



(12)发明专利

(10)授权公告号 CN 106063191 B

(45)授权公告日 2019.09.17

(21)申请号 201380081578.8

(22)申请日 2013.11.12

(65)同一申请的已公布的文献号
申请公布号 CN 106063191 A

(43)申请公布日 2016.10.26

(85)PCT国际申请进入国家阶段日
2016.06.13

(86)PCT国际申请的申请数据
PCT/CN2013/086982 2013.11.12

(87)PCT国际申请的公布数据
W02015/070381 EN 2015.05.21

(73)专利权人 皮沃塔尔软件公司
地址 美国加利福尼亚州

(72)发明人 李熊 高小明 李焯 韩冷 朱磊

(74)专利代理机构 中原信达知识产权代理有限
责任公司 11219

代理人 韩峰 孙志湧

(51)Int.Cl.
H04L 12/24(2006.01)

(56)对比文件
US 2006239271 A1,2006.10.26,
US 2006168080 A1,2006.07.27,
CN 103069402 A,2013.04.24,
US 2006230189 A1,2006.10.12,
CN 101014002 A,2007.08.08,

审查员 胡锐先

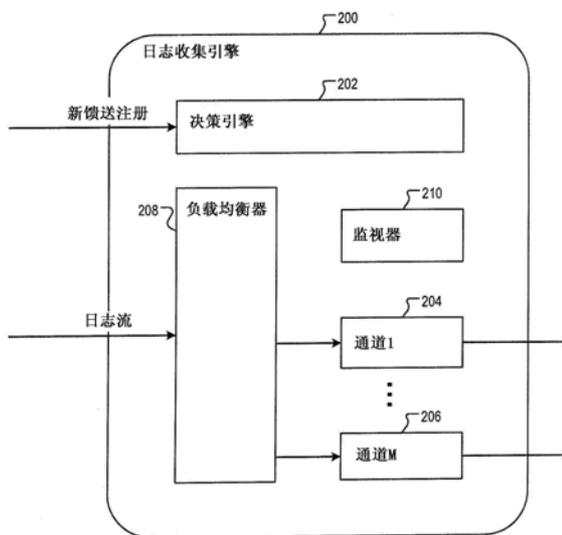
权利要求书4页 说明书6页 附图4页

(54)发明名称

流事件数据收集

(57)摘要

用于分布式数据管理的方法、系统以及设备,包括在计算机存储介质上编码的计算机程序。所述方法中的一个包括接收流事件数据的多个馈送并且将来自所述馈送中的每一个的馈送数据路由到多个通道中的相应通道,包括将来自第一馈送的馈送数据路由到第一通道,所述通道中的每一个被配置成存储馈送数据直到所述馈送数据被数据汇点消耗为止。所述第一通道的负载度量被确定为超过阈值。作为响应,为所述第一馈送分配第二通道,并且馈送数据被从所述第一馈送重定向到所述第二通道而不是所述第一通道。



1. 一种计算机实现的方法,所述方法包括:

在包括多个计算机的分布式计算系统中接收具有多个租户的多租户应用的流事件数据的多个馈送,其中,所述多个馈送中的每个馈送由所述多租户应用的所述多个租户中的相应租户生成;

将来自所述多租户应用的租户中的每一个的馈送数据路由到多个通道中的不同的相应通道,其中,每个通道包括在所述多个计算机中的一个上执行的软件进程,其中,所述多个通道中的每一个通道被配置成接收并存储馈送数据直到该馈送数据被数据汇点消耗为止;

计算在所述多个计算机中的第一计算机上执行的第一通道的负载度量,其中,所述第一通道被分配以接收由所述多租户应用的所述多个租户中的第一租户生成的第一馈送数据;

确定所述第一通道的所述负载度量超过阈值;

响应于确定所述负载度量超过所述阈值,为所述第一馈送数据分配第二通道;以及将来自所述第一租户的第一馈送数据重定向到所述第二通道而不是所述第一通道。

2. 根据权利要求1所述的方法,还包括:

从第一租户接收用于注册所述第一馈送请求;

为所述第一租户分配所述第一通道,使得所述第一通道仅接收来自所述第一租户的第一馈送数据;

为所述第一租户分配网络化连接,所述网络化连接被配置成接收来自所述第一租户的第一馈送数据;以及

使所述网络化连接与所述第一通道相关联,使得所述分布式计算系统将来自所述第一租户的所述第一馈送数据从所述网络化连接路由到所述第一通道。

3. 根据权利要求2所述的方法,其中,将由所述多租户应用的所述多个租户中的所述第一租户生成的所述第一馈送数据重定向到所述第二通道包括:

使所述网络化连接与所述第二通道相关联。

4. 根据权利要求3所述的方法,还包括,在使所述网络化连接与所述第二通道相关联之后,解除分配所述第一通道,使得由所述第一通道使用的所述分布式计算系统的一个或多个计算资源变得可用于一个或多个其它通道。

5. 根据权利要求1所述的方法,其中,

所述第一通道的所述负载度量指定由所述第一通道使用的存储器的量或网络业务的量。

6. 根据权利要求1所述的方法,还包括:

响应于确定所述负载度量超过所述阈值,基于所述负载度量超过所述阈值的量来确定有多少新通道是对于所述第一租户足够数目的新通道;

为所述第一租户分配所述足够数目的新通道;以及

将来自所述第一租户的第一馈送数据重定向到所述新通道而不是所述第一通道。

7. 根据权利要求1所述的方法,还包括:

使来自所述通道中的每一个的相应输出沉入到所述分布式计算系统的分布式文件系统。

8. 根据权利要求1所述的方法,还包括:

使来自所述通道中的每一个的相应输出沉入到被配置成对所述馈送执行数据分析的数据解析引擎。

9. 根据权利要求1所述的方法,其中,所述多个通道中的每一个包括软件实例,所述软件实例被配置成将馈送数据存储在对于所述分布式计算系统中的一个或多个计算机或一个或多个计算机的集群为本地的本地数据存储器中,其中,所述本地数据存储器包括随机存取存储器 (RAM) 或大容量存储部或两者。

10. 一种分布式计算系统,所述分布式计算系统包括多个物理计算机,所述多个物理计算机被配置成执行包括以下步骤的操作:

接收具有多个租户的多租户应用的流事件数据的多个馈送,其中,所述多个馈送中的每个馈送由所述多租户应用的所述多个租户中的相应租户生成;

将来自所述多租户应用的租户中的每一个的馈送数据路由到多个通道中的不同的相应通道,其中,每个通道包括在所述多个计算机中的一个上执行的软件进程,其中,所述多个通道中的每一个通道被配置成接收并存储馈送数据直到该馈送数据被数据汇点消耗为止;

计算在所述多个计算机中的第一计算机上执行的第一通道的负载度量,其中,所述第一通道被分配以接收由所述多租户应用的所述多个租户中的第一租户生成的第一馈送数据;

确定所述第一通道的所述负载度量超过阈值;

响应于确定所述负载度量超过所述阈值,为所述第一馈送数据分配第二通道;以及将来自所述第一租户的第一馈送数据重定向到所述第二通道而不是所述第一通道。

11. 根据权利要求10所述的分布式计算系统,所述操作还包括:

从第一租户接收用于注册所述第一馈送请求;

为所述第一租户分配所述第一通道,使得所述第一通道仅接收来自所述第一租户的第一馈送数据;

为所述第一租户分配网络化连接,所述网络化连接被配置成接收来自所述第一租户的第一馈送数据;以及

使所述网络化连接与所述第一通道相关联,使得所述分布式计算系统将来自所述第一租户的所述第一馈送数据从所述网络化连接路由到所述第一通道。

12. 根据权利要求11所述的分布式计算系统,其中,将由所述多租户应用的所述多个租户中的所述第一租户生成的所述第一馈送数据重定向到所述第二通道包括:

使所述网络化连接与所述第二通道相关联。

13. 根据权利要求12所述的分布式计算系统,所述操作还包括,在使所述网络化连接与所述第二通道相关联之后,解除分配所述第一通道,使得由所述第一通道使用的所述分布式计算系统的一个或多个计算资源变得可用于一个或多个其它通道。

14. 根据权利要求10所述的分布式计算系统,其中,

所述第一通道的所述负载度量指定由所述第一通道使用的存储器的量或网络业务的量。

15. 根据权利要求10所述的分布式计算系统,所述操作还包括:

响应于确定所述负载度量超过所述阈值,基于所述负载度量超过所述阈值的量来确定有多少新通道是对于所述第一租户足够数目的新通道;

为所述第一租户分配所述足够数目的新通道;以及

将来自所述第一租户的第一馈送数据重定向到所述新通道而不是所述第一通道。

16. 根据权利要求10所述的分布式计算系统,所述操作还包括:

使来自所述通道中的每一个的相应输出沉入到所述分布式计算系统的分布式文件系统。

17. 根据权利要求10所述的分布式计算系统,所述操作还包括:

使来自所述通道中的每一个的相应输出沉入到被配置成对所述馈送执行数据分析的数据解析引擎。

18. 根据权利要求10所述的分布式计算系统,其中,所述多个通道中的每一个包括软件实例,所述软件实例被配置成将馈送数据存储在于所述分布式计算系统中的一个或多个计算机或一个或多个计算机的集群为本地的本地数据存储器中,其中,所述本地数据存储器包括随机存取存储器(RAM)或大容量存储部或两者。

19. 一种被编码有计算机程序的计算机存储介质,所述程序包括当由多个物理计算机的分布式计算系统执行时使所述分布式计算系统执行包括以下步骤的操作的指令:

接收具有多个租户的多租户应用的流事件数据的多个馈送,其中,所述多个馈送中的每个馈送由所述多租户应用的所述多个租户中的相应租户生成;

将来自所述多租户应用的租户中的每一个的馈送数据路由到多个通道中的不同的相应通道,其中,每个通道包括在所述多个计算机中的一个上执行的软件进程,其中,所述多个通道中的每一个通道被配置成接收并存储馈送数据直到所述馈送数据被数据汇点消耗为止;

计算在所述多个计算机中的第一计算机上执行的第一通道的负载度量,其中,所述第一通道被分配以接收由所述多租户应用的所述多个租户中的第一租户生成的第一馈送数据;

确定所述第一通道的所述负载度量超过阈值;

响应于确定所述负载度量超过所述阈值,为所述第一馈送数据分配第二通道;以及

将来自所述第一租户的第一馈送数据重定向到所述第二通道而不是所述第一通道。

20. 根据权利要求19所述的计算机存储介质,所述操作还包括:

从第一租户接收用于注册所述第一馈送请求;

为所述第一租户分配所述第一通道,使得所述第一通道仅接收来自所述第一租户的第一馈送数据;

为所述第一租户分配网络化连接,所述网络化连接被配置成接收来自所述第一租户的第一馈送数据;以及

使所述网络化连接与所述第一通道相关联,使得所述分布式计算系统将来自所述第一租户的所述第一馈送数据从所述网络化连接路由到所述第一通道。

21. 根据权利要求20所述的计算机存储介质,其中,将由所述多租户应用的所述多个租户中的所述第一租户生成的所述第一馈送数据重定向到所述第二通道包括:

使所述网络化连接与所述第二通道相关联。

22. 根据权利要求21所述的计算机存储介质,所述操作还包括,在使所述网络化连接与所述第二通道相关联之后,解除分配所述第一通道,使得由所述第一通道使用的所述分布式计算系统的一个或多个计算资源变得可用于一个或多个其它通道。

23. 根据权利要求19所述的计算机存储介质,其中,
所述第一通道的所述负载度量指定由所述第一通道使用的存储器的量或网络业务的量。

24. 根据权利要求19所述的计算机存储介质,所述操作还包括:
响应于确定所述负载度量超过所述阈值,基于所述负载度量超过所述阈值的量来确定有多少新通道是对于所述第一租户足够数目的新通道;

为所述第一租户分配所述足够数目的新通道;以及

将来自所述第一租户的第一馈送数据重定向到所述新通道而不是所述第一通道。

25. 根据权利要求19所述的计算机存储介质,所述操作还包括:
使来自所述通道中的每一个的相应输出沉入到所述分布式计算系统的分布式文件系统。

26. 根据权利要求19所述的计算机存储介质,所述操作还包括:
使来自所述通道中的每一个的相应输出沉入到被配置成对所述馈送执行数据分析的数据解析引擎。

27. 根据权利要求19所述的计算机存储介质,其中,所述多个通道中的每一个包括软件实例,所述软件实例被配置成将馈送数据存储在对所述分布式计算系统中的一个或多个计算机或一个或多个计算机的集群为本地的本地数据存储器中,其中,所述本地数据存储器包括随机存取存储器 (RAM) 或大容量存储部或两者。

流事件数据收集

技术领域

[0001] 本说明书涉及云计算。

背景技术

[0002] 在云计算中,能够将数据分发给由一个或多个通信网络所连接的多个计算机的系统。每个计算机能够托管一个或多个服务器,其中的每一个对数据的一部分进行处理。服务器能够并行地对数据进行处理。这样的分布式系统能够处置被部署在虚拟化环境中的面向web且数据密集型应用。例如,分布式系统能够被配置成通过互联网托管多租户计算服务。例如,软件即服务(SaaS)提供商能够在分布式系统上运行其应用的实例并且向多个租户提供访问。

发明内容

[0003] 本说明书描述了给日志收集引擎提供高可用性和负载均衡能力的分布式计算系统。该分布式计算系统能够针对在多租户环境中托管多个应用的系统执行日志收集。

[0004] 一般而言,本说明书中所描述的主题的一个创新方面能够用包括以下各项的动作的方法加以具体化:接收流事件数据的多个馈送;将来自所述馈送中的每一个的馈送数据路由到多个通道中的相应通道,包括将来自第一馈送的馈送数据路由到第一通道,所述通道中的每一个被配置成存储馈送数据直到所述馈送数据被数据汇点消耗为止;确定所述第一通道的负载度量超过阈值;响应于确定所述负载度量超过所述阈值,为所述第一馈送分配第二通道;以及将来自所述第一馈送的馈送数据重定向到所述第二通道而不是所述第一通道。这个方面的其它实施例包括各自被配置成执行所述方法的动作的对应的计算机系统、设备以及记录在一个或多个计算机存储装置上的计算机程序。为使一个或多个计算机的系统被配置成执行特定操作或动作意味着该系统已经将在操作中使该系统执行所述操作或动作的软件、固件、硬件或它们的组合安装在其上。为使一个或多个计算机程序被配置成执行特定操作或动作意味着一个或多个程序包括当由数据处理设备执行时使该设备执行所操作或动作的指令。

[0005] 上述和其它实施例能够单独或者相结合地各自可选地包括以下特征中的一个或多个。所述动作还包括:从第一馈送源接收用于注册所述第一馈送请求;为所述第一馈送分配所述第一通道,使得所述第一通道仅接收来自所述第一馈送的数据;为所述第一馈送分配网络化连接,所述网络化连接被配置成接收来自所述第一馈送源的馈送数据;以及使所述网络化连接与所述第一通道相关联,使得所述分布式计算系统将来自所述第一馈送的所述馈送数据从所述网络化连接路由到所述第一通道。将馈送数据从所述第一馈送重定向到所述第二通道包括使所述网络化连接与所述第二通道相关联。所述动作还包括,在使所述网络化连接与所述第二通道相关联之后,解除分配所述第一通道,使得由所述第一通道使用的所述分布式计算系统的一个或多个计算资源变得可用于一个或多个其它通道。所述分布式计算系统至少对被配置成为多个租户服务的第一多租户应用进行托管,并且其中,

接收用于注册所述第一馈送的请求包括从第一租户接收所述请求。所述第一通道的所述负载度量指定由所述第一通道使用的存储器的量或网络业务的量。所述动作还包括：响应于确定所述负载度量超过所述阈值，基于所述负载度量超过所述阈值的量来确定有多少新通道是对于所述第一馈送足够数目的新通道；为所述第一馈送分配所述足够数目的新通道；以及将来自所述第一馈送的馈送数据重定向到所述新通道而不是所述第一通道。所述动作还包括使来自所述通道中的每一个的相应输出沉入到所述分布式计算系统的分布式文件系统。所述动作还包括使来自所述通道中的每一个的相应输出沉入到被配置成对所述馈送执行数据分析的数据解析引擎。所述通道中的每一个包括软件实例，所述软件实例被配置成将馈送数据存储在对所述分布式计算系统中的一个或多个计算机或一个或多个计算机的集群为本地的本地数据存储器中，其中，所述本地数据存储器包括随机存取存储器 (RAM) 或大容量存储部或两者。

[0006] 能够实现本说明书中所描述的主题的特定实施例以实现一个或多个优点。日志收集引擎能够在多租户环境中收集流事件数据。日志收集引擎能够借助于能够响应于检测到高负载而为日志收集分配新通道的负载均衡引擎而具有高可用性。日志收集引擎能够支持大数据服务的日志收集以用于分析。

[0007] 在附图和以下描述中阐述本说明书中所描述的主题的一个或多个实施例的细节。本主题的其他特征、方面和优点从本说明书、附图和权利要求书将变得显而易见。

附图说明

[0008] 图1是示例分布式计算系统的框图。

[0009] 图2是被配置成执行负载均衡的示例日志收集引擎的框图。

[0010] 图3是由图2的决策引擎所执行的示例过程的流程图。

[0011] 图4是由图2的负载均衡器所执行的示例过程的流程图。

[0012] 各个附图中的相同的附图标记和名称指示相同的元件。

具体实施方式

[0013] 图1是示例分布式计算系统100的框图。该系统包括在由一个或多个数据通信网络所连接的一个或多个位置中的多个计算机。每个计算机可以是物理计算机或虚拟计算机。该系统能够处置被部署在虚拟化环境中的数据密集型应用。

[0014] 该系统执行N个应用，包括第一应用102、第i个应用104以及第N个应用106。应用能够在不用了解底层系统架构的情况下访问系统中的分布式计算资源。第i个应用是托管第一租户110至第k个租户112的多租户应用。

[0015] 对于每个租户，应用能够分配安全且排他的虚拟计算环境。该环境能够包括软件架构的一个或多个层，例如，从存储层到用户接口层。在一些实施方式中，系统被配置成向租户提供例如有关用户接口元件或商业规则的可定制性，而无需提供底层应用代码的可定制性。

[0016] 该系统包括日志收集引擎114。该日志收集引擎能够收集、聚集并且移动大量的流事件数据。该日志收集引擎被配置成收集来自各种源的流事件数据。

[0017] 流事件数据可以是日志数据，例如，错误或状态日志数据，以及其它类型的事件数

据,例如,网络业务数据、社交媒体数据、电子邮件消息等。日志收集引擎能够收集来自应用的以及来自应用的个别租户的流事件数据。在一些实施方式中,日志收集引擎被配置成从一个或多个其它计算系统116收集流事件数据。

[0018] 该系统包括分布式文件系统118,其通常被实现在非易失性大容量存储存储器上,例如,闪存或磁盘存储器。分布式文件系统104的示例是Hadoop™分布式文件系统,Hadoop™ Distributed File System (HDFS™)。(“Hadoop”和“HDFS”是Apache软件基金会的商标。)日志收集引擎被配置成将收集到的流事件数据移动到分布式文件系统中。日志收集引擎还能够被配置成将收集到的流事件数据移动到其它类型的数据汇点中,例如,被配置成对馈送执行数据分析的数据解析引擎。

[0019] 收集流事件数据能够使用大量的计算资源,例如,存储器和网络带宽。在一些情况下,日志收集引擎上的负载可能难以预测。特别地,在系统正在托管多租户应用的情况下,日志收集引擎上的负载可能随着各种租户发送流事件数据而显著地波动。日志收集引擎能够被配置成执行负载均衡以在托管多租户应用的系统中提供高可用性,例如,在通道正在经历高负载时通过监视通道负载并且分配新通道。

[0020] 图2是被配置成执行负载均衡的示例日志收集引擎200的框图。能够在图1的分布式计算系统100中使用该日志收集引擎。

[0021] 该日志收集引擎包括决策引擎202。该决策引擎被配置成注册流事件数据的新引入的馈送。该决策引擎能够从流事件数据提供方接收用于注册新馈送的请求,例如,应用以及由应用在多租户环境中托管的租户。能够例如使用针对分布式配置服务的Apache ZooKeeper™架构来实现该决策引擎。

[0022] 响应于接收到用于注册新馈送的请求,决策引擎202为新馈送分配网络化连接。例如,决策引擎能够为新馈送分配前端端口或IP地址或两者。决策引擎然后为新馈送分配通道并且使所分配的网络化连接与所分配的通道相关联。日志收集引擎包括第一通道204至第M个通道206。

[0023] 每个通道被配置成对来自流事件数据源的事件进行分级,直到那些事件被汇点(sink)——例如,分布式文件系统——消耗为止。在一些实施方式中,通道是存储引入的事件数据的无源存储结构。例如,通道可以是这样的软件实例,其将事件数据存储在本地的存储器结构或本地文件系统中,直到日志收集引擎能够将该事件数据移动到具有比本地存储器结构或本地文件系统更大的存储容量的分布式文件系统为止。

[0024] 通道能够被实现为Apache Flume™通道。在那种情况下,系统能够创建多个Flume实例,并且每个flume实例能够分配多个通道。当系统分配新通道时,它能够试图例如通过将新通道分配到具有最少数目的通道的Flume实例中来在Flume实例之间均匀地分发通道。

[0025] 日志收集引擎包括负载均衡器208。该负载均衡器被配置成接收来自流数据的注册馈送的数据。该负载均衡器将数据从所分配的网络化连接重定向到针对馈送所分配的通道。能够使用Nginx™服务器以及可从netfilter.org得到的iptables应用来实现该负载均衡器。

[0026] 日志收集引擎包括监视器210。该监视器被配置成监视通道的状态。例如,该监视器能够从每个通道周期性地请求该通道正在使用或者已经在一时间段内使用的存储器或网络带宽的量。能够使用Java管理扩展,Java Management Extensions (JMX) 来实现该监视

器。

[0027] 基于通道的状态,例如,通过将由通道使用的存储器或网络带宽的量与阈值进行比较并且如果所述量超过阈值则确定该通道正在经历高负载,所述监视器来确定通道中的任一个是否正在经历高负载。阈值可以是固定的或可变的,并且能够基于系统的可用计算资源的总量。存储器或网络带宽的量可以是绝对量或相对量,例如,相对于总系统存储器或分布式系统中的单个计算机上的存储器的量。

[0028] 监视器向决策引擎报告馈送是否正被定向至被确定为正在经历高负载的通道。决策引擎然后向负载均衡器发送再均衡命令。作为响应,负载均衡器为馈送分配一个或多个新通道并且使为该馈送所分配的网络化连接与一个或多个新通道相关联。负载均衡器然后能够将来自馈送的流事件数据重定向到代替或者除旧通道之外的一个或多个新通道。在对给定馈送来说存在多个通道的情况下,负载均衡器能够并行地或者通过一个接一个地向每个通道顺序地发送一定量的数据来将馈送数据重定向到通道中的每一个。

[0029] 在一些实施方式中,负载均衡器能够通过创建新Flume实例并且在该新Flume实例内分配新通道来为馈送分配新通道。在一些实施方式中,负载均衡器能够使用在分布式计算系统中不同的计算机或计算机的集群来适应新通道或新Flume实例。不同的计算机或计算机的集群对于日志收集引擎而言可能是新的,例如,不是由日志收集引擎先前或最近使用的。

[0030] 在一些实施方式中,被分配给馈送的新通道的数目基于旧通道上的负载超过阈值负载的量。例如,负载均衡器能够确定旧通道上的负载的负载度量之间的差并且将该负载度量减去阈值负载,然后确定新通道的估计数目以基于所述差来处置负载。

[0031] 在一些实施方式中,在负载均衡器使为馈送所分配的网络化连接与一个或多个新通道相关联之后,负载均衡器使旧通道停用。例如,负载均衡器能够标记待由日志收集引擎停用的旧通道,所述日志收集引擎然后等待直到旧通道中的事件中的全部已被一个或多个汇点消耗为止。在所有事件已被消耗之后,日志收集引擎然后能够对被分配给通道的计算资源解除分配,从而为系统的其余部分释放那些资源。

[0032] 图3是由图2的决策引擎202所执行的示例过程300的流程图。决策引擎接收用于注册流事件数据的新馈送的请求(302)。决策引擎为新馈送分配网络化连接(304)。决策引擎为新馈送分配通道(306)。决策引擎使网络化连接与通道相关联(308),使得馈送的流事件数据将被重定向到通道。

[0033] 当决策引擎从监视器接收到通道正在经历高负载的报告(310)时,决策引擎向负载均衡器发送用于执行负载均衡的命令(312),即,用于为馈送分配一个或多个新通道。

[0034] 图4是由图2的负载均衡器208所执行的示例过程400的流程图。负载均衡器接收馈送的流事件数据(402)。负载均衡器将流事件数据从它分配的网络化连接重定向到它分配的通道(404)。负载均衡器能够继续重定向流事件数据,直到馈送被取消注册为止或者直到负载均衡器接收到再均衡命令为止。

[0035] 在一些点上,负载均衡器从决策引擎接收用于为馈送分配一个或多个新通道的命令(406)。作为响应,负载均衡器为馈送分配一个或多个新通道(408)。负载均衡器使网络连接与一个或多个新通道相关联(410)。在使网络化连接与一个或多个新通道相关联之后,负载均衡器可选地停用旧通道(412)。

[0036] 本说明书中所描述的主题和功能操作的实施例能够用数字电子电路、用有形地具体化的计算机软件或固件、用包括本说明书中所公开的结构及其结构等同物的计算机硬件或者用它们中的一个或多个的组合加以实现。本说明书中所描述的主题的实施例能够作为一个或多个计算机程序被实现,即,在有形非暂时性程序载体上编码以用于由数据处理设备执行或者控制数据处理设备的操作的计算机程序指令的一个或多个模块。替换地或此外,能够将程序指令编码在人工生成的传播信号上,例如,被生成来对信息进行编码以便传输到适合的接收器设备以用于由数据处理设备执行的机器生成的电、光学或磁信号。计算机存储介质可以是机器可读存储装置、机器可读存储基底、随机或串行访问存储器装置或它们中的一个或多个的组合。

[0037] 术语“数据处理设备”是指数据处理硬件,并且包含用于对数据进行处理的所有类型的设备、装置和机器,作为示例包括可编程处理器、计算机或多个处理器或计算机。所述设备也可以是或者还包括专用逻辑电路,例如,FPGA(现场可编程门阵列)或ASIC(专用集成电路)。所述设备除了包括硬件之外,还能够可选地包括为计算机程序创建执行环境的代码,例如,构成处理器固件、协议栈、数据库管理系统、操作系统或它们中的一个或多个的组合的代码。

[0038] 计算机程序——其也可以被称为或者描述为程序、软件、软件应用、模块、软件模块、脚本或代码——能够用任何形式的编程语言——包括编译或解释语言——或者描述性或过程语言编写,并且它能够被以任何形式部署,包括作为独立程序或者作为适合于在计算环境中使用的模块、组件、子例行程序或其它单元。计算机程序可以但不必对应于文件系统中的文件。程序能够被存储在保持其它程序或数据的文件的一部分中,例如,存储在标记语言文档中的一个或多个脚本,以在专用于所述的程序的单个文件中或者在多个协调文件的方式,例如,存储一个或多个模块、子程序或代码的部分的文件中。能够将计算机程序部署成在一个计算机上或在位于一个站点处或者跨越多个站点分布并通过通信网络互连的多个计算机上执行。

[0039] 本说明书中所描述的过程和逻辑流程能够由执行一个或多个计算机程序的一个或多个可编程计算机来执行以通过对输入数据进行操作并且生成输出来执行功能。过程和逻辑流程还能够由专用逻辑电路执行,并且设备还能够作为专用逻辑电路被实现,所述专用逻辑电路例如FPGA(现场可编程门阵列)或ASIC(专用集成电路)。

[0040] 作为示例,适合于执行计算机程序的计算机能够基于通用微处理器或专用微处理器或两者,或任何其它类型的中央处理单元。通常,中央处理单元将从只读存储器或随机存取存储器或两者接收指令和数据。计算机的必要元件是用于依照指令来执行或者实行指令的中央处理单元以及用于存储指令和数据的一个或多个存储器装置。通常,计算机还将包括用于存储数据的一个或多个大容量存储装置——例如磁盘、磁光盘或光盘,或者在操作耦合以从用于存储数据的一个或多个大容量存储装置——例如磁盘、磁光盘或光盘接收数据或者向用于存储数据的一个或多个大容量存储装置转移数据或两者。然而,计算机不必具有这样的装置。而且,能够将计算机嵌入在另一装置,例如,移动电话、个人数字助理(PDA)、移动音频或视频播放器、游戏控制台、全球定位系统(GPS)接收器或便携式存储装置——例如通用串行总线(USB)闪存驱动器——等等中。

[0041] 适合于存储计算机程序指令和数据的计算机可读媒体包括所有形式的非易失性

存储器、媒体以及存储器装置,作为示例包括:半导体存储器装置,例如,EPROM、EEPROM和闪存存储器装置;磁盘,例如,内部硬盘或可移动盘;磁光盘;以及CD-ROM盘和DVD-ROM盘。处理器和存储器能够由专用逻辑电路补充或者并入专用逻辑电路。

[0042] 计算系统能够包括客户端和服务端。客户端和服务端通常远离彼此并且典型地通过通信网络相互作用。客户端和服务端的关系借助于在相应计算机上运行并且彼此具有客户端-服务端关系的计算机程序来产生。

[0043] 虽然本说明书包含许多特定实施方式细节,但是这些不应该被解释为对任何发明的范围或者对可能要求保护的构成限制,而是相反被解释为可能特定于特定本发明的特定实施例的特征的描述。在本说明书中在单独的实施例背景下所描述的某些特征也能够地在单个实施例中相结合地实现。相反地,在单个实施例背景下所描述的各种特征也能够地在独立地多个实施例中或者在任何适合的子组合中实现。而且,尽管特征可以在上面被描述为在特定组合中起作用并且同样地甚至最初要求保护,但是来自所要求保护的组合中的一个或多个特征也能够地在一些情况下从组合中删除,并且所要求保护的组合可以针对子组合或子组合的变化。

[0044] 类似地,虽然按特定次序在附图中描述操作,但是这不应该被理解为要求这样的操作被按所示出的特定次序或按顺序次序执行,或者要求执行所有图示的操作,以实现所希望的结果。在特定情况下,多任务和并行处理可能是有利的。而且,在上面所描述的实施例中独立的各系统模块和组件不应该被理解为在所有实施例中要求这样的独立,并且应该理解,所描述的程序组件和系统能够通常被一起集成单个软件产品中或者封装到多个软件产品中。

[0045] 因此,已经对本主题的特定实施例进行了描述。其它实施例在以下权利要求的范围内。例如,本主题是在科学论文的背景下描述的。本主题能够适用于将深度方面添加到搜索的其它索引工作。在一些情况下,权利要求中所记载的动作能够被按不同次序执行并且仍然实现所希望的结果。此外,附图中所描绘的过程未必要求所示出的特定次序或顺序次序来实现所希望的结果。在特定实施方式中,多任务处理和并行处理可能是有利的。

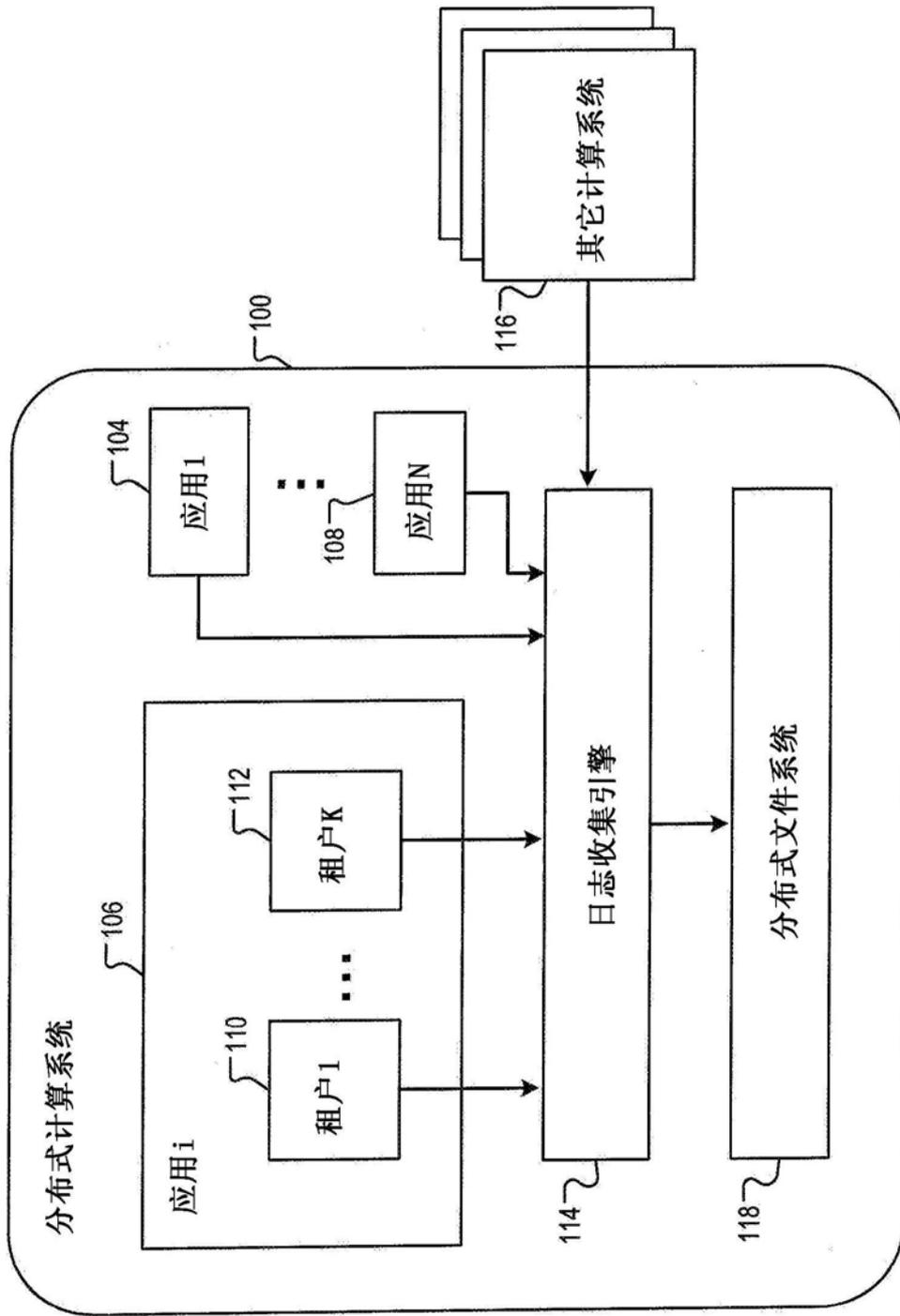


图1

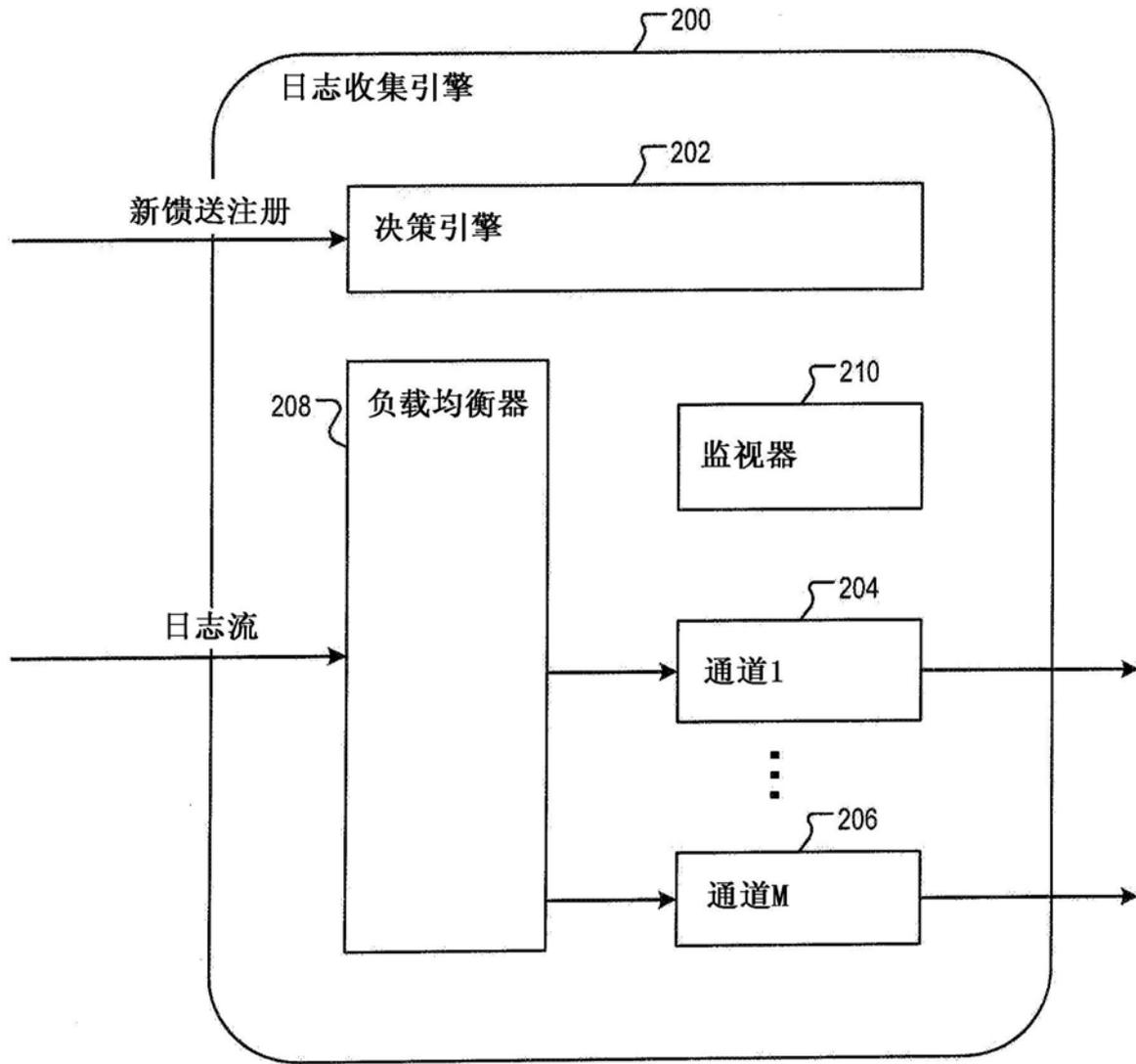


图2

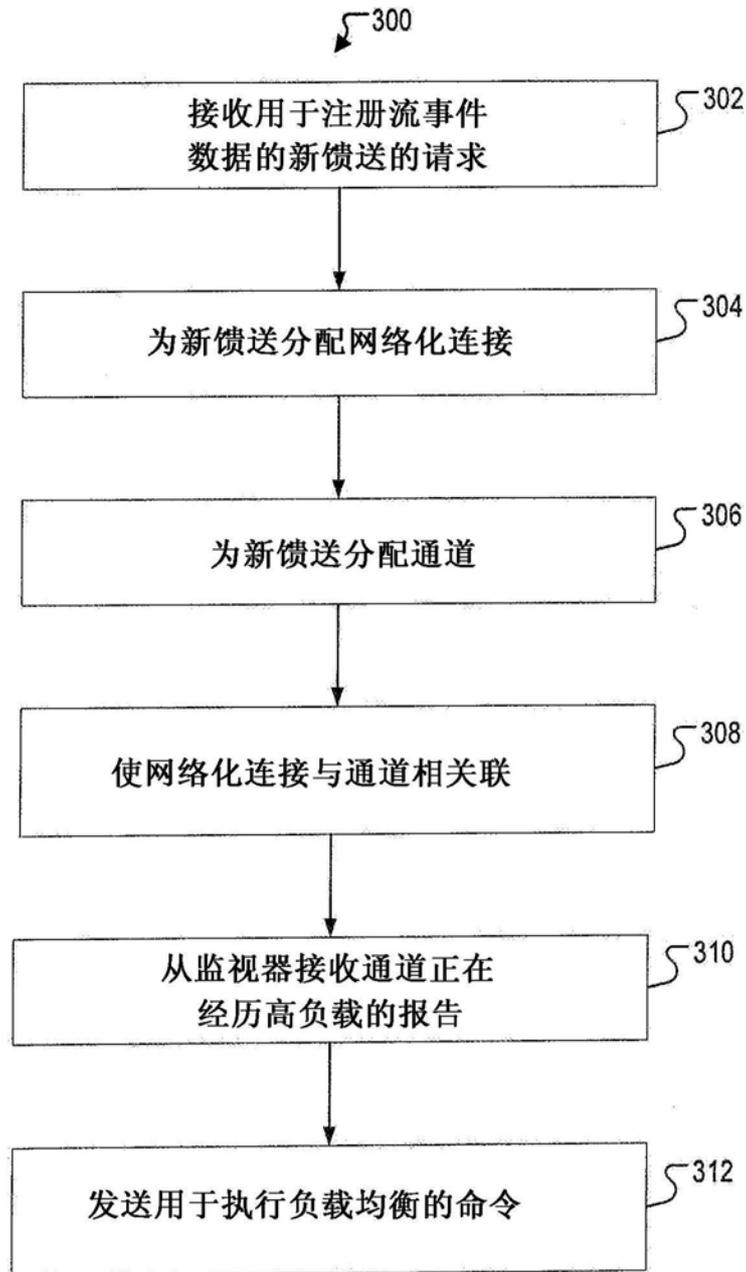


图3

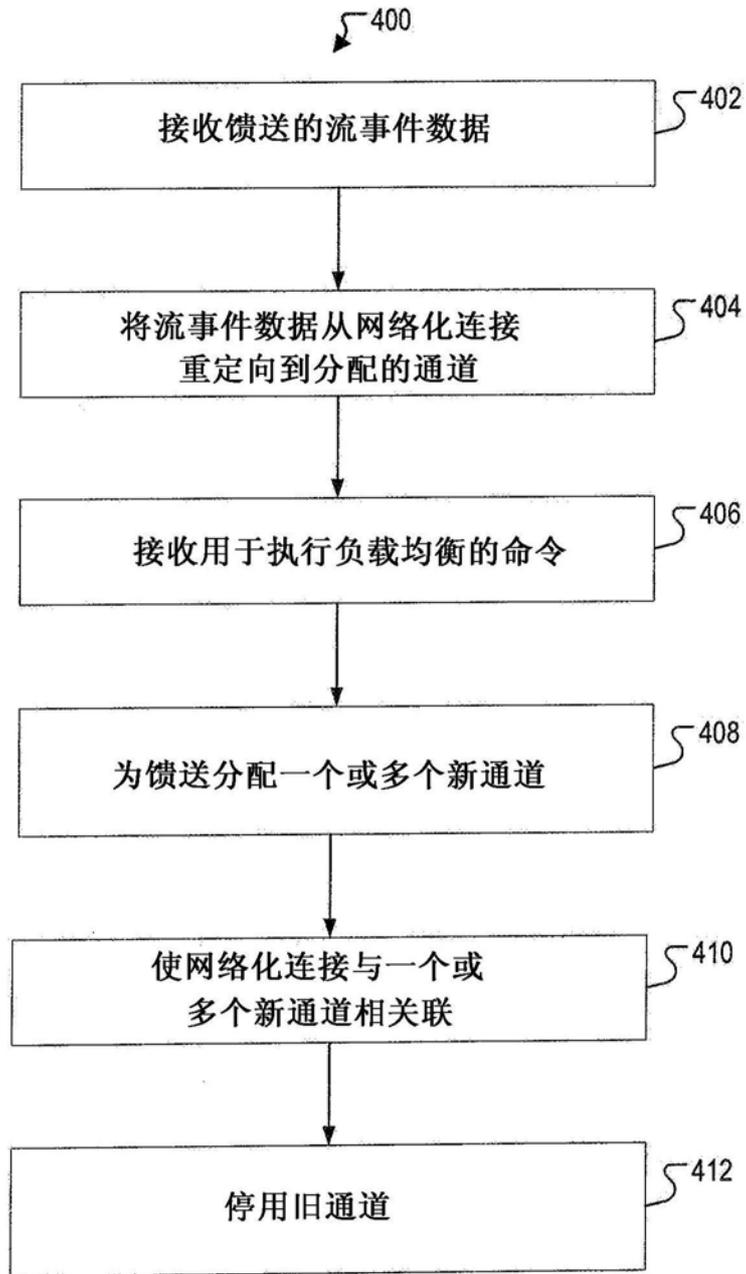


图4