应于带外检测所述文件系统内容访问事件而调用应用, 其中所述带外检测发生在所述文件系统内容访问事件已经完成之后。

16. 按照权利要求 9 所述的方法, 还包括 :

查询所述文件系统内容。

17. 一种基于文件的存储系统, 包括 :

配置成保存文件系统内容的存储装置, 所述文件系统内容包括多个文件 ;

文件系统, 被配置成管理对于所述存储装置的应用访问, 将所述文件系统内容存储在所述存储装置中, 执行文件系统内容访问事件的带内检测, 响应地生成指示所述文件系统内容访问事件的对应事件记录, 并且将所述对应事件记录保存在所述文件系统内容中 ; 以及

内容处理器, 配置成执行所述事件记录的带外检测 ;

其中所述文件系统所作的文件系统内容访问事件的带内检测与所述检测到的文件系统内容访问事件的发生同步 ; 以及

其中所述内容处理器所作的文件系统访问事件的带外检测发生在所述文件系统内容访问事件已经完成之后。

18. 按照权利要求 17 所述的系统, 其中所述内容处理器还被配置成响应于执行所述事件记录的带外检测而调用应用。

19. 按照权利要求 17 所述的系统, 其中所述内容处理器还被配置成向与所述存储系统相连接的查询系统通知所述文件系统内容访问事件中的一个给定的文件系统内容访问事件, 响应于此, 所述查询系统被配置成对于所述给定的文件系统内容访问事件, 通过更新所述文件系统内容的索引以使得所述索引随着所述文件系统内容而变化, 而保持所述索引的参考完整性。

20. 按照权利要求 17 所述的系统, 其中一个或多个所述对应事件记录被保存在与给定文件相对应的命名流中。

21. 按照权利要求 17 所述的系统, 其中按照可扩展的自描述数据格式保存所述对应事件记录。

22. 按照权利要求 21 所述的系统, 其中所述可扩展的自描述数据格式符合可扩展置标语言 XML 格式的版本。

23. 一种基于文件的存储方法, 包括 :

文件系统将文件系统内容保存在存储装置上, 其中所述文件系统内容包括多个文件, 所述文件系统被配置成管理对于所述存储装置的应用访问 ;

所述文件系统执行文件系统内容访问事件的带内检测，其中所述文件系统所作的文件系统内容访问事件的带内检测与所述检测到的文件系统内容访问事件的发生同步；

响应于所述带内检测，所述文件系统生成指示所述文件系统内容访问事件的对应事件记录，并且将所述对应事件记录保存在所述文件系统内容中；

内容处理器执行所述事件记录的带外检测，其中所述内容处理器所作的文件系统访问事件的带外检测发生在所述文件系统内容访问事件已经完成之后。

24. 按照权利要求 23 所述的方法，还包括：

响应于执行所述事件记录的带外检测而调用应用。

25. 按照权利要求 23 所述的方法，还包括：

所述内容处理器向查询系统通知所述文件系统内容访问事件中的一个给定的文件系统内容访问事件，

响应于此，所述查询系统被配置成对于所述给定的文件系统内容访问事件，通过更新所述文件系统内容的索引以使得所述索引随着所述文件系统内容而变化，而保持所述索引的参考完整性。

26. 按照权利要求 23 所述的方法，其中一个或多个所述对应事件记录被保存在与给定文件相对应的命名流中。

27. 按照权利要求 26 所述的方法，其中按照可扩展的自描述数据格式保存所述对应事件记录。

28. 按照权利要求 27 所述的方法，其中所述可扩展的自描述数据格式符合可扩展置标语言 XML 格式的版本。

## 产生可扩展的文件系统元数据和文件系统内容处理的系统 和方法

### 技术领域

[0001] 本发明涉及计算机系统,更具体地说,涉及基于文件的存储系统。

### [0002] 背景技术

[0003] 计算机系统经常处理大量的信息,包括应用数据和配置成处理这样的数据的可执行代码。在众多的实施例中,计算机系统提供各种配置成存储数据的大容量存储装置,比如磁盘驱动器和光盘驱动器,磁带驱动器等。为了提供常规并且系统的接口,通过所述接口访问它们保存的数据,这样的存储装置经常被软件,比如操作系统组织成文件的分级结构。文件通常定义存储装置内,用户能够操作的数据粒度的最小级别,不过各种应用和操作系统可在比整个文件更低的粒度级别上处理文件内的数据。

[0004] 在一些基于文件的计算机系统中,除了文件本身之外,还可保存关于文件的各种信息,也称为元数据。但是,在典型的常规计算机系统中,对元数据的支持局限于如果需要额外的文件信息,那么不会被增加的少量的固定类型的文件属性。此外,访问元数据的程序局限于识别给定文件,随后取回其相关元数据的程序,而不是更灵活的程序。

[0005] 另外,在复杂的计算环境中,可提供众多的应用,每个应用可与文件和存储装置交互作用,从而实现特定的功能。在一些情况下,几个应用中的每一个可被配置成执行复杂的多步骤事务的一个特殊步骤。但是,这样的应用相互交互作用或协调的能力有限,为了协调事务处理,在系统用户一方,需要相当程度的干预。

### [0006] 发明内容

[0007] 公开了产生可扩展的文件系统元数据的系统和方法的各种实施例。在一个实施例中,系统可包括配置成保存数据的存储装置,和配置成管理对于存储装置的访问,以及保存文件系统内容的文件系统。文件系统还可被配置成检测文件系统内容访问事件,并响应检测文件系统内容访问事件,产生元数据记录,按照可扩展的自描述数据格式保存元数据记录。

[0008] 在系统的一种具体实施方式中,如下更详细所述,检测文件系统内容访问事件可带内执行。在系统的另一种具体实施方式中,可扩展的自描述数据格式可符合可扩展置标语言 (XML) 格式的版本。

[0009] 另外还预期一种方法,在一个实施例中,所述方法包括保存文件系统内容,检测文件系统内容访问事件,响应检测文件系统内容访问事件,产生元数据记录,并按照可扩展的自描述数据格式保存该元数据记录。

[0010] 根据本发明,提供了一种基于文件的存储系统,该存储系统中的文件系统将文件系统内容保存在与该存储系统相连接的存储装置中,所述文件系统内容包括多个文件,并且所述文件系统被配置成管理对于所述存储装置的应用访问,所述存储系统包括:用于检测文件系统内容访问事件的装置;以及用于响应于检测所述文件系统内容访问事件,产生指示所述文件系统内容访问事件的元数据记录的装置,其中按照可扩展的自描述数据格式,生成所述元数据记录,其中所述元数据记录中包含的各个数据元素被一个或多个对应

的标记字段定界,其中所述元数据记录中包含的给定的各个数据元素具有如下的通用语法 :<descriptive\_tag>data element</descriptive\_tag>,其中“descriptive\_tag”定界符描述“data element”字段的某一方面 ;以及用于将所述元数据记录保存在所述存储装置的文件系统内容中的装置。

[0011] 根据本发明,还提供了一种基于文件的存储方法,其中文件系统将文件系统内容保存在存储装置中,所述文件系统内容包括多个文件,并且所述文件系统被配置成管理对于所述存储装置的应用访问,该方法包括 :检测文件系统内容访问事件 ;响应于检测所述文件系统内容访问事件,所述文件系统产生指示所述文件系统内容访问事件的元数据记录,其中按照可扩展的自描述数据格式,生成所述元数据记录,其中所述元数据记录中包含的各个数据元素被一个或多个对应的标记字段定界,其中所述元数据记录中包含的给定的各个数据元素具有如下的通用语法 :<descriptive\_tag>data element</descriptive\_tag>,其中“descriptive\_tag”定界符描述“data element”字段的某一方面 ;以及所述文件系统将所述元数据记录保存在所述存储装置的文件系统内容中。

[0012] 根据本发明,还提供了一种用于生成可扩展文件系统元数据以及文件系统内容处理的系统,包括 :配置成保存文件系统内容的存储装置,所述文件系统内容包括多个文件 ;文件系统,被配置成管理对于所述存储装置的应用访问,将所述文件系统内容存储在所述存储装置中,执行文件系统内容访问事件的带内检测,响应地生成指示所述文件系统内容访问事件的对应事件记录,并且将所述对应的事件记录保存在所述文件系统内容中 ;以及内容处理器,配置成执行所述事件记录的带外检测 ;其中所述文件系统所作的文件系统内容访问事件的带内检测与所述检测到的文件系统内容访问事件的发生同步 ;以及其中所述内容处理器所作的文件系统访问事件的带外检测发生在所述文件系统内容访问事件已经完成之后。

[0013] 根据本发明,还提供了一种用于生成可扩展文件系统元数据以及文件系统内容处理的方法,包括 :文件系统将文件系统内容保存在存储装置上,其中所述文件系统内容包括多个文件,所述文件系统被配置成管理对于所述存储装置的应用访问 ;所述文件系统执行文件系统内容访问事件的带内检测,其中所述文件系统所作的文件系统内容访问事件的带内检测与所述检测到的文件系统内容访问事件的发生同步 ;响应于所述带内检测,所述文件系统生成指示所述文件系统内容访问事件的对应事件记录,并且将所述对应的事件记录保存在所述文件系统内容中 ;内容处理器执行所述事件记录的带外检测,其中所述内容处理器所作的文件系统访问事件的带外检测发生在所述文件系统内容访问事件已经完成之后。

#### [0014] 附图说明

[0015] 图 1 是图解说明存储系统的一个实施例的方框图。

[0016] 图 2 是图解说明基于软件的存储系统体系结构的一个实施例及其与存储装置的接口的方框图。

[0017] 图 3 是图解说明存储管理系统的一个实施例的方框图。

[0018] 图 4 是图解说明配置成保存文件和相关元数据的文件系统的一个实施例的方框图。

[0019] 图 5 是图解说明元数据产生方法的一个实施例的流程图。

- [0020] 图 6 是图解说明内容处理系统的一个实施例的方框图。
- [0021] 图 7 是图解说明包括内容处理系统的存储系统的操作方法的一个实施例的流程图。
- [0022] 图 8 是图解说明查询系统的一个实施例的方框图。
- [0023] 图 9 是图解说明包括查询系统的存储系统的操作方法的一个实施例的流程图。
- [0024] 虽然本发明容许各种修改和备选形式,不过附图中举例表示了本发明的具体实施例,这里将具体描述这些实施例。但是,应明白附图及其详细说明并不意图将本发明局限于公开的特定形式,相反,目的是覆盖落入由附加的权利要求限定的本发明的精神和范围内的所有修改、等同物和替代物。

[0025] **具体实施方式**

[0026] 计算机系统概述

[0027] 现在参见图 1,图中表示了计算机系统的一个实施例的方框图。在图解说明的实施例中,系统 10 包括通过系统互连 40 与多个存储装置 30a 和 30b 耦接的多个主机设备 20a 和 20b。此外,在图解说明的实施例中,主机设备 20b 包括系统存储器 25。为了便于引用,这里由跟有字母的附图标记所指的部件将总体由附图标标记单独表示。例如,主机设备 20a 和 20b 以及存储装置 30a 和 30b 可被总体称为主机设备 20 和存储装置 30。

[0028] 在系统 10 的各个实施例中,主机设备 20 被配置成访问保存在一个或多个存储装置 30 上的数据。在一个实施例中,系统 10 可在单一的计算机系统内被实现成例如集成的存储服务器。在这样的实施例中,主机设备 20 可以是单独的处理器,系统存储器 25 可以是高速缓冲存储器,例如静态 RAM(SRAM),存储装置 30 可以是大容量存储装置,比如硬盘驱动器或者其它可写或可重写的媒体,系统互连 40 可包括外围总线互连,例如外设部件互连(PCI)总线。在一些这样的实施例中,系统互连 40 可包括主机设备 20 和存储装置 30 之间的几种互连。例如,系统互连 40 可包括配置成耦接主机设备 20 的一个或多个处理器总线(未示出),配置成使处理器总线和一个或多个外围总线耦接的一个或多个总线桥接器(未示出),和配置成使外围总线与存储装置 30 耦接的一个或多个存储装置接口(未示出)。各个实施例中的存储装置接口类型可包括小型计算机系统接口(SCSI),AT 附加包接口(ATAPI),火线,和 / 或通用串行总线(USB),不过包括其它接口类型的众多的备选实施例也是可能的。

[0029] 在单一计算机系统内实现的系统 10 的一个实施例中,系统 10 可被配置成为一个或多个其它计算机系统(未示出)提供大多数数据存储需求,并且可被配置成与这样的其它计算机系统通信。在一个备选实施例中,系统 10 可被配置成分布式存储系统,例如存储区域网(SAN)。在这样的实施例中,主机设备 20 可以是单独的计算机系统,例如服务器系统,系统存储器 25 可由一种或多种动态 RAM(DRAM)构成,存储装置 30 可以是均包括一个或多个硬盘驱动器或者其它类型的存储器的独立存储节点,系统互连 40 可以是诸如以太网或光纤通道之类的通信网络。系统 10 的分布式存储配置可方便存储系统容量,以及主机设备和存储装置之间的数据带宽的按比例缩放。

[0030] 在另一实施例中,系统 10 可被配置成混合存储系统,一些存储装置 30 作为一些主机设备 20 被集成到相同计算机系统内,而其它存储装置 30 被配置成通过网络与其它主机设备 20 耦接的独立设备。在这种混合存储系统中,系统互连 40 可包含各种互连机构,例如

上面说明的外围总线和网络互连。

[0031] 注意虽然图1中图解说明了两个主机设备20和两个存储装置30,不过在备选实施例中,系统10可具有任意数目的每种这些设备。另外,在系统10的一些实施例中,在其它主机设备20或存储装置30中可以采用系统存储器25的一个以上的实例。此外,在一些实施例中,给定的系统存储器可存在于主机设备20和存储装置30之外,并且可直接与给定的主机设备20或存储装置30耦接,或者通过系统互连40与给定的主机设备20或存储装置30间接耦接。

[0032] 在系统10的许多实施例中,一个或多个主机设备20可被配置成执行程序指令和参考数据,从而实现计算功能。在一些实施例中,系统存储器25可以是配置成保存这样的程序指令和数据的计算机可访问媒体的一个实施例。但是,在其它实施例中,程序指令和/或数据可被接收、发送或保存在不同类型的计算机可访问媒体上。一般来说,计算机可访问媒体可包括作为存储装置30包括在系统10中的存储媒体或存储器媒体,例如磁性媒体或光学媒体,比如磁盘或CD-ROM。计算机可访问媒体还可包括可作为系统存储器25包括在系统10的一些实施例中的易失性或非易失性媒体,比如RAM(例如SDRAM、DDR SDRAM、RDRAM、SRAM等),ROM等。此外,计算机可访问媒体可包括可作为系统互连40包括在系统10的一些实施例中的通过通信媒体,比如网络和/或无线链路传送的传输媒体或信号,比如电信号、电磁信号或数字信号。

[0033] 在一些实施例中,保存在如上所述的计算机可访问媒体内的程序指令和数据可实现操作系统,操作系统再提供各种应用的执行环境。例如,给定的主机设备20可被配置成执行一种版本的Microsoft Windows操作系统,Unix/Linux操作系统,Apple Macintosh操作系统,或者另一适当的操作系统。另外,给定的主机设备可被配置成执行应用,比如字处理器,web浏览器和/或服务器,电子邮件客户机和/或服务器,和多媒体应用,以及其它许多可能的应用。

[0034] 在给定主机设备20上运行期间,操作系统或给定的应用可产生从给定存储装置30装入数据或者将数据保存到给定存储装置30的请求。例如,与操作系统或应用本身的多个部分对应的代码可被保存在给定存储装置30上,从而响应所需操作系统例程或应用的调用,可收回对应的代码以便执行。类似地,操作系统或应用可产生待保存的数据。

[0035] 在一些实施例中,保存在存储装置30上的数据的移动和处理可由基于软件的存储系统管理。图2中图解说明了一个这样的实施例,图2表示通过内容感知存储系统150与多个存储装置230A-C连接的应用层100。图2内图解说明的一些模块可被配置成按照用户执行模式或“用户空间”运行,而其它一些模块可被配置成按照内核执行模式或“内核空间”运行。在图解说明的实施例中,应用层100包括多个用户空间软件进程112A-C。每个进程通过多个应用编程接口(API)114A-C连接存储系统150,以致每个进程112可访问每个API114A-C。存储系统150包括内核空间存储管理系统200,内核空间存储管理系统200被表示成通过API 114A与应用层100连接;用户空间内容处理系统300,用户空间内容处理系统300被表示成通过API114B与应用层100连接;和用户空间查询系统400,用户空间查询系统400被表示成通过API 114C与应用层100连接。在图解说明的实施例中,存储管理系统200还与存储装置230A-C连接。

[0036] 预期在一些实施例中,可以实现任意数目的进程112和/或存储装置230。还预期

在一些实施例中,处理系统 300 和 / 或查询系统 400 的全部或者部分可在内核空间中实现,并且在一些实施例中,配置成在内核空间中运行的进程可被配置成通过 API 114 或者专用于内容空间进程的其它 API,访问存储系统 150。

[0037] 在一个实施例中,每个进程 112 可对应于给定的用户应用,并且每个进程 112 可被配置成通过对 API 114 的调用,访问存储装置 230A-C。API 114 向进程 112 提供对存储系统 150 的各个组件的访问。例如,在一个实施例中,API 114 可包括给定进程 112 可调用的由存储系统 150 暴露的函数调用,而在其它实施例中,API 114 可支持其它类型的进程间通信。在一个实施例中,存储装置 230 可作为图 1 的存储装置 30 的例证。另外,在一个实施例中,存储系统 150 的任意组件和 / 或任意进程 112 可被配置成作为保存在计算机可访问媒体,比如图 1 的系统存储器 25 内的程序指令和数据,在一个或多个图 1 的存储装置 20 上运行。

[0038] 如下结合图 3-9 的描述更详细所述,存储系统 150 可被配置成提供各种存储相关服务。例如,在一个实施例中,存储管理系统 200 可被配置成利用文件系统组织存储装置 230 保存的数据,以致各种进程 112 可以分级文件的形式保存和操作数据。另外,存储系统 150 可被配置成监视访问和 / 或操作保存的数据的操作,并产生这种操作的记录。查询系统 400 可被配置成向进程 112 提供接口,通过所述接口,查询存储装置 230 保存的数据,以及访问这种数据的操作的记录。最后,内容处理系统 300 可被配置成监视存储管理系统 200 的活动,和创建这种活动的附加记录。如下更详细所述,内容处理系统 300 还可被配置成响应检测到这样的活动,调用一个或多个应用,从而提供根据存储管理系统 200 的活动,而不是直接根据应用层 100 的活动(例如,根据进程 112 的数据请求),发起内容处理活动的机构。预期在存储系统 150 的一些实施例中,在保留存储管理系统 200 提供的服务的同时,内容处理系统 300,查询系统 400,或者这两个系统可被省略。

#### [0039] 存储管理系统和文件系统

[0040] 如上所述,在一些实施例中,存储管理系统 200 可提供将存储装置 20 提供的存储空间组织成文件的数据结构和控制结构。在各个实施例中,数据结构可包括一个或多个表格,所述一个或多个表格被配置成保存信息,比如每个文件的身份,其在存储装置 230 内的位置(例如对特定存储装置内的特定物理位置的映射),以及关于每个文件的其它信息,如下更详细所述。另外,在各个实施例中,控制结构可包括操作文件的可执行例程,比如改变文件身份和修改文件内容的函数调用。总的来说,这里可将这些数据结构和控制结构称为文件系统,由给定的文件系统实现的特定数据格式和协议可被称为文件系统的格式。

[0041] 在一些实施例中,文件系统可被集成到操作系统中,以致对保存在存储装置 230 上的数据的任何访问由文件系统的控制结构和数据结构管理。不同的操作系统可实现利用不同格式的不同的本机文件系统,但是在一些实施例中,给定的操作系统可包括支持多种不同的文件系统格式,包括其它操作系统特有的文件系统格式的文件系统。在这样的实施例中,文件系统支持的各种文件系统格式被称为本地文件系统。另外,在一些实施例中,可利用分层排列的多层功能,实现文件系统,如图 3 中所示。

[0042] 图 3 图解说明存储管理系统 200 的一个实施例。在图解说明的实施例中,存储管理系统包括配置成与一个或多个设备驱动器 224 连接的文件系统 205,一个或多个设备驱动器 224 又被配置成与存储装置 230 连接。如在图 2 的存储系统 150 内所示,存储管理系统

200 的组件可被配置成在内核空间中执行；但是，预期在一些实施例中，存储管理系统 200 的一些组件可被配置成在用户空间中执行。另外，在一个实施例中，存储管理系统 200 的任意组件可被配置成作为保存在计算机可访问媒体，比如图 1 的系统存储器 25 内的程序指令和数据，在图 1 的一个或多个主机设备 20 上执行。

[0043] 如同关于图 1 的系统 10 所述那样，给定的主机设备 20 可和给定的存储装置 30 存在于不同的计算机系统上，并可通过网络访问该存储装置。类似地，就存储管理系统 200 来说，在一个实施例中，给定的进程，比如进程 112A 可远程执行，并可通过网络访问存储装置 230。在图解说明的实施例中，文件系统 205 包括支持远程进程对文件系统的访问的网络协议 225。在一些实施例中，网络协议 225 可包括对网络文件系统 (NFS) 协议或者公用因特网文件系统 (CIFS) 协议的支持，不过预期可以采用任何适当的网络协议，并且在一些实施例中可以支持多个这样的协议。

[0044] 文件系统 205 可被配置成支持多个本地文件系统。在图解说明的实施例中，文件系统 205 包括 VERITAS (VxFS) 格式本地文件系统 240A, Berkeley 快速文件系统 (FFS) 格式本地文件系统 240B, 和专有 (X) 格式本地文件系统 240X。但是，预期在一些实施例中，文件系统 205 可支持许多本地文件系统格式或者它们的组合。为了向各个本地文件系统 240 提供公共接口，文件系统 205 包括一个虚拟文件系统 222。在一个实施例中，虚拟文件系统 222 可被配置成将源于进程 112 的文件系统操作转换成适合于每个操作所针对的特定本地文件系统 240 的格式。另外，在图解说明的实施例中，存储管理系统 200 包括设备驱动器 224，通过设备驱动器 224，本地文件系统 240 可访问存储装置 230。设备驱动器 224 可实现特定于存储装置 230 采用的接口的类型的数据传输协议。例如，在一个实施例中，设备驱动器 224 可提供通过 SCSI 和 ATAPI 接口传送数据的支持，不过在其它实施例中，设备驱动器 224 可支持其它类型的接口和它们的组合。

[0045] 在图解说明的实施例中，文件系统 205 还包括过滤驱动器 221。在一些实施例中，过滤驱动器 221 可被配置成监视进入文件系统 205 的每个操作，随后检测特定类型的操作，导致进行另外的操作或者改变检测到的操作的行为。例如，在一个实施例中，过滤驱动器 221 可被配置成将多个写操作组合成单一的写操作，从而提高文件系统性能。在另一实施例中，过滤驱动器 221 可被配置成在检测到对文件的写操作之后，计算文件的签名。在另一实施例中，过滤驱动器 221 可被配置成在检测到对特定文件的某些类型的操作之后，保存与这些文件相关的信息，比如记录，如下更详细所述。预期在一些实施例中，过滤驱动器 221 可被配置成实现前述操作，包括未明确提及的其它过滤操作的一个或多个组合。

[0046] 配置成当文件系统操作被请求或处理时，检测文件系统操作的过滤驱动器 221 的一个实施例可被认为进行这种操作的“带内”检测。另一方面，这样的检测可被认为是与所检测的操作或事件的发生同步。在一些实施例中，响应操作的带内检测而采取的处理动作可能影响该操作被如何完成。例如，文件读取操作的带内检测可能导致该操作的取消，如果该操作的发起者未被充分特许访问所请求的文件。在一些实施例中，操作的带内检测可能不会对操作本身的完成带来任何影响，但是可能产生另外的操作，比如将检测到的操作的发生记录在元数据记录中，如下所述。

[0047] 相反，在文件系统操作或事件发生之后，可检测文件系统操作或事件，以致可在所述操作或事件完成之后发生检测。这样的检测可被称为“带外”检测或者与所检测的操作或

事件异步。例如，用户进程 112 可定期检查文件以确定其长度。文件长度可能已在自用户进程 112 最后一次检查之来的任何时间发生变化，但是对于已改变文件长度的操作来说，所述检查可以是带外检查。在一些情况下，带外检测可能不能检测某些事件。参见前面的例子，自用户进程 112 最后一次检查以来，文件长度可能已变化多次，但是只有最后一次变化才能被检测。

[0048] 注意虽然可带内检测操作或事件，不过响应这样的检测而采取的动作可在所检测的操作完成之前或之后发生。参见前面的例子，在一个实施例中，修改所检查文件的长度的每个操作中被带内检测和记录。用户进程 112 可被配置成定期检查记录以确定文件长度。由于长度修改操作被带内检测和记录，因此用户进程 112 可考虑每个这样的操作，即使它可在这些操作发生之后这样做。

[0049] 注意过滤驱动器 221 是文件系统 205 的一部分，而不是用户空间 210 内的应用或进程。从而，过滤驱动器 221 可被配置成独立于用户空间 210 内的应用和进程工作。另一方面或者除了上述之外，过滤驱动器 221 可被配置成响应从用户空间 210 内的应用或进程收到的请求执行操作。

[0050] 另外注意在一些实施例中，内核空间 220 可包括产生对存储装置 230 的进程（未示出），该进程类似于用户空间进程 112。在这样的实施例中，在内核空间 220 中执行的进程可被配置成按照为用户空间进程 112 类似的方式，通过内核模式 API（未示出）访问文件系统 205。从而，在一些实施例中，对存储装置 230 的所有访问可由文件系统 205 处理，与发起访问操作的进程的类型或空间无关。

[0051] 存储管理系统 200 和文件系统 205 的众多备选实施例是可能的。例如，文件系统 205 可支持不同数目和格式的本地文件系统 240，或者可只支持单一的本地文件系统 240。在一些实施例中，网络协议 225 可被省略或者被集成到在文件系统 205 外的存储管理系统 200 的一部分中。类似地，在一些实施例中，虚拟文件系统 222 可被省略或者禁用，例如如果只有单一的本地文件系统 240 在使用。另外，在一些实施例中，可在文件系统 205 的不同的层中实现过滤驱动器 221。例如，在一个实施例中，过滤驱动器 221 可被集成到虚拟文件系统 222 中，而在另一实施例中，过滤驱动器 221 的一个实例可在每个本地文件系统 240 中实现。

#### [0052] 文件和元数据

[0053] 如上所述，文件系统 205 可被配置成管理对保存在存储装置 230 上的多个文件的访问。在许多实施例中，每个保存的文件可具有由文件系统用于将每个文件与其它文件区分开的相关身份。在文件系统 205 的一个实施例中，文件的身份可以是文件名，所述文件名可包括诸如“filename.txt”之类的一串字符。但是，在实现文件分层结构，比如分层的文件夹或目录的文件系统 205 的实施例中，文件分层结构的所有或部分可被包括在文件身份中。例如，名为“file1.txt”的给定文件可存在于目录“smith”中，而目录“smith”又存在于目录“users”中。目录“users”可存在于目录“test1”中，目录“test1”是文件系统 205 内的顶级或根级别目录。在一些实施例中，文件系统 205 可定义单一的“根目录”来包括所有的根级别目录，没有任何更高级别的目录包括该根目录。在其它实施例中，多个顶级目录可同时存在，以致没有任何更同级别的目录包括任何顶级目录。给定文件位于其中的特定文件夹或目录的名称可被称为给定文件的路径或路径名。

[0054] 在实现文件分层结构的文件系统 205 的一些实施例中,通过在文件的路径中列出每个目录以及文件名,可规定给定文件的身份。参见上面给出的例子,名为“file1.txt”的文件的给定实例的身份可被规定为“/test1/users/smith/file1.txt”。注意在文件系统 205 的一些实施例中,文件名独自不足以唯一地识别给定文件,而包括路径信息的完全规定的文件身份足以唯一地识别给定文件。例如,可存在识别为“/test2/users/smith/file1.txt”的文件,虽然该文件与前面提及的文件共用相同的文件名,不过借助其路径,该文件是清楚的。注意利用路径和文件名信息表示给定文件身份的其它方法是可能的。例如,不同的字符可被用于对目录 / 文件夹名称和文件名定界,或者可按照不同的顺序规定目标 / 文件夹名称和文件名。

[0055] 文件系统 205 管理的文件可保存采取许多编码格式中的任意一种的应用数据或程序信息,这些应用数据或程序信息可被总称为文件数据。例如,给定的文件可保存 ASCII- 编码格式的纯文本或者专有应用格式,例如特定的字处理器或者电子数据表编码格式的数据。另外,给定文件可保存视频或音频数据或者二进制格式的可执行程序指令。预期众多的其它类型的数据和编码格式,以及数据和编码格式的组合可在作为文件数据的文件中使用。

[0056] 除了如上所述管理对存储装置的访问,保存在存储装置上的各个文件,以及这些文件中的文件数据之外,在一些实施例中,文件系统 205 可被配置成保存和一个或多个给定文件对应的信息,该信息可被称为元数据。一般来说,元数据可包含与文件相关的任意类型的信息。在各个实施例中,元数据可包括诸如(但不限于)文件身份,大小,所有权和文件访问许可之类的信息。元数据还可包括自由形式或者用户定义的数据,比如对应于文件系统操作的记录,如下更详细所述。在一些实施例中,包括元数据的信息可被预定义(即硬编码)到文件系统 205 中,例如作为由文件系统 205 的卖方或集成者定义的一批元数据类型。在一些实施例中,文件系统 205 可被配置成在操作期间,产生新类型的元数据定义。在其它实施例中,在文件系统 205 之外的一个或多个应用进程 112 可定义将由文件系统 205 管理的新的元数据,例如通过为此目的定义的 API 的实例。预期在一些实施例中可以采用定义元数据的这种技术的组合。无论如何定义的对应于文件的元数据,以及文件的数据内容可被总称为文件系统内容。

[0057] 图 4 图解说明配置成保存文件和相关元数据的文件系统的一个实施例。图 4 中所示的文件系统 205 的实施例可包括在图 3 的实施例中图解说明的那些部件;但是,为了清楚起见,这些部件中的一些未被示出。在图解说明的实施例中,文件系统 205 包括过滤驱动器 221,任意数目的文件 250a-n,目录 255,与每个文件 250a-n 相关的相应的命名流 260a-n,与目录 255 相关的相应命名流 260,和事件日志 270。注意文件 250a-n 或命名流 260a-n 之一的类属实例可被分别称为文件 250 或命名流 260,文件 250a-n 或命名流 260a-n 可分别被称为文件 250 和命名流 260。如上所述,文件 250 和命名流 260 可被总称为文件系统内容。在一些实施例中,目录 255 也可被包括在文件系统内容中。

[0058] 文件 250 可代表由文件系统 205 管理的文件,在各个实施例中可被配置成保存如上所述的各种数据和程序指令。在文件系统 205 的分层实现中,一个或多个文件 250 可被包括在一个目录 255 中(它也可被称为文件夹)。在各个实施例中,可提供任意数目的目录 255,一些目录 25 可被配置成分层包括其它目录 255 以及文件 250。在图解说明的实施例

中,每个文件 250 和目录 255 具有对应的命名流 260。每个命名流 260 可被配置成保存属于其对应文件的元数据。注意文件 250,目录 255 和命名流 260 可被物理保存在一个或多个存储装置,例如图 2 的存储装置 230 上。但是,为了便于举例说明,文件 250,目录 255 和命名流 260 被表示成概念上驻留在文件系统 205 内。另外,预期在一些实施例中,从元数据生成的观点来看,目录 255 类似于文件 250,显然在这样的实施例中,在下面的讨论中对文件 250 的引用也适用于目录 255。

[0059] 在一些实施例中,过滤驱动器 221 可被配置成访问保存在给定文件 250 中的文件数据。例如,过滤驱动器 221 可被配置成检测文件系统 205 收到的读取和 / 或写入操作,并且作为响应,可导致从对应于接收的操作的给定文件 250 读出文件数据或者将文件数据写入给定文件 250。在一些实施例中,过滤驱动器 221 可被配置成产生和给定文件 250 对应的带内元数据,并将产生的元数据保存在对应的命名流 260 中。例如,当检测到针对给定文件 250 的文件写入操作时,过滤驱动器 221 可被配置成更新和给定文件 250 的最后修改时间对应的元数据,并将更新的元数据保存在命名流 260 中。另外,在一些实施例中,过滤驱动器 221 可被配置成代表特定的应用,取回和给定文件对应的元数据。

[0060] 可响应由图 2 的进程 112 发起的各种文件系统活动,产生元数据。在一些实施例中,产生的元数据可包括任意复杂性的记录。例如,在一个实施例中,过滤驱动器 221 可被配置成检测各种文件处理操作,例如文件创建,删除,重命名,和 / 或复制操作,以及文件读取和写入操作。在一些实施例中,可如上所述带内检测这样的操作。在检测特定文件操作之后,过滤驱动器 221 可被配置成产生操作的记录,并将记录保存在适当的命名流 260 中,作为该操作所针对的文件 250 的元数据。

[0061] 更一般地,访问文件系统内容的任何方面的任何操作,比如文件数据或元数据的读取或写入可被称为文件系统内容访问事件。在一个实施例中,过滤驱动器 221 可被配置成响应检测到文件系统内容访问事件,产生元数据记录。预期在一些实施例中,针对元数据的访问事件可自己产生另外的元数据。如下更详细所述,在图解说明的实施例中,事件日志 270 可被配置成保存所检测的文件系统内容访问事件的记录,与响应事件检测,是否将另外的元数据保存在特定的命名流 260 中无关。

[0062] 在各个实施例中,保存的元数据记录可包括和文件 250 及检测的操作相关的各类信息,例如产生该操作的进程的身份,文件身份,文件类型,文件大小,文件所有者,和 / 或文件许可。在一个实施例中,记录可包括表示文件 250 的内容的文件签名。文件签名可以是所有或部分文件内容的散列函数,并且具有文件内容方面的微小差异产生明显不同的文件签名的性质。例如,文件签名可以采用消息摘要 5(MD5) 算法,消息摘要 5 算法可在内容方面仅仅单一比特不同的文件产生不同的签名,不过预期可以采用任何适当的签名产生算法。记录还可包括未明确列出的另外的信息。

[0063] 在一个实施例中,在检测特定的文件操作之后,可按照可包括数据字段以及描述相关数据字段的重要性的标记的格式,产生和保存由过滤驱动器 221 保存的元数据记录。这样的格式可被称为“自描述”数据格式。例如,可利用通用语法,由这样的标记字段对元数据记录内的数据元素定界:

[0064] <descriptive\_tag>data element</descriptive\_tag>

[0065] 其中“descriptive\_tag”定界符可描述“data element”字段的某一方面,从而用

于构成元数据记录内的各种数据元素。预期在各个实施例中,自描述数据格式可采用各种语法中的任意一种,所述各种语法可包括将标记和数据元素区分开的不同约定。

[0066] 在一些实施例中,自描述数据格式也可以是可扩展的。即,可根据需要将数据格式扩展到包含额外的结构元素。例如,不可扩展的格式可规定固定的结构,数据元素必须符合该固定结构,例如,表格式行列数据格式或者其中标记字段的数目和类型是固定的格式。相反,在一个实施例中,可扩展的自描述数据格式可容许对数据定界和构成数据的任意数目的任意定义的标记。在另一实施例中,可扩展的自描述数据格式可容许用于规定给定数据元素的语法的修改。在一些实施例中,在数据正被产生和使用时,可扩展的自描述数据格式可由用户和应用扩展。

[0067] 在一个实施例中,可扩展置标语言 (XML) 格式,或者服从任何版本的 XML 的任何数据格式可被用作保存元数据记录的可扩展的自描述格式,不过预期在其它实施例中,可以使用任何适当的格式,包括不可扩展或自描述的格式。根据待记录的所需元数据,XML 格式可容许记录字段的任意定义。XML 格式记录的一个例如如下:

```
[0068] <record sequence = " 1 " >
[0069]   <path>/test1/foo.pdf/</path>
[0070]   <type>application/pdf/</type>
[0071]   <user id = 1598>username</user>
[0072]   <group id = 119>groupname</group>
[0073]   <perm>rw-r--r--</perm>
[0074]   <md5>d41d8cd98f00b204e9800998ecf8427e</md5>
[0075]   <size>0</size>
[0076] </record>
```

[0077] 在文件创建操作之后,这样的记录可被附到和具有文件身份“test1/foo.pdf”的文件(例如文件 250a)相关的命名流(例如命名流 260a)上。这种情况下,与“record sequence”字段相关的编号指示该记录是与文件 250a 相关的第一记录。“path”字段包括文件身份,“type”字段指示文件类型,在一个实施例中,它可由发出文件创建操作的进程提供,在其它实施例中,可根据文件名的扩展名,或者根据文件内的报头信息确定。“user id”字段记录与发出文件创建操作的进程相关的用户的数字用户 id 和文本用户名,“group id”字段记录该用户的数字组 id 和文本组名称。“perm”字段按照特定于文件系统 205 和 / 或操作系统的格式,记录与文件 250a 相关的文件许可。“md5”字段记录与文件内容对应的 MD5 签名,“size”字段以字节为单位记录文件 250a 的长度。预期在备选实施例中,过滤驱动器 221 可保存与所检测的操作对应的记录,这里所述记录包括更多或更少的字段,以及具有不同定义和内容的字段。还预期在一些实施例中,过滤驱动器 221 可将从给定文件 250 读取的数据封闭在 XML 格式内,以致对文件的读取操作可返回 XML 数据,而不管基本的文件数据格式。同样地,在一些实施例中,过滤驱动器 221 可被配置成接收将被写入给定文件 250 的 XML 格式数据。在这样的实施例中,过滤驱动器 221 可被配置成在将文件数据写入给定文件 250 之前,先消除 XML 格式化。

[0078] 注意在一些实施例中,元数据可被保存在不同于命名流的结构中。例如,在一个实施例中,对应于一个或多个文件的元数据可被保存在呈数据库格式或另一格式的另一文件

中。另外，预期在一些实施例中，文件系统 205 的其它软件模块或组件可被配置成产生，保存和 / 或收回元数据。例如，过滤驱动器 221 的元数据例程可被包含到另一软件模块中，或者由另一软件模块复制。

[0079] 在图解说明的实施例中，文件系统 205 包括事件日志 270。事件日志 270 可以是和命名流 260 类似的命名流；但是，与其与特定文件相关，倒不如事件日志 270 可直接与文件系统 205 相关。在一些实施例中，文件系统 205 可以只包括一个事件日志 270，而在其它实施例中，可提供一个以上的事件日志 270。例如，在包括如图产 2 中所示的多个本地文件系统 240 的文件系统 205 的一个实施例中，可提供每个本地文件系统 240 一个历史流。

[0080] 在一些实施例中，过滤驱动器 221 可被配置成响应检测文件系统操作或事件，将元数据记录保存在事件日志 270 中。例如，针对特定文件 250 的读取或写入操作可被检测，随后过滤驱动器 221 可将表示该操作的记录保存在事件日志 270 中。在一些实施例中，过滤驱动器 221 可被配置成将元数据记录保存在事件日志 270 内，而不管对应的元数据记录是否已被保存在命名流 260 中。在一些实施例中，事件日志 270 可起在文件系统 205 内发生的所有检测到的操作和事件的集中历史的作用。

[0081] 类似于保存在命名流 260 内的记录，在一个实施例中，可按照可扩展的自描述数据格式，比如可扩展置标语言 (XML) 格式产生事件日志 270 中由过滤驱动器 221 保存的记录，不过预期在其它实施例中，可以使用任何适当的格式。例如，在文件系统 205 的操作过程中，名为 “/test1/foo.pdf” 的给定文件 250a 可被创建、修改，随后被重命名为文件 250b “/test1/destination.pdf”。在一个实施例中，在重命名操作之后，事件日志 270 可包括下述例证记录

```
[0082] <record>
[0083]   <op>create</op>
[0084]   <path>/test1/foo.pdf</path>
[0085] </record>
[0086] <record>
[0087]   <op>modify</op>
[0088]   <path>/test1/foo.pdf</path>
[0089] </record>
[0090] <record>
[0091]   <op>rename</op>
[0092]   <path>/test1/destination.pdf</path>
[0093]   <oldpath>/test1/foo.pdf</oldpath>
[0094] </record>
```

[0095] 在本例中，每个记录的 <op> 字段指示执行的操作，而 “path” 字段指示处理的文件 250a 的文件身份。就文件重命名操作来说，“path” 字段指示重命名操作的目的地文件 250b 的文件身份，“oldpath” 字段指示源文件 250a 的文件身份。预期在备选实施例中，过滤驱动器 221 可将包括更多或更少字段，以及具有不同定义和内容的字段的记录保存在事件日志 270 内。

[0096] 在图 5 中所示的流程图中更一般地图解说明了元数据产生的前述讨论。参见图

1- 图 5, 操作始于方框 500, 在方框 500 中, 文件系统内容, 例如文件数据或元数据被保存。在一个实施例中, 文件系统内容可包括文件数据和元数据。在保存文件系统内容之后, 检测文件系统内容访问事件 (方框 502)。在一个实施例中, 如上所述, 过滤驱动器 221 可带内检测文件系统内容访问事件。注意在一些实施例中, 文件系统内容的初始保存本身可以是可检测的文件系统内容访问事件。例如, 文件创建操作可以是可在步骤 502 中检测的文件系统内容访问事件。

[0097] 响应文件系统内容访问事件的检测, 产生元数据记录 (方框 504)。例如, 与访问给定文件的操作有关的各种元数据元素可被集合, 例如文件身份, 文件所有权, 产生访问事件的进程的身份等。

[0098] 在元数据记录的产生之后, 按照可扩展的自描述格式保存元数据记录 (方框 506)。在一个实施例中, 数据格式可符合一种版本的 XML 格式。在各个实施例中, 元数据记录可被保存在与给定文件相关的命名流中, 例如命名流 260 之一中, 或者被保存在诸如事件日志 270 之类的事件日志中。

#### [0099] 内容处理系统

[0100] 如上所述, 在一些实施例中, 存储系统可被配置成响应各种检测的文件系统操作或事件, 产生带内元数据。在一些情况下, 由于各种应用进程的执行, 可能发生这样的操作或事件。例如, 给定的应用可使文件按照各种方式中的任意一种被处理 (例如打开, 关闭, 读取, 写入, 复制, 重命名或者任何其它类型的文件活动), 对于所述各种方式可产生对应的元数据记录。在这样的实施例中, 所得到的元数据记录能够实现由给定的应用或进程产生的文件系统活动的系统跟踪, 可按照任意专一程度进行这样的跟踪, 并且对于应用来说, 这样的跟踪可以是透明的。

[0101] 在一些实例中, 众多的应用可与作为复杂, 异质的数据处理系统一部分的存储系统 150 交互作用。例如, 企业可使用数据库应用来管理库存和生产, 使用会计应用来跟踪账单和收据, 使用财务应用来产生季报, 使用人力资源应用来识别个人详情。在各个实施例中, 可提供另外的或者不同的应用。

[0102] 这些应用中的一些可以是多个版本的相同应用 (例如, 会计和财务可使用共同的或者相关应用), 或者它们可以是紧密耦接的应用, 即, 例如通过共用公共 API, 它们可知晓彼此的存在和数据, 通过所述公共 API, 它们可直接通信和协调。例如, 当处理账单时, 会计应用可直接通知财务应用更新预算。其它应用可由不同的卖方提供, 并且可以松散地耦接, 即, 它们共用公共的数据格式, 但是拥有的直接通信和协调彼此的操作的能力有限。例如, 财务应用能够响应用户的干预, 输入由人力资源应用产生的薪水和津贴信息, 但是在没有这种干预的情况下, 不能直接请求和接收这样的信息。最后, 在一些情况下, 一些应用可能完全不兼容, 缺少共享数据或者直接交互作用的能力。

[0103] 某些复杂的企业操作可能不只涉及一个应用, 而是涉及几个应用, 不是所有的这些应用都可紧密耦接。这样的操作 (可被称为事务) 可包括将按照特定的顺序或者响应特定的事件, 由一个或多个应用采取的一系列操作。构成事务的所述一系列操作也可被称为由事务实现的进程或程序, 并且可按照可用的各种应用的能力任意定义。

[0104] 根据单一组成应用的活动, 事务的状态可能不明显; 相反, 事务可以是和关于定义所述事务的进程的信息联系在一起的所有相关应用的活动的功能。例如, 根据在给定企业

中定义的程序,定购单的处理涉及几个步骤。定购单首先被输入,例如通过专用应用或者电子邮件接口。一旦被输入,就可确认请求者的身份和权限,例如通过利用人力资源应用核实请求者是具有适当签字权限的雇员。随后,可得到财务批准,这可包括使用财务应用来核实请求在请求该定购单的个人或组织的预算内。根据这些步骤的结果和企业政策的复杂性,可获得另外的核实,比如管理人员批准。当所有要求被满足时,定购单可被传送给卖方,完成定购单事务处理。

[0105] 当处理文件系统内容时,支持复杂事务的任何应用可在存储系统 150 内产生活动。如上所述,响应这样的活动可产生文件系统内容,包括文件数据和 / 或对应于该活动的元数据记录。但是,如前所述,在一些情况下,从给定应用的活动来看,给定事务通过它的规定进程的进展可能不如在对应于其活动的文件系统内容中所反映的那样明显。例如,人力资源应用可反映个人数据,但是不反映预算数据。从而,查询人力资源应用会导致对于特定进货,给定个人具有适当签名权限的核实,该查询的结果可在文件系统内容中显示出。但是,人力资源应用不能确定是否存在足够的进货预算。事实上,在一些情况下,任何给定的应用,例如人力资源应用一般不知晓适合于其操作的更宽的事务环境。即,应用不能区分给定的查询是否是可能跨越多个应用的给定事务的进程的一部分。

[0106] 配置可能参于给定事务的每个应用,以便能够直接与其它应用交互作用是困难或者不切实际的。例如,如果外部供应商固定了一个或多个应用的功能,那么不可能进行这样的配置。在图 2 中图解说明的实施例中,存储系统 150 包括内容处理系统 300,在各个实施例中,内容处理系统 300 可被配置成监视存储管理系统 200 保存的文件系统内容,根据这样的监视产生另外的元数据,和响应这样的监视调用应用。在一些实施例中,内容处理系统 300 可被配置成协调或监视其组成应用可能未知的复杂事务。

[0107] 图 6 中表示了内容处理系统 300 的一个实施例。在图解说明的实施例中,连同图 4 中图解说明的文件系统 205 的实施例一起表示了内容处理系统 300。内容处理系统 300 包括配置成与多个内容类型专用处理器 330a-c(也可被简称为内容处理器 330)交互作用的一个内容处理后台驻留程序 320。内容处理后台驻留程序 320 还被配置成与文件系统 205 的文件 250,命名流 260 和事件日志 270 交互作用。另外,在包括查询系统的存储系统 150 的实施例中,内容处理后台驻留程序 320 可被配置成与这样的查询系统,例如查询系统 400 交互作用。

[0108] 在图解说明的实施例中,内容处理后台驻留程序 320 可被配置成进行已由过滤驱动器 221 带内检测和记录的操作和事件的带外检测。例如,内容处理后台驻留程序 320 可偶尔扫描事件日志 270,以确定自从最后一次扫描以来,发生了什么文件系统内容访问事件。响应检测到的事件,内容处理后台驻留程序 320 可产生另外的文件系统内容,如下更详细所述。在一些实施例中,预期内容处理后台驻留程序 320 可直接扫描文件 250 和 / 或命名流 260,而在其它实施例中,内容处理后台驻留程序 320 可使用事件日志 270 来访问和记录的事件对应的那些文件 250 和命名流 260。另外,预期在一些实施例中,内容处理系统 300 可包括它自己的响应过滤驱动器 221 的通知而更新的事件的日志,而不是使用事件日志 270。

[0109] 内容处理后台驻留程序 320 可与文件系统 205 交互作用,从而代表一个或多个内容处理器 330 产生或修改文件系统内容,包括文件数据和元数据。在一个实施例,内容处理器 330 可包括配置成监视特殊事务的规定进程的过程代码或逻辑。例如,给定的内容处

理器 330 可实现描述定义成特定事务的一部分的一系列操作的算法或状态机。内容处理器 330 还可包括识别与事务相关的文件系统内容,例如在事务过程中可被访问的特定文件 250 的信息。另外,内容处理器 330 可包括识别和各个操作对应的特定应用的信息。例如,如果给定的事务包括将由会计应用执行的步骤,那么对应的内容处理器 330 可包括特定应用识别信息,比如应用名称或识别代码,所述特定应用识别信息可被包括在当会计应用访问给定文件 250 时,过滤驱动器 221 为该给定文件 250 产生的元数据记录中。

[0110] 在一些实施例中,给定的内容处理器 330 可被配置成处理特定事务的所有实例。例如,配置成监视上面描述的定购单事务的内容处理器 330 可被配置成处理在给定时间进行中的所有定购单事务。在这样的实施例中,给定的内容处理器 330 可包括数据结构,借助该数据结构,在处理器内可区分各个事务,例如依据时间戳记或者唯一的标识符。例如,当检测到新事务时,可从现有实例(例如模板),或者从内容处理后台驻留程序 320 产生内容处理器 330 的新实例。预期在一些实施例中,内容处理器 330 和内容处理后台驻留程序 320 可被实现成单一的处理实体,例如单一的软件模块。

[0111] 给定内容处理器 330 的操作可由它结合通过内容处理后台驻留程序 320 接收的文件系统内容访问事件信息实现的算法确定。在一个实施例中,内容处理器 330 最初可能处于无效或空闲状态,直到被特定的文件系统内容访问事件触发为止。例如,在通过用电子邮件将定购单寄给特定电子邮件账户,发起定购单的系统中,响应将接收的定购单的内容附到与所述特定电子邮件账户样的文件上,过滤驱动器 221 可在事件日志 270 内创建元数据记录。随后,内容处理后台驻留程序 320 可检测该记录,并将记录的指示传送给定购单内容处理器 330,作为响应,定购单内容处理器 330 可激活。在一个备选实施例中,响应恰当的激活事件的检测,内容处理器后台驻留程序 320 可产生内容处理器 330 的各个实例。

[0112] 在一个实施例中,内容处理器 330 可以是检测给定的一系列事件何时发生,或者文件系统内容的给定状态何时产生的被动监视器。例如,内容处理器 330 可被配置成通过响应在批准进程中采取的应用活动,检查过滤驱动器 221 产生的元数据记录,检测是否已按照正确的顺序执行了一系列的事件,例如批准进程的步骤。在另一情况下,内容处理器 330 可被配置成按照特定的语法或方案,确定文件系统内容是否合适。例如,内容处理器 330 可检查更新之后的元数据记录或文件数据,以确定元数据记录或文件数据在句法上是否正确,是否被恰当地构成(即,所需数据存在)等。

[0113] 在另一实施例中,内容处理器 330 可被配置成响应检测各种事件或内容的状态,主动修改文件系统内容,和 / 或调用其它应用。例如,在文档发布环境中,可以几种不同的格式(例如可移植文档格式(PDF),HTML 格式,Microsoft Word 格式)使给定的文档供用户之用。在这样的环境中,内容处理器 330 可被配置成使必要版本的给定文档的产生自动化。例如,内容处理器 330 可被配置成通过检测命名流 260 和 / 或事件日志 270 中,对给定文件 250 的更新的元数据记录,检测给定文件 250 中文档的原版何时已被更新。当检测到更新时,控制处理器 330 可调用适当的发生器或转换器应用,将更新的原版转换成每种所需的格式。对于更新原版文档的用户或应用来说,这样的转换可透明地发生。

[0114] 作为另一例子,内容处理器 330 可被配置成主动进行复杂的事务,例如上面说明的进货批准进程。在这种情况下,内容处理器 330 可被配置成调用在对应于该事务的过程中规定的各种应用。例如,内容处理器 330 可被配置成检测已如上所述通过电子邮件收到

了新的定购单请求。随后,通过访问保存请求数据的文件 250,内容处理器 330 可从该请求提取信息,例如请求者的身份和请求的美元数额。利用提取的信息,内容处理器 330 可与各种应用交互作用,从而获得批准,如上所述。例如,内容处理器 330 可产生查询来核实人力资源应用内的签字权限,财务应用内的预算状态等。在需要用户交互作用的实例中,内容处理器 330 可被配置成产生提示,所述提示将通过例如电子邮件被传送给用户用于响应。

[0115] 在各个实施例中,内容处理器 330 可被配置成产生不同形式的输出。在一个实施例中,内容处理器 330 可响应其处理,产生带外元数据记录。例如,配置成执行给定文件 250 中的结构数据的方案确认的内容处理器 330 可在对应的命名流 260 内产生指示其检查的状态的元数据记录。在其它实施例中,代替或者除了元数据之外,内容处理器 330 可被配置成产生或修改文件数据。例如,前面提及的方案确认者可被配置成纠正正在确认结构数据时检测到的某些缺陷,例如通过截去残缺的记录或者填写缺少的数据字段。作为另一例子,内容处理器 330 可被配置成与查询引擎交互作用,以保持由查询引擎索引的数据的参考完整性,如下更详细所述。在其它实施例中,内容处理器 330 可被配置成与应用或用户交互作用。例如,如上所述,内容处理器 330 可被配置成响应检测特定事件,比如文档内容更新,调用应用的 API。预期在一个实施例中,一个或多个内容处理器 330 可被配置成按照如上所述的可扩展的自描述数据格式产生元数据记录,所述可扩展的自描述数据格式可包括与任何版本的 XML 格式相符的格式。

[0116] 注意虽然内容处理系统 300 及其各个组件可与如上所述为应用层 100 内的进程 112 的应用交互作用,不过内容处理系统 300 和其各个组件不同于应用。一般来说,应用不知晓其它应用的活动,并且不能访问在文件系统 205 的操作过程中产生的元数据。但是,在图解说明的实施例中,内容处理系统 300 确实可以访问这样的元数据,由于这样的访问,内容处理系统 300 可被配置成检测未被特定应用的操作充分表现的事务事件。

[0117] 图 7 中图解说明了包括内容处理系统 300 的存储系统 150 的操作方法的一个实施例。参见图 1- 图 4,图 6 和图 7,操作开始于方框 700,在方框 700 中,保存文件系统内容,例如文件数据或元数据。在一个实施例中,文件系统内容可既包括文件数据又包括元数据。在保存文件系统内容之后,带内检测文件系统内容访问事件(方框 702)。例如,如上所述,文件系统内容访问事件可由过滤驱动器 221 带内检测。如上所述,在一些实施例中,文件系统内容的初始保存本身就是一个可检测的文件系统内容访问事件。例如,文件创建操作可以是可在步骤 702 中检测的一个文件系统内容访问事件。

[0118] 响应文件系统内容访问事件的检测,产生一个事件记录(方框 704)。例如,在一个实施例中,过滤驱动器 221 可被配置成产生包括关于文件系统内容访问事件的信息,例如事件的类型和所访问的文件系统内容的事件记录。在一个实施例中,这样的记录可被保存在事件日志 270 中。在一些实施例中,还可产生包括关于文件系统内容访问事件的其它信息的元数据记录。例如,在一个实施例中,过滤驱动器 221 可被配置成集合和访问给定文件的操作有关的各种元数据元素,例如文件身份,文件所有权,产生访问事件的进程的身份等。在一个实施例中,如前所述,可按照可扩展的自描述数据格式,例如 XML 格式保存响应事件检测而产生的事件记录和 / 或任何另外的元数据记录。

[0119] 在产生了事件记录之后,它可被带外检测(方框 706)。例如,在一个实施例中,内容处理后台驻留程序 320 或内容处理器 330 可被配置成扫描事件日志 270,以检测事件记

录。在一些实施例中,检测事件记录可包括检测包含几个应用的事务。例如,特定的内容处理器 330 可发现给定的事件记录表示在涉及几个应用方面的活动的特定类型的事务中的初始或连续步骤。

[0120] 响应带外事件记录检测,产生另外的文件系统内容(方框 708)。在一个实施例中,产生的另外的文件系统内容可包括保存在多个文件 250 之一中的文件数据,保存在一个或多个命名流 260 中的元数据记录,或者这两者。在一个实施例中,响应带外事件记录检测,可另外调用一个应用。例如,在处理事务的过程中,给定的内容处理器 330 可调用一个应用,如上所述。

#### [0121] 查询文件系统内容

[0122] 如上所述,在一些实施例中,文件系统 205 可被配置成保存各种文件系统内容。文件系统 205 可保存一个或多个文件内的各种文件数据,可保存与给定文件对应的任意复杂性的元数据。文件系统 205 还可被配置成消耗文件系统内容。例如,文件系统 205 可实现特定的存储策略,由此具有如在它们的元数据中指出的某些使用特性的文件被分配给特定类型的存储器。例如,在一个实施例中,最近被某些类型的进程使用或者访问过的文件可被分配给更快的存储器,而其它文件可被分配给较慢的存储器。

[0123] 在一些实施例中,在文件系统 205 之外的应用或操作系统进程(例如图 2 的进程 112)也可被配置成消耗系统内容。例如,编写应用模块的程序员可能希望创建和处理特定的文件和它们的相关文件数据,以便保存或取回应用数据。另外,这样的程序员可能希望对某些文件的元数据特性进行偶然的动作,例如配置备份程序,以便只选择自从最后一次备份以来已被修改的文件。在一些实施例中,可通过查询文件系统内容识别满足特定规则的内容,来给定特定的文件系统内容。查询文件系统内容的可用规则取决于保存文件系统内容的格式。例如在一个实施例中,可以固定的不可扩展的格式,例如表格式数据结构保存文件系统内容,在表格式数据结构中,数据项目的描述继承自行定义和列定义,而不是继承于自描述格式标记。在这样的实施例中,选择文件系统内容的标准可由格式的规定结构确定,例如可用的行定义和列定义。在以可扩展的自描述格式,例如上面说明的 XML 格式保存文件系统内容的实施例中,可用于选择所需的文件系统内容的标准可包括该内容的任何自描述特征。

[0124] 在图 2 中图解说明的存储系统 10 的实施例中,查询系统 400 被配置成通过 API 114C 向各个进程 112 提供文件系统内容查询能力。图 8 中图解说明了查询系统 400 的一个具体实施例。在该实施例中,查询系统 400 包括连接管理器 420,查询引擎 430,索引 / 提交引擎 440,和数据布局管理器 450,下面更详细地分别对它们进行说明。

[0125] 一般来说,查询可规定如何从较大的一组数据中选择数据的子集,例如通过按照自描述格式保存的记录的一个或多个数字字段的评估。例如,用户可能希望选择和文件 / test1/foo.pdf 对应的所有保存记录以便进一步分析。相应地,用户可构成一个查询,所述查询规定具有数据字段标记的“path”的所有记录的选择,这里该数据字段等于特定的值,比如“/test1/foo.pdf”。在一些实施例中,预期整个文件系统内容可查询,包括保存在文件 250 内的文件数据以及保存在命名流 260 内的元数据,不论元数据是由例如过滤驱动器 221 带内产生的,还是由例如特定的内容处理器 330 带外产生的,或者不论如前所述,该元数据是由应用进程 112 通过 API 114 在文件系统 205 之外定义和 / 或产生的。另外,在如上所

述,文件系统内容访问事件被记录在事件日志 270 内的一些实施例中,预期这些记录事件也可被查询。

[0126] 可用查询语言构成查询,所述查询语言可根据一个或多个标记的数据字段的值,为多组数据的选择提供句法结构。在一些实施例中,除了组选择特征之外,给定的查询语言可支持过程特征,例如函数。此外,在一些实施例中,给定的查询语言可支持在用其它编程语言,例如 Java 或 C 编码的过程例程内的嵌入。在 XML 格式被用于构成文件系统内容的情况下,给定的应用可用由万维网联盟 (W3C) 规定的 XML 查询 (XQuery) 语言,或者任何未来的 XQuery 标准或其变型构成选择特定文件系统内容的查询。但是,预期可以采用任何适当的查询语言。

[0127] 如前所述,在图解说明的实施例中,进程 112 可产生查询,并通过 API 14 将其传送给查询系统 400。在一些实施例中,查询系统 400 可被配置成支持具有在给定时间同时待解决的查询的几个进程 112。此外,在一些实施例中,查询进程 112 可通过网络从远程计算机系统提交其查询。另外,在一些实施例中,查询进程 112 需要被验证,以限制对查询系统 400 的访问。在一个实施例中,连接管理器 420 可被配置成管理建立和保持查询进程 112 和查询系统 400 之间的连接的开销。例如,连接管理器 420 可被配置成提供验证界面(例如用户名和口令界面),从而查询进程 112 可建立它们的进行查询的权限。另外,在一个实施例中,连接管理器 420 可配置成保持支持查询进程 112 的基于连接的或者基于会话的语义学所必需的信息。例如,连接管理器 420 可保持将进行中的查询映射到它们的相关请求者的数据结构,以便查询结果被引向正确的查询进程 112。在一些实施例中,连接管理器 420 还可被配置成通过在查询引擎 430 的多个实例间分布请求,实现查询请求的负载均衡,从而提高总的查询吞吐量。

[0128] 在一个实施例中,查询引擎 430 可被配置成解析和评估通过连接管理器 420 提交给查询系统 400 的查询。例如,查询引擎 430 可接收请求在某一时间范围内,已被特定用户修改的所有文件 250 的名称的查询。查询引擎 430 可关于句法正确性解析查询,如果查询不对,那么可返回错误条件。在一些实施例中,查询引擎 430 还可对查询进行结构变换,例如将查询分解成多个查询和 / 或关于性能优化查询。随后,查询引擎 430 可检查保存在命名流 260 中的元数据记录,以识别满足规定标准的文件 250,将这些文件的名称返回给查询进程 112。配置成解析和评估查询的查询引擎 430 的众多实现是可能的。

[0129] 在一些实施例中,查询引擎 430 可直接与存储管理系统 200 交互作用,从而响应查询的评估访问文件系统内容。但是,在一些情况下,通过创建文件系统内容的一个或多个索引,并使用这些索引来帮助查询评估,可提高查询评估性能。在图解说明的实施例中,索引 / 提交引擎 440 可被配置成产生和保持这些索引,并在查询的评估期间,将索引信息提供给查询引擎 430。

[0130] 一般来说,索引可以是按照某一特征或属性组织一批数据,借助被索引的特征或属性便利数据的查询的任何数据结构。例如,在一个实施例中,索引可以是按照字母顺序组织的借助文件系统 205 定义的所有文件 250 的名称的列表。在一些实施例中,可以采用文件系统内容的多个索引。例如,如果频繁地用名称,相关用户和内容创建 / 修改时间查询文件系统内容,那么可创建依据这些属性中的每个属性分类或组织文件系统内容的单独索引。在一些实施例中,可以采用更复杂的索引方案,包括将多个内容属性组合成复杂的状态

空间的索引。另外，预期可利用任何适当的数据结构，包括列表，表格，树形结构和更高级的数据结构实现索引。

[0131] 索引 / 提交引擎 440 创建的索引本身可被保存在文件系统 205 内。在一些实施例中，可独立于其它文件系统内容保存这些索引。在这样的实施例中，数据布局管理器 450 可被配置成跟踪文件系统 205 内的索引的位置。在一个实施例中，数据布局管理器 450 可被配置成在访问与索引相关的存储器的时候，绕过过滤驱动器 221，以致不产生与索引访问对应的带内元数据。在这样的实施例中，可避免涉及标引和元数据的一些不一致的情形。例如，如果索引 / 提交引擎 440 试图将给定索引的元数据，例如修改时间戳记包括在给定索引内，随后通过过滤驱动器 221 将给定索引写入存储器，那么在写入之后的给定索引的元数据可能不再与给定索引的内容一致，例如如果过滤驱动器 221 响应写入操作，创建了新的修改时间戳记。

[0132] 在一些实施例中，查询进程 112 可使用查询通过查询系统 400 来修改文件系统内容。例如，查询可被用于从可用的文件系统内容中选择一组数据项，例如文件 250。选择的数据项随后可被修改，而不是直接将修改提交给文件系统 205，以便传送给存储装置 230，查询系统 400 可协调数据更新，从而为文件系统内容修改提供一条备选的路径。但是，在存在修改文件系统内容的多个路径的实施例中，为了防止对公共数据的冲突修改，这些路径间的协调是必需的。在一个实施例中，索引 / 提交引擎 440 可被配置成实现提交协议（例如两阶段提交），以确保对文件系统内容的更新是一致的。

[0133] 索引 / 提交引擎 440 保持的索引一般可以是文件系统内容的衍生物，以致当文件系统内容变化时（不论是归因于通过查询系统 400 的更新，归因于内容处理器 300 的活动，或者源于直接与文件系统 205 交互作用的进程 112），与修改后的内容对应的一个或多个索引可能不再准确反映内容的新状态。例如，如果索引 / 提交引擎 440 包括文件系统内容的文件名索引，并且给定文件 250 被进程 112 重命名，那么在重命名之后，基于文件名的索引可能是陈旧的。一般来说，如果相对于被索引数据的状态，索引是最近的，那么可认为索引保持了关于被索引数据的参考完整性。

[0134] 如果被索引的文件系统内容变化，那么索引 / 提交引擎 440 可被配置成通过更新相关索引来反映所述变化，保持其索引的参考完整性。在一个实施例中，索引 / 提交引擎 440 可被配置成与内容处理系统 300 内的内容处理后台驻留程序 320 或者特定内容处理器 330 协调，以反映文件系统内容变化，和相应地更新索引。例如，在一个实施例中，内容处理后台驻留程序 320 可被配置成检测何时发生了对文件系统内容的任何改变，例如通过扫描事件日志 270。响应检测可能导致内容变化的文件系统内容访问事件（例如，导致文件系统内容的创建，修改或删除的事件），内容处理后台驻留程序 320 可被配置成将检测到的事件通知索引 / 提交引擎 440。作为响应，索引 / 提交引擎 440 可更新受检测到的事件影响的一个或多个索引，以保持它们的参考完整性。例如，如果特定文件 250 被删除，那么内容处理后台驻留程序 320 可检测事件日志 270 内与删除对应的事件，并将该事件和删除的文件 250 通知索引 / 提交引擎 440。索引 / 提交引擎 440 随后可修改它保持的索引，以除去对被删除文件 250 的参考。

[0135] 在另一实施例中，一个或多个特定的内容处理器 330 可被配置成监视和索引 / 提交引擎 440 保持的特殊索引对应的文件系统内容。例如，在一个实施例中，索引 / 提交引擎

440 可将特殊索引的结构和内容,例如与索引相关的自描述数据字段(例如文件名,文件大小,文件签名等)和它们的当前值通知给定的内容处理器 330。内容处理器 330 随后可被配置成关于和特殊索引的结构和内容相关的事件,监视文件系统内容访问事件,并只将这些相关事件通知索引 / 提交引擎 440。当被通知时,索引 / 提交引擎 440 可通过增加信息或者从索引中删除信息,更新相关的索引,以保持其参考完整性。

[0136] 图 9 中图解说明了包括查询系统 400 的存储系统 150 的操作方法的一个实施例。参见图 1- 图 4, 图 8 和图 9, 操作开始于方框 900, 在方框 900, 保存文件系统内容, 例如文件数据或元数据。在一个实施例中, 文件系统内容既可包括文件数据又可包括元数据。在一些实施例中, 查询系统 400 可被配置成如上所述产生文件系统内容的一个或多个索引(方框 902), 不过在其它实施例中, 该步骤可被省略。

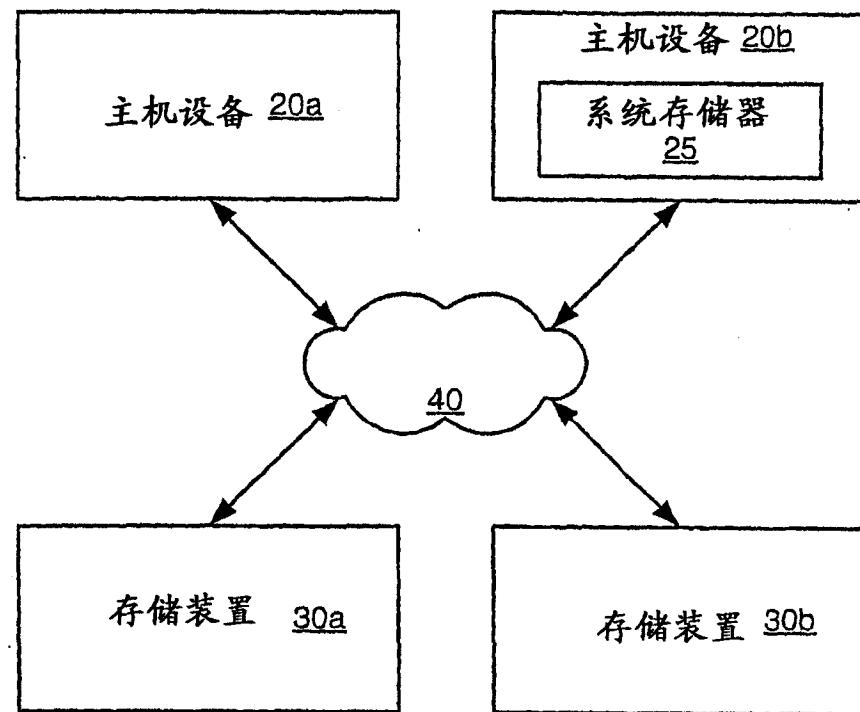
[0137] 在保存文件系统内容之后, 带内检测文件系统内容访问事件(方框 904)。例如, 可如上所述由过滤驱动器 221 检测文件系统内容访问事件。如上所述, 在一些实施例中, 文件系统内容的初始保存本身可以是一个可检测的文件系统内容访问事件。例如, 文件创建操作可以是在步骤 902 中可检测的文件系统内容访问事件。

[0138] 响应文件系统内容访问事件的检测, 产生一个元数据记录(方框 906)。例如, 在一个实施例中, 过滤驱动器 221 可被配置成产生包含和事件或正被访问的文件系统内容相关的信息, 例如时间戳记, 文件签名, 产生事件的进程的身份等的元数据记录。在一个实施例中, 这样的记录可被保存在特定的命名流 260 中。另外, 在一些实施例中, 可响应事件检测产生事件记录。这样的事件记录可被保存在事件日志 270 中。在一个实施例中, 如前所述, 可按照可扩展的自描述数据格式, 例如 XML 格式, 保存元数据记录和 / 或响应事件检测产生的事件记录。

[0139] 在文件系统内容被索引的实施例中, 文件系统内容访问事件的通知被接收(方框 908), 并且可关于文件系统内容访问事件保持相关索引的参考完整性(方框 910)。例如, 内容处理后台驻留程序 320 可被配置成将所有文件系统内容访问事件通知索引 / 提交引擎 440, 或者特殊的内容处理器 330 可被配置成监视和索引 / 提交引擎 440 保持的索引相关的那些事件, 如上所述。响应收以相关事件的通知, 索引 / 提交引擎 440 可被配置成更新相关索引以保持其参考完整性, 如上所述。预期在不采用标引的实施例中, 这些步骤可被省略。

[0140] 在元数据记录的产生之后, 文件系统内容可被查询(方框 912)。例如, 查询系统 400 可被配置成通过查询文件数据和 / 或元数据, 答复进程 112 提交的查询。在按照 XML 格式保存文件系统内容的实施例中, 可按照 XML 查询(XQuery 格式)产生查询。

[0141] 虽然相当详细地说明了上面的实施例, 不过一旦充分理解上面的公开内容, 那么对本领域的技术人员来说, 各种变化和修改将变得明显。下面的权利要求意图包含所有这样的变化和修改。



10

图 1

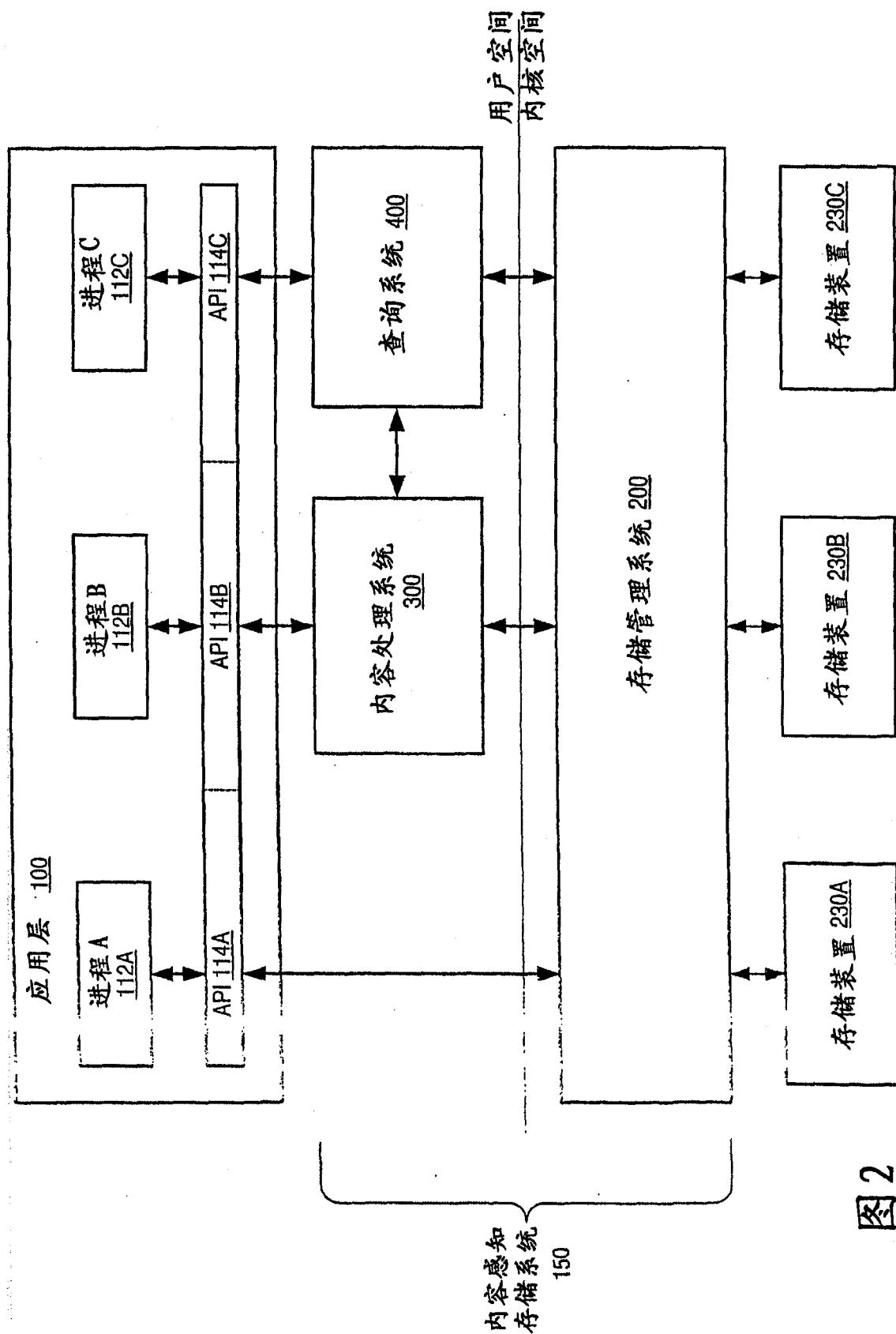


图 2

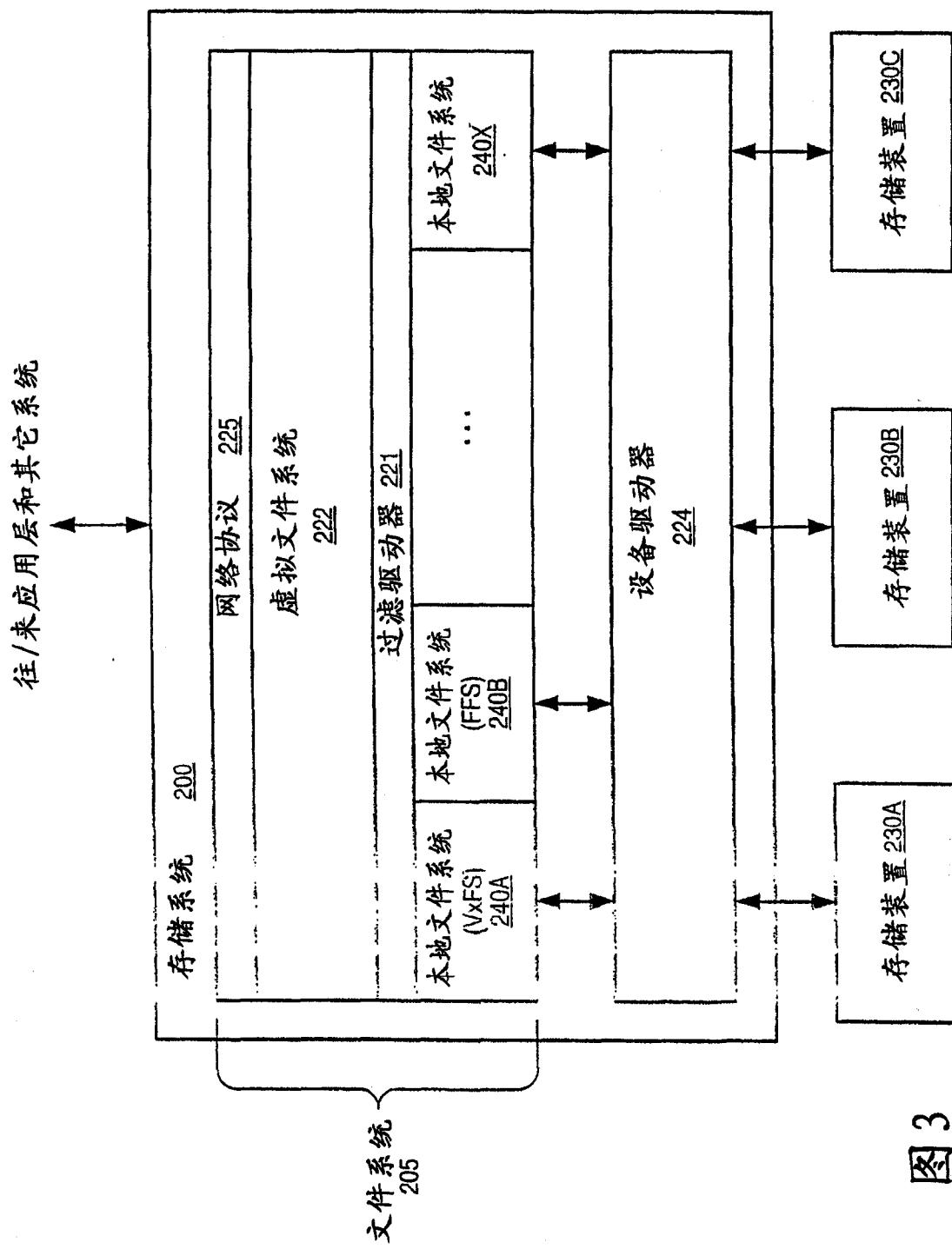


图 3

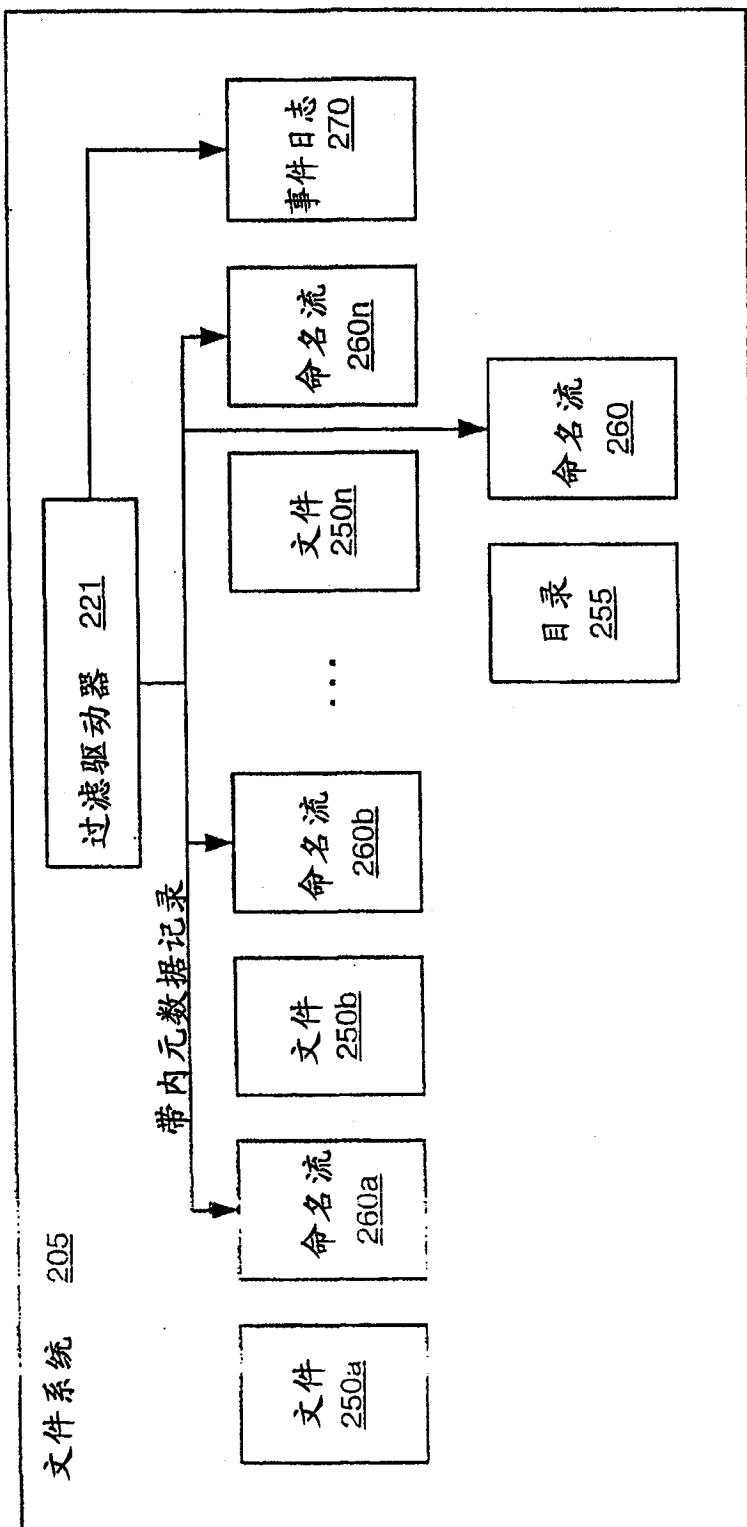


图 4

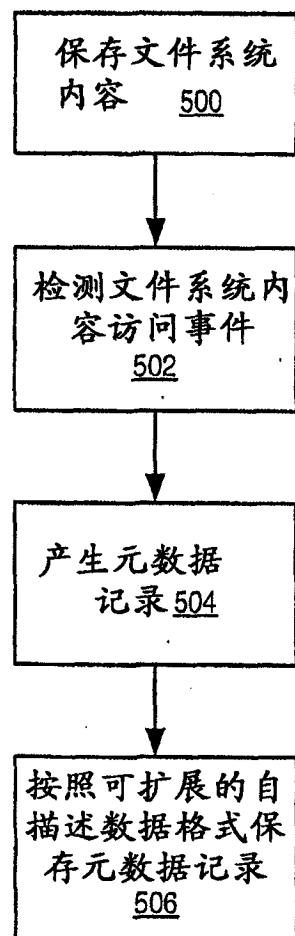


图 5

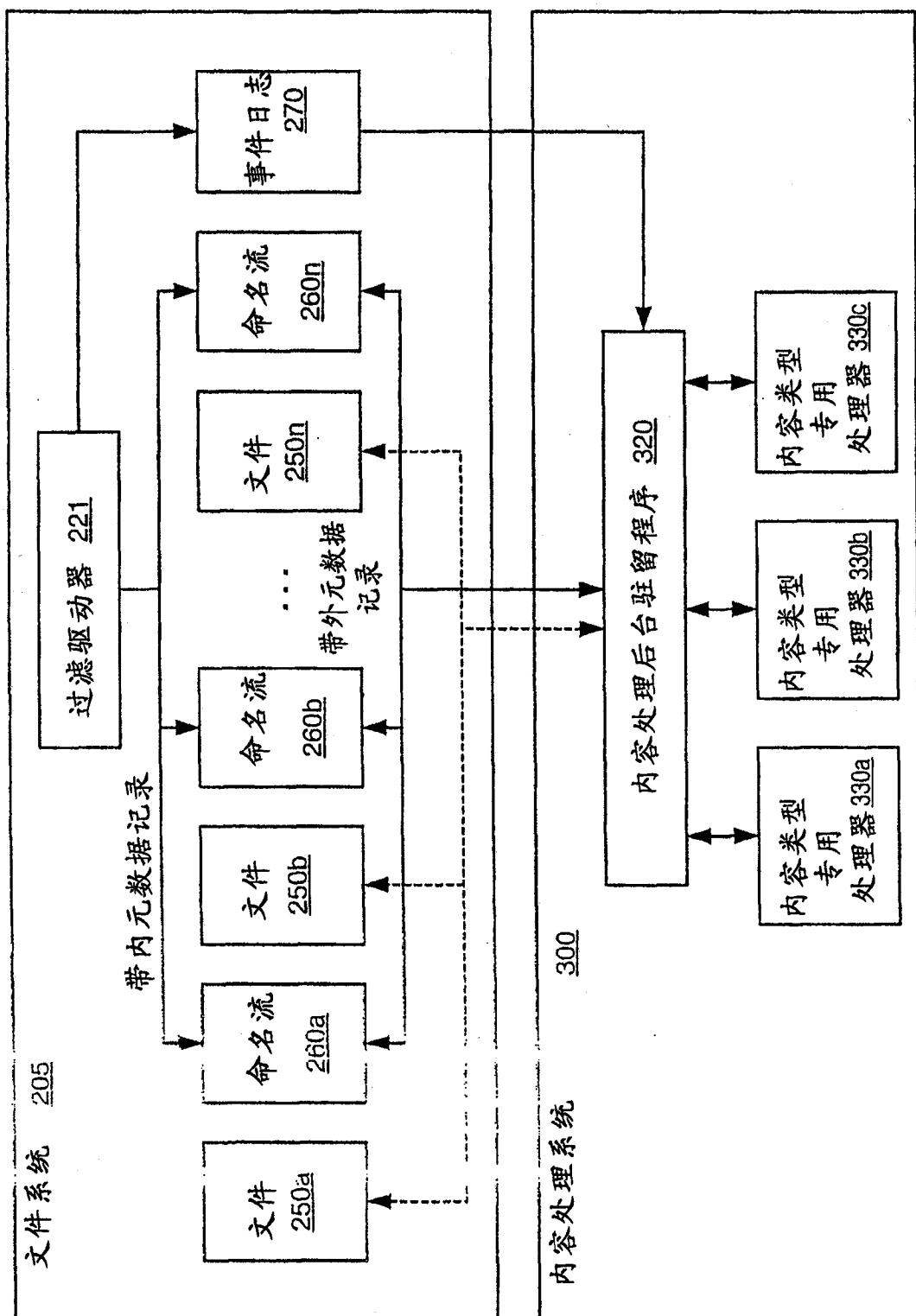


图 6

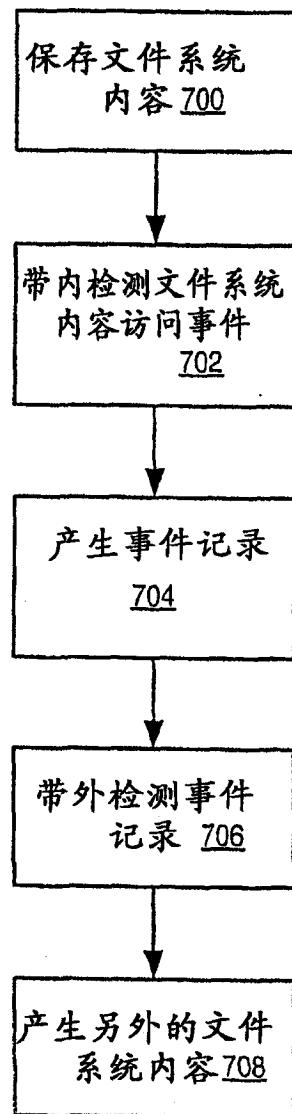


图 7

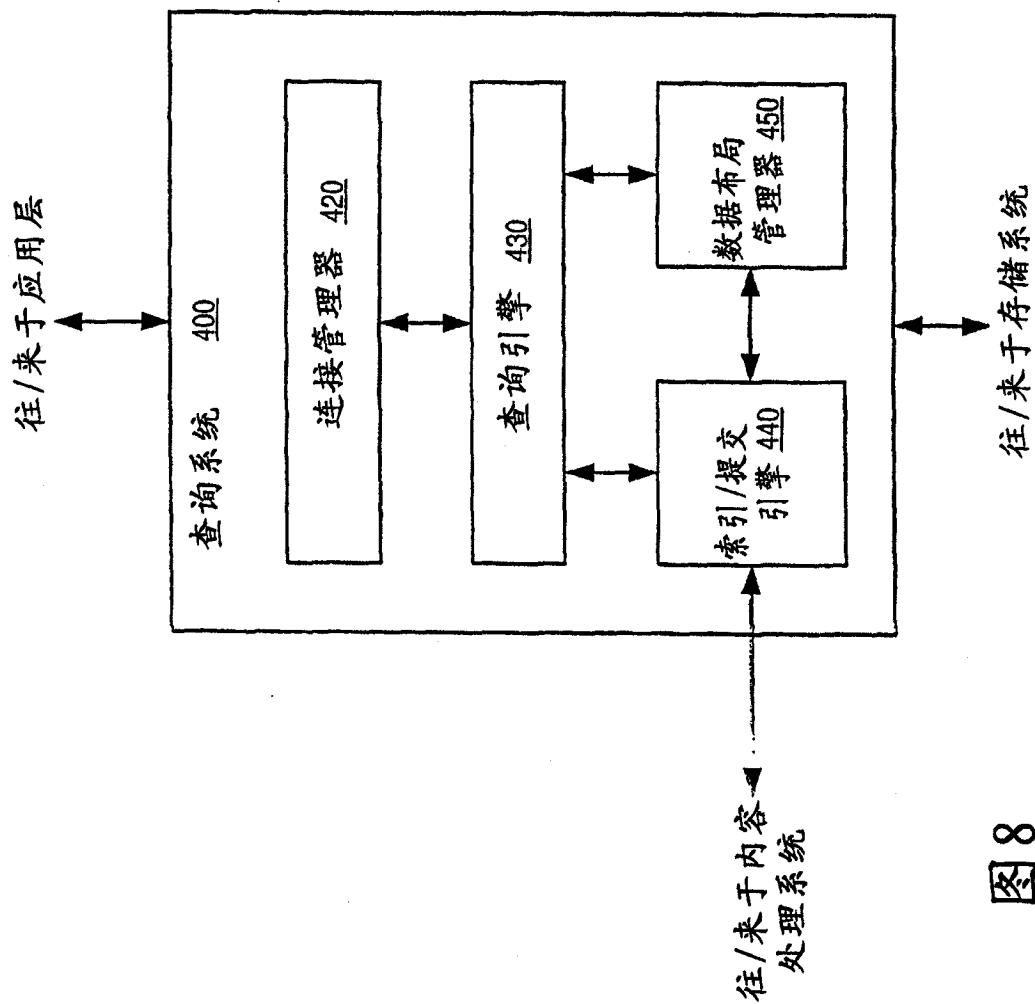


图 8