



(12) 发明专利

(10) 授权公告号 CN 111263938 B

(45) 授权公告日 2024. 11. 08

(21) 申请号 201880068978.8

(22) 申请日 2018.09.28

(65) 同一申请的已公布的文献号
申请公布号 CN 111263938 A

(43) 申请公布日 2020.06.09

(30) 优先权数据
62/566,156 2017.09.29 US
16/135,769 2018.09.19 US

(85) PCT国际申请进入国家阶段日
2020.04.23

(86) PCT国际申请的申请数据
PCT/US2018/053574 2018.09.28

(87) PCT国际申请的公布数据
W02019/068002 EN 2019.04.04

(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 H·亨 R·库马尔 J·李
S·翟因 G·兰维帕缇
J·麦克休 C·D·李 A·陈
J-C·黄

(74) 专利代理机构 中国贸促会专利商标事务所
有限公司 11038

专利代理师 刘前红

(51) Int.Cl.
G06F 16/21 (2019.01)
G06F 16/27 (2019.01)

(56) 对比文件
US 2008016123 A1, 2008.01.17

审查员 李欢

权利要求书3页 说明书25页 附图11页

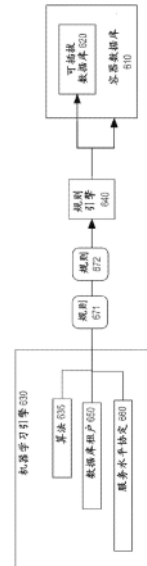
(54) 发明名称

基于规则的自主数据库云服务框架

(57) 摘要

本文提供了通过使用规则引擎来实现自主性的数据云管理技术,该规则引擎通过自主提交重新配置数据库的异步作业来对数据库系统事件做出反应。在实施例中,规则引擎从DBMS接收事件。基于该事件,规则引擎执行规则以生成指示数据库的配置细节的请求。规则引擎将请求发送到请求代理。请求代理基于请求调度异步作业。异步作业基于配置细节来配置数据库。因此,可以基于可预见和不可预见的动态条件来自主管理(无需人工干预)云、数据网格或数据中心中的数据库。

图100



1. 一种计算机实现的方法,包括:
 - 基于历史上由一个或多个数据库管理系统DBMS生成的多个过去事件来训练机器学习引擎;
 - 机器学习引擎生成并且规则引擎加载包括特定规则的第一多个数据库规则,所述特定规则用于生成指示数据库配置细节的请求;
 - 规则引擎从特定DBMS接收事件;
 - 规则引擎基于该事件执行所述特定规则,其中执行所述特定规则包括生成指示数据库配置细节的请求;
 - 规则引擎将该请求发送到请求代理;
 - 请求代理基于该请求来调度异步作业;
 - 异步作业基于数据库配置细节来配置数据库;
 - 基于比所述多个过去事件更近的第二多个过去事件来重新训练机器学习引擎;
 - 基于所述第二多个过去事件中的过去事件的计数来增加置信度得分;以及
 - 当置信度得分超过阈值时,机器学习引擎生成替换所述第一多个数据库规则的第二多个数据库规则。
2. 如权利要求1所述的方法,其中:
 - 规则引擎包括模板引擎;
 - 执行所述特定规则包括模板引擎基于所述事件来选择所述特定规则;
 - 生成请求包括模板引擎从模板生成文本。
3. 如权利要求1所述的方法,其中,所述事件指示从由以下构成的组中选择的至少一项:所述数据库的完整性检查的结果,所述数据库的资源的使用超过阈值,或者阈值超过所述数据库的资源可用性。
4. 如权利要求1所述的方法,其中,所述特定规则是基于以下中的至少一项生成的:特定数据库租户,或特定服务水平协定SLA。
5. 如权利要求1所述的方法,其中,所述第一多个数据库规则至少包括:适用于可插拔数据库的规则,以及适用于容器数据库的规则。
6. 如权利要求1所述的方法,其中,生成所述特定规则包括执行以下中的至少一种算法:最近邻、人工神经网络、线性回归或逻辑回归。
7. 如权利要求1所述的方法,其中:
 - 所述多个过去事件中的每个过去事件包括一个或多个属性;
 - 训练机器学习引擎包括针对所述一个或多个属性中的每个属性,调整与该属性相关联的权重。
8. 如权利要求1所述的方法,还包括:
 - 规则引擎加载所述第一多个数据库规则和多个人工制作的规则;
 - 规则引擎加载所述第二多个数据库规则的子集,其中所述第二多个数据库规则的所述子集中的每个特定规则替换以下中的对应规则:
 - 当置信度得分超过第二阈值时,所述多个人工制作的规则,或者
 - 所述第一多个数据库规则。
9. 如权利要求1所述的方法,其中,所述数据库配置细节指定调谐所述数据库以便于更

快的查询执行。

10. 如权利要求9所述的方法,其中,所述数据库配置细节指定调整以下中的至少一项:打开数据库游标的最大计数、连接会话的最大计数、会话相关变量的存储器存储的最大尺寸、会话无关变量的存储器存储的最大尺寸、或并行性的量。

11. 如权利要求9所述的方法,其中:

所述数据库配置细节指定搜集对于所述数据库的一个或多个对象的数据分布或存储特性的多个测量,所述一个或多个对象包括以下中的至少一项:表、列、索引、分区或查询;所述方法还包括基于所述多个测量来生成一个或多个统计。

12. 如权利要求11所述的方法,还包括:

基于所述一个或多个统计来计算查询谓词的选择性;

基于所述选择性,针对多个替代查询机制中的每个查询机制计算相应的估计的成本。

13. 如权利要求12所述的方法,其中:

查询机制的相应的估计的成本基于以下中的至少一项的估计量:存储器,或输入/输出 I/O;

查询机制包括以下中的至少一项:多个替代访问方法中的特定访问方法,或多个替代加入排序中的特定加入排序。

14. 如权利要求11所述的方法,还包括基于所述一个或多个统计来执行以下中的至少一项:

基于所述一个或多个统计,将优化器提示插入查询中,

创建或丢弃数据库索引,

对列进行编码或取消编码,

对列或表进行碎片整理或预加载,

实体化或去实体化数据库视图的对象,

调整数据库文件或表空间的尺寸,或者

将所述一个或多个统计从所述数据库的数据字典复制到第二数据库的数据字典。

15. 如权利要求1所述的方法,其中,异步作业配置数据库包括:

异步作业检测所述事件的原因或条件是否已停止;

仅在所述原因或条件尚未停止的情况下,异步作业才配置数据库。

16. 如权利要求1所述的方法,其中,所述数据库是能够存储在容器数据库内的可插拔数据库。

17. 如权利要求16所述的方法,其中:

所述第一多个规则进一步包括第二规则,该第二规则适用于存储在存储所述可插拔数据库的同一容器数据库内的第二可插拔数据库;

所述事件识别所述可插拔数据库并指示原因或条件;

所述特定规则包含与所述原因或条件匹配的第一动作模板和第一条件模板;

第二规则包含与所述原因或条件匹配的第二动作模板和第二条件模板;

所述方法还包括:

从所述特定DBMS接收第二事件,该第二事件识别第二可插拔数据库并指示所述原因或条件;

基于检测到以下来应用第二动作模板：
所述事件和第二事件包括所述原因或条件，
异步作业已完成配置所述可插拔数据库，以及
所述原因或条件尚未停止。

18. 如权利要求16所述的方法，其中，异步作业导致以下中的至少一项：

将所述可插拔数据库从第一计算机上的第一容器数据库克隆到第二计算机上的第二容器数据库，

调整数据库的最大连接速率，

基于动态条件生成警报，

调整重做日志或快速恢复区域的尺寸，

对所述数据库或所述特定DBMS应用软件升级或补丁。

19. 如权利要求1所述的方法，其中，所述特定规则基于时间周期来指定以下中的至少一个资源的分配量：CPU持久性存储装置，或RAM。

20. 如权利要求1所述的方法，其中，执行所述特定规则包括生成使第二规则执行的第二事件。

21. 如权利要求1所述的方法，其中，所述事件指示以下中的至少一项：不同查询的执行统计、数据操纵语言DML写的执行统计、数据定义语言DDL命令的执行统计、特定查询的执行统计、所述数据库的完整性检查的结果、所述数据库的资源的使用超过阈值，或者所述数据库的资源的可用性超过阈值。

22. 如权利要求21所述的方法，还包括：在异步作业配置数据库之后，异步作业发起生成第二事件的数据库健康检查，第二事件指示以下中的至少一项：不同查询的执行统计、DML写的执行统计、DDL命令的执行统计、特定查询的执行统计、所述数据库的完整性检查的结果、所述数据库的资源的使用超过阈值，或者所述数据库的资源的可用性超过阈值。

23. 如权利要求1所述的方法，其中，所述请求包括以下中的至少一项：代表性状态转移REST请求、JavaScript对象符号JSON、可扩展标记语言XML，或可执行脚本。

24. 一种用于存储数据的装置，包括用于执行权利要求1-23中的任一项所述的方法的部件。

基于规则的自主数据库云服务框架

技术领域

[0001] 本公开涉及多租户数据网格的管理。本文给出了通过使用规则引擎来实现自主性的数据云管理技术,该规则引擎通过自主提交重新配置数据库的异步作业来对数据库系统事件做出反应。

背景技术

[0002] 当今的前沿组织通过从所有数据源中提取值来经过分析使自己与众不同,从而进一步提高竞争优势。其它公司希望通过其数据管理部署的现代化而成为数据驱动的。这些策略包括诸如不断增长的数据量的管理之类的挑战。当今的数字世界已经以爆炸性的速度创建数据,并且随着物联网 (IoT) 数据源的出现,下一波浪潮即将到来。过去的物理数据仓库非常适合跨整个企业收集数据以进行分析,但是支持它们所需的存储和计算资源却无法跟上爆炸性增长的步伐。此外,由于人为错误,人工的繁琐的补丁、更新、升级任务也会给数据带来风险。为了降低风险、成本、复杂性和时间以实现价值,许多组织都将其数据仓库带到云中。无论是本地托管在私有云中、外包给公共云服务还是两者的混合,云都在确保性能的情况下提供更简单的管理、更高的可扩展性和可用性,并提供降低数据存储和处理成本的新方法。

[0003] 传入数据的速度和数量对传统数据集市、企业数据仓库和分析系统提出了苛刻的要求。典型的数据仓库云解决方案可能无法满足此类需求。许多客户通过“沙盒”环境、业务线数据集市和数据库备份证明了云中的数据仓库的价值。更高级的货币化用例包括高性能数据管理项目、与云计算分析耦合的数据仓库以及大数据云实现方式。

[0004] 生产运营人员通常包括进行数据库生命周期管理和数据库管理员 (DBA) 和其它系统管理员。数据库和计算机之间的托管关系差不多严格地布置在网络拓扑中,使得软件升级、水平缩放以及计划的或紧急的供给 (provision) /配置可能是繁重的且容易出错的,尤其是在云规模比简单的计算机集群大几个数量级的情况下。

[0005] 严格的拓扑可能会阻碍或阻止水平缩放 (弹性) 和容量规划。数据库和数据中心的管理通常是自组织的 (ad hoc), 并且通常基于预期差不多立即执行的必要命令。因此,管理杂务本质上往往是同步的,使得并行化的机会有限。因此,管理作业提交和作业执行趋于紧密耦合,从而需要大量的人工看管来实现相关管理活动的定序。因此,对于云规模的管理自动化,常规的机会可能受到限制。因此,安装的可插拔数据库的演进和迁移也可能受到限制,诸如当容忍中等拓扑比优化重新布置可插拔数据库更容易时。例如,将可插拔数据库从一个容器数据库和/或计算机重新定位到另一个容器数据库和/或计算机可能需要重新打包和重新部署许多客户端和/或服务,这可能不利于这种重新定位。人为数据库调谐和管理既缓慢又容易出错。管理质量也取决于人类数据库管理员 (DBA) 的知识水平而非常不同。行业解决该问题的一种方式聘请专业的DBA来在某种程度上减少错误并在某种程度上提高性能,这或多或少可以缓解但不能完全解决与规模相关的各种问题。所需要的是一种实现具有无与伦比的可靠性的业务洞察力的解决方案,该解决方案实际上可以确保避免容易出

错的人工管理处理。因此,数据网络的可靠性、可用性和可维护性(RAS)以及效率和性能应当被提高到以前无法达到的程度。

附图说明

[0006] 附图中:

[0007] 图1是描绘实施例中的示例云的框图,该云使用异步作业来最终执行对于可插拔数据库的管理配置请求;

[0008] 图2是描述实施例中的示例异步配置处理的流程图,该示例异步配置处理最终执行对于可插拔数据库的管理请求;

[0009] 图3是描绘实施例中的具有自动反射以治愈拓扑的示例云的框图;

[0010] 图4是描绘实施例中的具有规则引擎的示例云的框图,该规则引擎通过自主提交重新配置数据库的异步作业来对数据库系统事件做出反应;

[0011] 图5是描绘实施例中的使用规则引擎的示例自主处理的流程图,该规则引擎通过提交重新配置数据库的异步作业来对数据库系统事件做出反应;

[0012] 图6是描绘实施例中的示例云的框图,该云使用机器学习来自动调谐规则以对容器数据库及其可插拔数据库进行最优管理;

[0013] 图7是描绘实施例中的示例云的框图,该云用历史事件进行训练以改进规则;

[0014] 图8是描绘实施例中避免规则之间的干扰的示例云的框图;

[0015] 图9是描绘实施例中用于更快的查询执行的示例数据库调谐处理的流程图;

[0016] 图10是图示可以在其上实现本发明的实施例的计算机系统的框图;

[0017] 图11是图示可以用于控制计算系统的操作的基本软件系统的框图。

具体实施方式

[0018] 在下面的描述中,出于解释的目的,阐述了许多具体细节以便提供对本发明的透彻理解。但是,将显而易见的是,可以在没有这些具体细节的情况下实践本发明。在其它情况下,以框图形式示出了众所周知的结构和设备,以避免不必要地使本发明模糊。

[0019] 本文根据以下大纲描述各实施例:

[0020] 1.0 总体概述

[0021] 2.0 示例数据网格

[0022] 2.1 容器数据库和可插拔数据库

[0023] 2.2 管理请求

[0024] 2.3 配置细节

[0025] 2.4 配置描述符

[0026] 2.5 异步

[0027] 2.6 托管元数据(Hosting Metadata)

[0028] 2.7 后处理

[0029] 3.0 异步配置处理

[0030] 4.0 自主重新配置

[0031] 4.1 监视器

- [0032] 4.2 升级/降级
- [0033] 5.0 示例自主数据库云
- [0034] 5.1 数据库系统事件
- [0035] 5.2 事件类别
- [0036] 5.3 规则引擎
- [0037] 5.4 规则
- [0038] 6.0 示例自主管理处理
- [0039] 6.1 刺激
- [0040] 6.2 反应
- [0041] 6.3 连锁反应
- [0042] 7.0 机器学习
- [0043] 7.1 优化算法
- [0044] 7.2 规则API
- [0045] 8.0 训练
- [0046] 8.1 事件交付
- [0047] 8.2 历史重放
- [0048] 8.3 加权的属性
- [0049] 8.4 训练置信度
- [0050] 8.5 重新训练
- [0051] 8.6 保留优先级
- [0052] 8.7 示例技术堆栈
- [0053] 9.0 规则干扰
- [0054] 9.1 条件识别
- [0055] 9.2 模板化
- [0056] 9.3 整体效率
- [0057] 9.4 示例管理和维护任务
- [0058] 10.0 示例自动调谐处理
- [0059] 10.1 统计
- [0060] 10.2 谓词选择性
- [0061] 10.3 成本-效益分析
- [0062] 10.4 示例调谐场景
- [0063] 11.0 数据库系统概述
- [0064] 12.0 可插拔数据库和容器数据库系统概述
- [0065] 13.0 硬件概述
- [0066] 14.0 软件概述
- [0067] 15.0 云计算
- [0068] 1.0 总体概述

[0069] 本文给出的是用于管理数据库云的自主技术。本文中的示例系统的两个主要组件是异步保证的执行作业框架和基于事件的自动规则引擎。这些主要组件促进可包括多租户

数据库和/或数据库的层次结构或联盟的(一个或多个)数据库管理系统(DBMS)的生态系统的自动化管理和性能优化。在下面的“数据库系统概述”部分中更详细地说明DBMS。在一些情况下,可能存在数据库的结构聚合(例如,嵌套),诸如当可插拔数据库(PDB)驻留在容器数据库(CDB)内时。在下面的“可插拔数据库和容器数据库系统概述”部分中更详细地说明CDB内的PDB的嵌套。

[0070] 诸如数据云之类的在线云的弹性性质需要动态拓扑,并且至少在公共云的情况下需要多租户。利用本文的技术,数据云可以被配置成资源(包括数据存储库和分析)的动态生态系统,其可能经历诸如需求尖峰、网络天气、维护和容量增长之类的具有不同程度的不可预测性的干扰。本文是基于各种机制和技术的用于异步框架的DBMS集成的规则引擎,用于配置在变化的环境条件下优化云性能的智能反应。

[0071] 本文中的DBMS可以提供数据库可移植性机制或与数据库可移植性机制合作,使得数据库可以被自动复制或重新定位。因此,云的逻辑拓扑和负载分布可以是动态的且可优化的。例如,交付点(PoD)可以是服务的逻辑共址(co-location)单元。在实施例中,计算机可以托管一个或多个PoD,每个PoD可以托管一个或多个容器数据库,每个容器数据库可以托管一个或多个可插拔数据库。(一个或多个)数据库实例的创建、复制、移动和移除可以根据数据库的明确定义的(例如,统一的)生命周期而发生。

[0072] 规则引擎可以提交异步作业,以跨一组交付点(PoD)、容器数据库或托管中间件的其它数据库执行一些或所有可插拔数据库(PDB)生命周期任务。在实施例中,异步作业框架为客户端提供代表性状态转移(REST)API以提交请求,这些请求然后被转换成异步作业。异步作业是可以被推迟的工作单元,诸如计划或排队。异步作业可以以一种排序提交,然后以另一种时间排序执行。作业管理的所有方面,包括持久性、计划、分配、执行、恢复和错误处理,都在系统内处理。作业被路由到正确的PoD,其中它使用PoD API被执行。包括请求代理(request broker)、事件代理和规则引擎的主要组件是可独立地且水平地扩展的。

[0073] 规则引擎接受事件并将其转换成作业规范以供执行。来自PoD数据库的事件由事件代理中摄取,事件代理然后对事件进行过滤和分类,并将其馈送到规则引擎和机器学习引擎中。规则既是自动又是人工生成的,以使系统能够自我调谐和管理。

[0074] 一般而言,规则引擎可以通过将其代码库划分为可编辑/可替换的规则和经强化的引擎来解释规则的逻辑解释器,以实现可定制性和可维护性。规则可以符合引擎可解析的形式语法。在某种程度上,即使规则对逻辑进行编码,规则也可以作为数据被处理和/或存储。因此,可以诸如根据分析和/或机器学习来自动修改规则。在实施例中,规则可以是自我修改的。规则可以包含要评估的谓词,并且如果满足该谓词,那么结果会发生。谓词可以评估诸如变量和/或事件之类的条件的组合。规则可以通过更新变量和/或引发附加事件来更改条件。因此,相关规则可以相互作用以实现推论,该推论得出复合行为的信息和/或定序。规则、规则激活和规则演变将在本文稍后针对数据云的自主管理和性能优化进行讨论。本文稍后将讨论异步数据库管理作业的基于规则的生成和编排。本文稍后将讨论基于机器学习的规则调谐和演进。

[0075] 本文提出的解决方案具有配置或重新配置可插拔数据库的异步作业,诸如异步数据库管理作业。例如,异步数据库管理作业可以创建或丢弃关系表的索引,或者创建、复制、移动或移除数据库实例。一般而言,数据库配置或重新配置是定义、移除或调整(例如重新

定义) 诸如数据库实例、模式、名称空间、表、关系、索引、文件或逻辑之类的数据库对象的系统更改。配置可以是诸如在从另一个计算机或DBMS进行重新定位期间在给定计算机处和/或在给定数据库或DBMS内供给或部署数据库或数据库对象的一部分。重新配置可以是对数据库或数据库对象进行调整或移除。配置或重新配置可能需要执行数据定义语言 (DDL)、调整DBMS的数据字典的内容和/或添加、更改或移除诸如文件之类的资源。异步可以促进计划、工作负载平衡和优先化。

[0076] 本文中的每个解决方案都具有规则引擎,该规则引擎针对上下文适当的行为编排异步作业。规则引擎具有处理在模板中编码的规则的模板引擎。规则以事件为条件。这些解决方案中的每一个都具有动态地发出指示操作条件的事件的数据库管理系统 (DBMS)。事件可以使规则生成配置或重新配置可插拔数据库的管理请求。例如,可插拔数据库可能耗尽盘空间。作为响应,DBMS可以将向规则引擎发送事件。规则引擎对规则的执行可以生成可将可插拔数据库从容器数据库或计算机上移动到另一个容器数据库或计算机的管理请求。规则引擎将管理请求作为REST请求发送到请求代理,该REST请求指示识别可插拔数据库并指定逐出(重新定位)可插拔数据库的配置细节。请求代理基于REST请求调度(dispatch)异步作业。基于配置细节,异步作业将可插拔数据库移动到另一个计算机上的另一个容器数据库。

[0077] 在实施例中,规则引擎从DBMS接收事件。基于该事件,规则引擎执行规则以生成指示数据库配置细节的请求。规则引擎将请求发送到请求代理。请求代理基于请求来调度异步作业。异步作业基于配置细节来配置数据库。因此,可以基于可预见和不可预见的动态条件来自自主管理(无需人工干预)云、数据网格或数据中心中的数据库。

[0078] 解决方案可以具有造成异步的组件,诸如队列、线程池、泳道和执行器。如下所示,图1-3描绘了为数据库管理作业的异步执行而定制的云基础设施。图4-9描绘了那个异步数据库管理云基础设施的提炼,这些提炼基于规则引擎实现自主性,规则引擎通过引发异步作业对来自(一个或多个)DBMS的自发事件做出智能反应。异步数据库管理云基础设施可以组合图10-11所描绘的计算系统。在实践中,实施例可以组合本文讨论的图的数据库云和计算系统中的一些或全部的整体或部分,以及在相关的美国申请16/055,468中提出的那些。

[0079] 2.0示例数据网格

[0080] 图1是描绘实施例中的示例云100的框图。云100使用异步作业最终执行对于可插拔数据库的管理请求。为了作为数据网格进行操作,云100由诸如托管诸如121-122和130的数据库的111-112的(一个或多个)计算机集群组成。例如,数据库121-122和130中的每一个可以由同一数据库管理系统(DBMS)管理的关系数据库。计算机111-112中的每一个可以是个人计算机、诸如刀片之类的机架服务器、大型机、虚拟机或能够托管数据库的任何联网的计算设备。计算机111-112可以经由诸如局域网(LAN)之类的网络或诸如全球互联网之类的网络的彼此以及与其它计算机(未示出)通信。计算机111-112可以共址,例如在同一机架上,或者可以被分开,诸如在不同的数据中心中。

[0081] 2.1容器数据库和可插拔数据库

[0082] 云100可以向客户端(未示出)隐藏拓扑细节的方式托管诸如可插拔数据库130之类的数据库。例如,客户端可能在不知道哪个数据中心、计算机和容器数据库实际托管可插拔数据库130的情况下使用可插拔数据库130。诸如121-122之类的容器数据库可以包含诸如130之类的可插拔数据库。例如,容器数据库121可以由多个租户共享,这些租户可以共

享或可以不共享由容器数据库121托管的各种可插拔数据库。

[0083] 2.2管理请求

[0084] 在操作中,数据库管理员(DBA)人员或管理软件可以提交管理请求140,以创建、配置或重新配置可插拔数据库130,而无需知道可插拔数据库130在哪里被托管或将在哪里被托管。例如,云100可以在差不多任意时间重复地隐式或自发地将可插拔数据库130重新定位到不同的容器数据库和/或计算机。因此,管理请求140内的配置细节150不需要识别容器数据库或计算机。

[0085] 例如,DBA可以从驻留或不驻留在云100内的差不多任意的计算机向云100提交管理请求140以供执行。管理请求140可以作为超文本传输协议(HTTP)请求(诸如公共网关接口(CGI) post)、代表性状态转移(REST)请求、远程过程调用(RPC)、简单对象访问协议(SOAP)调用、异步JavaScript XML(可扩展标记语言)(AJAX)请求或其它远程数据传递机制交付到云100。

[0086] 2.3配置细节

[0087] 管理请求140是面向目标/结果的,并且不需要指定要执行的战略或策略。例如,配置细节可以指定可插拔数据库130应该被创建或调整为适合某些高级需求。

[0088] 例如,如果可插拔数据库130要耗尽盘空间,则配置细节150可以指定更大量的盘空间。云100在履行管理请求140方面具有灵活性。例如,可以通过将可插拔数据库130的表空间文件从一个文件系统移动到另一个文件系统来实现更大的盘空间,就像以前(交叉)安装在同一个计算机上那样。替代地,可以通过调整其中托管可插拔数据库130的虚拟机分配(并可能重新启动虚拟机)来实现更多的盘空间。替代地,云100可以实际上必须将可插拔数据库130从一个容器数据库和/或计算机移动到另一个容器数据库和/或计算机。

[0089] 同样,如果可插拔数据库130的性能不佳,则配置细节150可以指定更多的处理能力,诸如更多的处理器、处理器核心、线程或并行性。云100可以通过向托管可插拔数据库130的虚拟机授予更多处理器核心来履行管理请求140。替代地,云100可以实际上必须将可插拔数据库130从一个容器数据库和/或计算机移动到另一个容器数据库和/或计算机,以获取更多处理能力。

[0090] 云100不会立即(同步地)执行管理请求140。例如,履行管理请求140可能需要使可插拔数据库130暂时退出服务。因此,配置细节150可以指定时间、时间窗口、空闲或轻工作负载的量,或者其它应该在执行管理请求140之前发生的动态条件。可替代地,云100、可插拔数据库130或任何涉及的容器数据库或计算机可能暂时太忙(例如,需求尖峰),以至于不能立即执行管理请求140。因此,云100可以排队、缓冲或以其它方式推迟管理请求140的执行。

[0091] 2.4配置描述符

[0092] 云100使用配置描述符160和异步作业170来推迟管理请求140的执行。在接收到管理请求140后,云100基于管理请求140和配置细节150生成配置描述符160。配置描述符160以适合于存储并最终调度给异步作业170以供执行的格式对管理请求140的规格化规范进行编码。配置描述符160可以是经编码的文档,诸如XML或JavaScript对象符号(JSON)。规格化可能需要转换和/或缩放数字单位(如本文后面讨论的)或范围限制。例如,配置细节150可以指定夜间执行,该云100可以将其转换成作为特定深夜时段的时间窗口的配置描述符

160。例如,可以舍入对少于最小资源量的请求。同样如本文稍后讨论的,指定数量的资源可以以可强制的方式相互依赖,诸如分配的易失性存储器不应超过分配的持久性存储装置。

[0093] 云100可以将配置描述符160缓冲或存储在存储器、数据库或文件中。云100可以(例如,同步地)向任何提交了管理请求140的客户端确认该请求被接受并且未决。例如,如果管理请求140作为HTTP请求到达,则云100可以同步发送确认管理请求140至少已经被排队的HTTP响应。确认可以包括跟踪标识符,客户端随后可以使用该跟踪标识符来取消管理请求140、暂停管理请求140或轮询管理请求140的状态。确认可以包括可以轮询和/或等待管理请求140实际被履行的面向对象的接收器(诸如计算未来(computational future)、远程未来(remote future)或智能存根(smart stub))。接收器可以是面向对象的类的实例。接收器可以是主动的,诸如轮询状态的智能存根。存根是远程代理。智能存根是包含额外逻辑的存根,用于灵活地定位(一个或多个)服务和/或管理诸如轮询或重试之类的重复操作。接收器可以是被动的,诸如在等待来自云100的状态或完成消息或信号时挂起。

[0094] 2.5异步

[0095] 在生成配置描述符160之后,云100可以立即或最终创建(或重新使用)异步作业170以基于配置描述符160执行管理请求140。例如,异步作业170可以从轻量级线程池、重量级操作系统进程或其它执行程序中重新使用。在将配置描述符160分配给异步作业170之后,异步作业170可以立即或最终根据配置描述符160执行(履行)管理请求140。例如,异步作业170可以空闲直到配置描述符160被分配给它。例如,配置描述符160可以指定管理请求140的执行在午夜之前不应该开始。

[0096] 可替代地,异步作业170可以从未决的配置描述符的队列中一次执行一个其它请求。当配置描述符160最终到达队列的开头时,异步作业170可以退出队列并执行配置描述符160。例如,配置描述符160可以是Java消息服务(JMS)队列中的JSON文档。在实施例中,队列可以是基于以下中的任何一项的优先级队列:a) 诸如160的配置描述符中指示的优先级,b) 配置描述符中指示的期限(完成时间)或生存时间(TTL),或者c) 请求类型,诸如创建数据库、移动数据库、提高数据库服务质量、复制数据库等。例如,备份和副本创建自然是低优先级的。但是,提高服务质量自然是高优先级的,因为数据库正在变得不足,诸如等待时间增加或盘空间耗尽。从备份还原数据库可能是紧急的。

[0097] 2.6托管元数据

[0098] 异步作业170可以加载、解析或以其它方式处理配置描述符160,以发现要针对可插拔数据库执行哪个或哪些配置更改。如果配置描述符160未识别容器数据库或计算机,则云100可从托管元数据180检索所需的标识符。托管元数据180提供目录信息,该目录信息将可插拔数据库的标识符映射到托管可插拔数据库的容器数据库和/或计算机的标识符。托管元数据180可以用查找表、数据库表或其它参考数据结构(诸如映射)来实现。托管元数据可以部署为文件、网络服务或其它方式。托管元数据180可以具有附加信息,诸如资源容量和利用率测量、资源需求和数据库相互依赖性。例如,托管元数据180可以指示诸如以下的事实:a) 特定容器数据库具有特定的可插拔数据库,b) 特定的可插拔数据库使用了一定量的盘空间,c) 特定计算机具有的处理核心的计数,d) 特定的物理计算机托管特定的虚拟计算机,e) 特定的计算机托管特定的容器数据库,f) 特定的容器数据库具有特定的可插拔数据库的副本,g) 特定的容器数据库托管特定的多实例可插拔数据库的数据分区,以及h)

特定的计算机和数据库具有特定的风格、发行级别和补丁程序级别。

[0099] 例如,异步作业170可以检测到配置描述符160为可插拔数据库130指定了三个处理器核心。异步作业170可以咨询托管元数据180,以发现可插拔数据库130驻留在计算机111上的当前容器数据库121中。托管元数据180可以指示:a) 当前容器数据库121限于两个处理核心,即使b) 计算机111具有四个处理核心。因此,异步作业170可以决定可插拔数据库130需要被移动到不同的容器数据库。例如,托管元数据180可以指示:a) 计算机112也具有四个处理核心,并且b) 目标容器数据库122对使用处理核心没有限制。因此,对于要将可插拔数据库130移动到哪里,异步作业170可以选择目标容器数据库122。因此,异步作业170可以通过将可插拔数据库130移动到目标容器数据库122来履行管理请求140。

[0100] 2.7后处理

[0101] 在履行管理请求140之后,异步作业170可以以各种方式附加地进行后处理。异步作业170可以通知云100或客户端管理请求140完成。异步作业170可以更新托管元数据180以反映由异步作业170做出的改变。异步作业170可能空闲、返回到工作器池、使另一个管理请求出队、或等待破坏,诸如被垃圾回收或以其它方式被取消分配。

[0102] 3.0异步配置处理

[0103] 图2是描绘实施例中的示例异步配置处理的流程图,该示例异步配置处理最终使用异步作业执行对于可插拔数据库的管理请求。参考图1讨论图2。

[0104] 在步骤202中,接收指示可插拔数据库的配置细节的管理请求。例如,云100(从外部客户端或从内部设施)接收到来自想要重新配置可插拔数据库130的管理人员的管理请求140。例如,管理员想要通过给予可插拔数据库130更多的存储器(例如,更大的数据库高速缓存)来减少可插拔数据库130的等待时间。管理员可以使用web浏览器来创建在管理请求140中指定更大量的存储器的配置细节150。

[0105] 在步骤204中,云基于管理请求的配置细节生成指定异步作业的配置描述符。例如,云100在生成配置描述符160时参考配置细节150和管理请求140。例如,配置描述符160可以是定义未决工作单元的工作票据。云100可以:a) 准备立即执行配置描述符160,b) 使描述符排队以便以后执行,或者c) 调度配置描述符160以在将来的相对或绝对时间执行。

[0106] 在步骤206中,云访问托管元数据以检测已经或将要托管可插拔数据库的(一个或多个)容器数据库和/或(一个或多个)计算机的(一个或多个)位置。虽然步骤206被示出为发生在步骤204和208之间,但是步骤206的一些或所有活动实际上可以在步骤204期间急切地发生,或者在步骤208期间懒惰地发生。例如,可插拔数据库130移动到的目标容器数据库或计算机的选择可以在步骤204处发生。而可插拔数据库130的当前容器数据库的识别可以在步骤208处发生,反之亦然。步骤204可以差不多在接收到管理请求140时发生。而步骤208可以被推迟到实际履行管理请求140的方便时间。那些时间中的任何时间可能适合于执行步骤206的部分或全部。例如,在恰好及时创建异步作业170以实际执行管理请求140之前,访问托管元数据180可以由异步作业170或由云100执行。

[0107] 在步骤208中,云执行异步作业以配置或重新配置可插拔数据库。例如,云100可以检测到可插拔数据库130和/或当前容器数据库121可能已增加其存储器分配,并且因此云100不需要将可插拔数据库130移动到另一个容器数据库。因此,在步骤208中,异步作业170可以简单地调整可插拔数据库130和/或当前容器数据库121以保留更多的存储器。

[0108] 在异步作业170的生命周期中的任何时候(包括上述任何步骤),异步作业170都可以产生附加的管理请求。例如,在履行管理请求140的同时,异步作业170可以通过提交从当前容器数据库121创建可插拔数据库130的备份的第二管理请求以及将可插拔数据库130的备份恢复到目标容器数据库122中的第三管理作业来克隆可插拔数据库130。

[0109] 4.0自主重新配置

[0110] 图3是描绘实施例中的示例云300的框图。云300具有动态优化拓扑的性能的自动反射功能。在这个示例中,云300响应于盘满而自动重新定位和升级可插拔数据库E。云300可以是云100的实现。

[0111] 4.1监视器

[0112] 云300包含组件310、321-322、330、360和370,每个组件可以是计算机或共享计算机的可共址的软件服务。监视器310从云300的组件接收遥测,诸如测量和警报。监视器310可以配置有反射,诸如基于规则的反应,这些反射:a)以可配置的方式检测环境条件,诸如警报、阈值穿越和其它条件;b)对如何反应做出可配置的决定;以及c)发出管理请求以执行其决定。

[0113] 盘满A可以是如下警报:指示盘已满,或者根据穿越的阈值,盘已接近满,诸如当容器数据库(CDB)321具有生成大量审核日志的繁忙时段时。响应于盘满A,监视器310可以决定:a)切换为对一些文件(诸如审核文件)使用存储区域网络(SAN),b)将容器数据库(及其可插拔数据库)移动到具有更大盘的交付点(PoD),或c)从容器数据库中逐出一些可插拔数据库。如图所示,监视器310决定通过生成请求B从CDB 321中逐出可插拔数据库E。

[0114] 4.2升级/降级

[0115] 请求B的执行需要对托管储存库360进行询问,托管储存库360通过公告指示应将可插拔数据库E移动到CDB 322中的前一个和下一个托管C来应答。前一个和下一个托管C还指示软件版本不匹配,为此工作器370应该在移动期间向可插拔数据库E应用软件升级(或降级)。CDB 321-322中的每一个都有大量的系统和应用软件层,并且每个软件层都有其自己的发行版本和补丁生命周期。因此,版本不匹配可能涉及容器数据库DBMS、PoD框架、字节码解释器、虚拟化中间件、操作系统或其它已安装的软件基础设施中的一个、一些或全部。

[0116] 工作器370可以使用前一个和下一个托管C来检测版本不匹配,并执行迁移脚本以将可插拔数据库E迁移到目标发行版本。迁移脚本可以包括工件(artifacts),诸如shell脚本、python脚本、本机可执行程序、存储的过程,以及其它结构化查询语言(SQL),诸如数据定义语言(DDL)或数据操纵语言(DML)。迁移脚本可以调整工件,诸如模式、表内容和文件。可能运行一些迁移脚本来准备可插拔数据库E以便重新定位。可以运行其它迁移脚本来准备目标CDB 322以托管可插拔数据库E。在将可插拔数据库E移动到CDB 322中之后,还有其它迁移脚本也可以对可插拔数据库E进行调整。

[0117] 在相关的美国申请16/055,468中给出了用于实现和操作自主多租户数据库云服务框架的架构和技术。

[0118] 5.0示例自主数据库云

[0119] 图4是描绘实施例中的示例云400的框图。云400具有规则引擎,该规则引擎通过自主提交用于重新配置数据库的异步作业来对数据库系统事件做出反应。云400可以是云100的实现方式。云400包括诸如410之类的DBMS、规则引擎440和请求代理470(其可以是相关的

美国申请16/055,468的图6的WebLogic代理630的任何实现方式)。DBMS 410包含诸如420之类的数据库,其可以是关系数据库、NoSQL数据库或其它托管数据库。

[0120] 5.1数据库系统事件

[0121] DBMS 410的操作条件可以动态波动,诸如盘空间、处理负载或查询混合。DBMS 410可以配置有阈值和其它触发器,这些阈值和其它触发器检测操作条件的重大改变。例如,未使用的盘空间可以减少到阈值百分比或绝对量以下。响应于操作条件的重大改变,DBMS 410可以发出诸如430的事件,以针对给定的可插拔数据库、容器数据库或DBMS 410本身公告这种改变。例如,DBMS 410可以发送JMS消息或操作系统信号以将事件430交付到其它(一个或多个)软件程序。因此,根据一系列发出的事件,其它软件可以容易地看到DBMS 410的操作条件。

[0122] 5.2事件类别

[0123] 事件430可以表示可被检查以确定是否正在发生重大条件的属性的集合。在实施例中,事件代理处置向订户的事件交付。每个事件订阅都由事件类型和事件订阅者组成。事件可以广义地分为诸如操作、监视器和与SQL相关之类的类别。

[0124] 操作事件指示数据库的生命周期内的状态过渡。对于特定的数据库,操作事件可以指示以下中的至少一项:供给、取消供给、重新定位、克隆和/或水平缩放、刷新、备份、还原、升级、打补丁、注册或注销。

[0125] 监视事件指示数据库的性能状态。监视事件可以指示一般或重要的统计(诸如在健康检查期间测得的)中SQL命令的性能度量。

[0126] 与SQL相关的事件指示特定子句、谓词或整个或部分语句的统计。语句可以是DDL或DML,诸如查询。例如,SQL事件可以指示特定谓词被不同的查询共享,这可以意味着该谓词是实质化为虚拟列的候选。

[0127] 5.3规则引擎

[0128] 规则引擎440可以接收诸如430之类的事件以监视诸如410之类的DBMS的状态。例如,规则引擎440可以根据发布/订阅框架来订阅DBMS 410所发布的事件。在规则引擎440和DBMS 410之间的可以是事件总线、队列或代理(未示出),其可以提供诸如缓冲、持久性、可靠性和解耦合之类的功能,使得DBMS 410和规则引擎440不需要彼此直接接口。

[0129] 规则引擎440可以是推理引擎,其加载并应用诸如450之类的规则以实现可定制的行为。例如,规则引擎440可以将规则450加载为可以被编译或解释为扩展规则引擎440的行为的数据。例如,规则450可以是符合语法的文本,该语法用于对哪些事件可以匹配该规则以及当匹配发生时采取什么动作进行编码。可以加载多个规则以配置规则引擎440,从而将特定行为反应与事件可以指示的特定操作条件相关联。例如,规则450可以指定当事件430指示空闲盘空间低于1GB时,应当将数据库420移动至另一个计算机。

[0130] 5.4规则

[0131] 但是,规则引擎440不直接执行由与事件430匹配的规则所指定的动作。相反,规则450生成并发送请求460,以引起异步作业480的提交,如下所示。规则450可以在请求460内发出配置细节465,以直接或间接指定针对诸如420之类的数据库的配置调整。配置细节465可以将数据库420识别为重新配置目标。

[0132] 规则450可以诸如通过使用REST请求通过HTTP发送JSON、XML或其它CGI数据,将请

求460发送到请求代理470。在实施例中,请求460包括诸如JavaScript、SQL、python或shell脚本之类的可执行脚本,以供异步作业480执行。基于请求460和配置细节465,请求代理470可以创建异步作业480,异步作业480的执行可以例如根据相关的美国申请16/055,468的图6的泳道、线程池和队列来推迟。异步作业480最终执行,这包括根据配置细节465重新配置数据库420。因此,规则引擎440可以根据诸如410之类的DBMS的操作条件的改变来自主地管理诸如420之类的数据库。因此,规则引擎440提供维护云400的数据库所需的自主行为的中央储存库。

[0133] 6.0示例自主管理处理

[0134] 图5是描绘实施例中的使用规则引擎的示例自主处理的流程图,该规则引擎通过提交用于重新配置数据库的异步作业来对数据库系统事件做出反应。参考图4讨论图5。

[0135] 6.1刺激

[0136] 在步骤501中,规则引擎从DBMS接收事件。在实施例中,DBMS 410直接将事件430发送到规则引擎440。在实施例中,DBMS 410和规则引擎440由单独的计算机或操作系统进程托管,并且事件430根据诸如HTTP、REST、简单对象访问协议(SOAP)和/或远程过程调用(RPC)之类的远程协议直接交付。在实施例中,DBMS 410间接发送事件430。例如,企业服务总线(ESB)、消息队列(MQ)或其它代理传达、中继和/或缓冲事件430以最终交付到规则引擎440。在实施例中,规则引擎440被嵌入在DBMS 410的操作系统进程和地址空间内,使得事件430可以在子例程调用期间被交付。

[0137] 一些事件可以指示以下中的至少一项:不同查询的执行统计,诸如平均响应时间、数据操纵语言(DML)写的执行统计、数据定义语言(DDL)命令的执行统计、特定查询的执行统计、数据库的完整性检查的结果、数据库的资源的使用超过阈值或数据库的资源可用性低于阈值。

[0138] 6.2反应

[0139] 在步骤502中并且基于事件,规则引擎执行规则以生成指示数据库的配置细节的请求。例如,事件430与规则450匹配,该规则450生成针对数据库420的配置细节465和请求460。例如,配置细节465可以指定应当通过克隆/复制到附加计算机来水平缩放数据库420。在实施例中,配置细节465可以指定调整以下中的至少一个DBMS调谐参数:打开数据库游标(open database cursor)的最大计数、连接会话的最大计数、会话相关变量的存储器存储的最大尺寸、会话无关变量的存储器存储的最大尺寸,或用于处理查询的并行性的量(例如CPU核心)。配置细节465还可以具有用于DBMS调谐的其它可调属性,诸如:事务日志的尺寸、数量和主机,恢复的检查点间隔,执行计划高速缓存(也称为过程高速缓存)的尺寸,数据分区的尺寸、键和主机,列索引或编码,查询高速缓存的尺寸,行高速缓存的尺寸,数据块高速缓存的尺寸,事务锁定粒度,日志缓冲区尺寸,日志刷新频率或粒度,视图实质化,索引重建,索引数据结构,模式非规范化,复制量和高速缓存受害者策略。配置细节465还可以具有用于表空间调谐的可调属性,诸如尺寸和数量、存储器的量、撤消表空间尺寸、扩展尺寸和预取尺寸。如本文稍后所讨论的,模板机制可以生成请求460和/或配置细节465。

[0140] 在步骤503中,规则引擎将请求发送到请求代理。例如,规则引擎440可以向请求代理470发送请求460。针对相关的美国申请16/055,468中的图4-6讨论了向请求代理发送请求的示例。理想地,请求代理470最初缓冲或最小程度地处理请求460,使得可以快速吸收入

站请求的需求尖峰,并且不会对规则引擎440或诸如DBA之类的交互式用户造成网络背压。例如,与请求460的摄入相关的一些处理可以或可以不推迟到下一步骤504。

[0141] 在步骤504中,请求代理基于该请求创建并分配异步作业。例如,请求代理470可以基于配置细节465生成异步作业480。在一些实施例中,异步作业480可以是诸如python或JavaScript之类的逻辑脚本或者Java作业,诸如以下中的至少一种:由Java.util.concurrent包定义的Runnable、Callable或Future。可以将异步作业480的执行推迟到计划的时间,诸如时间维护窗口,直到执行资源变得可用(诸如池化的线程),或者直到数据库420的事务负载恰好降到阈值以下。

[0142] 在步骤505中,异步作业基于配置细节来配置数据库。例如,异步作业480最终执行,其根据配置细节465来重新配置数据库420。例如,异步作业480可以调用数据库420的存储的过程以引起重新配置。

[0143] 6.3连锁反应

[0144] 异步作业480可以在执行之后进行各种后处理,诸如评估DBMS 410和/或数据库420的系统状态和条件,这可能导致(一个或多个)附加事件被发送到规则引擎440。例如,异步作业480可以创建数据库420,这引起指示在本地驱动器上剩余多少可用盘空间的事件,该事件可以触发或可以不触发其它规则激活(即,被应用)。异步作业480可以发送指示异步作业480的成功或不成功完成的事件,这可以触发或可以不触发其它规则激活。

[0145] 例如,在异步作业480配置数据库420之后,异步作业480可以发起生成指示以下中的至少一项的附加事件的数据库条件检查:不同查询的执行统计、DML写的执行统计、DDL命令的执行统计、特定查询的执行统计、数据库的完整性检查的结果、数据库的资源的使用超过阈值或数据库的资源的可用性低于阈值。在相关的美国专利9,692,819中给出了用于自动监视远程系统健康的实时遥测技术。

[0146] 7.0机器学习

[0147] 图6是描绘实施例中的示例云600的框图。为了容器数据库及其可插拔数据库的最优管理,云600使用机器学习来自动调谐规则。云600可以是云100的实现方式。云600包括机器学习引擎630、规则引擎640和容器数据库(诸如610)。

[0148] 7.1优化算法

[0149] 机器学习引擎630包含使用自适应调谐来生成改进的规则的优化算法635,诸如671-672。算法635可以是以下中的至少一项:最近邻、人工神经网络、线性回归或逻辑回归。算法635可以生成针对特定云客户(诸如数据库租户650)和/或特定服务水平协定(SLA)(诸如660)自定义调谐的规则。例如对于豪华租户,机器学习引擎630可以生成如下规则:规则将可插拔数据库620视为弹性的,并且在高需求时将可插拔数据库620复制到其它容器数据库。而对于经济租户,机器学习引擎630可以生成仅在高需求期间保留更多盘空间以容纳更大的重做日志(redo log)的规则。

[0150] 虽然未示出,但是机器学习引擎630可以周期性地或连续地从云600和/或数据库610和620接收事实、测量和统计,以用作算法635的训练输入,如本文稍后讨论的。因此,机器学习引擎630可以周期性地或连续地重新生成规则,诸如671-672,这些规则可以被发送到规则引擎640以补充或替换规则引擎640先前加载的其它规则。因此,规则引擎640可以具有根据算法635的深度学习和训练而改进的演进的规则库。因此,随着更多的事实和测量被

提供给机器学习引擎630,根据训练算法635生成的不断演进的规则,云600的数据库的配置将更好。

[0151] 7.2规则API

[0152] 规则引擎640可以具有许多规则,包括一些人工制作的规则和其它自动生成的规则。因此,规则引擎640的规则库的管理可能或多或少是复杂的。规则引擎640可以具有用于规则管理自动化的应用编程接口(API)。例如,规则API可以支持诸如生成的和/或人工制作的规则的创建、读取、更新和删除(CRUD)的操作。规则API可以具有用于将事件注入到规则引擎640中以进行规则处理的操作。

[0153] 规则API可以被实现为远程过程调用(RPC)、Web服务(诸如REST和/或公共网关接口(CGI)post)、存储的过程或其它接口协议。

[0154] 8.0训练

[0155] 图7是描绘实施例中的示例云700的框图。云700用历史事件进行训练以改进规则。云700可以是云100的实现方式。云700包括机器学习引擎740、规则引擎770、发布-订阅总线720和DBMS(诸如711-712)。

[0156] 8.1事件交付

[0157] DBMS 711-712发出事件,诸如731-733。但是,事件没有直接交付到消费者,诸如规则引擎770和机器学习引擎740。代替地,事件由将事件中继到感兴趣的(一个或多个)消费者的发布-订阅总线720来传达。发布-订阅总线720可以是企业服务总线(ESB)、发布-订阅代理或其它形式的事件队列。在实施例中,将事件按专用于事件源或事件类型的主题进行分类,诸如按可插拔数据库、按容器数据库、按客户或以其它方式。感兴趣的消费者(诸如引擎740和770)可以订阅感兴趣的主题以接收相关事件。在实施例中,每个客户具有专用的规则引擎(诸如770),其通过发布-订阅总线720订阅特定于客户的事件主题。

[0158] 发布-订阅总线720可以仅将事件中继给(一个或多个)订户。例如,事件733被中继而没有对规则引擎770的附加传达,该规则引擎可以匹配引发(一个或多个)异步作业的规则引擎770的规则。发布-订阅总线720也可以持久地将事件记录到事件历史725中。在实施例中,事件历史725是所记录的事件的数据库。例如,事件的关系表可以具有指示每个事件的来源和性质的列,诸如通过识别以下各项之一:客户、可插拔数据库、容器数据库、事件类型、事件参数或时间戳。

[0159] 8.2历史重放

[0160] 诸如机器学习引擎740之类的事件消费者可以向发布-订阅总线720订阅实时事件或检索事件历史725中已记录的事件。例如,可以从事件历史725中检索事件731,以训练机器学习引擎740。诸如732-733之类的实时事件可以差不多实时地中继到诸如引擎740和770之类的消费者,在它们通过发布-订阅总线720时将其记录或不记录在事件历史725中。因此,可以基于来自事件历史725的大批事件来对机器学习引擎740进行批量训练,或者可以基于现场事件流自发地且增量地对机器学习引擎740进行训练。

[0161] 8.3加权的属性

[0162] 每个事件(诸如731)可以具有属性(诸如791-792),这些属性指示在生成事件时存在的单独的操作条件。例如,属性791可以诸如根据重做日志的尺寸来指示积压深度。例如,属性792可以指例如CPU负载、一天中的时间、事件类型或事件源。其它事件可以具有相同

的或其它属性。一些属性可以比其它属性更重要。机器学习引擎740具有根据重要性对相应属性791-792的值进行缩放的权重751-752。因此,属性791的值和权重751的值的乘积可以产生缩放的属性值,该属性值可以在训练期间作为输入被注入到学习算法中。权重751-752可以是可调的。例如,训练可以优化权重751-752。

[0163] 8.4训练置信度

[0164] 已训练好的算法的准确性可以取决于发生了多少训练。因此,在训练期间消耗的事件越多,那么由机器学习引擎740生成的规则(诸如782)就越适合。机器学习引擎740的置信度得分760随着机器学习引擎740在训练期间消耗事件而增加。在实施例中,当置信度得分760超过阈值765时,训练完成。在另一个实施例中,当机器学习引擎740已经消耗了所有训练事件并且迭代学习算法收敛时,训练完成。

[0165] 8.5重新训练

[0166] 阈值765还可以用于重新训练,其中机器学习引擎740消耗附加的事件以进行附加的调谐,诸如权重751-752的附加优化。当置信度得分760超过阈值765时,重新训练结束,该阈值可能具有与初始训练不同的值。每次重新训练开始时,可以降低置信度得分760或将其重置为零和/或可以增加阈值765。在实施例中,重新训练是连续的,并且在每个新事件到达时发生。

[0167] 8.6保留优先级

[0168] 当训练或重新训练结束时,机器学习引擎740重新生成规则,诸如782。规则引擎770可以获得并加载重新生成的规则。后续规则的加载可以补充或替换先前加载的规则。诸如DBA 701之类的客户端应用或人类用户可以将诸如781之类的人工制作的规则加载到规则引擎770中。因此,规则引擎770可以具有人工和自动起源的规则的混合。可以在规则引擎770内为人工制作的规则赋予任期,使得仅当置信度得分760超过阈值(诸如765)时,人工制作的规则781才由生成的规则782替换。

[0169] 在实施例中,机器学习(ML)引擎740订阅其规则推荐模型(ML模型)因训练所需要的所有事件类型。机器学习引擎740采用各种ML技术,包括k最近邻、神经网络和线性回归,以基于订阅事件中包括的属性列表构建各种规则推荐模型。每个规则推荐模型都基于历史事件数据调整指派给订阅事件的不同属性的权重。由于系统中存在多个租户数据库,因此机器学习引擎740对事件数据执行跨租户分析以提高其模型置信度。随着被分析事件数量的增加,模型的置信度得分随时间提高。推荐的规则包括条件和动作/作业,如上面所解释的。推荐的作业可以是自调谐作业或者自维护作业。

[0170] 在实施例中,云700中的每个数据库租户可以指定以下两个置信度得分阈值。这包括基本置信度阈值,使得一旦规则推荐模型的置信度得分超过这个阈值,就生成机器得到的规则并将其发布到规则引擎770中。当置信度随时间提高时,机器得到的规则被更新。这还包括覆盖置信度阈值,如果机器得到的规则与人工制作的规则之间发生冲突,那么仅当其置信度超过覆盖置信度阈值时,机器得到的规则才会覆盖人工制作的规则。

[0171] 8.7示例技术堆栈

[0172] 以下是可以是本文给出的任何云的实现方式的特定实施例。这个实施例是针对云400和700而定制的,并参考云400和700进行讨论。

[0173] 请求代理470包括WebLogic Java-2企业版(J2EE)应用服务器,其包括接受用于请

求提交和管理的REST API的Web服务器。REST API是用SmartBear的Swagger OpenAPI实现的。客户端可以使用CURL或Web浏览器。因此,请求可以是交互式的、脚本化的或动态生成的,诸如请求460。异步作业(诸如480)通过以下任一种或两种方式实现:Java消息服务(JMS)和java.util.concurrent Java程序包(例如线程池、执行器和计算未来)。通过发布/订阅总线720进行的事件分发是用JMS发布/订阅消息主题实现的,其中诸如711-712之类的(一个或多个)DBMS作为发布者并且规则引擎770作为订户。规则引擎770是使用Mustache模板引擎实现的。稍后本文将在图8中进一步讨论模板化。机器学习引擎740是通过机器学习库(诸如Apache Spark MLlib)实现的,该库具有诸如逻辑回归之类的预装算法。

[0174] 这个特定实施例具有以下好处。这个实施例可水平缩放以用于:a) 诸如多核的对称多处理(SMP),和/或b) 诸如集群或云计算之类的分布式计算。这个实施例仅需要一个代码库(例如,纯Java)用于异构计算(例如,广域网、WAN)。

[0175] 9.0规则干扰

[0176] 图8是描绘实施例中的示例云800的框图。云800避免了规则之间的干扰。云800可以是云100的实现方式。云800包括规则引擎810和诸如820之类的DBMS。

[0177] 云800不期望个别规则(诸如861-862)导致对于云800全局最佳的重新配置。代替地,规则861-862中的每一个可以识别对于特定可插拔数据库(诸如841-842)或租户而言不便的特定条件(诸如835)。同样,当应用个别规则(诸如861-862)时,该规则导致对可插拔数据库或租户有利、但不一定对整个云800有利的特定动作。

[0178] 9.1条件识别

[0179] 例如,原因或条件835可以是在容器数据库830本地并且由可插拔数据库841-842共享的盘的存储容量的耗尽。DBMS 820可以检测磁盘差不多已满,并发出指示盘已满的一个事件或冗余的事件(诸如851-852)。在实施例中,针对原因或条件835的相同出现,为相应的可插拔数据库841-842生成事件851-852。在实施例中,可插拔数据库841-842中的每一个在共享或相应的规则引擎810中具有其自己单独的规则集。例如,规则861-862可以或多或少相似,但是与相应的可插拔数据库841-842相关联。在未示出的实施例中,规则861-862驻留在单独的规则引擎中,因为规则861-862与单独的租户相关联。

[0180] 在各种实施例中,规则引擎810可以是由以下中的任何一个共享的单个规则引擎:a) 一个容器数据库的一些或所有可插拔数据库、b) 同一DBMS的一些或所有容器数据库和可插拔数据库、c) 多个DBMS的一些或所有容器数据库和可插拔数据库(未示出),或d) 一些或所有租户(例如客户和/或应用)的一些或所有容器数据库和可插拔数据库。即使当单独的可插拔数据库具有单独的规则集时,也可以将那些规则集合并为组合的整体式规则库以在一个或多个规则引擎中使用。例如,规则861-862中的每一个可以在用于单独的可插拔数据库841-842的单独的规则集中被维护,然后被组合成整体式规则库以用于加载到相同的规则引擎中。在这种情况下,一些实施例可以从原因或条件835仅生成一个(而不是如图所示得两个)事件,并且该单个事件可以触发规则861-862中的任一个或两个。

[0181] 在实施例中,规则引擎810可以被复制(即,水平缩放),使得任何事件都可以被交付到规则引擎的任何副本,并且仍然针对相同的可插拔数据库激活(即,应用)相同的规则。在复制的实施例中并且取决于实现方式,可插拔数据库841可以或不固定到规则引擎810的特定实例。例如,可插拔数据库841可以先前由一个规则引擎实例管理,并且随后由具

有相同规则的另一个规则引擎实例管理。在实施例中,规则引擎810的水平缩放可以是弹性的,诸如用于重新配置需求尖峰,诸如在同时影响许多可插拔数据库的操作紧急情况期间。

[0182] 规则861-862中的每一个都具有与当前事件(诸如852)匹配或不匹配的条件模板(诸如870)。例如,条件模板870可以对事件852的属性可能满足或可能不满足的谓词逻辑或其它过滤表达式进行编码。例如,规则862可能需要千兆字节的空闲本地盘空间,使得当事件852指示空闲空间不超过千兆字节时满足条件模板870。

[0183] 9.2模板化

[0184] 当满足条件模板870时,通过应用动作模板880来应用规则862。例如,动作模板880可以生成配置描述符,从该配置描述符可以构造并提交异步作业890以最终执行。在实施例中,模板870和880中的任一个或两个都是文本的。在实施例中,所生成的配置描述符在动作模板880内显示为模板化的文本,从而实现配置描述符的所见即所得(WYSIWIG)生成。在实施例中,文本模板被聚合到诸如Mustache或Java服务器页面(JSP)之类的模板引擎中。在实施例中,使用诸如CGI post之类的HTTP将事件852交付到规则引擎810之一。

[0185] 9.3整体效率

[0186] 因此,如上所述,原因或条件835可以是由触发规则861-862的事件851-852指示的盘满。因此,两个规则861-862的动作模板可以响应于相同的盘满而执行。例如,规则861-862都可以提交诸如890之类的异步作业,这些异步作业使每个相应的可插拔数据库841-842移出容器数据库830并进入具有更多本地盘空间的不同容器数据库中。实际上,通过规则决定离开容器数据库830的两个可插拔数据库841-842都可以解决盘空间不足的问题。但是,移动两个可插拔数据库841-842可能在全局上是次优的。例如,可以通过代替地仅移动两个可插拔数据库841-842之一来节省时间和精力。如下在各种实施例中可以避免由有争议的规则的干扰引起的这种次优性。拥有多个租户和/或管理不同可插拔数据库、不同范围的数据库(例如容器和可插拔数据库)或不同租户和/或应用的数据库的规则会加剧规则之间的干扰。

[0187] 在实施例中,诸如890之类的异步作业自身避免了干扰。例如,规则861可以引发从容器数据库830中逐出可插拔数据库841的异步作业(未示出),从而释放一些本地盘空间,从而使得原因或条件835停止。最终,异步作业890执行,并且它可以检测是否满足各种前提条件,如果不满足,那么中止或重新计划。异步作业890可以检测本地盘空间是否已经足够。如果盘空间足够,那么异步作业890可以提前终止,从而使可插拔数据库842保留在容器数据库830内。在实施例中,异步作业890可以通过重新调用条件模板870来评估其前提条件。

[0188] 在实施例中,来自规则861的异步作业可以正常执行(即,逐出可插拔数据库841),然后使得指示本地盘空间充足的另一个事件被发送到规则引擎810。该附加事件可以使得规则引擎810按规则或以其它方式取消不再需要的异步作业890。

[0189] 在实施例中,规则引擎810直到可插拔数据库841被逐出之后才消耗事件852,在这种情况下,当事件852最终被处理时将不满足条件模板870。因此,将不应用动作模板880,并且将不创建异步作业890。规则861-862中的哪一个先于另一个被评估取决于实施例。在实施例中,事件851-852以与它们到达规则引擎810相同的次序被处理。在实施例中,可以为规则、可插拔数据库和租户各自指派等级或优先级。在实施例中,首先评估与高优先级相关联的规则,以避免会导致高优先级数据库有害地等待低优先级数据库做出反应的优先级倒

置。在实施例中,首先评估与低优先级相关联的规则,并且例如在将低优先级数据库逐出的同时,暂时降级该低优先级数据库的服务质量,从而最小化高优先级数据库的维护相关的服务降级。

[0190] 在实施例中,容器数据库830可以在规则引擎810中具有其自己的规则(未示出)。在实施例中,容器数据库830的规则可以具有高优先级并且在可插拔数据库841-842的规则之前执行,从而促进多个可插拔数据库841-842的集中协调。例如,容器数据库830的规则可以通过选择和逐出可插拔数据库841来对盘耗尽做出反应,从而集中解决问题并避免需要应用否则会造成协调问题的规则861-862,诸如如上面所讨论的产生次优结果的竞争条件。

[0191] 在实施例中,可插拔数据库和/或容器数据库可以各自具有差不多统一的生命周期,其中形式化的操作状态对应于重要维护场景的典型阶段。例如,可插拔数据库可以处于表示重新配置活动的过渡状态,诸如按比例放大、按比例缩小或重新定位。在(例如Java或C)实施例中,一个或多个枚举声明一个或多个生命周期的状态。诸如870和880之类的规则模板可以检查一个或多个可插拔数据库的操作状态。例如,可插拔数据库842的规则862可以:a)检测另一个可插拔数据库841当前正在被逐出(即,重新定位),并且b)拒绝激活,因为正在进行的重新定位可以很快解决盘耗尽的共同问题。在一些实施例中,如果规则由于检测到的过渡状态而拒绝激活,那么在以后的时间(例如,按计划或在过渡状态的后续改变时),可以重新评估拒绝的规则,和/或可以通过规则引擎重新处理原始事件。因此,多个可插拔数据库的规则可以实现某种程度的协调以实现全局最优结果。

[0192] 9.4示例管理和维护任务

[0193] 异步作业890可以执行各种数据库管理任务,诸如891-898中的一个或多个。在实施例中,任务891-898中的任何一个由数据库(诸如830和841-842)中存储的过程调用来实现,这些过程可以由异步作业890来调用。在实施例中,任务891-898中的任何一个都被硬编码到异步作业890的代码库或脚本中。

[0194] 任务891在容器数据库830内创建可插拔数据库,诸如841-842。可插拔数据库的创建可以被模板化,使得可以将原型可插拔数据库克隆到诸如830的容器数据库中,以创建诸如841-842之类的可插拔数据库实例。相关的美国专利15/266,902中给出了使用中央储存库自动供给可插拔数据库的基于规则的技术。

[0195] 任务892创建或使用数据库的快照或备份,诸如830或841-842。数据库快照被广泛用于快速创建数据的时间点虚拟副本。数据库备份防止存储介质故障。在数据库还原或恢复期间使用(消耗)备份或快照。在相关的美国申请15/254,884中给出了可插拔数据库的自动备份技术。

[0196] 任务893调整新会话可以连接到数据库(诸如830或841-842)的最大速率。这可以诸如在拒绝服务(DOS)攻击期间减轻登录风暴。

[0197] 任务894基于动态条件生成警报或事件。例如,异步作业890可以消耗磁盘空间来创建列索引,然后生成事件(诸如851)以指示剩余多少空闲盘空间。这会导致应用诸如861之类的规则。

[0198] 任务895调整重做日志或快速恢复区域的尺寸。例如,重做日志可以扩展以容纳更多的进行中的事务,诸如需求尖峰期间。

[0199] 任务896将软件升级或补丁应用于数据库(诸如830或841-842),或应用于DBMS(诸

如820)。补丁可以创建或修改包含逻辑或数据的文件,可以执行Shell或DDL或DML脚本,和/或可以调用存储的过程。在相关的美国专利9,396,220和美国专利申请15/215,446中给出了将可插拔数据库从一个容器数据库自动重新定位到另一个容器数据库的技术。在相关的美国申请15/266,917中给出了自动升级可插拔数据库的技术。

[0200] 任务897调整分配给数据库(诸如830或841-842)的至少一种资源(诸如CPU、持久性存储装置或RAM)的量。例如,可以在需求尖峰期间扩展盘块的易失性高速缓存。任务897还可以影响诸如特定持续时间或直到特定条件的临时分配,此后资源分配恢复为先前的量。

[0201] 任务898将可插拔数据库从第一计算机上的第一容器数据库重新定位或克隆到相同或不同的计算机上的第二容器数据库。例如,异步作业890可以为可插拔数据库841分配更多的处理核心,比容器数据库830的可用核心还要多。因此,异步作业890还可以将可插拔数据库841-842之一移动到不同计算机上的不同容器数据库。在相关的美国专利申请15/215,435中给出了用于克隆可插拔数据库的技术。

[0202] 10.0示例自主调谐处理

[0203] 图9是描绘实施例中的用于更快的查询执行的示例数据库调谐处理的流程图。参考图8讨论图9。

[0204] 10.1统计

[0205] 步骤901-903是准备性的。他们收集并分析所需的统计,以决定如何调谐数据库以获得更好的查询性能。在步骤901中,搜集数据库对象的数据分布和/或(一个或多个)存储特性的测量。例如,DBMS 820可以监视和记录与查询语句、查询表达式、表、分区和/或列相关联的资源使用和/或数据模式。

[0206] 在步骤902中,基于测量来生成统计。例如,可以生成列或表达式的值的直方图。可以生成基于查询表达式或存储工件(例如列或表)的CPU、存储器或I/O使用情况的热图。可以检测列或表达式之间的数据相关性。

[0207] 10.2谓词选择性

[0208] 在步骤903中,基于统计来计算查询谓词的选择性。选择性促进差不多准确估计结果集尺寸以及结果集行计数与源表行计数的比率,这改善了查询处理期间的查询计划。选择性对于级联连接或级联过滤项的重新排序以积极修剪中间结果特别有帮助。例如,即使两个列都需要过滤,首先在具有更多不同值的列上进行过滤也会比首先在具有更少不同值的列上进行过滤更高效。

[0209] 作为动作的过程,步骤904分析潜在步骤906-912的子集。潜在步骤906-912中的每一个是用于数据库的特定重新配置技术,其可以改善未来的查询性能。每种潜在的重新配置招致相应预期的重新配置成本,并且会产生相应的预期加速。重新配置成本可以基于重新配置期间的临时资源消耗(例如CPU、I/O、存储器)以及重新配置期间的临时服务降级。后台重新配置的成本可以比临时锁定数据库对象的重新配置的成本低。例如,创建列索引(906)的成本可以比将优化器提示插入查询(911)的成本高。而列索引可以预期比优化器提示更能加速查询,这可以取决于所涉及的数据访问方法,诸如表扫描与索引遍历。在步骤904中,基于所计算的选择性来计算每个备选数据库优化机制的各自的估计成本。

[0210] 10.3成本-效益分析

[0211] 步骤905是决定性的。根据成本-效益分析,选择906-912的一个(或多个,如果兼容的话)潜在优化。在实施例中,步骤904-905(和906-912)中的任何一个可以被实现为规则引擎810的规则。例如,当特定查询、表达式、表或列的重复使用超过阈值时,会发生原因或条件835。例如,事件852可以指示在使用热图中某一列已经变热。因此,当条件模板870识别出热列满足各种标准时,可以触发步骤904。在步骤905,选择潜在步骤906-912中的一个或多个,并将其合并或不合并为一个或多个异步作业。因此,云800可以针对任何或所有查询自主地优化可插拔数据库841或842。

[0212] 10.4示例调谐场景

[0213] 特别地,步骤906创建或删除基于表的一个或多个列的数据库索引。步骤907诸如用字典或其它压缩对列进行编码或解编码。步骤908诸如通过添加备用数据块来调整表空间或数据库文件的尺寸。步骤909实质化或去实质化数据库视图或虚拟列。步骤910可以对列或表进行碎片整理,或者将列或表预加载(预取)到RAM中。例如,列或表可以预加载到高速缓存中,或者加载并固定在诸如堆之类的存储器中。步骤911可以预留优化器提示以用于重复插入到后续查询中。步骤912可以将优化(或引起优化的统计)从一个数据字典传播到另一个数据字典。因此,优化可以分散到副本或碎片中,并有助于弹性缩放,而无需人工干预。在相关的美国申请15/215,443和15/266,030中给出了自动传播可插拔数据库之间的改变的技术。

[0214] 因此,可以经由调谐由规则引擎810生成的作业来自动优化云数据库以改善查询性能。以下是用于特征调谐的特定场景。

[0215] 可以调谐资源参数,诸如Oracle的系统全局区域(SGA)、程序全局区域(PGA)、会话和最大打开游标。SGA是读/写存储器区域,它与Oracle后台进程一起形成数据库实例。代表用户执行的所有服务器进程都可以读取实例SGA中的信息。在数据库操作期间,若干进程写到SGA。服务器进程和后台进程不驻留在SGA中,而是存在于单独的存储器空间中。每个数据库实例都有自己的SGA。Oracle在实例启动时自动为SGA分配存储器,并在实例关闭时回收存储器。PGA是特定于操作进程或线程的存储器,系统上其它进程或线程不共享该存储器。因为PGA是特定于进程的,因此永远不会在SGA中进行分配。PGA是存储器堆,其包含专用或共享的服务器进程所需的会话相关变量。服务器进程分配其在PGA中所需的存储器结构。PGA的类比是文件文员使用的临时台面工作空间。以此类推,文件文员是代表客户进行工作的服务器进程(客户端进程)。文员清理工作台面的一部分、使用工作空间存储有关客户请求的细节并对客户请求的文件夹进行排序,然后在完成工作时放弃空间。初始化参数可以设置实例PGA的目标最大尺寸。各个PGA可以根据需要增长到这个目标尺寸。

[0216] 异步调谐作业可以依赖于已经搜集的数据库对象统计。基于成本的优化方法使用统计来计算谓词的选择性并估计每个执行计划的成本。选择性是SQL语句的谓词挑选的表中的行的一部分。优化器使用谓词的选择性来估计特定访问方法的成本并确定最优加入次序。统计量化表、列、索引和分区的数据分布和存储特性。优化器使用这些统计来估计使用特定执行计划执行SQL语句需要多少I/O和存储器。统计被存储在数据字典中,并且它们可以从一个数据库导出并导入到另一个数据库中(例如,将生产统计传送到测试系统以模拟真实环境,即使测试系统可能只具有少量样本的数据)。应当定期搜集统计,以向优化器提供关于模式对象(schema object)的信息。在以使得之前的统计不准确的方式修改模式对

象的数据或结构之后,应当收集新的统计。例如,在将大量的行加载到表中之后,应当收集关于行数的新统计。在更新表中的数据之后,虽然可能需要有关平均行长度的新统计,但无需收集关于行数的新统计。

[0217] 可以调谐初始化参数,诸如`cursor_sharing`。私有SQL区域保存关于已解析的SQL语句的信息以及其它特定于会话的信息以供处理。当服务器进程执行SQL或PL/SQL代码时,该进程使用私有SQL区域存储绑定变量值、查询执行状态信息和查询执行工作区。用于语句的每次执行的私有SQL区域不是共享的并且可以包含不同的值和数据。游标是具体私有SQL区域的名称或句柄。游标包含特定于会话的状态信息,诸如绑定变量值和结果集。游标差不多是客户端侧上指向服务器侧上的状态的指针。因为游标与私有SQL区域紧密关联,所以有时可以互换使用这些术语。

[0218] 在相关的美国专利8,903,801中给出了用于关系数据库的自动性能调谐的技术。在相关的美国专利8,341,178中给出了用于关系数据库的自动性能分析的技术。

[0219] 11.0数据库系统概述

[0220] DBMS管理一个或多个数据库。DBMS可以包括一个或多个数据库服务器。数据库包括存储在持久性存储器机制(诸如一组硬盘)上的数据库数据和数据库字典。数据库数据可以存储在一个或多个数据容器中。每个容器包含记录。每条记录内的数据被组织到一个或多个字段中。在关系型DBMS中,数据容器被称为表,记录被称为行,并且字段被称为列。在面向对象的数据库中,数据容器被称为对象类,记录被称为对象,并且字段被称为属性。其它数据库架构可以使用其它术语。

[0221] 用户通过向数据库服务器提交使数据库服务器对存储在数据库中的数据执行操作的命令来与DBMS的数据库服务器交互。用户可以是在客户端计算机上运行的与数据库服务器交互的一个或多个应用。多个用户在本文中也可以统称为用户。

[0222] 数据库命令可以采用符合数据库语言的数据库语句的形式。用于表达数据库命令的数据库语言是结构化查询语言(SQL)。存在许多不同的版本的SQL,一些版本是标准版本,一些是专有版本,并且具有各种扩展。数据定义语言(“DDL”)命令被发到数据库服务器以创建或配置数据库对象,诸如表、视图或复杂数据类型。SQL/XML是在对象关系数据库中操纵XML数据时使用的SQL的常见扩展。

[0223] 多节点数据库管理系统由互连的节点组成,这些节点共享对相同的一个或多个数据库的访问。典型地,节点经由网络互连,并在不同程度上共享对共享存储装置的访问,例如对一组盘驱动器和存储在其上的数据块的共享访问。节点之间不同程度的共享访问可以包括什么都不共享、共享所有内容、按节点对数据库分区进行独占访问或其某种组合。多节点数据库系统中的节点可以是经由网络互连的一组计算机(例如,工作站、个人计算机)的形式。可替代地,节点可以是网格的节点,该网格由与机架上的其它服务器刀片互连的服务器刀片形式的节点组成。

[0224] 多节点数据库系统中的每个节点都托管数据库服务器。服务器(诸如数据库服务器)是集成软件组件和计算资源(诸如存储器、节点以及节点上的用于在处理器上执行集成软件组件的进程)的分配的组合、专用于代表一个或多个客户端执行特定功能的软件和计算资源的组合。

[0225] 可以分配来自多节点数据库系统中的多个节点的资源,以运行特定的数据库服务

器的软件。软件和节点中的资源的分配的每种组合都是在本文中被称为“服务器实例”或“实例”的服务器。数据库服务器可以包括多个数据库实例,其中的一些或全部在单独的计算机(包括单独的服务器刀片)上运行。

[0226] 12.0可插拔数据库和容器数据库系统概述

[0227] 数据库整合涉及在多个数据库之间分配和共享计算资源。可以使用容器数据库管理系统来整合数据库。整合的数据库(诸如多租户容器数据库(CDB))包括一个或多个可插拔数据库(PDB)。

[0228] 容器数据库包括数据字典,数据字典包含定义容器数据库中的数据库对象的元数据。例如,给定CDB的数据字典将包含描述给定CDB中包含的每个PDB的元数据,包括每个PDB中包含的数据库对象。此外,每个可插拔数据库包括特定于PDB的数据库字典,该数据库字典包含定义可插拔数据库中包含的数据库对象的元数据。数据库对象包括表、表列、索引、文件、表空间、数据类型、用户、用户特权以及用于存储数据库对象数据的存储结构等。

[0229] 容器数据库可以管理多个可插拔数据库,并且给定的数据库服务器实例可以管理容器数据库中的那些可插拔数据库并为其提供服务。由此,给定的容器数据库允许多个可插拔数据库在同一数据库服务器和/或数据库服务器实例上运行,从而允许单个数据库服务器或实例的计算资源在多个可插拔数据库之间共享。在容器数据库管理系统中,可以在容器数据库中独立于其它可插拔数据库打开或关闭每个可插拔数据库。

[0230] 应用可以通过在容器数据库管理系统上为可插拔数据库建立数据库会话来访问该可插拔数据库,其中数据库会话表示应用和容器数据库管理系统之间用于访问可插拔数据库的连接。通过例如将对于新连接的请求传输到容器数据库管理系统来发起用于可插拔数据库的数据库会话,该请求指定了可插拔数据库。响应于这样的请求,容器数据库管理系统建立所请求的数据库会话。容器数据库管理系统可以托管多个数据库会话,每个数据库会话用于多个可插拔数据库之一。

[0231] 基于适用于容器数据库的那些数据库管理系统(DBMS)特征的要求,配置给定的容器数据库。适用于容器数据库的DBMS特征是与容器数据库交互或影响容器数据库的特征,因此需要对容器数据库进行某种配置。可能适用于给定容器数据库的DBMS特征包括以下中的一项或多项:管理容器数据库的DBMS的版本(包括主版本、次版本和/或补丁级别);可以为容器数据库安装或实现的可选特征(诸如数据加密、允许在数据库的区域内进行多个级别的数据限制的特征、本地化启用);容器数据库中存在的普通用户;已为管理容器数据库的DBMS安装的独立安装的补丁;等等。

[0232] CDB的配置包含CDB的基于适用于CDB的DBMS特征被调整的方面。CDB的这些方面包括以下中的一项或多项:存储在CDB的可插拔数据库中的数据或数据库对象的结构;CDB的底层操作系统文件的布局或内容;由CDB要求的后台进程的数量;与CDB相关联的标识符;CDB功能所需的变量;初始化参数;用于对CDB中的数据进行编码的字符集;由CDB支持的时区;标准数据库块大小;表空间设置;撤消设置;由CDB支持的服务;为CDB实现的特殊特征;CDB的数据库服务器实例集群支持;等等。

[0233] 可插拔数据库可以被“插入”到容器数据库中,并且可以在数据库服务器和/或数据库管理系统之间传输。数据库服务器实例通过将描述可插拔数据库的元数据包括在容器数据库的数据库字典中,并通过将可插拔数据库作为容器数据库的一部分发起管理来将可

插拔数据库插入到容器数据库中。特定可插拔数据库被插入到其中的容器数据库的配置的任何数量的方面都会影响可插拔数据库。

[0234] 当可插拔数据库从源容器数据库移动到目的地容器数据库(其中目的地数据库和源容器数据库具有相同的配置)时,在使可插拔数据库可用于目的地容器数据库处的操作之前,无需重新配置可插拔数据库。但是,重新定位可插拔数据库的源容器数据库和目的地容器数据库并非始终针对同一组DBMS特性进行配置。当将可插拔数据库移动到具有与可插拔数据库的源容器数据库不同配置的目的地容器数据库时,重新定位的可插拔数据库被重新配置为符合目的地容器数据库的配置,这允许将重新定位的可插拔数据库配置为与适用于目的地容器数据库的DBMS特征兼容。

[0235] 13.0硬件概述

[0236] 根据一个实施例,本文描述的技术由一个或多个专用计算设备实现。专用计算设备可以被硬连线以执行这些技术,或者可以包括数字电子设备(诸如被持久地编程为执行这些技术的一个或多个专用集成电路(ASIC)或现场可编程门阵列(FPGA)),或者可以包括被编程为根据固件、存储器、其它存储装置或组合中的程序指令执行这些技术的一个或多个通用硬件处理器。这种专用计算设备还可以将定制的硬连线逻辑、ASIC或FPGA与定制编程相结合,以实现这些技术。专用计算设备可以是台式计算机系统、便携式计算机系统、手持设备、联网设备或者结合硬连线和/或程序逻辑以实现这些技术的任何其它设备。

[0237] 例如,图10是图示可以在其上实现本发明实施例的计算机系统1000的框图。计算机系统1000包括总线1002或用于传送信息的其它通信机制,以及与总线1002耦合以处理信息的硬件处理器1004。硬件处理器1004可以是例如通用微处理器。

[0238] 计算机系统1000还包括耦合到总线1002的主存储器1006,诸如随机存取存储器(RAM)或其它动态存储设备,用于存储将由处理器1004执行的信息和指令。主存储器1006还可以用于存储在由处理器1004执行的指令期间的临时变量或其它中间信息。当存储在处理器1004可访问的非瞬态存储介质中时,这些指令使计算机系统1000成为被定制以执行指令中指定的操作的专用机器。

[0239] 计算机系统1000还包括耦合到总线1002的只读存储器(ROM)1008或其它静态存储设备,用于存储用于处理器1004的静态信息和指令。提供存储设备1010(诸如磁盘或光盘)并耦合到总线1002,用于存储信息和指令。

[0240] 计算机系统1000可以经由总线1002耦合到显示器1012(诸如阴极射线管(CRT)),用于向计算机用户显示信息。包括字母数字键和其它键的输入设备1014耦合到总线1002,用于将信息和命令选择传送到处理器1004。另一种类型的用户输入设备是光标控件1016(诸如鼠标、轨迹球或光标方向键),用于将方向信息和命令选择传送到处理器1004并用于控制显示器1012上的光标移动。这种输入设备通常在两个轴上具有两个自由度,第一轴(例如,x)和第二轴(例如,y),这允许设备指定平面中的位置。

[0241] 计算机系统1000可以使用定制的硬连线逻辑、一个或多个ASIC或FPGA、固件和/或程序逻辑(它们与计算机系统相结合,使计算机系统1000成为或将计算机系统1000编程为专用机器)来实现本文所述的技术。根据一个实施例,响应于处理器1004执行主存储器1006中包含的一个或多个指令的一个或多个序列,计算机系统1000执行本文的技术。这些指令可以从另一个存储介质(诸如存储设备106)读入到主存储器1006中。主存储器1006中包含

的指令序列的执行使得处理器1004执行本文所述的处理步骤。在替代实施例中,可以使用硬连线的电路系统代替软件指令或与软件指令组合。

[0242] 如本文使用的术语“存储介质”是指存储使机器以特定方式操作的数据和/或指令的任何非瞬态介质。这种存储介质可以包括非易失性介质和/或易失性介质。非易失性介质包括例如光盘或磁盘,诸如存储设备106。易失性介质包括动态存储器,诸如主存储器1006。存储介质的常见形式包括例如软盘、柔性盘、硬盘、固态驱动器、磁带或任何其它磁数据存储介质、CD-ROM、任何其它光学数据存储介质、任何具有孔图案的物理介质、RAM、PROM和EPROM、FLASH-EPROM、NVRAM、任何其它存储器芯片或盒式磁带。

[0243] 存储介质不同于传输介质但可以与传输介质结合使用。传输介质参与在存储介质之间的信息传送。例如,传输介质包括同轴电缆、铜线和光纤,包括包含总线1002的导线。传输介质也可以采用声波或光波的形式,诸如在无线电波和红外数据通信期间生成的那些。

[0244] 各种形式的介质可以参与将一个或多个指令的一个或多个序列传送到处理器1004以供执行。例如,指令最初可以被携带在远程计算机的磁盘或固态驱动器上。远程计算机可以将指令加载到其动态存储器中,并使用调制解调器通过电话线发送指令。计算机系统1000本地的调制解调器可以在电话线上接收数据并使用红外发送器将数据转换成红外信号。红外检测器可以接收红外信号中携带的数据,并且适当的电路系统可以将数据放在总线1002上。总线1002将数据传送到主存储器1006,处理器1004从主存储器1006检索并执行指令。由主存储器1006接收的指令可以可选地在由处理器1004执行之前或之后存储在存储设备106上。

[0245] 计算机系统1000还包括耦合到总线1002的通信接口1018。通信接口1018提供耦合到网络链路1020的双向数据通信,其中网络链路1020连接到本地网络1022。例如,通信接口1018可以是集成服务数字网(ISDN)卡、电缆调制解调器、卫星调制解调器或者提供与对应类型的电话线的数据通信连接的调制解调器。作为另一个示例,通信接口1018可以是局域网(LAN)卡,以提供与兼容LAN的数据通信连接。还可以实现无线链路。在任何此类实现中,通信接口1018都发送和接收携带表示各种类型信息的数字数据流的电信号、电磁信号或光信号。

[0246] 网络链路1020通常通过一个或多个网络向其它数据设备提供数据通信。例如,网络链路1020可以提供通过本地网络1022到主计算机1024或到由互联网服务提供商(ISP)1026操作的数据设备的连接。ISP 1026进而通过全球分组数据通信网络(现在通常称为“互联网”1028)提供数据通信服务。本地网络1022和互联网1028都使用携带数字数据流的电信号、电磁信号或光信号。通过各种网络的信号以及网络链路1020上并通过通信接口1018的信号(其将数字数据携带到计算机系统1000和从计算机系统1000携带数字数据)是传输介质的示例形式。

[0247] 计算机系统1000可以通过(一个或多个)网络、网络链路1020和通信接口1018发送消息和接收数据,包括程序代码。在互联网示例中,服务器1030可以通过互联网1028、ISP 1026、本地网络1022和通信接口1018发送对应用程序的所请求代码。

[0248] 接收到的代码可以在被接收到时由处理器1004执行,和/或存储在存储设备1010或其它非易失性存储器中,以供稍后执行。

[0249] 14.0软件概述

[0250] 图11是可以用于控制计算机系统1000的操作的基本软件系统1100的框图。软件系统1100及其组件,包括它们的连接、关系和功能,仅仅是示例性的,并且不意味着限制(一个或多个)示例实施例的实现。适于实现(一个或多个)示例实施例的其它软件系统可以具有不同的组件,包括具有不同的连接、关系和功能的组件。

[0251] 提供软件系统1100用于指导计算机系统1000的操作。可以存储在系统存储器(RAM)1006和固定存储装置(例如,硬盘或闪存)106上的软件系统1100包括内核或操作系统(OS)1110。

[0252] OS 1110管理计算机操作的低级方面,包括管理进程的执行、存储器分配、文件输入和输出(I/O)以及设备I/O。表示为1102A、1102B、1102C...1102N的一个或多个应用可以被“加载”(例如,从固定存储装置106传送到存储器1006中)以供系统1100执行。意图在计算机系统1000上使用的应用或其它软件也可以被存储为可下载的计算机可执行指令集,例如,用于从互联网位置(例如,Web服务器、app商店或其它在线服务)下载和安装。

[0253] 软件系统1100包括图形用户界面(GUI)1115,用于以图形(例如,“点击”或“触摸手势”)方式接收用户命令和数据。进而,这些输入可以由系统1100根据来自操作系统1110和/或(一个或多个)应用1102的指令来操作。GUI 1115还用于显示来自OS 1110和(一个或多个)应用1102的操作结果,用户可以提供附加的输入或终止会话(例如,注销)。

[0254] OS 1110可以直接在计算机系统1000的裸硬件1120(例如,(一个或多个)处理器1004)上执行。可替代地,管理程序或虚拟机监视器(VMM)1130可以插入在裸硬件1120和OS 1110之间。在这个配置中,VMM 1130充当OS 1110与计算机系统1000的裸硬件1120之间的软件“缓冲”或虚拟化层。

[0255] VMM 1130实例化并运行一个或多个虚拟机实例(“客人机”)。每个客人机包括“客人”操作系统(诸如OS 1110),以及被设计为在客户操作系统上执行的一个或多个应用(诸如(一个或多个)应用1102)。VMM 1130向客人操作系统呈现虚拟操作平台并管理客人操作系统的执行。

[0256] 在一些实例中,VMM 1130可以允许客人操作系统如同其直接在计算机系统1000的裸硬件1120上运行一样运行。在这些实例中,被配置为直接在裸硬件1120上执行的客人操作系统的相同版本也可以在VMM 1130上执行而无需修改或重新配置。换句话说,VMM 1130可以在一些情况下向客人操作系统提供完全硬件和CPU虚拟化。

[0257] 在其它实例中,客人操作系统可以被专门设计或配置为在VMM 1130上执行以提高效率。在这些实例中,客人操作系统“意识到”它在虚拟机监视器上执行。换句话说,VMM 1130可以在某些情况下向客户操作系统提供半虚拟化。

[0258] 计算机系统进程包括硬件处理器时间的分配,以及存储器的分配(物理和/或虚拟)、用于存储由硬件处理器执行的指令的存储器的分配,用于存储由硬件处理器执行指令所生成的数据,和/或用于当计算机系统进程未运行时在硬件处理器时间的分配之间存储硬件处理器状态(例如,寄存器的内容)。计算机系统进程在操作系统的控制下运行,并且可以在计算机系统上执行的其它程序的控制下运行。

[0259] 15.0云计算

[0260] 本文一般地使用术语“云计算”来描述计算模型,该计算模型使得能够按需访问计算资源的共享池,诸如计算机网络、服务器、软件应用和服务,并且允许以最少的管理工作

或服务提供商交互来快速提供和释放资源。

[0261] 云计算环境(有时称为云环境或云)可以以各种不同方式实现,以最好地适应不同要求。例如,在公共云环境中,底层计算基础设施由组织拥有,该组织使其云服务可供其它组织或公众使用。相反,私有云环境一般仅供单个组织使用或在单个组织内使用。社区云旨在由社区内的若干组织共享;而混合云包括通过数据和应用可移植性绑定在一起的两种或更多种类型的云(例如,私有、社区或公共)。

[0262] 一般而言,云计算模型使得先前可能由组织自己的信息技术部门提供的那些职责中的一些代替地作为云环境内的服务层来输送,以供消费者使用(根据云的公共/私人性质,在组织内部或外部)。取决于特定实现,由每个云服务层提供或在每个云服务层内提供的组件或特征的精确定义可以有所不同,但常见示例包括:软件即服务(SaaS),其中消费者使用在云基础设施上运行的软件应用,同时SaaS提供者管理或控制底层云基础设施和应用。平台即服务(PaaS),其中消费者可以使用由PaaS的供应者支持的软件编程语言和开发工具,以开发、部署和以其它方式控制它们自己的应用,同时PaaS提供者管理或控制云环境的其它方面(即,运行时执行环境下的一切)。基础设施即服务(IaaS),其中消费者可以部署和运行任意软件应用,和/或提供进程、存储装置、网络和其它基础计算资源,同时IaaS提供者管理或控制底层物理云基础设施(即,操作系统层下面的一切)。数据库即服务(DBaaS),其中消费者使用在云基础设施上运行的数据库服务器或数据库管理系统,同时DbaaS提供者管理或控制底层云基础设施和应用。

[0263] 呈现上述基本计算机硬件和软件以及云计算环境是为了说明可以用于实现(一个或多个)示例实施例的基本底层计算机组件。但是,(一个或多个)示例实施例不必限于任何特定的计算环境或计算设备配置。相反,根据本公开,(一个或多个)示例实施例可以在本领域技术人员依据本公开将理解为能够支持本文呈现的(一个或多个)示例实施例的特征和功能的任何类型的系统体系架构或处理环境中实现。

[0264] 在前面的说明书中,已经参考众多具体细节描述了本发明的实施例,这些细节可以从实现到实现有所变化。因而,说明书和附图应被视为说明性而非限制性的。本发明范围的唯一和排他性指示,以及申请人意图作为本发明范围的内容,是以发布这种权利要求书的具体形式从本申请发布的权利要求书集合的字面和等同范围,包括任何后续更正。

云 100

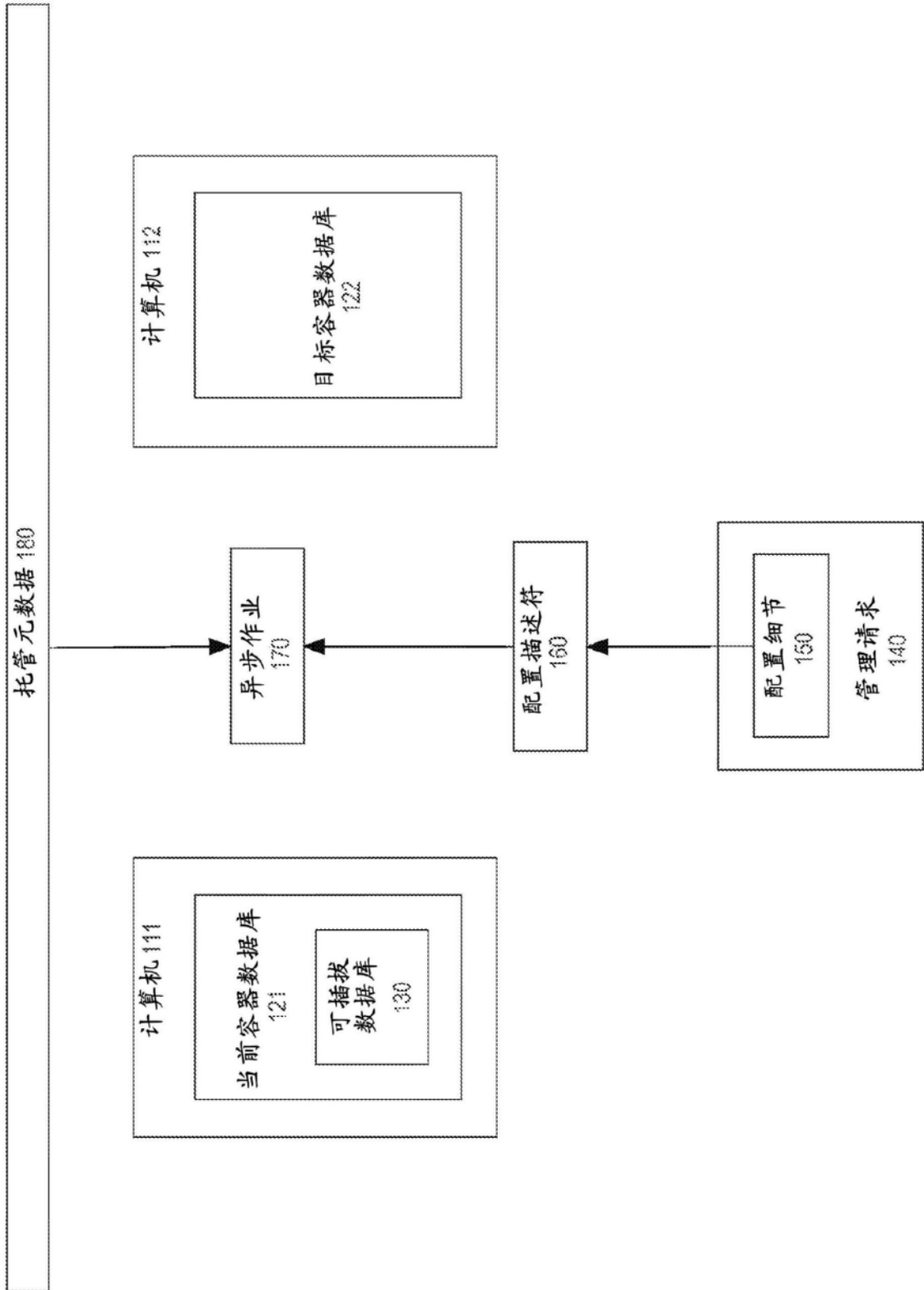


图1

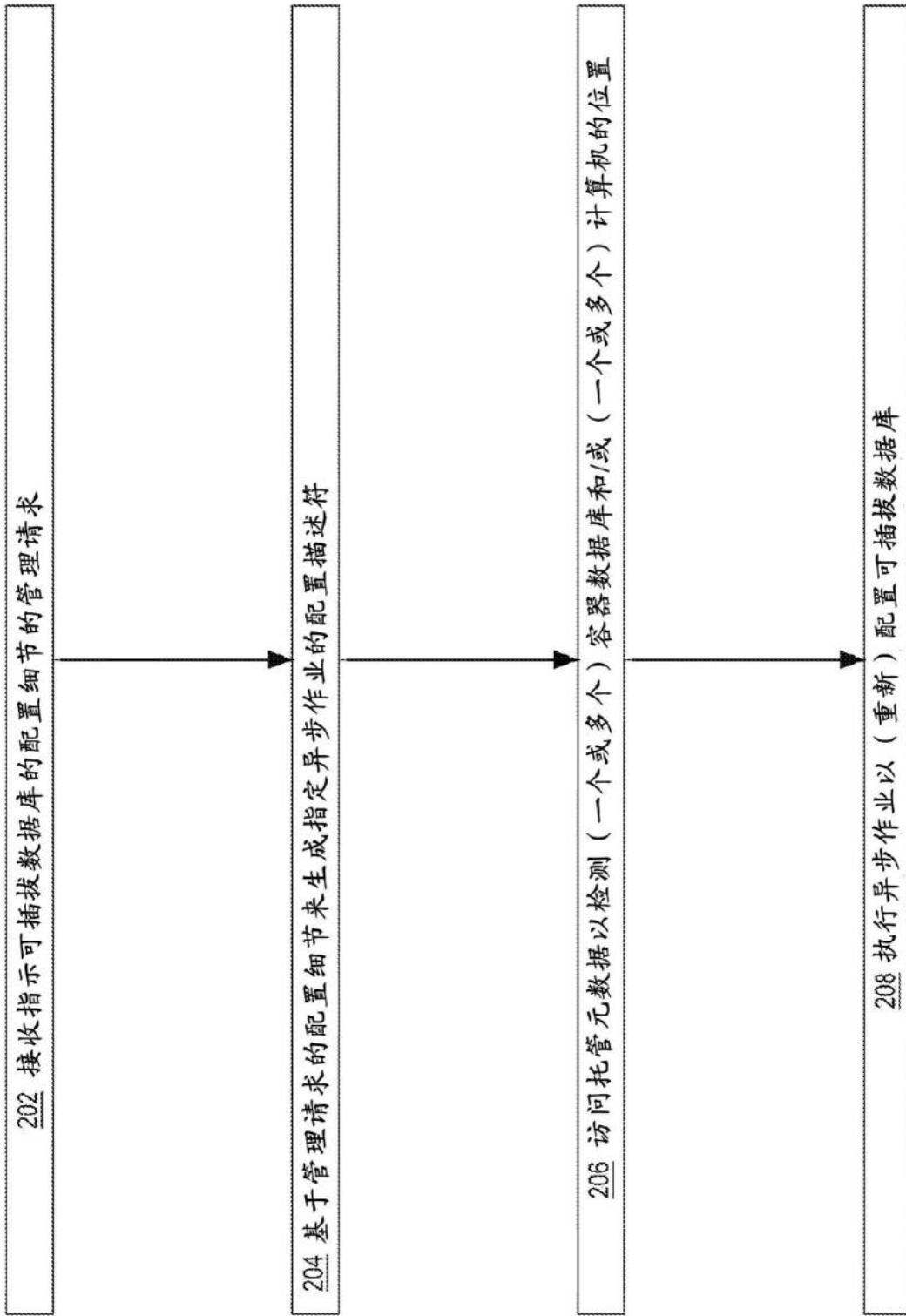


图2

云 300

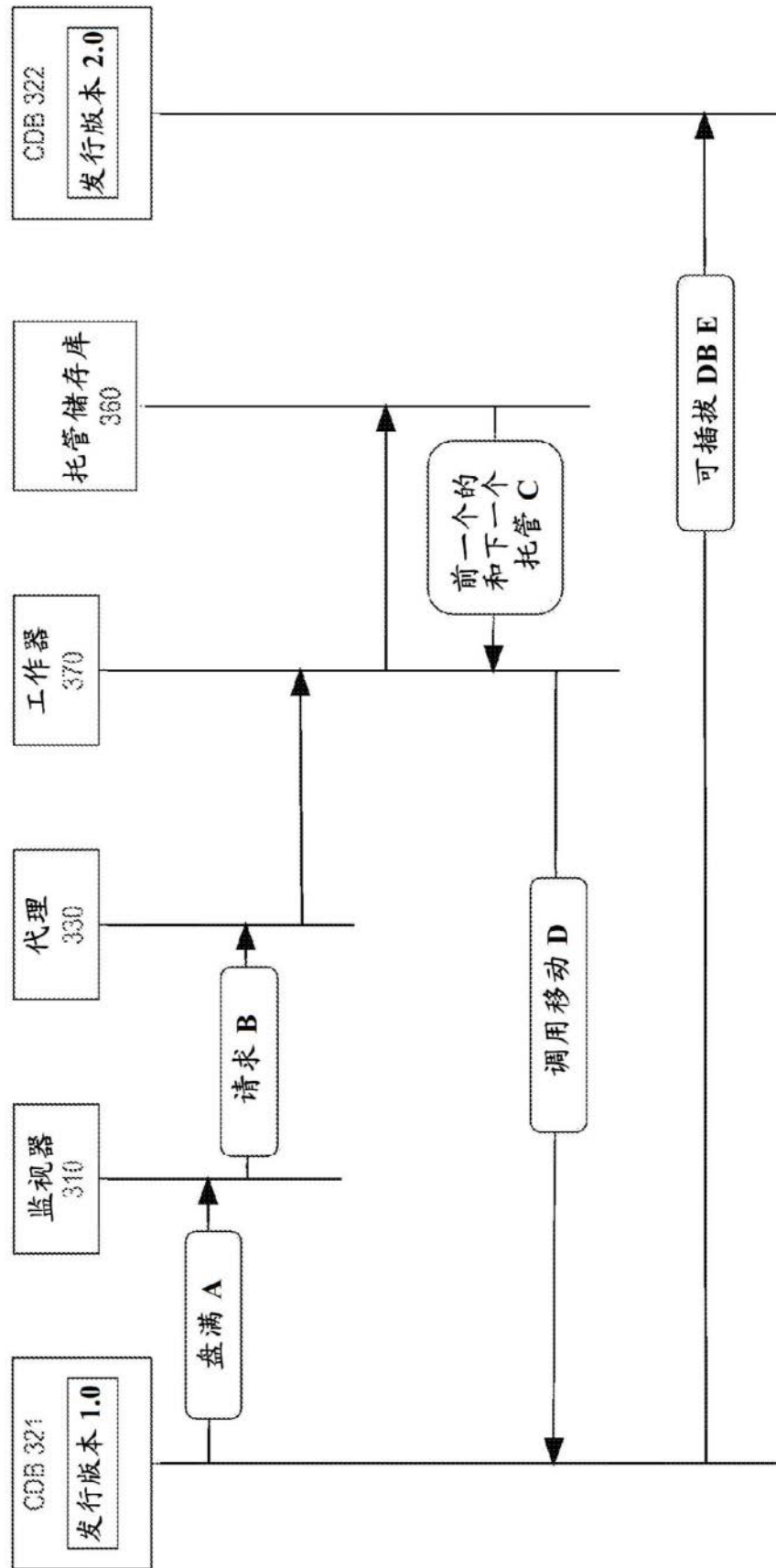


图3

云 400

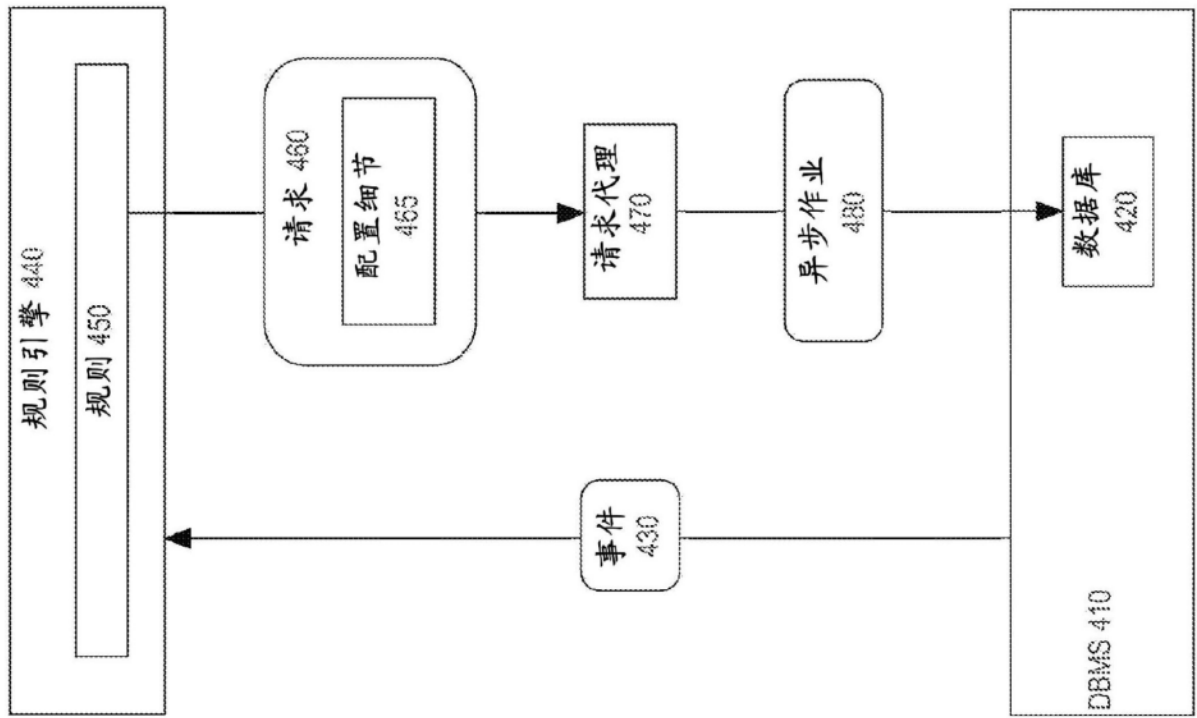


图4

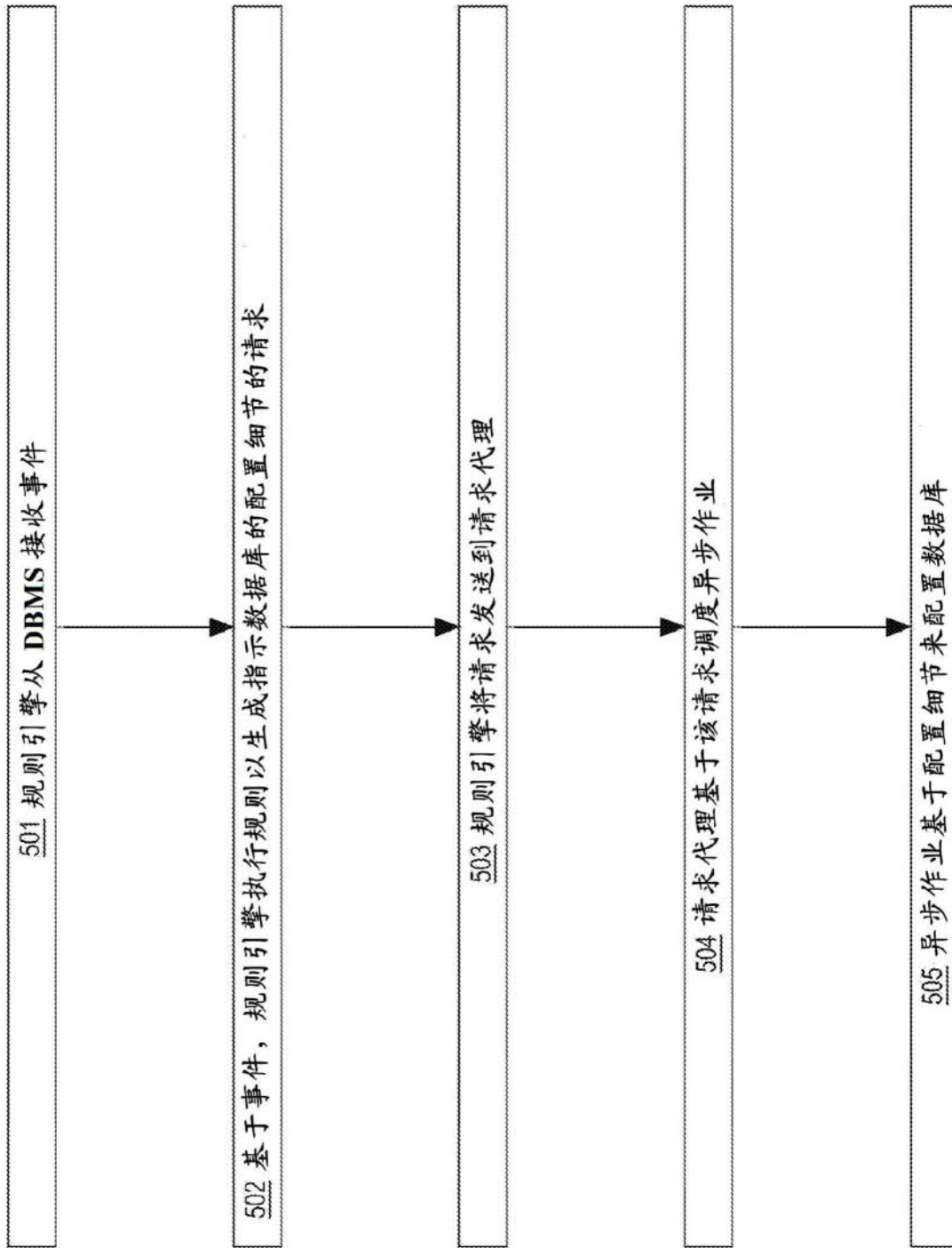


图5

云 600

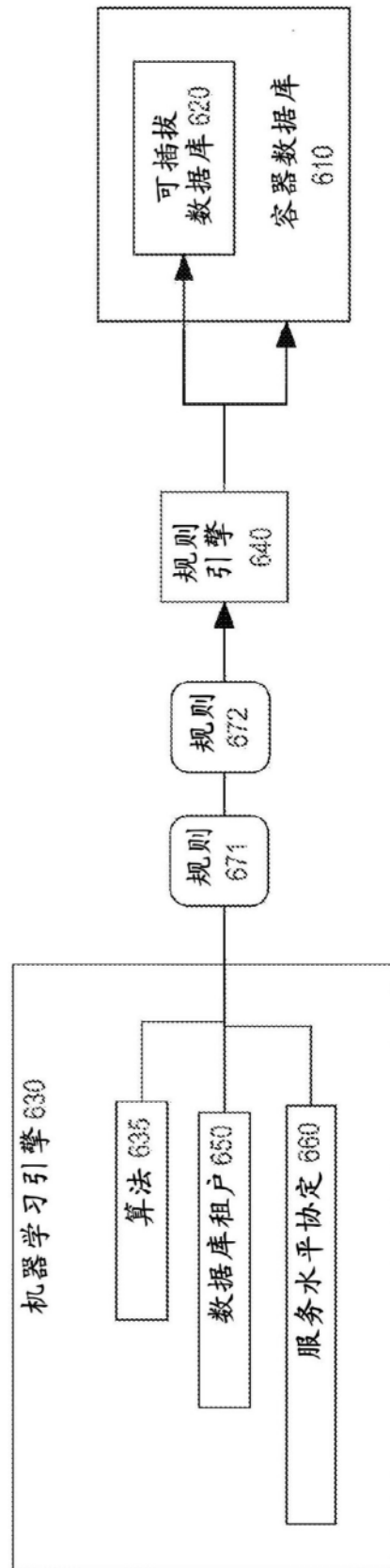


图6

云 700

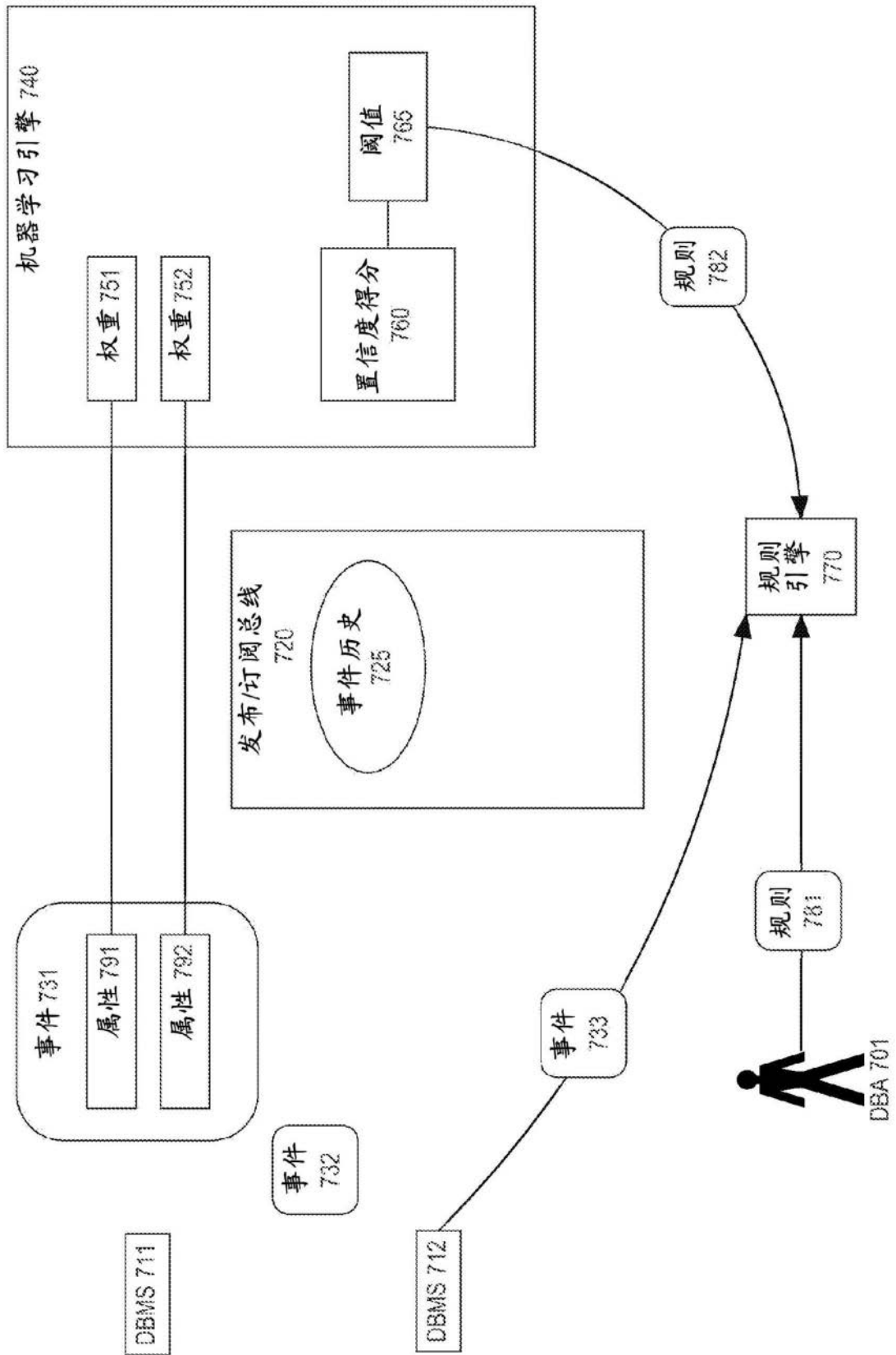


图7

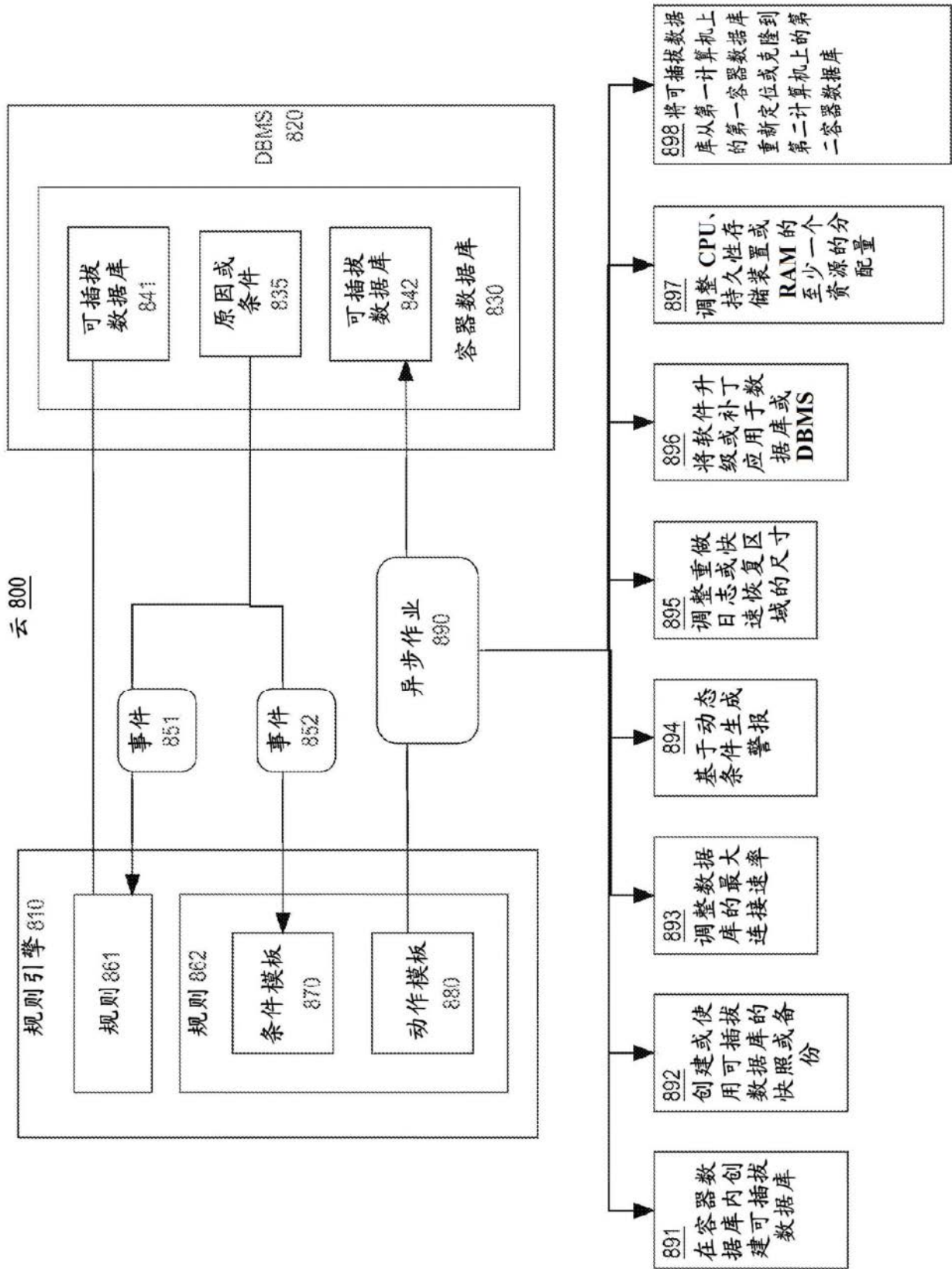


图8

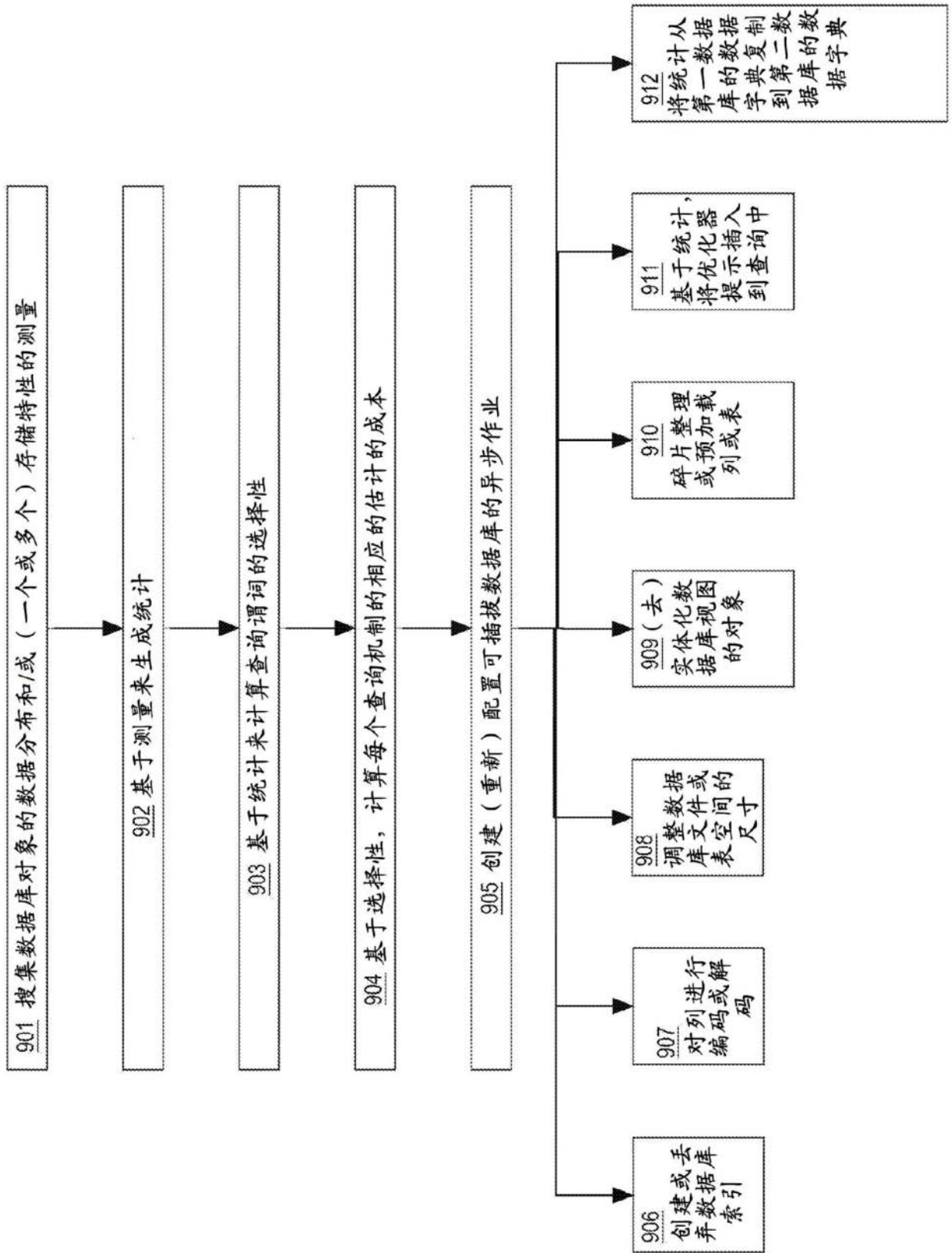


图9

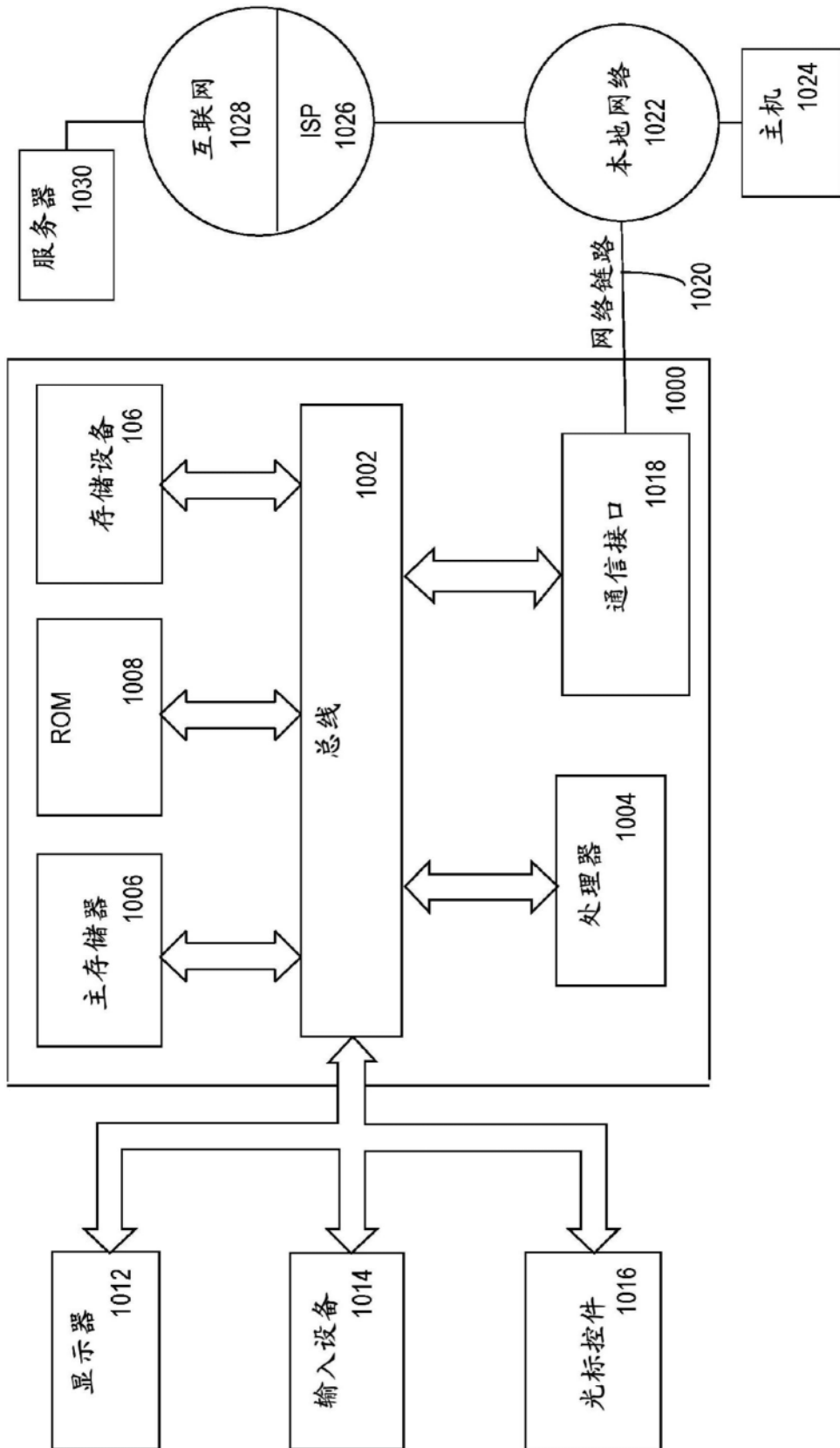


图10

软件系统 1100

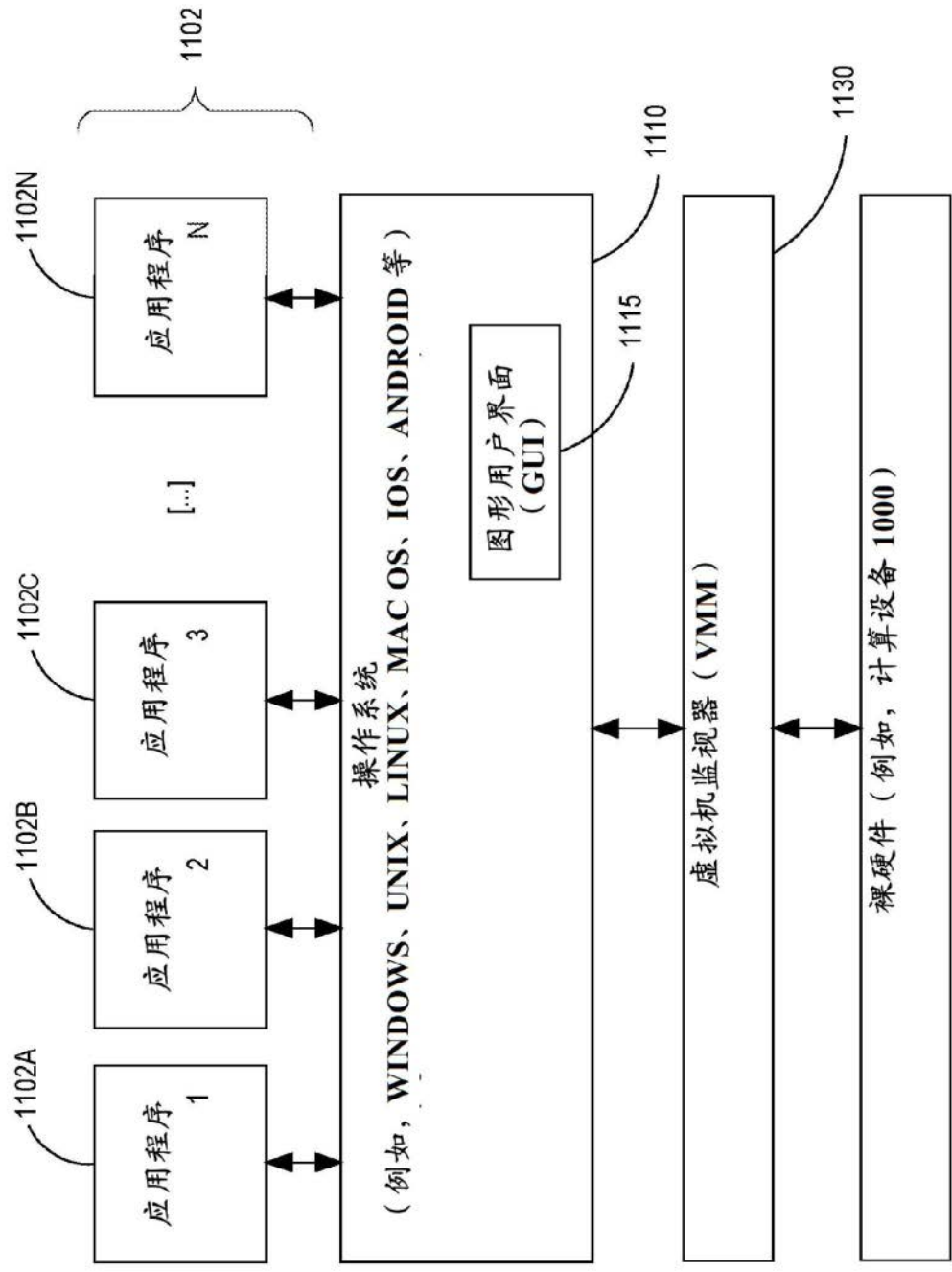


图11