(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 107079046 B
(45)授权公告日 2020.11.17

(21)申请号 201580056123.X

(22)申请日 2015.10.28

(65)同一申请的已公布的文献号
申请公布号 CN 107079046 A

(43)申请公布日 2017.08.18

(30)优先权数据
62/072,847 2014.10.30 US
62/075,000 2014.11.04 US
62/076,336 2014.11.06 US
62/121,294 2015.02.26 US
62/133,179 2015.03.13 US
14/924,281 2015.10.27 US

(85)PCT国际申请进入国家阶段日
2017.04.17

(86)PCT国际申请的申请数据
PCT/US2015/057860 2015.10.28

(87)PCT国际申请的公布数据
WO2016/069773 EN 2016.05.06

(73)专利权人 甲骨文国际公司
地址 美国加利福尼亚

(72)发明人 E·塔索拉斯 B·D·约翰森 E·G·格兰

(74)专利代理机构 中国贸促会专利商标事务所有限公司 11038
代理人 边海梅

(51)Int.Cl.
H04L 29/08(2006.01)
G06F 9/48(2006.01)

(56)对比文件
US 2008189432 A1,2008.08.07
CN 104094230 A,2014.10.08
CN 104115121 A,2014.10.22
CN 104094229 A,2014.10.08
Christopher Clark.
“Livemigrationofvirtualmachines”.《2nd Symposium on Networked Systems Design & Implementation》.2005,

审查员 颜光友

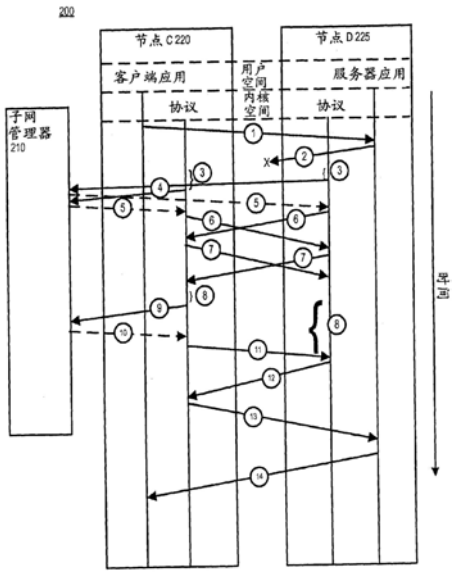
权利要求书4页 说明书10页 附图6页

(54)发明名称

用于为动态云提供子网经管(SA)查询高速缓存的系统和方法

(57)摘要

系统和方法可以在云环境中支持子网管理。在云环境中的虚拟机迁移期间,子网管理器会成为使高效服务延迟的瓶颈点。系统和方法可以通过确保虚拟机在迁移之后保留多个地址来缓解这个瓶颈点。该系统和方法还可以允许云环境中的每个主机节点与本地高速缓存关联,在重新建立与迁移后的虚拟机的通信时虚拟机可以利用该本地高速缓存。



CN 107079046 B

1.一种用于在云环境中支持子网管理的方法,包括:

在所述云环境内提供包括第一主机节点和第三主机节点的多个主机节点,第一主机节点与至少第一管理程序和第一主机信道适配器关联,其中所述多个主机节点中的每一个包括本地高速缓存,每个本地高速缓存包括一个或多个路径记录;

在第一主机节点上提供第一虚拟机,第一虚拟机与多个地址关联;

由第三主机节点向子网管理器发送查询,所述查询请求用于第一虚拟机的第一路径记录,第一路径记录包括与第一虚拟机关联的多个地址;

在与第三主机节点关联的本地高速缓存内存储从所述子网管理器接收到的第一路径记录;

将第一虚拟机从第一主机节点迁移到所述云环境内的所述多个主机节点中所提供的第二主机节点上的第二虚拟机,第二主机节点与至少第二管理程序和第二主机信道适配器关联;

当第一虚拟机从第一主机节点迁移到所提供的第二主机节点时,由第三主机节点检测通信中断;

由第三主机节点在与第三主机节点关联的本地高速缓存中查找第一路径记录;及

由第三主机节点基于第一路径记录在没有与所述子网管理器的进一步通信的情况下建立第二虚拟机和第三主机节点之间的通信。

2.如权利要求1所述的方法,其中所述将第一虚拟机从第一主机节点迁移到所述云环境内所提供的第二主机节点包括:

从第一管理程序分离第一虚拟机,所述从第一管理程序分离第一虚拟机包括从第一虚拟机分离与第一虚拟机关联的第一虚拟功能;

向第二主机节点提供与第一虚拟机关联的所述多个地址;

将所述多个地址指派给第二虚拟功能,第二虚拟功能与第二管理程序关联;

将第一虚拟机从第一主机节点迁移到第二主机节点上的第二虚拟机;及

将第二虚拟机暴露于与第一虚拟机关联的所述多个地址。

3.如权利要求2所述的方法,还包括:

在将第一虚拟机从第一主机节点迁移到所述云环境内所提供的第二主机节点之后,建立第二虚拟机和第三虚拟机之间的通信,第三虚拟机在所述多个主机节点中的第三主机节点上被提供;

其中在第一虚拟机从第一主机节点迁移到所述云环境内所提供的第二主机节点之前,第一虚拟机和第三虚拟机在进行通信。

4.如权利要求1所述的方法,其中所述一个或多个路径记录中的每一个包括多个特性,所述多个特性包括服务级别和最大发送单元。

5.如权利要求1所述的方法,其中,第一路径记录是基于由第三主机节点向所述子网管理器发送的与第一虚拟机关联的所述查询而创建的,所述子网管理器与所述云环境关联。

6.如权利要求5所述的方法,其中,在创建第一路径记录之后,第三主机节点不向所述子网管理器发送关于与第一虚拟机关联的所述多个地址的进一步查询。

7.如权利要求5或6中任一项所述的方法,其中,响应于由第三主机节点向所述子网管理器发送的与第一虚拟机关联的所述查询,所述子网管理器返回经标记的第一路径记录,

所述经标记的第一路径记录包括启用路径高速缓存的标记，该启用路径高速缓存的标记指示第一路径记录跨所述通信中断而存留。

8.如权利要求7所述的方法，其中，响应于对所述子网管理器的关于与在所述多个主机节点中另一个主机上提供的另一个虚拟机关联的另外多个地址的另一个查询，所述子网管理器返回另一个经标记的路径记录，所述另一个经标记的路径记录包括启用路径高速缓存的标记，该启用路径高速缓存的标记指示所述另一个路径记录跨另一个通信中断而存留。

9.如权利要求3至6或8中任一项所述的方法，其中第二虚拟机和第三虚拟机之间的通信基于InfiniBand协议。

10.如权利要求2所述的方法，还包括：

在将第一虚拟机从第一主机节点迁移到所述云环境内所提供的第二主机节点之后，建立第二虚拟机和第三实体之间的通信，第三实体是以下中的一者：物理主机、存储设备或先前经InfiniBand协议与迁移后的第一虚拟机进行通信的另一个实体；

在与第三实体关联的本地高速缓存中存储第一路径记录，第一路径记录至少包括与第一虚拟机关联的所述多个地址；

当第一虚拟机从第一主机节点迁移到所提供的第二主机节点时，由第三实体检测通信中断；

由第三实体在与第三实体关联的本地高速缓存中查找第一路径记录；及

至少基于第一路径记录，建立第二虚拟机和第三实体之间的通信，

其中在第一虚拟机从第一主机节点迁移到所述云环境内所提供的第二主机节点之前，第一虚拟机和第三实体在进行通信。

11.一种用于在云环境中支持子网管理的系统，所述系统包括：

一个或多个微处理器；及

在所述一个或多个微处理器上运行的处理器，其中所述处理器操作以执行包括以下操作的步骤：

在所述云环境内提供包括第一主机节点和第三主机节点的多个主机节点，第一主机节点与至少第一管理程序和第一主机信道适配器关联，其中所述多个主机节点中的每一个包括本地高速缓存，每个本地高速缓存包括一个或多个路径记录；

在第一主机节点上提供第一虚拟机，第一虚拟机与多个地址关联；

由第三主机节点向子网管理器发送查询，所述查询请求用于第一虚拟机的第一路径记录，第一路径记录包括与第一虚拟机关联的多个地址；

在与第三主机节点关联的本地高速缓存内存储从所述子网管理器接收到的第一路径记录；

将第一虚拟机从第一主机节点迁移到所述云环境内所述多个主机节点中所提供的第二主机节点上的第二虚拟机，第二主机节点与至少第二管理程序和第二主机信道适配器关联；

当第一虚拟机从第一主机节点迁移到所提供的第二主机节点时，由第三主机节点检测通信中断；

由第三主机节点在与第三主机节点关联的本地高速缓存中查找第一路径记录；及

由第三主机节点基于第一路径记录在没有与所述子网管理器的进一步通信的情况下

建立第二虚拟机和第三主机节点之间的通信。

12.如权利要求11所述的系统，其中所述处理器操作以执行进一步的步骤，所述步骤包括：

从第一管理程序分离第一虚拟机，所述从第一管理程序分离第一虚拟机包括从第一虚拟机分离与第一虚拟机关联的第一虚拟功能；

向第二主机节点提供与第一虚拟机关联的所述多个地址；

将所述多个地址指派给第二虚拟功能，第二虚拟功能与第二管理程序关联；

将第一虚拟机从第一主机节点迁移到第二主机节点上的第二虚拟机；及

将第二虚拟机暴露于与第一虚拟机关联的所述多个地址。

13.如权利要求12所述的系统，其中所述处理器操作以执行进一步的步骤，所述步骤包括：

在将第一虚拟机从第一主机节点迁移到所述云环境内所提供的第二主机节点之后，建立第二虚拟机和第三虚拟机之间的通信，第三虚拟机在所述多个主机节点中的第三主机节点上被提供；及

其中在第一虚拟机从第一主机节点迁移到所述云环境内所提供的第二主机节点之前，第一虚拟机和第三虚拟机在进行通信。

14.如权利要求11所述的系统，其中所述一个或多个路径记录中的每一个包括多个特性，所述多个特性包括服务级别和最大发送单元。

15.如权利要求11所述的系统，其中，第一路径记录是基于由第三主机节点向所述子网管理器发送的与第一虚拟机关联的所述查询而创建的，所述子网管理器与所述云环境关联。

16.如权利要求15所述的系统，其中，在创建第一路径记录之后，第三主机节点不向所述子网管理器发送关于与第一虚拟机关联的所述多个地址的进一步查询。

17.如权利要求13至16中任一项所述的系统，其中第二虚拟机和第三虚拟机之间的通信基于InfiniBand协议。

18.一种具有存储在其上的指令的非暂时机器可读存储介质，用于在云环境中支持子网管理，所述指令在被执行时使系统执行包括以下操作的步骤：

在所述云环境内提供包括第一主机节点和第三主机节点的多个主机节点，第一主机节点与至少第一管理程序和第一主机信道适配器关联，其中所述多个主机节点中的每一个包括本地高速缓存，每个本地高速缓存包括一个或多个路径记录；

在第一主机节点上提供第一虚拟机，第一虚拟机与多个地址关联；

由第三主机节点向子网管理器发送查询，所述查询请求用于第一虚拟机的第一路径记录，第一路径记录包括与第一虚拟机关联的多个地址；

在与第三主机节点关联的本地高速缓存内存储从所述子网管理器接收到的第一路径记录；

将第一虚拟机从第一主机节点迁移到所述云环境内所述多个主机节点中所提供的第二主机节点上的第二虚拟机，第二主机节点与至少第二管理程序和第二主机信道适配器关联；

当第一虚拟机从第一主机节点迁移到所提供的第二主机节点时，由第三主机节点检测

通信中断；

由第三主机节点在与第三主机节点关联的本地高速缓存中查找第一路径记录；及

由第三主机节点基于第一路径记录在没有与所述子网管理器的进一步通信的情况下建立第二虚拟机和第三主机节点之间的通信。

19.如权利要求18所述的非暂时机器可读存储介质，所述步骤还包括：

从第一管理程序分离第一虚拟机，所述从第一管理程序分离第一虚拟机包括从第一虚拟机分离与第一虚拟机关联的第一虚拟功能；

向第二主机节点提供与第一虚拟机关联的所述多个地址；

将所述多个地址指派给第二虚拟功能，第二虚拟功能与第二管理程序关联；

将第一虚拟机从第一主机节点迁移到第二主机节点上的第二虚拟机；及

将第二虚拟机暴露给与第一虚拟机关联的所述多个地址。

20.如权利要求19所述的非暂时机器可读存储介质，所述步骤还包括：

在将第一虚拟机从第一主机节点迁移到所述云环境内所提供的第二主机节点之后，建立第二虚拟机和第三虚拟机之间的通信，第三虚拟机在所述多个主机节点中的第三主机节点上被提供；及

其中在第一虚拟机从第一主机节点迁移到所述云环境内所提供的第二主机节点之前，第一虚拟机和第三虚拟机在进行通信。

21.如权利要求18所述的非暂时机器可读存储介质，其中所述一个或多个路径记录中的每一个包括多个特性，所述多个特性包括服务级别和最大发送单元。

22.如权利要求18所述的非暂时机器可读存储介质，其中，第一路径记录是基于由第三主机节点向所述子网管理器发送的与第一虚拟机关联的所述查询而创建的，所述子网管理器与所述云环境关联，及

其中，在创建第一路径记录之后，第三主机节点不向所述子网管理器发送关于与第一虚拟机关联的所述多个地址的进一步查询。

23.一种用于在云环境中支持子网管理的装置，包括用于实现如权利要求1-10中任一项所述的方法的单元。

用于为动态云提供子网经管(SA)查询高速缓存的系统和方法

[0001] 版权声明

[0002] 本专利文档公开的一部分包含受版权保护的素材。版权拥有者不反对任何人对专利文档或专利公开按照在专利商标局的专利文件或记录中出现那样进行的传真复制,但是除此之外在任何情况下都保留所有版权。

技术领域

[0003] 本发明一般而言涉及计算机系统,并且具体地涉及云环境。

背景技术

[0004] 在InfiniBand子网中,子网管理器(SM,Subnet Manager)是潜在的瓶颈。当InfiniBand子网的尺寸增加时,主机之间的路径数量会呈多项式地(polynomially)增加,并且当接收到许多并发的路径解析请求时,SM可能不能及时地为网络提供服务。这种可扩展性挑战在动态虚拟化云环境中进一步被放大。当具有InfiniBand互连的虚拟机(VM)实时迁移(live migrate)时,VM地址改变。这些地址改变导致对SM的附加负载,因为通信对等体(peer)向SM发送子网经管(SA,Subnet Administration)路径记录查询,以解析新的路径特性。

发明内容

[0005] 系统和方法可以支持云环境中的子网管理。在云环境中的虚拟机迁移期间,子网管理器可能成为使高效服务延迟的瓶颈点。系统和方法可以通过确保虚拟机在迁移之后保留多个地址来缓解这个瓶颈点。系统和方法还可以允许云环境内的每个主机节点与在重新建立与迁移后的虚拟机的通信时虚拟机可以利用的本地高速缓存关联。

附图说明

[0006] 图1示出了根据实施例的在云环境中支持VM实时迁移的图示。

[0007] 图2示出了根据实施例的在两个主机之间建立连接的协议的图示。

[0008] 图3示出了根据实施例的当连接丢失时在两个节点之间正在进行的通信的图示。

[0009] 图4示出了根据实施例的在云环境中支持SA路径高速缓存的图示。

[0010] 图5示出了根据本发明实施例的在云环境中支持SA路径高速缓存的图示。

[0011] 图6示出了根据实施例的用于在云环境中支持子网管理的方法的图示。

具体实施方式

[0012] 本发明通过示例而非限制的方式在附图的图中示出,附图中相似的标号指示相似的元件。应当指出,在本公开中对“一”或“一个”或“一些”实施例的引用不一定是对相同的实施例,并且这种引用意味着至少一个。

[0013] 下面对本发明的描述使用InfiniBand(IB)网络协议作为高性能网络协议的示例。

对于本领域技术人员将清楚的是,可以使用其它类型的高性能网络协议而不受限制。

[0014] 本文描述的是可以在云环境中支持虚拟机(VM)迁移子网经管(SA)路径高速缓存的系统和方法。

[0015] 根据实施例,可以提供能够提供高性能计算(HPC)的云计算。这种HPC即服务可以在计算云内被提供,并且允许及适于使用高性能互连解决方案的虚拟HPC(vHPC)集群。

[0016] 根据实施例,每个IB子网可以利用子网管理器(SM)。每个SM可以负责网络初始化、拓扑发现、路径计算以及主机信道适配器(HCA)和交换机上的IB端口的配置。在大型子网中,节点之间的可用路径可以呈多项式地增长,并且当收到许多并发的对于路径解析的请求时,SM会成为潜在的瓶颈。当具有IB互连的虚拟机实时迁移时,这种可扩展性挑战在动态虚拟化云环境中进一步被放大。

[0017] 为了支持高效的虚拟化,在维持高带宽和低延迟的同时,IB主机信道适配器(HCA)可以支持单根I/O虚拟化(SR-IOV)。每个IB连接的节点具有三个不同的地址。当实时迁移发生时,无论由分离(detach)直通的接口造成的停机时间如何,IB地址中的一个或多个会改变。与在迁移中的VM进行通信的其它节点失去连接,并且尝试通过向SM发送子网经管(SA)路径记录查询来找出要重新连接的新地址。所导致的在底层网络中朝向SM的通信可能是大量的。在大型网络中,由VM迁移引起的这种朝向SM的消息泛洪(message flooding)会增加总体网络延迟,因为SM上的负载增加。

[0018] 于是,根据实施例,期望通过减少SM接收的由于VM迁移引起的SA请求量来减小SM上的负载。方法和系统可以通过实现VM可以在迁移后保留其相同地址的系统来达成此目的。此外,在建立两个节点之间的初始连接之后,可以使用SA路径高速缓存机制来大大减少SA查询的数量。

[0019] 根据实施例,InfiniBand一般使用三种不同类型的地址。首先是16位的本地标识符(LID)。至少一个LID由SM指派给每个HCA端口和每个交换机。LID可以被用于在子网内路由流量。由于LID为16位长,因此可以进行65536个唯一的地址组合,其中仅有49151(0x0001-0xBFFF)个可以被用作单播地址。因此,可用的单播地址的数量定义了IB子网的最大尺寸。

[0020] 第二种类型的地址是一般由制造商指派给每个设备(例如,HCA和交换机)和每个HCA端口的64位全局唯一标识符(GUID)。SM可以向HCA端口指派附加的子网唯一GUID,该附加的子网唯一GUID在启用SR-IOV VF时可以是有用的。

[0021] 第三种类型的地址是128位全局标识符(GID)。GID一般是有效的IPv6单播地址,并且至少一个被指派给每个HCA端口和每个交换机。GID是通过组合由架构经管者指派的全局唯一的64位前缀和每个HCA端口的GUID地址而形成的。

[0022] 下面对本发明的描述使用Infiniband网络作为高性能网络的示例。对于本领域技术人员来说将清楚的是,可以使用其它类型的高性能网络而不受限制。而且,下面对本发明的描述使用KVM虚拟化模型作为虚拟化模型的示例。对于本领域技术人员来说将清楚的是,可以使用其它类型的虚拟化模型(例如,Xen)而不受限制。

[0023] 下面对本发明的描述另外还利用OpenStack、OpenSM和RDS Linux内核模块。OpenStack是云计算软件平台,包括一组相互关联的项目,这些项目通过数据中心控制处理、存储和联网资源的池。OpenSM是可以在OpenIB之上运行的、兼容InfiniBand的子网管理

器和经管。RDS(可靠数据报套接字)是用于输送数据报的高性能、低延迟、可靠的无连接协议。对于本领域技术人员来说将清楚的是,可以利用其它类似的平台而不受限制。

[0024] 根据本发明的实施例,虚拟化可以有益于云计算中高效的资源利用和弹性的资源分配。实时迁移使得有可能通过以应用透明的方式在物理服务器之间移动虚拟机(VM)来优化资源使用。因此,利用单根I/O虚拟化(SR-IOV)方法的虚拟化可以通过实时迁移来使得能够实现整合、资源的按需供应以及弹性。

[0025] IB体系结构是串行的点对点全双工技术。IB网络可以被称为子网,其中子网由使用交换机和点对点链路互连的一组主机组成。IB子网可以包括至少一个子网管理器(SM),该子网管理器可以负责初始化和唤醒(bring up)网络,包括对子网中所有交换机、路由器和主机信道适配器(HCA)的配置。

[0026] IB支持丰富的传输服务的集合,以便提供远程直接存储器访问(RDMA)和传统的发送/接收语义这二者。IB HCA使用队列对(QP)进行通信,而与所使用的运输服务无关。QP在通信设立期间创建,并且可以具有一组初始属性,诸如QP号、HCA端口、目的地LID、队列尺寸和所供应的传输服务。HCA可以处理许多QP,每个QP由一对队列(诸如发送队列(SQ)和接收队列(RQ))组成,并且在参与通信的每个端节点处有一个这样的对存在。发送队列保持要传送到远程节点的工作请求,而接收队列保持关于如何处理从远程节点接收的数据的信息。除了QP以外,每个HCA还具有与一组发送和接收队列关联的一个或多个完成队列(CQ)。CQ保持对于发布到发送和接收队列的工作请求的完成通知。即使通信的复杂性相对于用户是隐藏的,QP状态信息也保存在HCA中。

[0027] 网络I/O虚拟化:

[0028] 根据实施例,可以使用I/O虚拟化(IOV)来共享I/O资源并且提供对来自各个虚拟机的资源的受保护的访问。IOV可以将可以暴露给虚拟机的逻辑设备与其物理实现解耦。一种这样类型的IOV是直接设备指派。

[0029] 根据实施例,直接设备指派可以涉及I/O设备到VM的耦合而在VM之间没有设备共享。直接指派(或者说设备直通)可以以最小的开销提供接近本机的(near to native)性能。物理设备直接附连到虚拟机从而绕过管理程序(hypervisor),并且访客OS可以使用未经修改的驱动器。消极面是有限的扩展性,因为没有共享;一个物理网卡与一个VM耦合。

[0030] 根据实施例,单根IOV(SR-IOV)可以允许物理设备通过硬件虚拟化表现为同一设备的多个独立的轻量级实例。这些实例可以被指派给VM作为直通设备,并作为虚拟功能(VF)被访问。SR-IOV缓解了纯直接指派的可扩展性问题。

[0031] 不幸的是,如果所实现的系统为了数据中心优化而使用透明的实时迁移(VM迁移),则诸如SR-IOV的直接设备指派技术会给云提供商带来问题。实时迁移的实质是虚拟机的存储器内容被复制到远程管理程序。然后虚拟机在源管理程序处暂停,并且虚拟机的操作在其被复制到的目的地处恢复。当底层系统利用直接设备指派(诸如SR-IOV)时,网络接口的完整内部状态不能被复制,因为它被绑定到硬件。指派给虚拟机的SR-IOV VF被分离,实时迁移将运行,并且新的VF将在目的地处被附连。

[0032] 在使用IB VF的VM被实时迁移的情景中,由于VM的所有三种地址的改变,会引入对底层网络架构和SM的明显影响。因为VM被移动到具有不同LID的不同物理主机,所以LID改变。由SM指派给源VF的虚拟GUID(vGUID)也可以改变,因为在目的地处将附连不同的VF。随

后，由于vGUID被用于形成GID，因此GID也将改变。因此，迁移后的VM会突然与新的一组地址关联，并且迁移后的VM的通信对等体可以开始向SM发送并发的SA路径记录查询突发(burst)从而尝试重新建立与迁移后的VM的丢失的连接。这些查询会对SM造成附加开销，并且其副作用是追加的停机时间。如果迁移后的节点与网络中的许多其它节点通信，则SM会成为瓶颈并妨碍整体网络性能。

[0033] 根据实施例，本文所述的方法和系统可以减少和/或消除与使用向云提供商呈现的直接设备指派技术(诸如SR-IOV)的虚拟机的实时迁移关联的问题。这些方法和系统可以克服在使用IB VF的VM正在实时迁移的情景中存在的问题。

[0034] 虚拟机(VM)实时迁移

[0035] 图1示出了根据实施例的在云环境中支持VM实时迁移的图示。如图1所示，InfiniBand(IB)子网100可以包括支持不同管理程序111-113的多个主机节点A-C(101-103)。

[0036] 此外，每个管理程序111-113允许各个虚拟机(VM)在其上运行。例如，主机节点A 101上的管理程序111可以支持VM A 104，并且主机节点B上的管理程序112可以支持VM B 105。VM A和VM B在其上运行的这些节点可以在进行通信。

[0037] 此外，主机节点A-C(101-103)中的每一个可以与一个或多个主机信道适配器(HCA)117-119关联。如图1所示，主机节点A101上的HCA 117可以利用可由VM A 104使用的队列对(QP)，诸如QP a 108，而主机节点B 102上的HCA 118可以利用可由VM B 105使用的QP b 107。

[0038] 根据本发明的实施例，可以使用输入/输出虚拟化(IOV)来向VM提供I/O资源，并且提供对来自多个VM的共享的I/O资源的受保护的访问。IOV可以将暴露于VM的逻辑设备与其物理实现解耦。例如，单根I/O虚拟化(SR-IOV)是用于在IB网络上的虚拟化中实现高性能的I/O虚拟化方法。

[0039] 而且，IB子网100可以包括子网管理器110，该子网管理器可以负责网络初始化、HCA和交换机上的IB端口的配置、拓扑发现以及路径计算。

[0040] 如图1所示，VM B 105可以从管理程序112迁移到管理程序113(例如，在与管理程序111上的VM A 105通信的同时)。

[0041] 在迁移之后，新VM B'106会突然暴露于在目的地主机节点C 103处的新的一组地址。此外，对等体VM(例如，VM A 104)可以开始向SM 110发送子网经管(SA)路径记录查询，同时尝试重新建立丢失的连接(一旦VM B'在新的主机节点上运行，VM B'就也可以向SM发送SA路径请求)。这是由于以下事实：一般而言，一旦VM迁移，比如诸如VM B从主机节点B迁移到主机节点C，VM的地址(LID、GUID、GID)就相应地改变，因为在使用SR-IOV时它们一般是绑定到硬件的。对子网管理器的这些SA路径查询会造成显著的停机时间，以及对InfiniBand SM 110的附加开销。如果在大型数据中心内在相当短的时间内发生许多迁移，或者如果迁移后的节点与网络中的许多其它节点在进行通信，则SM 110会成为瓶颈，因为它可能不能及时响应。

[0042] 根据本发明的实施例，当VM B 104迁移并且IB地址信息改变时，系统可以减少由参与的主机节点A-C(101-103)生成的SA查询量。

[0043] 如图1所示，系统可以首先从管理程序112分离VM B 104，例如通过从VM B 104分

离虚拟功能(VF)115。然后，系统可以向目的地主机节点C 103提供与VM B 104关联的地址信息120，例如通过将地址指派给在主机节点C 103上的管理程序113上的下一个可用的虚拟功能(即VF'116)。最后，在VM B 104迁移到管理程序113成为VM B'106之后，系统可以将VM B'106暴露给地址信息120以便重新建立与对等体VM的通信(例如，经由QP b'109)。

[0044] 因此，在迁移到目的地主机节点C 103之后，新VM B'106可以暴露于原始的一组地址，并且不需要对等体VM A 104向SM 110发送SA路径记录查询。

[0045] 根据实施例，系统可以支持具有附连的IB SR-IOV VF的VM的VM实时迁移。远程直接存储器访问(RDMA)可以经诸如可靠数据报套接字(RDS)协议的协议被用来在VM的迁移之后重新建立通信。

[0046] 根据实施例，系统可以利用OpenStack、OpenSM和RDS Linux内核模块。此外，可以使用可称为LIDtracker的程序来跟踪与每个VM关联的IB地址，并且可以编排迁移过程。

[0047] 在实施例中，该程序可以启用OpenSM的选项honor_guid2lid_file。然后，由OpenSM生成的文件guid2lid可以由该程序解析，并以一定次序(诸如升序)按GUID排序。LID从一开始被指派给GUID。指派给GUID的每个LID可以被称为用于物理主机的基本LID。

[0048] 在实施例中，一旦指派了基本LID，就可以为运行的VM扫描每个启用IB的OpenStack计算节点。可以从49151(最上面的单播LID)开始以降序给被发现在运行的每个VM指派LID。指派给VM的这些LID可以被称为浮动LID。

[0049] 在实施例中，浮动LID可以替代VM正在其上运行的OpenStack计算节点中的基本LID。管理程序与VM共享LID。在某些实施例中，每个管理程序可以运行一个VM，并且VM可以迁移到没有其它VM当前正在运行的管理程序。在其它实施例中，多个VM可以在一个管理程序上运行，并且VM可以迁移到另一个管理程序，而不管其它VM当前是否在目的地管理程序上运行。

[0050] 在实施例中，当从诸如OpenStack API的API命令(order)对于VM_x的迁移时，可以从VM分离SR-IOV VF。当完成了设备的去除并且正在进行迁移时，OpenStack可以通知程序VM_x正在从一个管理程序(诸如$Hypervisor_y$)移动到目的地管理程序(诸如$Hypervisor_z$)。然后，程序可以将$Hypervisor_y$的LID改回其基本LID，并且$Hypervisor_z$可以获得与VM_x关联的浮动LID。该程序还可以将与VM_x关联的vGUID指派给在目的地管理程序$Hypervisor_z$处的下一个可用的SR-IOV VF。在迁移期间，该VM没有网络连接性。

[0051] 根据实施例，可以经由重新启动来应用改变。然后，当迁移完成时，OpenStack可以在$Hypervisor_z$上将下一个可用的SR-IOV VF添加到VM_x，并且VM可以得回其网络连接性。VM可以暴露给它在迁移之前所具有的相同的IB地址(LID、vGUID和GID)。从VM的角度来看，看起来就像是IB适配器被分离了迁移所需的时间并且同一个IB适配器被重新附连，因为地址没有改变。

[0052] 子网经管(SA)路径高速缓存

[0053] 根据实施例，在两个节点之间建立初始连接之后，端节点处的本地SA路径高速缓存机制可以减少或消除SA查询。高速缓存方案可以是通用的并且，当高速缓存方案被启用时，在发生或不发生实时迁移的情况下都可以缓解SM上的负载。

[0054] 图2示出了根据实施例的在两个主机之间建立连接的协议的图示。更具体地，图2示出了使用诸如RDS的协议在两个主机之间建立连接。

[0055] 根据实施例,在建立连接之前,可以在所有通信对等体中设立经IB的IP(IPoIB)。诸如RDS的协议可以使用具体IB端口的IPoIB地址来确定端口的GID地址。在解析GID地址之后,该协议可以具有足够的信息来执行路径记录查找并建立IB通信。

[0056] 如图2所示,在InfiniBand子网200内,子网管理器210可以提供节点C 220和节点D 225之间(更具体地,节点C上的客户端侧应用和节点D上的服务器侧应用之间)的路径通信。在图2中,上层应用的客户端侧在节点C中运行,并且该应用的服务器侧在节点D中运行。应用的客户端侧可以创建诸如RDS套接字的套接字,并尝试与应用的服务器侧进行通信(步骤1)。诸如RDS的协议可以从节点C向SM发送SA路径记录请求(步骤2)。子网管理器可以向协议提供响应(步骤3)。这个响应可以包括用于客户端侧应用的目标的地址信息。在从子网管理器接收到响应之后,该协议可以尝试通过发送连接请求来发起与节点D的连接(步骤4)。如果连接成功,则协议可以例如经由两侧中的RDMA_CM_EVENT_ESTABLISHED事件来建立通信信道(步骤5)。在这个时候,上层应用可以进行通信(步骤6)。

[0057] 在初始连接时某些事情出错的情况下,客户端侧(节点C)的协议可以尝试使用随机退避机制重试建立连接。服务器还没有意识到客户端要进行通信的意图。如果在连接建立之后出现任何问题,则两个RDS侧(从应用的角度来看是客户端和服务器)将主动参与与对等体进行重新连接。连接过程中的随机退避机制对于当双方都参与连接时避免出现竞争状况是有用的。

[0058] 图3示出了根据实施例的在连接丢失时两个节点之间正在进行的通信的图示。

[0059] 在图3中,在InfiniBand子网200内,子网管理器210可以提供节点C 220和节点D 225之间的路径通信,并且当连接丢失时(步骤2)节点C和节点D之间存在正在进行的通信(步骤1)。连接的丢失可以与例如在节点上运行的应用之一的实时迁移关联。两个协议端都可以确定连接断开(down)并且在尝试重新连接之前等待某个随机时间(即,退避时间)(步骤3)。如图3所示,在尝试重新连接之前每一侧等待的时间可以相同或不同。节点可以尝试通过向SM发送SA路径记录请求来重新连接(步骤4)。在接收到SA路径记录响应(步骤5)之后,可以发送连接请求(步骤6)。

[0060] 在图3所示的情况下,在步骤3中由两个节点选择的退避时间几乎相同。因此,即使节点D比节点C稍快地获得SA路径记录响应,并且尝试在步骤6中首先发起连接,该连接请求也不会在节点C自己发送连接请求之前到达节点C。在这种情况下,两个协议端都有未完成的连接请求。然后,当节点接收到来自它们的对等体的连接请求时,节点将拒绝该连接请求(步骤7)。在步骤8中,两个节点在它们重试重新连接之前再次选择了随机退避时间。这次,由节点D选择的随机退避时间明显长于由节点C选择的随机退避时间。因此,节点C获得优先并重复连接建立过程;发送SA路径记录请求(步骤8)、接收来自子网管理器的响应(步骤10)、向节点D发送连接请求(步骤11),并且该连接请求在节点D尝试自己发起与节点C的连接之前到达节点D。在图3中绘出的情景中,节点D接受传入的连接(步骤12)。然后在步骤13和步骤14中可以为上层应用恢复通信。

[0061] 从图3外推,变得清楚的是,在VM迁移(断开通信)的情况下,子网管理器会被SA路径请求轰炸。在具有数以千计的节点的大型子网中,即使从每个节点只发送一个附加的SA查询,SM也会最终被数以千计的消息淹没。当在动态的基于IB的云中发生实时迁移时,可能发送过多的SA查询。随着网络中节点数量的增加,SA查询的数量会呈多项式地增加。所公开

的方法和系统提供了高速缓存机制，该机制可以减少由子网中的节点发送到子网管理器的SA查询的数量。

[0062] 图4示出了根据本发明实施例的在云环境中支持SA路径高速缓存的图示。如图4所示，InfiniBand(IB)子网400可以包括子网管理器(SM)410和多个主机节点A-B(401-402)。

[0063] 当源主机节点A 401(例如，VM A 411)首次尝试与目的地主机节点B 402(例如，VM B 412)通信时，源主机节点A 401可以向SM 410发送SA路径记录请求。然后，源主机节点可以使用本地高速缓存421来存储路径信息(例如，路径记录422)。

[0064] 此外，当源主机节点A 401尝试重新连接到相同的目的地主机节点B 402时，源主机节点A 401可以在本地高速缓存421中的高速缓存表中查找目的地主机节点的地址而不是向子网管理器发送请求。

[0065] 如果找到路径信息，则源主机节点A 401可以使用如路径记录422所指示的路径420连接到目的地主机节点B 402，而没有向SM 410发送SA查询。否则，源主机节点A 401可以向SM 410发送SA路径记录请求，以获得必要的路径信息。

[0066] 图5示出了根据本发明实施例的在云环境中支持SA路径高速缓存的图示。更具体地，图5示出了在InfiniBand环境的子网内支持SA路径高速缓存的图示。

[0067] 如图5所示，InfiniBand(IB)子网500可以包括支持不同管理程序511-512的多个主机节点A-B(501-502)。此外，每个管理程序512-513允许各个虚拟机(VM)在其上运行。例如，主机节点A 101上的管理程序511可以支持VM A 504，并且主机节点B上的管理程序512可以支持VM B 505。

[0068] 此外，主机节点A-B(501-502)中的每一个可以与一个或多个主机信道适配器(HCA)517-518关联。如图5所示，主机节点A 501上的HCA 517可以利用可由VM A 504使用的队列对(QP)，诸如QP a 508，而主机节点B 502上的HCA 518可以利用可由VM B 505使用的QP b 507。

[0069] 根据实施例，每个主机节点还可以支持存储器530、540，这些存储器各自可以包含高速缓存(诸如本地高速缓存)535、545，并且每个高速缓存继而可以包括一个或多个路径记录537、547，这些路径记录可以存储在高速缓存表中。

[0070] 而且，IB子网500可以包括子网管理器510，子网管理器510可以负责网络初始化、HCA和交换机上的IB端口的配置、拓扑发现以及路径计算。

[0071] 根据实施例，当源主机节点A 501(例如，VM A 504)首次尝试与目的地主机节点B 502(例如，VM B 505)通信时，源主机节点A 501可以向SM 510发送SA路径记录请求。然后，源主机节点可以使用本地高速缓存535来存储路径信息(例如，路径记录537)。

[0072] 此外，当源主机节点A 501尝试重新连接到相同的目的地主机节点B 502时，源主机节点A 501可以在高速缓存535中查找目的地主机节点的地址，而不是向子网管理器发送请求。

[0073] 根据实施例，如果找到路径信息，则源主机节点A 501可以通过使用在路径记录537中指示的路径连接到目的地主机节点B 502，而没有向SM 510发送SA查询。否则，源主机节点A 501可以向SM 410发送SA路径记录请求，以获得必要的路径信息。

[0074] 根据实施例，在主机节点A 501向子网管理器510发送SA路径记录请求的情景中，所接收的响应可以包括高速缓存标志，该高速缓存标志可以向主机节点A 501指示使用本

地高速缓存表(在高速缓存535内)来存储与目的地主机节点B 502的给定GID地址(DGID)关联的路径特性。

[0075] 图6示出了根据实施例的用于支持云环境中的子网管理的方法的图示。示例性方法600可以在步骤601开始于在云环境内提供包括第一主机节点在内的多个主机节点,第一主机节点与至少第一管理程序和第一主机信道适配器关联。在步骤602,该方法可以继续在第一主机节点上提供第一虚拟机,第一虚拟机与多个地址关联。在步骤603,该方法继续将第一虚拟机从第一主机节点迁移到云环境内多个主机节点中提供的第二主机节点,第二主机节点与至少第二管理程序和第二主机信道适配器关联,其中多个主机节点中的每一个都包括本地高速缓存;每个本地高速缓存包括一个或多个路径记录。

[0076] 根据实施例,迁移第一虚拟机可以包括:在步骤604,从第一管理程序分离第一虚拟机,从第一管理程序分离第一虚拟机包括从第一虚拟机分离与第一虚拟机关联的第一虚拟功能。在步骤605,该方法继续向第二主机节点提供与第一虚拟机关联的该多个地址。在步骤606,该方法可以将该多个地址指派给第二虚拟功能,第二虚拟功能与第二管理程序关联。在步骤607,该方法可以将第一虚拟机从第一主机节点迁移到第二主机节点上的第二虚拟机。在步骤608,该方法可以以将第二虚拟机暴露于与第一虚拟机关联的该多个地址来结束。

[0077] 根据实施例,SA路径记录高速缓存机制可以以诸如RDS协议的协议实现,并且高速缓存表可以存储在每个节点的存储器中。可以使用如下面的伪代码中所示的程序:

[0078] 1:private bool SA PathCachingEnabled

[0079] 2:private list SA PathRecordCacheTable

[0080] 3:

[0081] 4:procedure RDSMODULEINITIALIZATION

[0082] 5://高速缓存表被初始化

[0083] 6:SA PathRecordCacheTable＝empty

[0084] 7:

[0085] 8://系统还不知道SA路径高速缓存是否被SM启用,

[0086] 9://所以我们假设还没有。

[0087] 10:SA PathCachingEnabled＝False

[0088] 11:end procedure

[0089] 12:

[0090] 13:procedure(RE-)CONNECTIONESTABLISHMENT(DGID)

[0091] 14:struct PathRecord DST Path＝NULL

[0092] 15:

[0093] 16://只有SA路径高速缓存被SM启用

[0094] 17://才使用高速缓存

[0095] 18:if SA PathCachingEnabled then

[0096] 19:if DGIDin SA PathRecordCacheTable

[0097] 20:DGIDs then

[0098] 21:DST Path＝Cached PathRecord

[0099] 22:end if

[0100] 23:end if

[0101] 24:

[0102] 25://如果DST Path在这个时候为NULL,

[0103] 26://或者高速缓存被SM禁用,

[0104] 27://或者用于具有给定DGID的主机的路径特性

[0105] 28://从未被检索过。在任何情况下,

[0106] 29://PathRecord查询可以被发送到SM。

[0107] 30:if DST Path＝＝NULL then

[0108] 31:SendAnewSA PathRecordQueryToTheSM

[0109] 32:WaitForTheReply

[0110] 33:DST Path＝PathRecordResponse

[0111] 34:

[0112] 35://如果SM启用了高速缓存,

[0113] 36://则回复将PathRecord中的保留字段设置为1。

[0114] 37://如果SM没有启用高速缓存,则保留字段为0

[0115] 38:if DST Path!Reserved Field!＝0then

[0116] 39:SA PathCachingEnabled＝True

[0117] 40:

[0118] 41://在高速缓存表中插入DST路径

[0119] 42:SA PathRecordCacheTable.append(

[0120] 43:DSTPath)

[0121] 44:end if

[0122] 45:end if

[0123] 46:连接到(DST路径)

[0124] 47:end procedure

[0125] 根据实施例,当源主机(SHost)首次尝试与目的地主机(DHost)通信时,SHost可以向子网管理器发送SA路径记录请求。如果响应具有升起的(raised)高速缓存标志,则SHost可以使用本地高速缓存表来存储与DHost的给定GID地址(DGID)关联的路径特性。而且,SHost现在意识到子网管理器支持高速缓存,所以在下次SHost尝试与任何DHost连接或重新连接时,它将首先在高速缓存表中查找。如果找到用于给定DHost的路径信息,则可以防止SHost向子网管理器发送SA查询,并且SHost可以替代性地尝试使用其高速缓存表内的信息与DHost连接。

[0126] 再次参考图3,参考中断的连接(步骤2),在启用上述高速缓存机制的系统中,不需要向子网管理器发送SA查询。在图3所描述的情况下,消除了步骤4、5、9和10,因此,连接重建更快并且子网管理器上的负载(例如,SA路径请求和响应)更低。

[0127] 本发明的许多特征可以在硬件、软件、固件或其组合中实现、利用硬件、软件、固件或其组合实现、或者在硬件、软件、固件或其组合的协助下实现。因此,本发明的特征可以利用(例如,包括一个或多个处理器的)处理系统来实现。

[0128] 本发明的特征可以在计算机程序产品中实现、利用计算机程序产品实现、或者在计算机程序产品的协助下实现，其中计算机程序产品是其上/其中存储有可用来编程处理系统以执行本文所呈现的任何特征的指令的(一个或多个)存储介质或计算机可读介质。存储介质可以包括但不限于任何类型的盘(包括软盘、光盘、DVD、CD-ROM、微驱动器、以及磁光盘)、ROM、RAM、EPROM、EEPROM、DRAM、VRAM、闪存存储器设备、磁卡或光卡、纳米系统(包括分子存储器IC)、或适于存储指令和/或数据的任何类型的介质或设备。

[0129] 在存储在任何一种机器可读介质的情况下，本发明的特征可以被结合到软件和/或固件中，以用于控制处理系统的硬件，以及用于使处理系统能够与其它机制交互，从而利用本发明的结果。这种软件或固件可以包括但不限于应用代码、设备驱动器、操作系统和执行环境/容器。

[0130] 本发明的特征也可以利用例如诸如专用集成电路(ASIC)的硬件部件在硬件中实现。实现硬件状态机以执行本文所描述的功能对相关领域的技术人员将是清楚的。

[0131] 此外，本发明可以使用一个或多个常规的通用或专用数字计算机、计算设备、机器或微处理器来方便地实现，该通用或专用数字计算机、计算设备、机器或微处理器包括一个或多个处理器、存储器和/或根据本公开的教导编程的计算机可读存储介质。如对软件领域的技术人员将清楚的，适当的软件编码可以容易地由熟练的程序员基于本公开的教导来准备。

[0132] 虽然以上已经描述了本发明的各种实施例，但是应该理解，它们已作为示例而不是限制呈现。对相关领域的技术人员将清楚的是，在不背离本发明的精神和范围的情况下，其中可以做出各种形式和细节上的变化。

[0133] 本发明已经借助说明具体功能及其关系的执行的功能构建块进行了描述。这些功能构建块的边界在本文中通常是为了方便描述而任意定义的。可以定义可替代的边界，只要具体的功能及其关系被适当地执行。任何这种可替代的边界因此在本发明的范围和精神之内。

[0134] 本发明的以上描述是为了说明和描述的目的提供。它不是旨在是穷尽的或者要把本发明限定到所公开的精确形式。本发明的广度和范围不应该由任何上述示例性实施例来限制。许多修改和变化对本领域技术人员来说将是清楚的。修改和变化包括所公开特征的任何相关组合。实施例的选择与描述是为了最好地解释本发明的原理及其实践应用，从而使本领域其他技术人员能够理解本发明用于各种实施例并且可以进行适于预期特定用途的各种修改。本发明的范围要由以下权利要求及其等价物来定义。

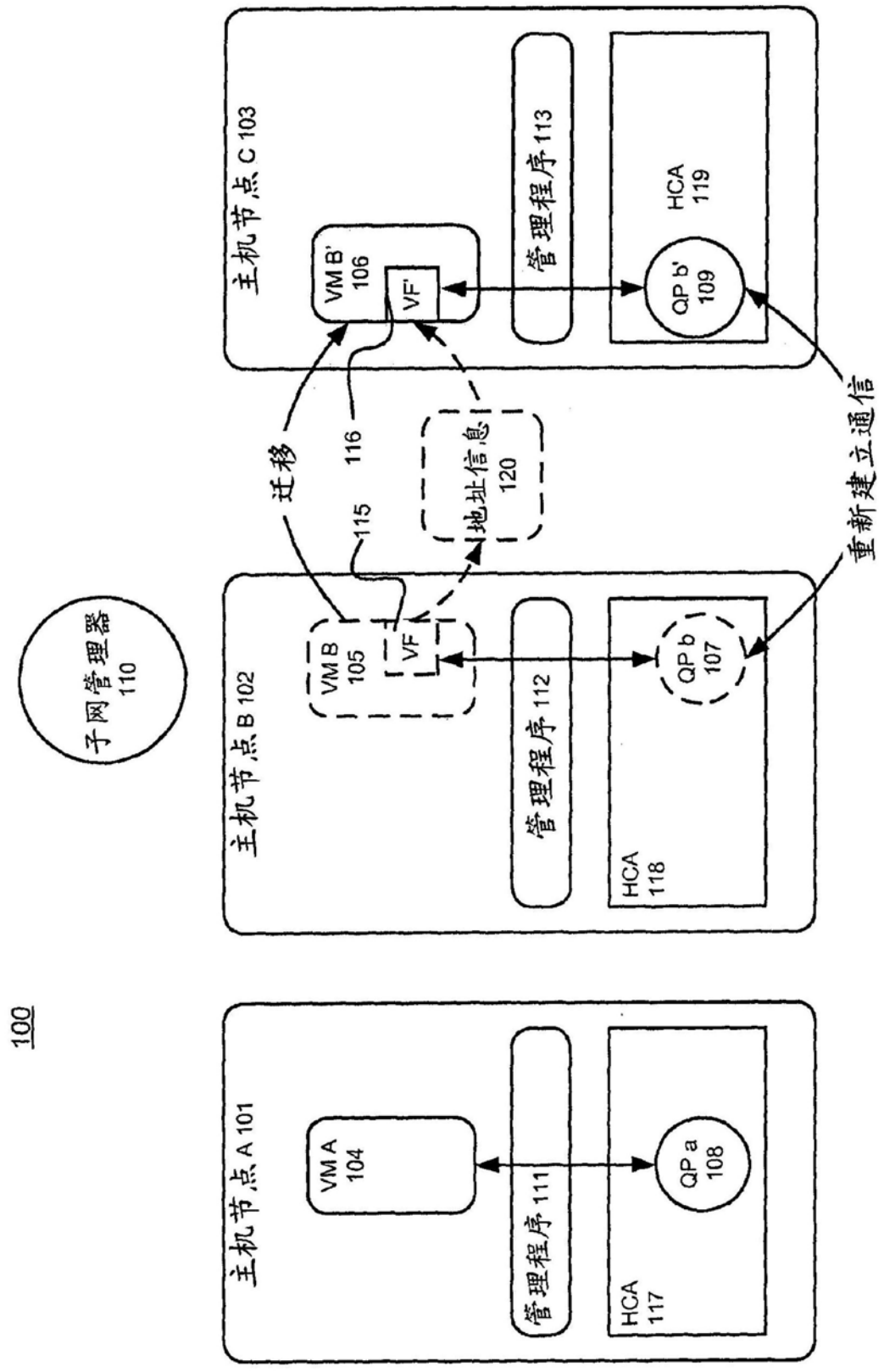

100
子网管理器 110
主机节点A 101
VM A 104
管理程序 111
HCA 117
QP a 108
主机节点B 102
VM B 105
VF
管理程序 112
HCA 118
QP b 107
主机节点C 103
VM B' 106
VF'
管理程序 113
HCA 119
QP b' 109
迁移
115
116
地址信息 120
重新建立通信

图1

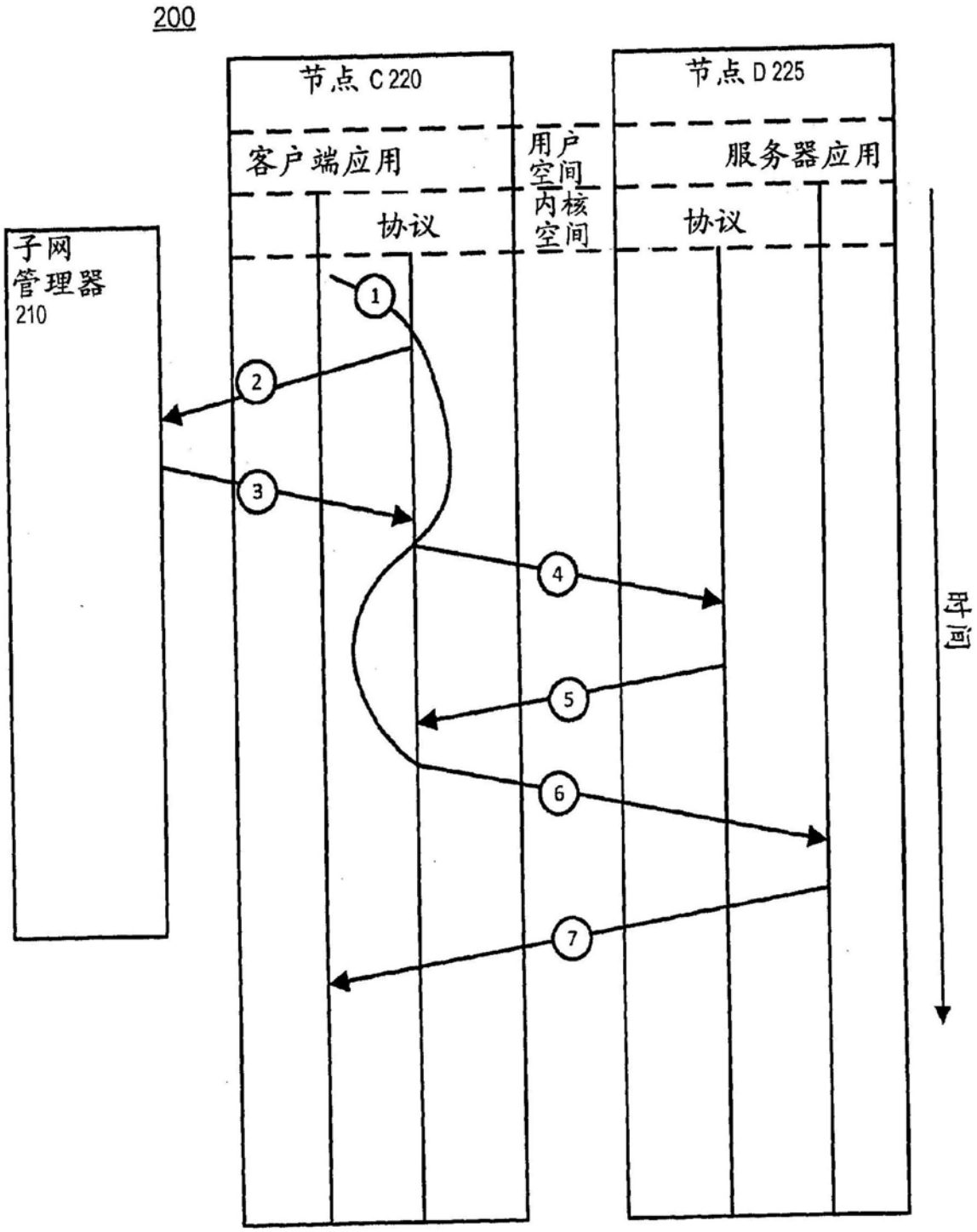

200
节点 C 220
节点 D 225
用户空间
内核空间
客户端应用
协议
服务器应用
协议
子网管理器
210
1
2
3
4
5
6
7
时间

图2

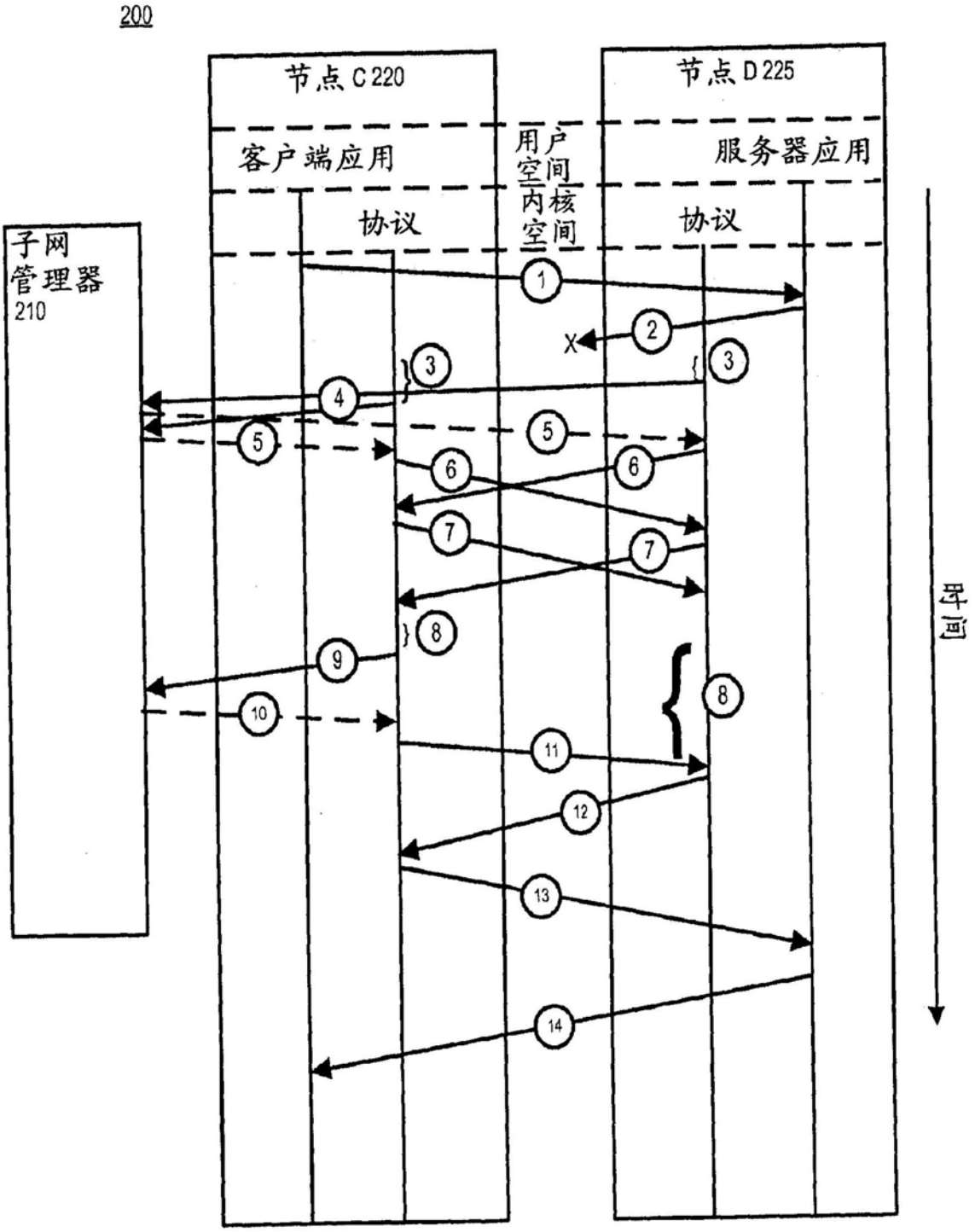

200
节点C 220
节点D 225
用户空间
内核空间
客户端应用
服务器应用
协议
协议
子网管理器
210
时间
1
2
X
3
3
4
5
5
6
6
7
7
8
8
9
10
11
12
13
14

图3

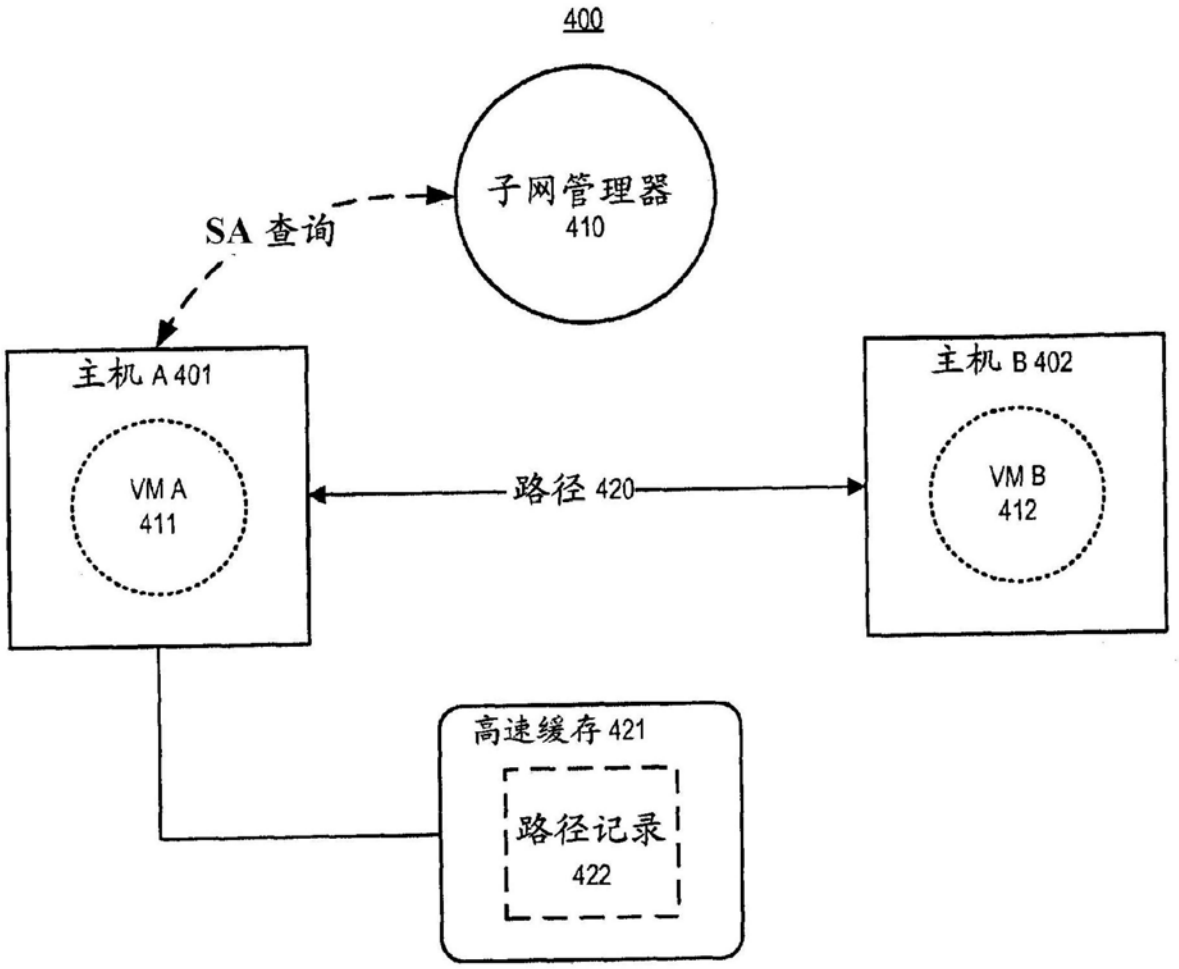

400
子网管理器
410
SA 查询
主机 A 401
VM A
411
路径 420
主机 B 402
VM B
412
高速缓存 421
路径记录
422

图4

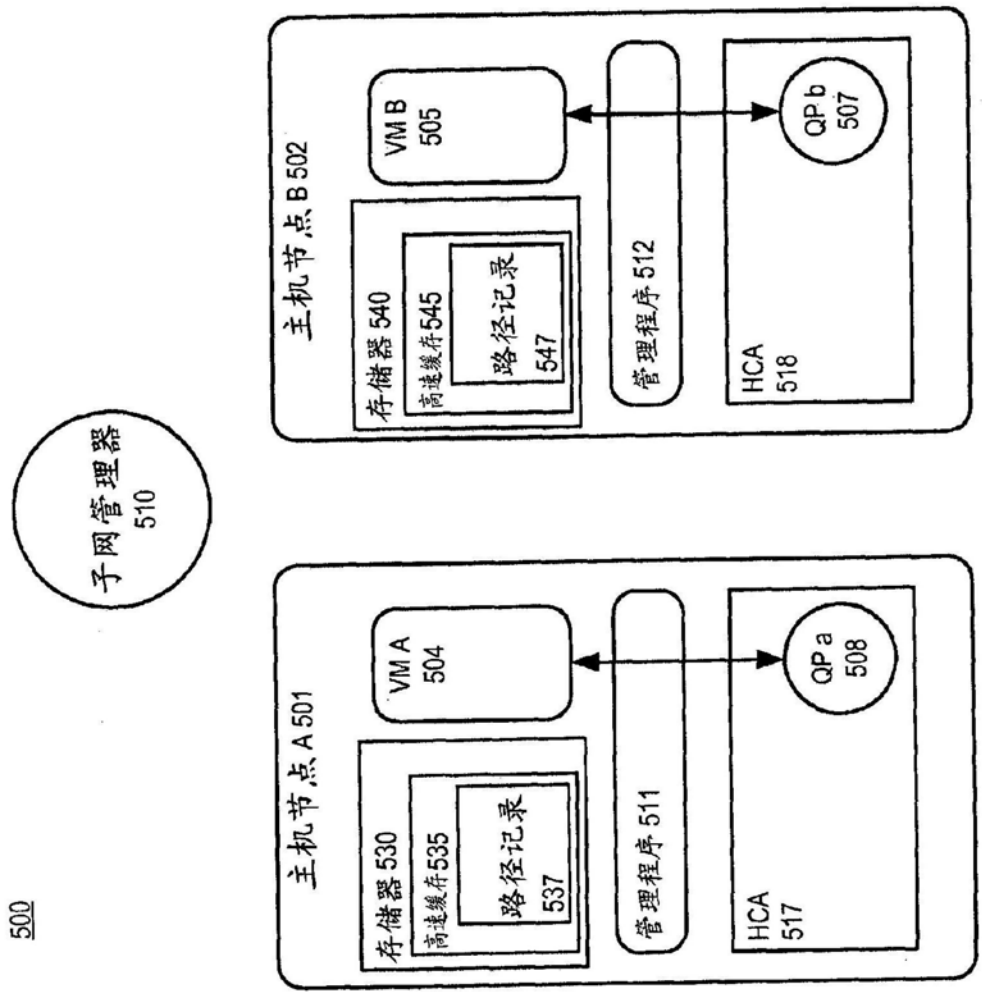

500
子网管理器
510
主机节点A 501
存储器 530
高速缓存535
路径记录
537
VM A
504
管理程序 511
HCA
517
QP a
508
主机节点B 502
存储器 540
高速缓存545
路径记录
547
VM B
505
管理程序 512
HCA
518
QP b
507

图5

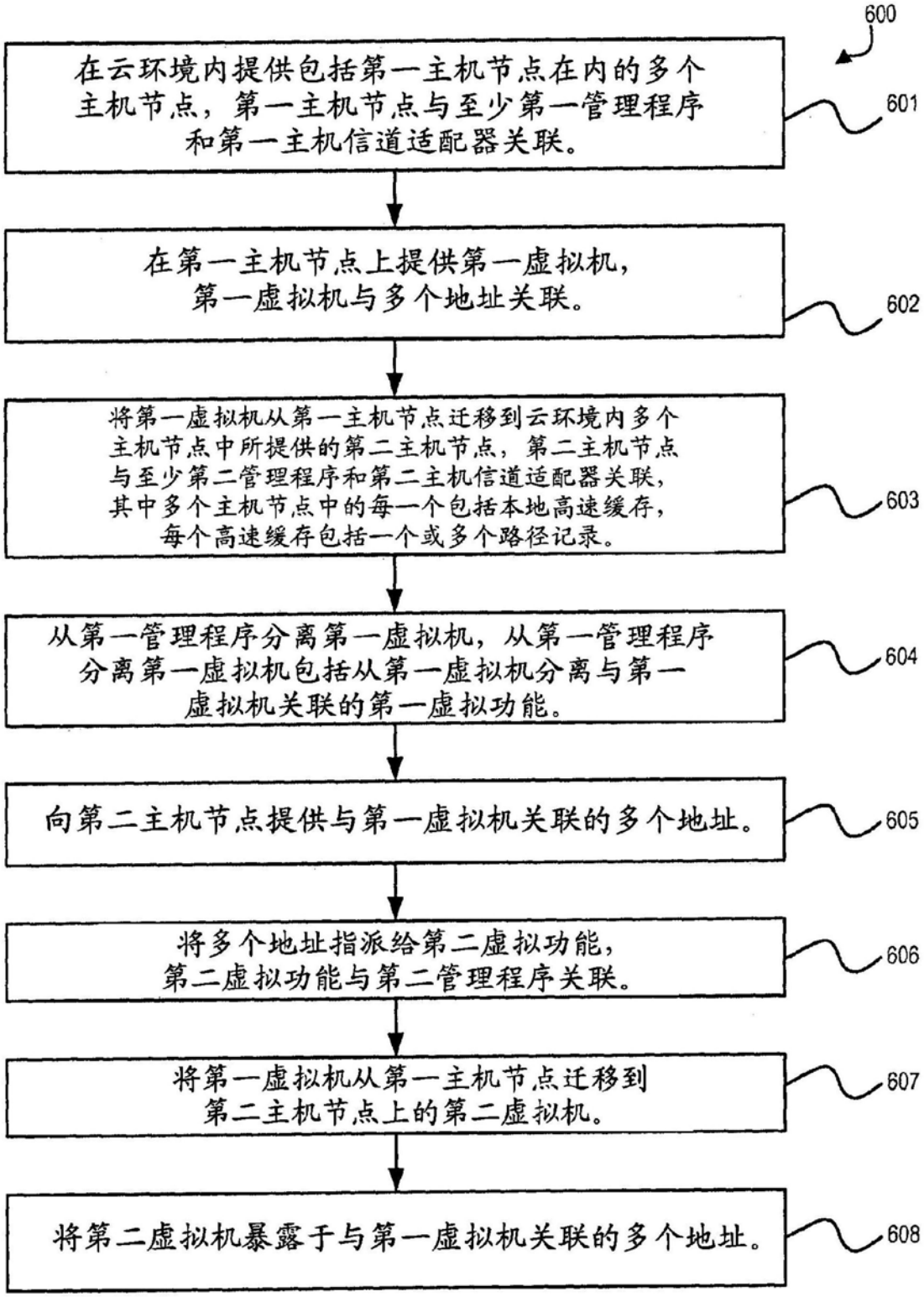

600
在云环境内提供包括第一主机节点在内的多个主机节点，第一主机节点与至少第一管理程序和第一主机信道适配器关联。
601
在第一主机节点上提供第一虚拟机，第一虚拟机与多个地址关联。
602
将第一虚拟机从第一主机节点迁移到云环境内多个主机节点中所提供的第二主机节点，第二主机节点与至少第二管理程序和第二主机信道适配器关联，其中多个主机节点中的每一个包括本地高速缓存，每个高速缓存包括一个或多个路径记录。
603
从第一管理程序分离第一虚拟机，从第一管理程序分离第一虚拟机包括从第一虚拟机分离与第一虚拟机关联的第一虚拟功能。
604
向第二主机节点提供与第一虚拟机关联的多个地址。
605
将多个地址指派给第二虚拟功能，第二虚拟功能与第二管理程序关联。
606
将第一虚拟机从第一主机节点迁移到第二主机节点上的第二虚拟机。
607
将第二虚拟机暴露于与第一虚拟机关联的多个地址。
608

图6