(19)中华人民共和国国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 111316251 A
(43)申请公布日 2020.06.19

(21)申请号 201780096526.6

(22)申请日 2017.11.06

(85)PCT国际申请进入国家阶段日
2020.04.30

(86)PCT国际申请的申请数据
PCT/US2017/060175 2017.11.06

(87)PCT国际申请的公布数据
WO2019/089057 EN 2019.05.09

(71)申请人 海量数据有限公司
地址 以色列特拉维夫

(72)发明人 R·哈拉克 A·利维 A·戈伦 A·霍雷夫

(74)专利代理机构 广州嘉权专利商标事务所有限公司 44205
代理人 赵学超

(51)Int.Cl.
G06F 13/00(2006.01)
G06F 15/16(2006.01)

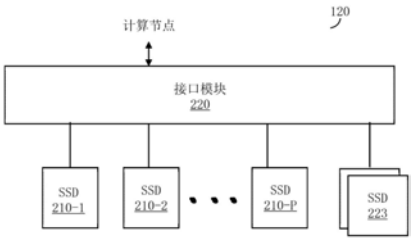
权利要求书3页 说明书8页 附图4页

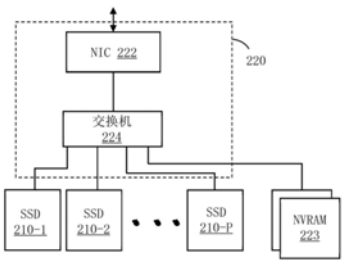
(54)发明名称

可扩展存储系统

(57)摘要

提供了一种大型存储系统。大型系统包括多个计算节点;以及多个存储节点;一种通信结构,用于在所述多个计算节点与所述多个存储节点之间提供通信基础设施;其中,多个计算节点中的每个计算节点被配置为以持久方式独立地对任何一个存储节点执行至少一个存储操作,并且其中,所述多个存储节点中的每个存储节点提供所述大型存储系统的物理存储空间。





CN 111316251 A

1.一种大型存储系统,包括:

多个计算节点;

多个存储节点;和

通信结构,用于在多个计算节点和多个存储节点之间提供通信基础设施;

其中,所述多个计算节点中的每个计算节点被配置为以持久方式独立地对任何一个存储节点执行至少一个存储操作,以及

其中,所述多个存储节点中的每个存储节点提供所述大型存储系统的物理存储空间。

2.根据权利要求1所述的系统,其中,所述大型存储系统连接到网络,以允许客户端设备访问大型存储系统。

3.根据权利要求2所述的系统,其中,所述多个计算节点中的每一个被配置为从客户端设备接收通过第一协议的请求,并将命令转换为所述至少一个存储操作,以通过通信结构的第二协议将所述至少一个存储操作传送到存储节点。

4.根据权利要求1所述的系统,其中,所述通信结构是以下之一:以太网结构和无限带宽结构。

5.根据权利要求4所述的系统,其中,所述通信结构支持通信协议,所述通信协议包括以下各项中的任一项:融合以太网(RoCE)上的远程直接存储器访问(RDMA),iWARP,以及非易失性存储器标准(NVMe)。

6.根据权利要求1所述的系统,其中,所述多个存储节点中的每一个包括:

多个固态持久性驱动器(SSDs),其中,所述多个SSDs中的每个是消费级SSD;

至少一个非易失性随机存取存储器(NVRAM),用于临时保持要写入多个SSDs的数据,从而减少每个SSDs的写入放大次数;和

至少一个接口模块,其被配置为控制多个SSDs和NVRAM并与多个计算节点通信。

7.根据权利要求6所述的系统,其中,每个存储节点中的SSDs的数量和每个SSD的容量是可配置的。

8.根据权利要求6所述的系统,其中,所述多个SSDs被放置在具有企业级外形尺寸的机柜中。

9.根据权利要求6所述的系统,其中,所述至少一个接口模块还包括:

网络接口卡(NIC),用于通过通信结构与多个计算节点接口;以及

允许连接到多个SSDs的交换机。

10.根据权利要求9所述的系统,其中,计算节点还被配置为通过聚合来自多个写入请求的写入请求,直到准备好将完整的数据块写入至少一个SSD,在至少一个NVRAM上执行垃圾回收过程。

11.根据权利要求6所述的系统,其中,所述计算节点被配置为维持与所述多个SSDs中的每一个的数据弹性。

12.根据权利要求6所述的系统,其中,所述计算节点被配置为通过将数据块写入确定数量的冗余SSDs来维持与所述多个SSDs的数据弹性。

13.根据权利要求6所述的系统,其中,所述计算节点还被配置为在将数据写入所述多个SSDs中的至少一个之前压缩数据。

14.根据权利要求1所述的系统,其中,所述多个计算节点中的每个计算节点是以下各

项中的任一项:虚拟机、软件容器和物理机。

15.根据权利要求1所述的系统,其中,所述多个计算节点中的每个计算节点访问所述大型存储系统的整个命名空间,并且其中,每个存储节点访问所述命名空间的预定范围。

16.根据权利要求1所述的系统,其中,所述至少一个存储操作包括执行读取请求。

17.根据权利要求16所述的系统,其中,所述计算节点还被配置为:

接收读取请求,其中,所述读取请求至少包括待读取的数据元素的标识;

确定所请求的数据元素的数据块;

确定数据块的位置,其中该位置在至少一个存储节点中;和

访问确定的位置以检索所请求的元素的数据块。

18.根据权利要求17所述的系统,其中,所述至少一个存储节点内的位置是以下之一:至少一个SSD和至少一个NVRAM。

19.根据权利要求18所述的系统,其中,使用令牌来确定所述位置,其中,所述令牌是表示指向以下各项中的任意一项的指针的抽象概念:数据和元数据的数据块。

20.根据权利要求1所述的系统,其中,所述存储操作至少包括执行写入操作。

21.根据权利要求20所述的系统,其中,所述计算节点还被配置为:

接收写入请求,其中,所述写入请求至少包括待写的数据元素;

确定写接收到的数据元素的位置,其中该位置在至少一个存储节点中;

将数据元素写入确定位置的写缓冲器;和

在将数据元素写入写缓冲器时收到确认。

22.根据权利要求21所述的系统,其中,所述至少一个存储节点内的位置是所述至少一个存储节点中的每一个的NVRAM。

23.根据权利要求21所述的系统,其中,使用哈希表确定位置,该哈希表将数据元素的处理程序映射到物理存储目标数据块和元数据。

24.根据权利要求21所述的系统,其中,所述计算节点还被配置为:在将所述数据元素写入所述写缓冲器之前,锁定所述写缓冲器。

25.一种在大型存储系统中执行写入请求的方法,包括:

由所述大型存储系统的计算节点接收写入请求,其中所述写入请求至少包括待写入的数据元素;

确定写接收到的数据元素的位置,其中该位置在大型存储系统的至少一个存储节点中;

在确定位置将数据元素写入写缓冲器;和

在所述计算节点处,在将所述数据元素写入所述写缓冲器时接收到确认。

26.根据权利要求25所述的方法,进一步包括:

在将数据元素写入写缓冲器之前,锁定写缓冲器。

27.根据权利要求25所述的方法,其中,确定位置还包括:

使用哈希表将数据元素的处理程序映射到物理存储目标数据块和元数据。

28.根据权利要求27所述的方法,其中,在所述至少一个存储节点内的位置是包括在所述至少一个存储节点中的至少一个NVRAM。

29.根据权利要求28所述的方法,进一步包括:

控制NVRAM以聚集完整的数据块;和

将完整的数据块写入包括在至少一个存储节点中的至少一个固态持久驱动器(SSD)。

30.一种非暂时性计算机可读介质,其上存储有用于使一个或多个处理单元执行根据权利要求25所述的方法的指令。

31.一种用于在大型存储系统中执行读取请求的方法,包括:

在大型存储系统的计算节点处接收读取请求,其中,读取请求至少包括要读取的数据元素的标识;

确定所请求的数据元素的数据块;

确定数据块的位置,其中该位置在大型存储系统的至少一个存储节点中;和

访问确定的位置以检索所请求的元素的数据块。

32.根据权利要求31所述的方法,其中,所述至少一个存储节点内的位置是以下至少之一:至少一个SSD和至少一个NVRAM。

33.根据权利要求31所述的方法,其中,确定位置还包括:

检查令牌的值,其中令牌是代表指向以下任意一项的指针的抽象概念:数据和元数据的数据块。

34.一种非暂时性计算机可读介质,其上存储有用于使处理电路执行根据权利要求31所述的方法的指令。

可扩展存储系统

技术领域

[0001] 本公开总体上涉及数据存储领域，并且更具体地涉及大型存储系统。

背景技术

[0002] 数据中心是一组联网的计算机服务器，通常由组织用于远程存储、处理或分发大量数据。传统上，使用四个不同的网络来安排数据中心：提供与数据中心之间的连接的广域网(WAN)，提供数据中心的服务器之间的连接的局域网(LAN)，提供服务器到存储系统的连接的存储区域网络(SAN)，以及用于连接各种存储元素(例如磁盘)的内部存储结构。随着网络技术的发展，传统的数据中心布置可能无法提供最佳性能。例如，在数据中心组件之间使用2种不同的连接类型是一种低效的配置。

[0003] 诸如闪存和NVRAM的固态持久性驱动器(SSDs)的先进技术，为传统的磁性硬盘驱动器提供了可靠，更快的替代方案。SSDs的缺点是它们的价格。因此，这种持久性媒介通常不用于备份或存档应用程序。此外，在安装在数据中心中的服务器和存储系统中，使用了企业级SSDs，可确保大量的写-擦除周期。这种企业级固态硬盘相对昂贵。

[0004] 为了满足数据中心对存储和性能的需求，引入了软件定义存储(SDS)。软件定义存储是指计算机数据存储技术，该技术将存储硬件与管理存储基础架构的软件分开。该软件实施策略管理，以进行包括重复数据删除、复制、快照和备份在内的操作。利用软件定义存储技术，可以满足灵活调整基础架构的要求。

[0005] 在软件定义的存储解决方案的典型配置中，存储驱动器直接连接到执行存储软件(逻辑)的服务器。从物理空间的角度来看，这是低效的，因为服务器正在过渡到更小的外形尺寸，并且容纳存储驱动器的空间越来越小。此外，连接到多个存储驱动器的服务器可能导致单点故障，即，导致连接到该存储驱动器的所有驱动器无法访问。

[0006] 软件定义存储解决方案的另一个主要缺点是计算和存储资源是耦合的。也就是说，增加计算资源以获得更好的性能将需要增加存储驱动器的数量(例如，作为服务器的一部分)。同样，增加存储驱动器以增加可用存储将需要增加服务器的数量。

[0007] 因此，提供一种可克服上述缺陷而用作存储解决方案的存储系统是具有优势的。

发明内容

[0008] 以下是本公开的几个示例实施例的概述。提供该概述是为了方便读者提供对这样的实施例的基本理解，并且不完全限定本公开的广度。该概述不是所有预期实施例的详尽概述，并且既不旨在标识所有实施例的关键或重要元素，也不旨在描绘任何或所有方面的范围。其唯一目的是以简化的形式呈现一个或多个实施例的一些概念，作为稍后呈现的更详细描述的序言。为了方便起见，术语“一些实施例”或“某些实施例”在本文中可以用来指本公开的单个实施例或多个实施例。

[0009] 本文公开的某些实施例包括大型存储系统，其包括：多个计算节点；以及多个存储节点；通信结构，用于在所述多个计算节点与所述多个存储节点之间提供通信基础设施；其

中，多个计算节点中的每个计算节点被配置为以持久方式独立地对任何一个存储节点执行至少一个存储操作，并且其中，多个存储节点中的每个存储节点提供大型存储系统的物理存储空间。

[0010] 本文公开的某些实施例还包括一种用于在大型存储系统中执行写入请求的方法。该方法包括：由大型存储系统的计算节点接收写入请求，其中，写入请求至少包括要写入的数据元素；以及确定写接收到的数据元素的位置，其中该位置在大型存储系统的至少一个存储节点中；在确定的位置将数据元素写入写缓冲器；在所述计算节点处，在将所述数据元素写入所述写缓冲器时接收到确认。

[0011] 本文公开的某些实施例还包括一种用于在大型存储系统中执行读取请求的方法。该方法包括在大型存储系统的计算节点处接收读取请求，其中，读取请求至少包括要读取的数据元素的标识符；以及确定所请求的数据元素的数据块；确定数据块的位置，其中该位置在大型存储系统的至少一个存储节点中；访问确定的位置以检索所请求元素的数据块。

[0012] 本文公开的某些实施例包括一种用于在大型存储系统中执行读取请求的方法。该方法包括在大型存储系统的计算节点处接收读取请求，其中，读取请求至少包括要读取的数据元素的标识符；以及确定所请求的数据元素的数据块；确定数据块的位置，其中该位置在大型存储系统的至少一个存储节点中；访问确定的位置以检索所请求元素的数据块。

附图说明

[0013] 在说明书的结尾处，在权利要求书中特别指出并明确要求保护本文公开的主题。根据结合附图的以下详细描述，所公开的实施例的前述和其他目的、特征和优点将是显而易见的。

[0014] 图1是用于描述根据各种公开的实施例的大型存储系统的网络图；

[0015] 图2A是根据一个实施例的存储节点的示例框图；

[0016] 图2B是根据一个实施例的存储节点的接口模块的示例框图；

[0017] 图3是根据一个实施例的计算节点的示例框图；

[0018] 图4是根据一个实施例的用于在大型存储系统中执行读取请求的方法的示例流程图；

[0019] 图5是根据一个实施例的用于在大型存储系统中执行写入请求的方法的示例流程图。

具体实施方式

[0020] 重要的是要注意，本文公开的实施例仅仅是本文创新教导的许多有利用途的例子。一般而言，在本申请的说明书中做出的陈述不必限制任何各种要求保护的实施例。而且，某些陈述可能适用于某些发明特征，而不适用于其他特征。通常，除非另外指出，否则单数元素可以是复数，反之亦然，而不会失去一般性。在附图中，相同的附图标记通过若干视图指代相同的部分。

[0021] 根据公开的实施例，公开了一种大型存储系统。该系统包括通过通信结构通信地连接到至少一个存储节点的至少一个计算节点。所公开的大型存储系统的布置提供了分解的软件定义的存储架构，其中计算和存储资源被解耦。

[0022] 如将在下面更详细地讨论的，系统中的每个存储节点可以包括多个消费者级固态驱动器(SSDs)。利用SSDs可以进行快速和随机访问的读写操作。计算节点被配置为控制写入操作，以便对每个SSD执行较少数量的写入-擦除周期，从而延长了每个SSDs的寿命。结果，可以利用低等级(和低成本)的SSDs，从而降低了系统的总成本。

[0023] 图1示出了根据所公开的实施例的大型存储系统100的示例图。存储系统100包括N个计算节点110-1至110-N(仅出于简化目的，以下分别称为计算节点110，统称为多个计算节点110，N是等于或大于1的整数)，M个存储节点存储节点120-1到120-M(仅出于简化目的，以下分别称为存储节点120，统称为存储节点120，M是等于或大于1的整数)。计算节点110和存储节点120通过通信结构130连接。

[0024] 在一实施例中，计算节点110可被实现为物理机或虚拟机。物理机可以包括计算机、服务器等。虚拟机可以包括任何虚拟化的计算实例(在计算硬件上执行)，例如虚拟机、软件容器等。

[0025] 应注意，在两种配置(物理或虚拟)中，计算节点110均不需要任何专用硬件。在图3中提供了计算节点110的示例布置。

[0026] 计算节点110被配置为执行与存储节点120的管理有关的任务。在一个实施例中，每个计算节点110经由网络150与客户端设备140(或安装在其中的应用程序)对接。最后，计算节点110被配置为接收请求(例如，读取或写入请求)并以持久的方式迅速地服务这些请求。网络150可以是但不限于因特网、万维网(WWW)、局域网(LAN)、广域网(WAN)等。

[0027] 在一个实施例中，计算节点110被配置为与由客户端设备或应用实现的不同协议(例如，HTTP，FTP等)对接，并且管理来自存储节点120的读取和写入操作。计算节点110还被配置为将协议命令转换成统一的结构(或语言)。然后，每个计算节点110还被配置为逻辑上寻址和映射存储在存储节点120中的所有元素。

[0028] 此外，每个计算节点110通过存储在存储节点120上的状态来维护元素的逻辑操作以及元素(例如，目录树)和元素属性(例如，元数据)之间的关系。元素可以包括文件、目录、对象等。元素的映射和寻址使计算节点110可以在存储节点120中维护元素的确切物理位置。

[0029] 在一个实施例中，为了有效地从物理层读取数据并将数据写入存储节点120，每个计算节点110执行多个进程，包括数据缩减、数据弹性和闪存管理行为(例如，碎片整理、磨损平衡等)。

[0030] 应注意，每个计算节点110以与所有其他计算节点110相同的方式操作。在发生故障的情况下，任何计算节点110都可以替换发生故障的节点。此外，每个计算节点都可以控制和管理一个或多个模式存储节点120，而与存储节点120的特定体系结构无关。因此，在计算节点110和存储节点120之间不存在耦合。在不增加存储节点数量(或容量)的情况下将其添加到系统100中，反之亦然，可以在不增加计算节点110数量的情况下添加存储节点。

[0031] 为了允许系统100的可扩展性，计算节点110之间不彼此通信以服务应用程序请求。此外，每个计算节点110可以被独立地升级，安装有不同的软件版本或两者。

[0032] 存储节点120在大型系统100中提供存储和状态。为此，每个存储节点120包括相对便宜的多个消费级SSDs。这种类型的SSDs具有许多缺点。特别是，SSDs的耐用性、数据完整性、写入延迟、电源保护、并行写入和垃圾回收均较差。使用消费级SSDs的关键缺点是耐用

性,这意味着此类驱动器的写入-擦除周期明显少于企业级驱动器。

[0033] 根据下面更详细讨论的公开的实施例,控制用于向SSDs写入和删除数据的操作以减少写入-擦除周期的次数,从而确保使用消费者级SSDs的企业级性能。

[0034] 存储节点120可以被配置为具有彼此相同的容量或彼此不同的容量。在一个实施例中,使在每个存储节点120中存储的数据在存储节点内部冗余,在不同的存储节点处冗余,或两者兼而有之。如将在下面参考图2A和图2B讨论的,每个存储节点120还包括非易失性随机存取存储器(NVRAM)和用于与计算节点110对接的接口模块。

[0035] 存储节点120通过通信结构130与计算节点110通信。应当注意,每个计算节点110可以通过通信结构130与每个存储节点120通信。计算节点110和存储节点120之间没有直接耦合。

[0036] 在实施例中,通信结构130可以包括以太网结构,无线带宽结构等。具体地,通信结构130可以启用诸如但不限于,基于融合以太网(RoCE)的远程直接存储器访问(RDMA),iWARP,非易失性存储器标准(NVMe)等的通信协议。应当注意,仅出于示例目的提供本文讨论的通信协议,并且在不脱离本公开的范围的情况下,根据本文公开的实施例可以等同地利用其他通信协议。

[0037] 应注意,在一个示例部署中,客户端设备140是计算节点110的一部分。在这样的部署中,系统100不与外部网络(例如,网络150)通信。进一步注意,计算节点110和存储节点120之间的通信总是通过结构130来促进。应该进一步注意,计算节点120之间可以通过结构130彼此通信。结构130是共享结构。

[0038] 图2A示出了根据一个实施例的存储节点120的示例框图。存储节点120包括多个SSDs 210-1至SSDs 210-P(以下仅为了简单起见,在下文中分别称为SSD 210,统称为SSDs 210),至少一个NVRAM和接口模块220。在一些配置中,两个接口模块220被提供用于冗余。如上所述,SSDs 210可以是消费级SSDs。SSDs的数量,SSDs的配置以及SSDs的容量在一个存储节点210与另一个存储节点之间以及在存储节点210内可能有所不同。SSDs 210放置在适用于以下情况的企业机柜(机架)中:具有消费级外形尺寸的主机SSDs。例如,企业级外形尺寸通常为2.5英寸,以符合企业机箱的期望,而低成本消费级SSDs的外形尺寸为M.2。为了弥合外形差异,可将多个SSDs放在一个企业机柜的单个插槽。

[0039] 根据公开的实施例,利用NVRAM 223来减少对SSDs 210的写访问和写放大的次数。根据一个实施例,数据首先被写入NVRAM 223,该NVRAM 223在每次这样的数据写入之后返回确认。然后,在后台处理期间,将数据从NVRAM 223传输到SSDs 210。将数据保留在NVRAM 223中,直到将数据完全写入SSDs 210。此外,该写入过程可确保断电时不会丢失任何数据。

[0040] 由于NVRAM 223支持低写入延迟和并行写入,因此整个存储节点120支持这些功能。具体而言,一旦将数据保存到NVRAM 223,就通过确认写入请求来实现低延迟。并行写入是通过NVRAM 223为多个并发写入请求提供服务,并且在后台过程中通过将数据保存到SSDs 210中以独立满足此类请求而实现的。

[0041] 在一个实施例中,NVRAM 223用于在计算节点110的控制下执行有效的垃圾收集处理。通常,这样的处理释放用于擦除和后续写入的块。在消费级SSDs中,垃圾收集很弱,并且会导致固态硬盘响应速度变慢。为了改进垃圾收集过程,将来自多个客户端设备(未显示)的写入请求进行汇总,直到准备好完整的数据块为止。然后,将完整的数据块保存在SSDs

210中。这样，“浪费”的存储空间量得以最小化，从而简化了垃圾收集过程的操作。应当注意，NVRAM 223可以是例如3D Xpoint或任何非易失性存储器(NVM)设备。

[0042] 图2B示出了接口模块220的示例框图。在示例实施例中，接口模块220包括网络接口卡(NIC)222和通过诸如PCIe总线的内部总线(未示出)连接的交换机224。

[0043] NIC 222允许存储节点120与计算节点(110，图1)通过通信结构(130，图1)进行通信。NIC 222可以允许经由以上讨论的协议中的至少一种进行通信。

[0044] 交换机224允许将多个SSDs 210和NVRAM 223连接到NIC 222。在示例实施例中，交换机224是PCIe交换机。在另一实施例中，利用多个PCIe交换机来支持与SSDs的更多连接。在一些配置中，在非PCIe的SSDs 210可用的情况下(例如，以太网SSDs)，交换机224可以是非PCIe交换机，例如以太网交换机。

[0045] 根据所公开的实施例，存储节点110被设计为即使在使用通常容易出现故障的消费级SSDs时也支持数据弹性。当SSD 210发生故障时，无法从SSD读取数据或将数据写入SSD。根据实施例，防止SSDs故障的弹性方案基于N+K擦除校正方案。也就是说，对于每“N”个数据SSDs，都会使用“K”个冗余SSDs，其中N和K是整数。应当理解，随着系统的发展，这种弹性方案提供了增加的冗余和减少的容量开销。

[0046] 在另一个实施例中，数据缩减过程由任何计算节点110管理，以便节省物理存储空间。为此，来自客户端设备的数据在被接收时即以未压缩或本地压缩的方式被写入NVRAM 223。然后，作为后台处理的一部分，全局压缩数据并将其保存到SSDs 210。如上所述，一旦将数据块完全写入SSDs 210，就将从NVRAM 223中删除该数据块。

[0047] 压缩可以包括主动检查数据以检测重复项，并消除此类重复项。重复项检测可以在块内部的块级别和字节级别。应当注意，压缩不会增加等待时间，因为当将数据保存在NVRAM 223中时，写入请求得到了确认。

[0048] 图3示出了示出根据实施例的计算节点110的示例框图。计算节点110包括处理电路310、存储器320、第一网络接口控制器(NIC)330和第二NIC 340。在一个实施例中，计算节点110的组件可以经由总线305通信地连接。

[0049] 处理电路310可以被实现为一个或多个硬件逻辑组件和电路。例如但不限于，可以使用的说明性类型的硬件逻辑组件包括FPGAs、ASICs、ASSPs、SOCs、通用微处理器、微控制器、DSPs等，或可以执行信息的计算或其他操作的任何其他硬件逻辑组件。

[0050] 存储器320可以是易失性的(例如，RAM等)、非易失性的(例如，ROM，闪存等)或其组合。在一种配置中，用于实现由计算节点110执行的一个或多个过程的计算机可读指令或软件可以存储在存储器320中。软件应被广义地解释为表示任何类型的指令，无论被称为软件、固件、中间件、微码、硬件描述语言或其他方式。指令可以包括代码(例如，以源代码格式、二进制代码格式、可执行代码格式或任何其他合适的代码格式)。

[0051] 第一NIC 330允许计算节点110经由通信结构130(参见图1)与存储节点通信，以提供对存储在存储节点中的数据的远程直接存储器访问。在一个实施例中，第一NIC 130可以经由诸如但不限于无线宽带、融合以太网上的RDMA(RoCE)、iWARP等的RDMA协议来启用通信。

[0052] 第二NIC 340允许计算节点110通过通信网络(例如，网络150，图1)与客户端设备(例如，客户端设备140，图1)进行通信。这样的网络的示例包括但不限于因特网、万维网

(WWW)、局域网(LAN)、广域网(WAN)等。应当理解,在一些配置中,计算节点110可以包括单个NIC。例如,在共享结构时,此配置适用。

[0053] 将参考图1-3中描述的元素来讨论大型系统的各种功能。

[0054] 作为任何存储系统,所公开的大型存储系统100支持数据流,包括但不限于写入和读取。根据所公开的实施例,以同步方式执行读取和写入数据流。当在计算节点110处接收到请求时,数据流开始,并且当指示请求完成的答复被发送回客户端设备140时,数据流结束。所有数据流以持久方式执行。也就是说,在计算节点110之一或存储节点120之一发生故障的情况下,不会丢失任何数据。

[0055] 根据所公开的实施例,每个计算节点110可以访问整个命名空间,而每个存储节点120负责命名空间的特定(预定义)范围,即,大型存储系统100实施分片机制。此外,每个计算节点110在存储节点120上实现持久锁定,以确保同步执行数据流,特别是写入请求,从而保持数据完整性。根据元素句柄以及根据相应存储节点210处理的偏移范围、命名空间范围或两者,将数据分片在NVRAM 223上。不论系统大小如何,碎片数都是恒定的,因此,在较小的系统中,多个碎片将驻留在同一NVRAM 223上。

[0056] 图4示出了示例流程图400,该流程图示出了根据实施例的用于在大型存储系统100中执行读取请求的方法。在一个实施例中,图4所示的方法由计算节点110执行。

[0057] 在S410,从客户端设备接收读取请求。这样的请求通常包括要读取的数据元素的标识符。如上所述,元素可以包括文件、目录、对象等。

[0058] 在S420,确定所请求的数据元素的数据块。例如,数据块可以是文件、文件的一部分、目录或目录中的文件等等。

[0059] 在S430,确定所请求的数据元素的位置。该位置是多个存储节点120中的一个。在每个存储节点120中,数据元素可以保存在NVRAM 223或一个或多个SSDs 210中。在一个实施例中,该位置是使用令牌确定的。

[0060] 令牌是代表指向数据或元数据块的指针的抽象概念。令牌可以是三种类型之一:直接令牌、映射器或数据缩减器。直接令牌保存数据块所在的物理地址(在NVRAM或SSD上)。映射器令牌用于从NVRAM检索物理地址。映射器令牌被提供给映射表以检索物理地址。数据缩减器令牌用于从接口模块检索物理地址。要使用的令牌类型取决于要读取的数据。例如,数据块具有直接令牌或数据缩减令牌。元数据块具有直接或映射器令牌。

[0061] 在S440,使用所确定的位置,从存储这种数据的存储节点访问和检索所请求元素的数据。在某些情况下,例如,由于所访问的存储节点或SSDs的故障,访问该位置的尝试可能不会成功。在这种情况下,无法从确定的位置读取数据。为了满足读取请求,基于弹性方案确定一个或多个备用SSDs的集合,并从此类SSD(s)读取和解码信息。当无法访问SSDs时,此解决方案可以正常工作。当无法访问整个存储节点,并且所需的冗余SSDs的集合位于同一存储节点上时,将返回错误消息。

[0062] 在另一个实施例中,对检索到的数据采用校验和机制以确保这样的数据不被破坏。如果检索到的数据已损坏,则将从弹性方案确定的冗余SSDs中再次读取数据。

[0063] 在S450,生成包括所检索到的数据的答复并将其发送到客户端设备。

[0064] 图5示出了示例流程图500,该流程图示出了根据实施例的用于在大型存储系统100中执行写入请求的方法。在一个实施例中,图5所示的方法由计算节点110执行。

[0065] 在S510,从客户端设备接收写入请求。这样的请求通常包括要写入存储节点的数据元素。如上所述,元素可以包括文件、目录、对象等。

[0066] 在S520,确定用于写入所请求的元素的位置。该位置是一个或多个存储节点120的位置,更具体地说是一个存储节点中的NVRAM 223的位置。在一个实施例中,S520还包括确定分配给所接收的数据元素的分片以及所确定的分片所驻留的NVRAM。例如,可以使用将元素处理程序映射到其物理存储目标的哈希表来确定(分片,NVRAM或两者的)位置。应当注意,在某些情况下,确定的位置可以分布在多个存储节点上的多个NVRAM上。如果一个NVRAM的空间不足,就会出现这种情况。

[0067] 在S530,将元素数据写入确定位置的写缓冲器。写缓冲器确保将完整的数据块刷新到SSDs 210。在预定义的时间窗口内,任何新数据都将写入SSDs。在一些实施例中,S530包括在写缓冲器中分配空间。在又一个实施例中,S530还包括更新指向缓冲器中的数据位置的指针以及用于指示NVRAM的占用的其他数据结构。

[0068] 根据又一个实施例,S530包括在写入元素数据并且更新指针/数据结构时锁定NVRAM(写缓冲器)。一旦完成对NVRAM的写入,将释放锁定。锁可以是局部锁或全局锁。

[0069] 在S540,接收写入完成的确认。在完成对NVRAM的写入后,会从存储节点接收到这样的确认。

[0070] 在S550,将指示写入请求已经完成的回复发送给客户端。

[0071] 参考图5所讨论的写入请求处理是同步流程。写缓冲器或NVRAM中的数据以异步方式保存到SSDs。当NVRAM中的数据达到预定义的阈值时,数据将从NVRAM迁移到SSDs。

[0072] 在从NVRAM迁移数据之前,执行数据缩减过程以减小数据大小。此外,应用弹性方案以将数据保存在多个位置。

[0073] 本文公开的各种实施例可以被实现为硬件、固件、软件或其任何组合。此外,软件优选地被实现为有形地体现在由部分或某些设备和/或设备的组合组成的程序存储单元或计算机可读介质上的应用程序。可以将应用程序上载到包括任何适当架构的机器并由其执行。优选地,机器在具有诸如一个或多个中央处理单元(“CPU”)、存储器和输入/输出接口的硬件的计算机平台上实现。该计算机平台还可以包括操作系统和微指令代码。本文描述的各种进程和功能可以是微指令代码的一部分,也可以是应用程序的一部分,或者是它们的任何组合,这些进程和功能可以由CPU执行,无论是否明确显示了这种计算机或处理器。另外,各种其他外围单元可以连接到计算机平台,例如附加数据存储单元和打印单元。此外,非暂时性计算机可读介质是除了暂时性传播信号之外的任何计算机可读介质。

[0074] 应该理解的是,本文中使用诸如“第一”,“第二”等之类的名称对元件的任何引用通常并不限制这些元件的数量或顺序。相反,这些指定在本文中通常用作区分两个或更多个元素或元素实例的便利方法。因此,对第一和第二元件的引用并不意味着在那里仅可以使用两个元件,或者第一元件必须以某种方式在第二元件之前。同样,除非另有说明,否则一组元素包括一个或多个元素。此外,在说明书或权利要求书中使用的,形式为“A、B或C中的至少一个”或“A、B或C中的一个或多个”或“由A、B和C构成的组中的至少一个”或“A、B和C中的至少一个”的术语,是指“A或B或C或这些元素的任意组合”。例如,该术语可以包括A、或B、或C、或A和B、或A和C、或A和B和C、或2A、或2B、或2C,依此类推。

[0075] 本文中引用的所有示例和条件语言旨在用于教学目的,以帮助读者理解所公开实

施例的原理和发明人为进一步发展本领域所贡献的概念，并且应解释为不限于此类具体引用的示例和条件。此外，本文中引用原理、方面和实施例及其特定示例的所有陈述旨在涵盖其结构和功能上的等同物。另外，意图是这样的等同物包括当前已知的等同物以及将来开发的等同物，即，开发的执行相同功能的任何元件，而与结构无关。

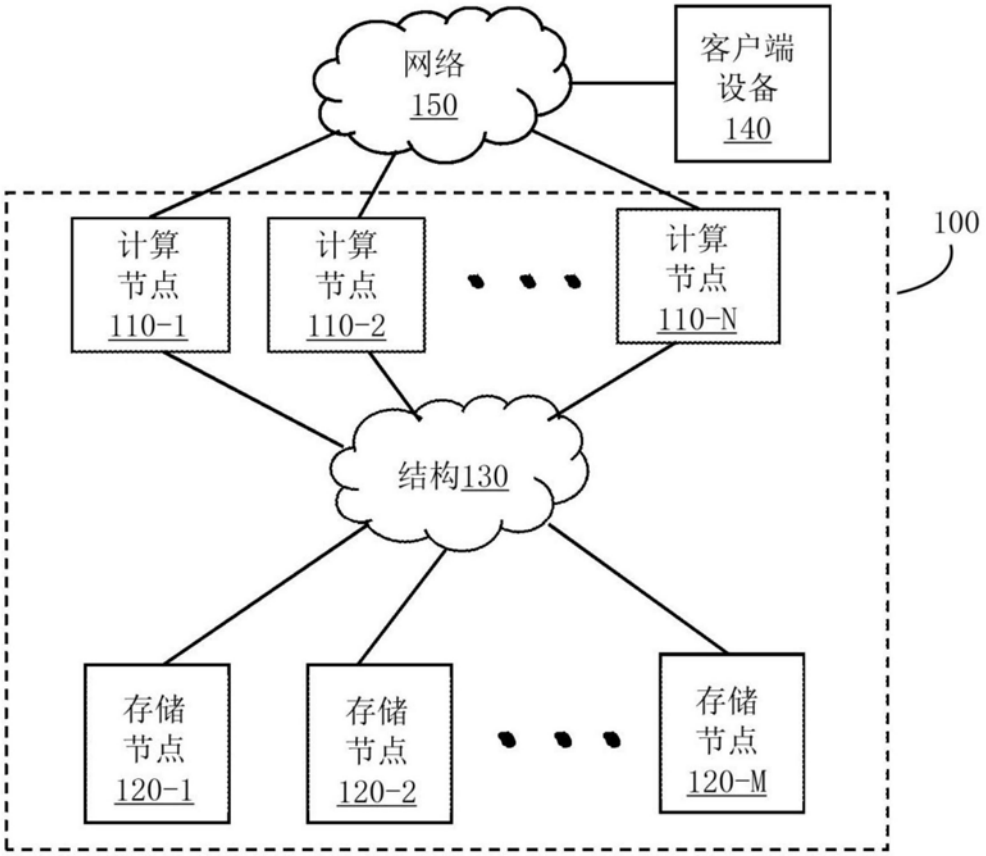

网络
150
客户端
设备
140
100
计算
节点
110-1
计算
节点
110-2
计算
节点
110-N
结构130
存储
节点
120-1
存储
节点
120-2
存储
节点
120-M

图1

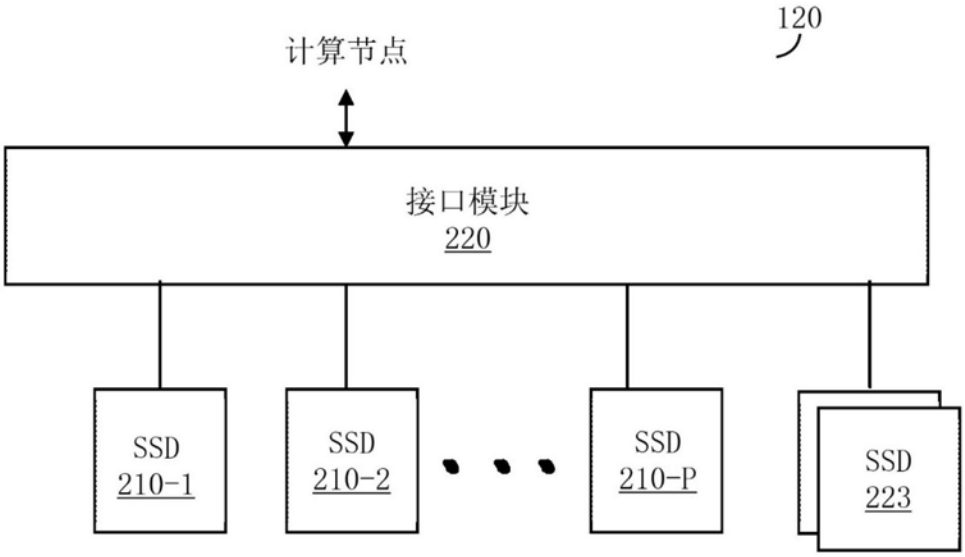

120
计算节点
接口模块
220
SSD
210-1
SSD
210-2
SSD
210-P
SSD
223

图2A

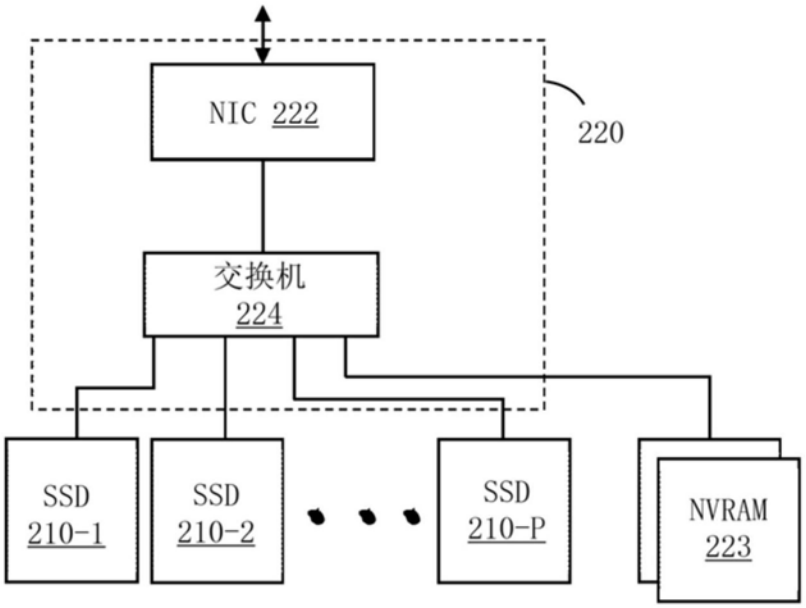

NIC 222
220
交换机
224
SSD
210-1
SSD
210-2
SSD
210-P
NVRAM
223

图2B

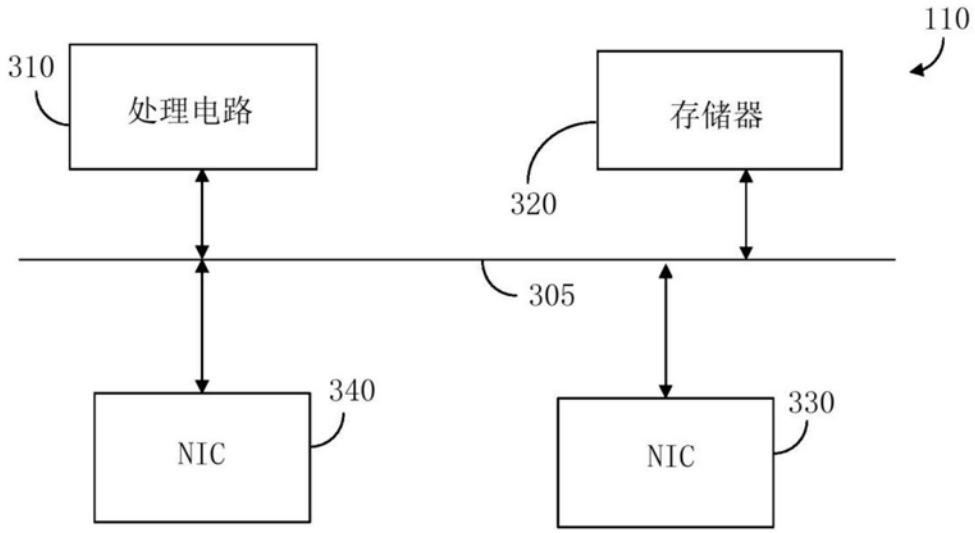

110
310
处理电路
存储器
320
305
340
NIC
330
NIC

图3

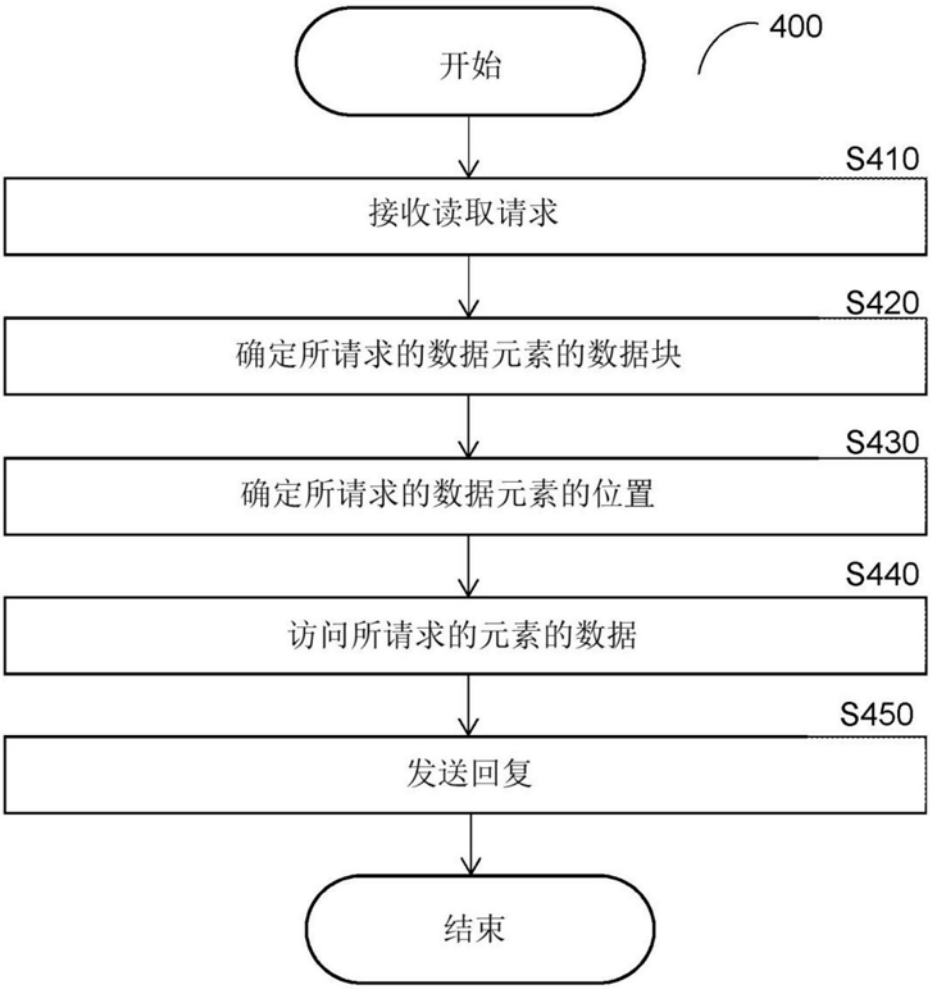

400
开始
S410
接收读取请求
S420
确定所请求的数据元素的数据块
S430
确定所请求的数据元素的位置
S440
访问所请求的元素的数据
S450
发送回复
结束

图4

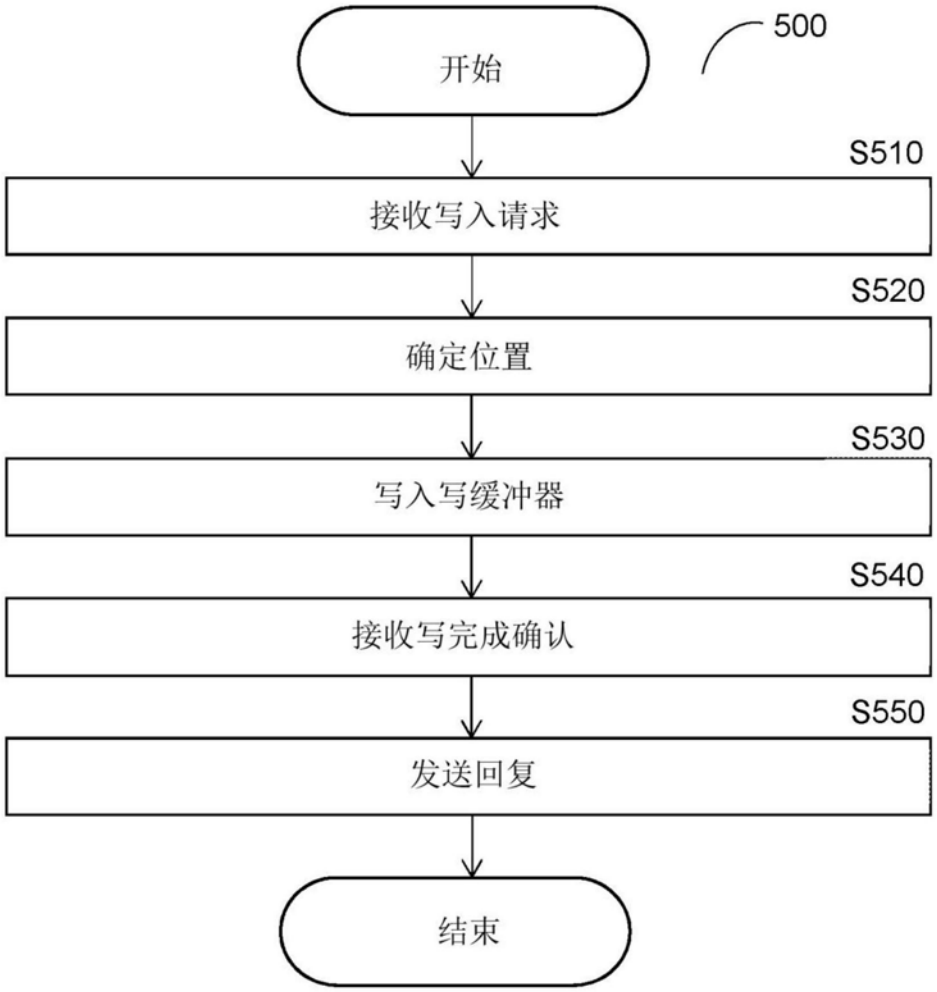

500
开始
S510
接收写入请求
S520
确定位置
S530
写入写缓冲器
S540
接收写完成确认
S550
发送回复
结束

图5