



(12) 发明专利

(10) 授权公告号 CN 101438280 B

(45) 授权公告日 2016. 01. 13

(21) 申请号 200580037483. 1

(22) 申请日 2005. 11. 01

(30) 优先权数据

10/979, 742 2004. 11. 02 US

(85) PCT国际申请进入国家阶段日

2007. 04. 29

(86) PCT国际申请的申请数据

PCT/US2005/039425 2005. 11. 01

(87) PCT国际申请的公布数据

W02006/050342 EN 2006. 05. 11

(73) 专利权人 起元科技有限公司

地址 美国马萨诸塞州

(72) 发明人 丹尼尔·史蒂文斯 唐纳德·博德里

(74) 专利代理机构 隆天知识产权代理有限公司

72003

代理人 王玉双

(51) Int. Cl.

G06F 17/30(2006. 01)

(56) 对比文件

US 5734886 A, 1998. 03. 31,

US 6609132 B1, 2003. 08. 19,

US 6763515 B1, 2004. 07. 13,

Henson Graves 等. “Lockheed

Environment for Automatic

Programming”. 《KNOWLEDGE-BASED SOFTWARE

ENGINEERING》. 1992, 第 15-20 页.

审查员 姜玲玲

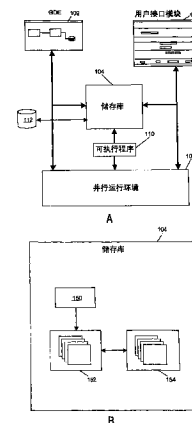
权利要求书4页 说明书11页 附图4页

(54) 发明名称

管理相关数据对象

(57) 摘要

确定数据库中第一组数据对象的相关信息，所述第一组数据对象包含该数据库中多个数据对象中的一个或多个数据对象。应用一个或多个处理组合来确定第一组数据对象的相关信息。所述处理的至少其中之一包括导航处理，该导航处理根据该导航处理的规范所确定的关系类型将一组数据对象映射到相关的另一组数据对象。应用一个或多个处理组合的步骤可以包括对该导航处理的循环应用。该处理可以包括第二处理，其执行基于与该导航处理返回的数据对象相关联的数值函数，其中每个数值表示该关联的数据对象的属性。



1. 一种方法,用于确定数据库中第一组数据对象的相关信息,所述第一组数据对象包含该数据库中多个数据对象中的一个或多个数据对象,所述数据对象为存储于储存库中的基于图形应用的数据对象,该方法包括以下步骤:

利用连接到该储存库的计算机系统,应用两个或多个处理组合以确定所述第一组数据对象的相关信息,所述处理的至少其中之一包括导航处理,该导航处理根据由该导航处理的规范确定的关系类型,将一组数据对象映射到相关的另一组数据对象,其中所述导航处理包括第一数值和第二数值,所述第一数值指定一个规则,所述规则定义与一给定类别成员的数据对象相关联的数值的一个或多个特性,以及所述第二数值指定一个类别,

其中所述导航处理针对包含一个或多个数据对象的每一组数据对象返回由所述第一数值指定的所述规则引用的、由所述第二数值指定的所述类别中的所述数据对象,或者返回由所述第二数值指定的所述类别中的所述数据对象并且所述数据对象具有引用包含一个或多个数据对象的所述组数据对象的任何成员的一个规则,以及

其中应用所述两个或多个处理组合的步骤包括对所述导航处理的循环应用。

2. 如权利要求 1 所述的方法,其中对该导航处理的循环应用包括:对包含所述导航处理的导航处理组合的循环应用。

3. 如权利要求 1 所述的方法,其中应用所述一个或多个处理组合的步骤还包括:标识在对该导航处理的循环应用中得到的数据对象。

4. 如权利要求 1 所述的方法,其中应用所述一个或多个处理组合的步骤还包括:标识在对该导航处理的循环应用的末尾得到的数据对象。

5. 如权利要求 1 所述的方法,还包括以下步骤:

接受该数据库中数据对象的相关信息的视觉表现的规范,该规范包括所述一个或多个处理组合的规范;以及

向用户呈现包括所述第一组数据对象的相关信息的视觉表现。

6. 如权利要求 5 所述的方法,其中该导航处理的规范包括一种函数,该函数的参数包括数据对象的类别以及用于指定该关系类型的注释规则的名称。

7. 如权利要求 1 所述的方法,其中该关系类型对应于多个关系类型的第一关系类型。

8. 如权利要求 7 所述的方法,其中所述多个关系类型对应于多个数据对象类别。

9. 如权利要求 8 所述的方法,其中该关系类型参照与第二数据对象相关联的数值来标识与第一数据对象相关联的数值。

10. 如权利要求 9 所述的方法,其中该第一数据对象属于第一类别,而该第二数据对象属于第二类别。

11. 如权利要求 9 所述的方法,其中与该第一数据对象相关联的数值是外关键字,而与该第二数据对象相关联的数值是主关键字。

12. 一种方法,用于确定数据库中第一组数据对象的相关信息,所述第一组数据对象包含该数据库中多个数据对象中的一个或多个数据对象,所述数据对象为存储于储存库中的基于图形应用的数据对象,该方法包括以下步骤:

利用连接到该储存库的计算机系统,应用处理组合以确定关于所述第一组数据对象的相关信息,所述处理包括:

第一处理,根据由该第一处理的规范确定的关系类型,将一组数据对象映射到相关的

另一组数据对象,其中所述第一处理包括一个导航处理,所述导航处理包括第一数值和第二数值,所述第一数值指定一个规则,所述规则定义与一给定类别成员的数据对象相关联的数值的一个或多个特性,以及所述第二数值指定一个类别;以及

第二处理,执行基于与该第一处理返回的数据对象相关联的数值的函数,其中每个数值表示关联的数据对象的属性,

其中所述导航处理针对包含一个或多个数据对象的每一组数据对象返回由所述第一数值指定的所述规则引用的、由所述第二数值指定的所述类别中的所述数据对象,或者返回由所述第二数值指定的所述类别中的所述数据对象并且所述数据对象具有引用包含一个或多个数据对象的所述组数据对象的任何成员的一个规则。

13. 如权利要求 12 所述的方法,其中该函数包括基于所述数值过滤返回的一组数据对象,以产生该返回的一组数据对象的子集。

14. 如权利要求 12 所述的方法,其中该函数包括对返回的一组数据对象进行排序。

15. 如权利要求 12 所述的方法,其中该函数包括至少某些所述数值的算术函数。

16. 如权利要求 12 所述的方法,其中该函数包括至少某些所述数值的逻辑函数。

17. 如权利要求 12 所述的方法,还包括:

接受该数据库中数据对象的相关信息视觉表现的规范,该规范包括该处理组合的规范;以及

向用户呈现包括所述第一组数据对象的相关信息视觉表现。

18. 如权利要求 17 所述的方法,其中该第一处理的规范包括一种函数,该函数的参数包括数据对象的类别以及用于指定该关系类型的注释规则的名称。

19. 如权利要求 12 所述的方法,其中该关系类型对应于多个关系类型中的第一关系类型。

20. 如权利要求 19 所述的方法,其中所述多个关系类型对应于多个数据对象类别。

21. 如权利要求 20 所述的方法,其中该关系类型参照与第二数据对象相关联的数值来标识与第一数据对象相关联的数值。

22. 如权利要求 21 所述的方法,其中该第一数据对象属于第一类别,而该第二数据对象属于第二类别。

23. 如权利要求 21 所述的方法,其中与该第一数据对象相关联的数值是外关键字,而与该第二数据对象相关联的数值是主关键字。

24. 一种系统,用于确定数据库中第一组数据对象的相关信息,所述第一组数据对象包含该数据库中多个数据对象中的一个或多个数据对象,该系统包括:

储存库,存储所述多个数据对象;以及

计算机系统,连接到该储存库,并且该计算机系统被配置为应用两个或多个处理组合以确定所述第一组数据对象的相关信息,所述处理的至少其中之一包括导航处理,该导航处理根据由该导航处理的规范确定的关系类型,将一组数据对象映射到相关的另一组数据对象,其中所述导航处理包括第一数值和第二数值,所述第一数值指定一个规则,所述规则定义与一给定类别成员的数据对象相关联的数值的一个或多个特性,以及所述第二数值指定一个类别,

其中所述导航处理针对包含一个或多个数据对象的每一组数据对象返回由所述第一

数值指定的所述规则引用的、由所述第二数值指定的所述类别中的所述数据对象,或者返回由所述第二数值指定的所述类别中的所述数据对象并且所述数据对象具有引用包含一个或多个数据对象的所述组数据对象的任何成员的一个规则,以及

其中应用所述两个或多个处理组合的步骤包括对所述导航处理的循环应用。

25. 如权利要求 24 所述的系统,其中对该导航处理的循环应用包括:对包含所述导航处理的导航处理组合的循环应用。

26. 如权利要求 24 所述的系统,其中该关系类型对应于多个关系类型中的第一关系类型。

27. 如权利要求 24 所述的系统,其中应用所述一个或多个处理组合的步骤还包括:标识在对该导航处理的循环应用中得到的数据对象。

28. 如权利要求 24 所述的系统,其中应用所述一个或多个处理组合的步骤还包括:标识在对该导航处理的循环应用的末尾得到的数据对象。

29. 如权利要求 24 所述的系统,其中所述计算机系统还被配置为:

接受该数据库中数据对象的相关信息的视觉表现的规范,该规范包括所述一个或多个处理组合的规范;以及

向用户呈现包括所述第一组数据对象的相关信息的视觉表现。

30. 如权利要求 29 所述的系统,其中该导航处理的规范包括一种函数,该函数的参数包括数据对象的类别以及用于指定该关系类型的注释规则的名称。

31. 如权利要求 26 所述的系统,其中所述多个关系类型对应于多个数据对象类别。

32. 如权利要求 31 所述的系统,其中该关系类型参照与第二数据对象相关联的数值来标识与第一数据对象相关联的数值。

33. 如权利要求 32 所述的系统,其中该第一数据对象属于第一类别,而该第二数据对象属于第二类别。

34. 如权利要求 32 所述的系统,其中与该第一数据对象相关联的数值是外关键字,而与该第二数据对象相关联的数值是主关键字。

35. 一种系统,用于确定数据库中第一组数据对象的相关信息,所述第一组数据对象包含该数据库中多个数据对象中的一个或多个数据对象,该系统包括:

储存库,存储所述多个数据对象;以及

计算机系统,连接到该储存库并且被配置为应用处理组合以确定关于所述第一组数据对象的相关信息,所述处理包括:

第一处理,根据由该第一处理的规范确定的关系类型,将一组数据对象映射到相关的另一组数据对象,其中所述第一处理包括一个导航处理,所述导航处理包括第一数值和第二数值,所述第一数值指定一个规则,所述规则定义与一给定类别成员的数据对象相关联的数值的一个或多个特性,以及所述第二数值指定一个类别;以及

第二处理,执行基于与该第一处理返回的数据对象相关联的数值的函数,其中每个数值表示关联的数据对象的属性,

其中所述导航处理针对包含一个或多个数据对象的每一组数据对象返回由所述第一数值指定的所述规则引用的、由所述第二数值指定的所述类别中的所述数据对象,或者返回由所述第二数值指定的所述类别中的所述数据对象并且所述数据对象具有引用包含一

个或多个数据对象的所述组数据对象的任何成员的一个规则。

36. 如权利要求 35 所述的系统,其中该函数包括基于所述数值过滤返回的一组数据对象,以产生该返回的一组数据对象的子集。

37. 如权利要求 35 所述的系统,其中该函数包括对返回的一组数据对象进行排序。

38. 如权利要求 35 所述的系统,其中该函数包括至少某些所述数值的算术函数。

39. 如权利要求 35 所述的系统,其中该函数包括至少某些所述数值的逻辑函数。

40. 如权利要求 35 所述的系统,其中该关系类型对应于多个关系类型中的第一关系类型。

41. 如权利要求 35 所述的系统,其中所述计算机系统还被配置为:

接受该数据库中数据对象的相关信息的视觉表现的规范,该规范包括该处理组合的规范;以及

向用户呈现包括所述第一组数据对象的相关信息的视觉表现。

42. 如权利要求 41 所述的系统,其中该第一处理的规范包括一种函数,该函数的参数包括数据对象的类别以及用于指定该关系类型的注释规则的名称。

43. 如权利要求 40 所述的系统,其中所述多个关系类型对应于多个数据对象类别。

44. 如权利要求 43 所述的系统,其中该关系类型参照与第二数据对象相关联的数值来标识与第一数据对象相关联的数值。

45. 如权利要求 44 所述的系统,其中该第一数据对象属于第一类别,而该第二数据对象属于第二类别。

46. 如权利要求 44 所述的系统,其中与该第一数据对象相关联的数值是外关键字,而与该第二数据对象相关联的数值是主关键字。

## 管理相关数据对象

### 技术领域

[0001] 本发明涉及管理相关数据对象并向相关数据对象提供接口。

### 背景技术

[0002] 通常根据文件系统对存储设备中的数据进行组织。文件系统通常以目录的层级结构来组织数据（例如文件），其中每一层级可以包含一组文件和 / 或其它目录。可使用能隐含地或明确地标识起始位置以及标识关于该起始位置的一个或多个文件和 / 或目录的路径来导航目录的层级结构。

[0003] 文件系统的层级结构可以表示为节点树，其中目录是具有用于该目录中的每个项目的分支的节点，而文件是叶节点。节点与其子节点之间的分支表示节点之间的“包含”关系。路径可以包括表达式，所述表达式包括用以过滤一组选择节点的通配符。

[0004] 其它数据系统可以用来导航数据对象的树结构的层级。例如，X 路径可以用于在部分 XML 文件中导航。X 路径将 XML 文件模拟为节点树。有不同类型的节点，包括在层级结构中关联的元素节点、属性节点和文本节点。

[0005] 用于以 X 路径导航的一种类型的表达式是位置路径。位置路径选择与上下文节点（context node）相关的一组节点。位置路径的计算结果是由位置路径选择的一组节点。位置路径也可以包括用于过滤所选择的一组节点的表达式。

### 发明内容

[0006] 一般地，本发明提供一种方法以及相应的软件和系统，用于确定数据库中第一组数据对象的相关信息，所述第一组数据对象包含该数据库中多个数据对象中的一个或多个数据对象。应用一个或多个处理组合以确定所述第一组数据对象的相关信息。所述处理的至少其中之一包括导航处理，该导航处理根据该导航处理的规范所确定的关系类型将一组数据对象映射到相关的另一组数据对象。

[0007] 本发明的方案可以包括一个或多个以下特征：

[0008] 应用一个或多个处理组合的步骤包括对导航处理的循环应用。对该导航处理的循环应用包括：对包含所述导航处理的导航处理组合的循环应用。应用该一个或多个处理组合的步骤还包括：标识在对该导航处理的循环应用中得到的数据对象，和 / 或标识在对该导航处理的循环应用的末尾得到的数据对象。

[0009] 接受该数据库中数据对象的相关信息的视觉表现的规范。该规范包括一个或多个处理组合的规范。向用户呈现包括所述第一组数据对象的相关信息的视觉表现。

[0010] 导航处理的规范包括一种函数，该函数的参数包括数据对象的类别和指定关系类型的注释规则的名称。

[0011] 所述处理包括第二处理，该第二处理执行基于与该导航处理返回的所述数据对象相关联的数值函数，其中每个数值表示该关联的数据对象的属性。该函数可以包括基于所述数值过滤所述返回的一组数据对象，以产生返回的一组数据对象的子集，对所述返回的

一组数据对象进行排序,至少某些数值的算术函数和 / 或至少某些数值的逻辑函数。

[0012] 关系类型对应于多个关系类型的其中之一,例如数据对象的类别。

[0013] 关系类型参照与第二数据对象相关联的数值来标识与第一数据对象相关联的数值。

[0014] 该第一数据对象属于第一类别,而该第二数据对象属于第二类别。

[0015] 与该第一数据对象相关联的数值是外关键字,而与该第二数据对象相关联的数值是主关键字。

[0016] 本发明的方案可以包括一个或多个以下优点:

[0017] 在此描述的技术能够在数据库(面向对象关系的数据库)中的数据对象中进行导航,所述数据对象通过多种类型的关系(例如多种类型的主关键字 / 外关键字关系)潜在关联。当导航处理的规范确定关系类型时,该导航处理可以根据该关系类型定位一组相关的数据对象。对导航处理的循环应用在定位相关的数据对象时能够具有非常大的灵活性。在从数据库中提取期望的信息时,执行基于与数据对象相关联的属性数值函数的功能也能够具有非常大的灵活性。

[0018] 本发明的其它特性和优点将从以下说明书以及权利要求书中显而易见。

## 附图说明

[0019] 图 1A 是计算机系统的框图。

[0020] 图 1B 是元数据储存库的框图。

[0021] 图 2 是示例性基本模型的实体关系图。

[0022] 图 3 是呈现的用于视窗的页面。

## 具体实施方式

[0023] 1 概要

[0024] 1.1 系统

[0025] 图 1A 示出用于开发、执行以及管理基于图形计算的计算系统 100 的各部分相互关系的框图。使用表示为有向图的“数据流图”实现基于图形的计算,其中图中的顶点表示组件(数据文件或过程),并且图中的有向链路或者“边界”表示组件之间的数据流。图形开发环境(GDE)102 提供用户接口,其用于指定可执行的图形并且为图形组件定义参数。GDE 可以是例如从本发明的受让人处得到的 CO > OPERATING SYSTEM® GDE。GDE 102 与储存库(repository)104 以及并行运行环境 106 进行通信。此外,储存库 104 以及并行运行环境 106 还连接有用户接口模块 108 和可执行程序 110。

[0026] 储存库 104 优选为可升级的面向对象的数据系统,该数据库系统被设计为支持基于图形的应用程序的开发和执行以及基于图形的应用程序与其它系统(例如其它操作系统)之间的元数据的相互交换。储存库 104 是用于各种元数据的存储系统,其中所述元数据包括文件、记录格式(例如表中记录的字段和数据类型)、传递函数、图形、作业以及监控信息等。储存库 104 也存储用于表示计算系统 100 要处理的实际数据的数据对象,该实际数据包括在外部数据存储 112 中存储的数据。

[0027] 并行运行环境 106 接受 GDE 102 中产生的数据流图的说明,并产生对应于处理逻辑

辑的计算机指令以及由图形定义的资源。然后并行运行环境 106 通常在多个处理器上执行这些指令（所述处理器无需为同构的处理器）。适合的并行运行环境可以从本发明的受让人处得到的 CO > OPERATINGSYSTEM®。

[0028] 用户接口模块 108 提供储存库 104 内容的基于网络浏览器的视窗。用户可以使用用户接口模块 108 来导航对象、产生新对象、改变现有的对象、指定应用程序参数、预定作业等。用户接口模块 108 为用户产生基于表格的浏览器屏幕，以搜索并且查看对象以及关于储存库 104 中存储的对象的信息。

[0029] 执行程序 110 是通过用户接口模块 108 访问的可选择的基于储存库的作业预定系统。执行程序 110 保留作业以及作业队列作为储存库 104 中的对象，而用户接口模块 108 提供作业和作业队列的视窗，从而有助于处理作业和作业队列。

[0030] 1.2 元数据分析

[0031] 储存库 104 存储包括基于图形应用的数据对象的元数据，所述数据对象包括用于构建计算图形的图形组件以及其它函数对象。存储在储存库 104 中的元数据也可以包括例如“技术”元数据（例如应用相关的业务规则、记录格式以及执行统计）以及“企业”或者“业务”元数据（例如关于作业功能、任务以及责任的用户定义文档）。

[0032] 以数据对象的形式在储存库 104 中存储的信息使得能够对应用程序以及由这些应用程序处理的数据进行各种分析。例如，用户可以得到关于数据血统问题的答案（例如，所给值从哪来？输出值是如何计算的？哪个应用程序产生该数据并且依赖该数据？）。开发者可以理解建议修改的结果（例如，如果这段信息改变，其它信息将会受到影响吗？如果该信息的源格式改变，哪个应用程序将受到影响？）。用户/开发者也可以得到既包括技术元数据又包括业务元数据的问题的答案（例如，哪个组负责产生和使用该数据？谁最后改变了这个应用程序？他们做了什么改变？）。

[0033] 储存库 104 能够跟踪存储数据的状态。储存库 104 中存储的数据对象被版本化，从而能够检查在最后一周、最后一月或最后一年时的状态，并且能够与今天的状态进行比较。储存库 104 收集能够进行趋势分析（例如我们的数据增长多快？）以及容量规划（例如该应用程序运行多长时间？处理多少数据以及以什么速率？该应用程序消耗什么资源？我们何时需要增加另一个服务器？）的作业跟踪或者执行信息。

[0034] 用户可以通过用户接口模块 108 查看（以及可选地，编辑）包含在存储的数据对象中的信息和/或与所述存储的数据对象相关联的信息。在一个实例中，用户可以定义一个或多个“视窗”，所述视窗提供关于网页中的数据对象的相关子集的信息。如以下更加详细描述，使用语法定义视窗，以基于这些对象之间的关系选择一个或多个数据对象。

[0035] 2. 数据模型

[0036] 2.1 基本模型

[0037] 图 2 示出用于与公司相关联的元数据的在基本模型 200 中的选择类别，该图表示为实体关系图。基本模型 200 定义在储存库 104 中存储的元数据的基本关系结构。基本模型 200 中的实体类型被称之为“类别”。每个类别表示一类数据对象。每个类别包括一组“注释规则”。如以下更加详细描述，类别的注释规则类似于表格的列定义。

[0038] 用户可以通过增加用户定义的类别扩充基本模型 200。用户能够基于已经存在的“父”类别增加“子”类别。子类别包括父类别的注释规则，并且还可选地包括针对子类别特



有的注释规则。

[0039] 数据对象可以存储在储存库 104 中作为类别的成员。作为类别的成員的数据对象类似于表格中的行。成员数据对象（例如包含雇员信息的数据结构）可以被存储在指定了类别（例如“雇员”类别）的目录中。或者，可以通过数据对象中的引用将成员数据对象与类别相关联。作为类别成员的数据对象存储与为该类别定义的注释规则相关联的数值。

[0040] 定义两个数据对象之间“关系”的一个方法是通过“主关键字 / 外关键字”关系。类别的“主关键字”是注释规则（或规则），所述注释规则（或规则）的数值（或者多个数值综合在一起）唯一地标识类别中的每个数据对象。当第二数据对象具有引用第一数据对象的主关键字数值的注释规则数值时，第一数据对象与第二数据对象（不同类别或相同类别）具有主关键字 / 外关键字关系。该第二数据对象的注释规则被称之为“外关键字”。

[0041] 可以使用其它技术定义两个数据对象之间的关系。例如，在某些情形，为了唯一地标识父数据对象，可能需要外关键字引用一个以上主关键字的注释规则数值。由一个以上注释规则数值组成的主关键字被称之为“复合关键字”。

[0042] 基本模型 200 描述可以存在于属于不同类别的数据对象之间的潜在关系。连接两个类别（或者连接一类别及其自身）的直线表示各类别的数据对象之间潜在关系的“连接性”。类别之间连接性的基本类型是：一对一，一对多以及多对多。

[0043] 当类别 A 中至多一个数据对象关联到类别 B 中的一个数据对象时，存在一对一连接性。例如，“雇员”类别 202 表示公司中的雇员，并且类别中的每个数据对象表示一个雇员。“办公室”类别 204 表示建筑物中占用的办公室，并且每个数据对象表示一间办公室。如果对每个雇员分配他们自己的办公室，则相应的数据对象将具有一对一的外关键字的关系。在图 2 中一对一连接性被描述为直线。

[0044] 当对于类别 A 中的一个数据对象在类别 B 中有零个、一个或多个相关的数据对象时，而对于类别 B 中的一个数据对象在类别 A 中有一个相关的数据对象时，存在一对多连接性。例如，如上所述，“雇员”类别 202 表示公司中的雇员。“部门”类别 206 表示公司中的部门。每个雇员关联到一个部门，而每个部门关联到很多雇员。因而，分别对应于“部门”和“雇员”的数据对象具有一对多的外关键字关系。图 2 中一对多连接性被描述为以鱼尾纹 (crow's foot) 结束的线。

[0045] 当对于类别 A 中的一个数据对象在类别 B 中有零个、一个或多个相关的数据对象，而对类别 B 中的一个数据对象在类别 A 中有零个、一个或多个相关的数据对象时，存在多对多连接性。例如，假设对多个雇员同时分配多个项目，而一个项目（在“项目”类别 208 中的数据对象）可以具有分配给该项目的多个雇员。来自“雇员”和“项目”类别的相应数据对象可以具有多对多外关键字的关系。在图 2 中多对多连接性被描述为以鱼尾纹开始并且结束的线。

[0046] 同一类别中的对象之间也可以存在关系。例如，“雇员”类别中的数据对象与该“雇员”类别中的其它数据对象之间具有一对多的相关关系。一个雇员与由一对多的外关键字关系表示的另一雇员可以具有“监督”关系。

[0047] 2.2 注释规则

[0048] 注释规则定义了与所给类别成员的数据对象相关联的数值的各种特性。注释规则可以是单值的或是多值的。每个注释规则具有使其被标识的唯一名称。通常多类别可以具

有一个或多个注释规则。

[0049] 在一个实例中有四种类型的注释规则,所述注释规则向用户(例如应用程序开发者)提供限制功能,其用于限制(例如最终用户)可以与数据对象相关联的数值的数目和种类(数据类型)。注释规则的四种类型是:string, choice, reference 和 record。所述四种类型的注释规则的特性如下。

#### [0050] 2.2.1string- 类型

[0051] string- 类型的注释规则使一个数据对象与一个或多个字符串相关联(例如 ASCII 字符串)。将一个或多个字符串附加到数据对象,并且使所述字符串与注释规则的名称相关联。具有一个字符串数值的 string- 类型的注释规则是单值的。例如,由于一个雇员具有一个社保号码,因此雇员数据对象可以具有单值的“社保号码”string- 类型的注释规则。具有一个以上字符串数值的字符串注释规则是多值的。例如,由于雇员可以具有一个以上电话号码,因此雇员数据对象可以具有多值的“电话号码”string- 类型的注释规则。

#### [0052] 2.2.2choice- 类型

[0053] choice- 类型的注释规则使一个数据对象与用户从可能数值列表中选择的一个或多个数值相关联。例如,当期望注释规则可以具有有效数值时,可以使用 choice- 类型规则。对于单值的 choice- 类型的注释规则,最终用户可以只选择多个选择的其中之一。单值的 choice- 类型的注释规则的实例是表示只具有两种可能的答案:“是”或“否”的问题答案的数值。

[0054] 对于多值的 choice- 类型的注释规则,最终用户可以从可能数值列表中选择任意个数的数值。例如,雇员可以工作在两个部门:“Development(开发部)”和“QA 部”。choice- 类型“Department”注释规则将使得最终用户从部门列表“Finance(财务部)”、“Development”、“Operations(运营部)”、“Documentation(文档部)”、“QA 部”以及“Executive(执行部)”中选择适当的数值。

[0055] 默认地,可能数值列表可以被限制为由用户定义的预定数值列表。或者,最终用户能够指定预定数值列表中没有的数值。可选地,当最终用户指定列表中没有的数值时,最终用户能够将该数值增加到预定数值列表中,从而其他最终用户可以看见该数值作为一个选项。或者,未列出的数值可以保留在储存库 104 中,但是对于其他最终用户并不显示为选项。

#### [0056] 2.2.3reference- 类型

[0057] reference- 类型注释规则使第一数据对象与一个或多个其它数据对象相关联。该数据对象可以是与该第一数据对象相同类别的成员或是不同类别的成员。多值的 reference- 类型的注释规则可以引用多数据对象。

[0058] reference- 类型注释规则可以用于指示数据对象之间的主关键字/外关键字关系。reference- 类型的注释规则的数值可以是引用不同类别中的数据对象的主关键字数值的外关键字数值。例如,“雇员”类别中的数据对象的注释规则包括 reference- 类型的注释规则“departmentKEY”,并且该 reference- 类型的注释规则“departmentKEY”具有与唯一地标识“部门”类别中的数据对象的 string- 类型注释规则“key”的数值匹配的数值,因而,将唯一的对象标识符“key”用作主关键字。

[0059] 不同类型的关系可以存在于一个数据对象与另一个数据对象之间。一个数据对象

可以具有分别表示不同关系类型的多个 reference- 类型的注释规则。例如,“雇员”类别中的数据对象也包括 reference- 类型注释规则“officeKEY”,其具有与唯一地标识“Office”类别中的数据对象的 string- 类型注释规则“key”的数值匹配的数值。

#### [0060] 2.2.4 record- 类型

[0061] record- 类型的注释规则使一个数据对象与来自另一类别中的对象的一组注释规则数值相关联。例如,“employee(雇员)”类别的“email\_info”注释规则的数值包括为“Email Info”类别(未示出)定义的注释规则的所有数值。如果“Email Info”类别中的数据对象具有数值为“joe@home.com”的 string- 类型的注释规则“home\_email”以及具有数值为“joe@work.com”的 string- 类型的注释规则“work\_email”,则“Employees”类别中的数据对象可以引用该数据对象以使数值“joe@home.com”以及“joe@work.com”与“email\_info”相关联。

#### [0062] 2.2.5 实例

[0063] 所有定义的类别可以共享一组共同的注释规则。例如,通常在基本模型 200 中的每个类别具有两个注释规则:key 和 name。以下将给出这些注释规则的描述。

[0064] key :唯一地标识储存库中每个数据对象的字符串。

[0065] name :数据对象的名称。

[0066] 每个类别也可以具有针对该类别的注释规则。例如,以下给出基本模型 200 中对于类别“Employee”以及“Department(部门)”的某些注释规则。

[0067] 表示公司中雇员的“Employee”类别的注释规则包括:

[0068] start\_date :雇员开始工作的日期。

[0069] name :雇员的姓名。

[0070] start\_date :雇员的性别,“F”表示女性,“M”表示男性。

[0071] managed\_byKEY :该值指作为该雇员管理者的另一雇员的关键字数值。

[0072] departmentKEY :该值指用于表示该雇员工作所在部门的部门类别成员的关键字数值。

[0073] email\_info :表示雇员的电子邮件信息。

[0074] ProjectKEY :该值指用于表示分配给该雇员的项目的项目类别成员的关键字数值。

[0075] OfficeKEY :该值指用于表示该雇员工作所在的办公室的办公室类别成员的关键字数值。

[0076] 表示公司中的部门的“部门”类别的注释规则包括:

[0077] name :指部门的名称。

[0078] location :指部门办公室的位置。

#### [0079] 3 视窗

[0080] 如上所述,用户可以通过用户接口模块 108 使用在网络浏览器中呈现给用户的用户定义的视窗来查看(可选地,编辑)储存库 104 中的元数据。参照图 1B,元数据储存库 104 包括:相关数据 150、数据对象 152 以及存储视窗 154,其中相关数据 150 定义数据对象类别和注释规则,数据对象 152 对应于相关数据 150 定义的对象实例,由用户定义存储视窗 154,以用于访问包含在数据对象 152 中和 / 或与数据对象 152 相关联的期望的信息。用户接口

模块 108 根据视窗从储存库 104 访问正确的数据对象,并且呈现具有期望信息(例如数据对象的标签或者符合指定标准的数据对象的数目的计数)的 HTML 网页。每个视窗均与类别相关联,并且包括用以定义待呈现的元数据的一个或多个“视觉元素”(“View-element”)。每个视觉元素包括元素描述符(element-descriptor),其能够基于该关联类别中或者通过一个或多个主关键字/外关键字关系与该关联类别相关的类别中的一个或多个数据对象来检索元数据。

[0081] 图 3 示出作为视窗的呈现的 HTML 网页的实例。网页 300 包括头部 302、内容部分 304 以及底部 306,其中头部 302 包括导航条。内容部分 304 包括视窗被定义的类别的名称 308(例如雇员)以及与该视窗中的信息相关联的类别中主要数据对象的名称 310。内容部分 304 也包括与该主要数据对象相关联的信息,例如用户定义的标签 312 以及该主要数据对象的注释规则的数值 314。网页 300 也包括视觉元素,所述视觉元素能够显示与关于该主要数据对象的其它数据对象相关联的信息。在这个实例中,网页 300 包括视觉元素 318 和视觉元素 320,其中视觉元素 318 显示“Applications this employee is responsible for(该雇员负责的应用)” (“Applications(应用)”类别的成员),而视觉元素 320 显示“Data files this employee is responsible for(该雇员负责的数据文件)” (“数据文件”类别的成员)。

[0082] 视觉元素的三种类型是: simple, view, decorate。simple- 类型的视觉元素包括这样一种元数据,其基于该关联类别中或者通过一个或多个主关键字/外关键字关系与该关联的类别相关的类别中的数据对象的注释规则数值。view- 类型的视觉元素包括在对一类别预先定义的视窗内指定的元数据,其中该类别通过一个或多个主关键字/外关键字关系与该关联的类别相关。decorate- 类型的视觉元素包括将呈现在用于视窗的网页上的文本(例如对网页的介绍)。

### [0083] 3.1 指定视觉元素

[0084] add-view-element 命令用于定义一个或多个视觉元素,并且使所述视觉元素与关于指定类别的现有视窗相关联。用户以视觉元素将在用户接口模块 108 中显示的顺序增加视觉元素。该命令的语法是: add-view-element categoryview element-type label element-descriptor element-view。该命令参数定义如下:

[0085] category: 现有类别的名称。

[0086] view: 根据要增加的该视觉元素的 category 定义的现有视窗的名称。

[0087] element-type: 指定 simple- 类型、view- 类型或者 decorate- 类型的其中之一。

[0088] label: 描述元数据项目的文本字符串。除了 decorate- 类型视觉元素的 label 之外,该 label 的文本显示在呈现的浏览器页面中。

[0089] element-descriptor: 描述将被检索的元数据(除了 decorate- 类型)。

[0090] element-view: (只用于 element-type 是 view- 类型的情况) 将包括的视窗的名称。

### [0091] 3.2 元数据选择处理(action)

[0092] 用户接口模块 108 为用户提供语法,以指定能够使用户接口模块 108 检索期望的元数据的元素描述符。该元素描述符包括多个处理的组合。用户接口模块 108 执行与标识包含一个或多个数据对象的一组数据对象的起始上下文相关的处理组合。例如,起始上下

文可以包括用户通过屏幕链接选择的数据对象,或者起始上下文可以包括与用户选择的类别相关联的一组数据对象。

[0093] 第一处理作用于该起始的一组数据对象。每个后面的处理(假设有)作用于从在先处理返回的输出。从处理返回的输出可以包括一个或多个数据对象的一组数据对象(例如以数据对象的主关键字数值的形式)、注释规则的一个或多个数值或者计算结果(例如从在先处理输出的数据对象的数目的计数)。返回的一个或多个数据对象的一组数据对象和/或注释规则数值成为后面处理的“起始组”。某些类型的处理(例如以下描述的“导航处理”)作用于只包含数据对象的起始组。

[0094] 处理组合的语法包括处理标识符(标识处理的关键字)和任何参数(arguments)(某些处理没有参数)。表示对应处理组合的处理标识符的序列可以选择地包括单个处理或由间隔(“.”符号)分开的多个处理,例如:

[0095] `action1(<arguments>).action2(<arguments>).action3()`

[0096] 3.2.1 导航处理

[0097] “漫游”(“Stroll”)是一个或多个“导航处理”的组合,并且表示从起始的一个或多个数据对象的一组数据对象到目的的一个或多个数据对象的一组数据对象的导航规范。导航处理包括 walk 处理和 inv\_walk 处理,所述 walk 处理和 inv\_walk 处理被定义为:

[0098] `walk(arule, cat)`:返回每个起始数据对象的注释规则 arule 引用的在类别 cat 中的数据对象。

[0099] `inv_walk(arule, cat)`:返回具有注释规则 arule 的在类别 cat 中的数据对象,其中所述注释规则 arule 引用该起始的一组数据对象的任何成员。

[0100] 例如,如果起始对象是管理者并且期望的数据是直接向该管理者报告的雇员,则对应的元素描述符是以下具有单个导航处理的漫游:

[0101] `inv_walk(managed_by KET, Employee)`

[0102] 如果起始对象是项目并且所期望的数据是做该项目的雇员的所有管理者,则对应的元素描述符是以下具有多个导航处理的漫游:

[0103] `inv_walk(projectKEY, Employee).\`

[0104] `walk(managed_byKEY, Employee)`

[0105] 这个实例示出用反斜线(“\”符号)指示元素描述符在新的一行的延续。

[0106] 某些导航处理通过执行指定的循环(recursively)漫游在数据对象中导航。例如,两个循环导航处理是 gather\_all 和 find\_ultimate,描述如下。

[0107] `gather_all(stroll)`:执行 stroll 指定的循环导航处理并且返回在循环的每个步骤发现的所有数据对象。

[0108] `find_ultimate(stroll)`:执行 stroll 指定的循环导航处理并且返回在循环末尾发现的所有“叶节点”数据对象(所述指定的导航处理返回空输出的数据对象)。

[0109] 3.2.2 表达式处理

[0110] 某些处理利用与在脚本语言(例如 Python)中使用的表达式类似的“表达式”来作用于起始组。以下是作用于起始的一个或多个起始数据对象的一组数据对象和/或注释规则数值的表达式处理的实例。

[0111] `sort(expr)`:返回通过表达式 expr(例如 python 表达式)排序的起始组。

[0112] `rsort(expr)` :返回通过表达式 `expr` 以逆序排序的起始组。

[0113] `filter(expr)` :通过表达式 `expr` 过滤起始组。

[0114] `eval(expr)` :对起始组计算表达式 `expr`。

[0115] `avg(expr)` :返回起始组的表达式 `expr` 的平均值。

[0116] 作用于起始组的其它处理没有任何参数。例如：

[0117] `count()` :返回起始组中项目的数目。

[0118] 除了例如标准 Python 语法的语法之外,表达式可以包括可以对起始组执行功能的功能语法(例如关键字)。以下是这种功能语法的实例。

[0119] 关键字 `ann` 与注释规则 `arule` 组合为 `ann.arule`,以返回注释规则 `arule` 的数值。例如出现在表达式 `expr` 中的 `ann.key` 返回该表达式应用的数据对象的 `key` 注释规则的数值。在另一实例中,对于是雇员类别的成员的一组数据对象,以下元素描述符返回女性的雇员:`filter(ann.gender == "F")`。

[0120] 在被应用到起始组中每个项目的表达式中出现的关键字 `position` 返回起始组中项目的有序列表中每个项目的位置。例如,以下元素描述符返回起始组中五个最老的雇员：

[0121] `Sort(ann.start_date).filter(position < 5)`

[0122] 首先, `sort` 处理按照升序排列起始日期的数值列表。最后过滤处理将表达式 '`positon < 5`' 应用到 `sort` 返回的数值列表并且返回最开始的五个数值(位置 0 至 4)。

[0123] 关键字 `look_ahead` 提供基于起始组中不能直接得到的数据对象过滤的功能。包括在 `filter` 处理中的表达式 `look_ahead(stroll)` 将 `stroll` 中的导航处理应用到起始组中的每个数据对象,并且返回能用于过滤该起始组中的数据对象的一组数据对象。如果过滤处理只包含 `look_ahead(stroll)` 表达式,则基于 `stroll` 中的导航处理是否从该起始数据对象返回任一数据对象来过滤该起始组中的起始数据对象。假如是这样,则过滤器返回起始数据对象。如果不是,则过滤器不返回(“过滤掉”)该起始数据对象。

[0124] 例如,“雇员”类别的视窗可以包括关于雇员管理者的信息。为了基于雇员管理者的信息进行过滤,过滤器将使用关键字 `look_ahead` 查看关于雇员管理者的信息。描述其上至少有两级管理的所有雇员组的元素描述符是：

[0125] `filter(look_ahead(walk(managed_byKEY, Employee). \`

[0126] `walk(managed_byKEY, Employee).count()) > 0)`

[0127] 4. 实例

[0128] 以下给出几个实例,所述实例示出用于检索视窗的期望信息的元素描述符的结构。

[0129] 4.1 实例 1

[0130] 从包括该雇员的上下文起始的雇员部门中检索雇员数目的元素描述符为：

[0131] `walk(departmentKEY, Department).inv_walk(departmentKEY,`

[0132] `Employee).count()`

[0133] `walk` 导航处理得到包括该雇员的部门对象。然后 `inv_walk` 导航处理得到在该部门的所有雇员。然后 `count` 处理得到这种图形的数目。

[0134] 4.2 实例 2

[0135] 在以下实例中,导航处理得到起始上下文给定的部门中的最老的雇员:

```
[0136] inv_walk(departmentKEY, Employee).sort(ann.start_date).\
```

```
[0137] filter(position < 1)
```

[0138] inv\_walk 导航处理对该部门中的所有雇员后向应用 departmentKEY 注释规则。然后 rsort 处理按照他们的起始日期—从最老到最新的顺序逆序排序这些雇员。然后 filter 处理作用于已排序的雇员列表,并且只选择列表中的第一个对象。结果是部门中最老的雇员。

[0139] 4.3 实例 3

[0140] 在以下实例中,导航处理得到起始部门中最老的女性雇员:

```
[0141] inv_walk(departmentKEY, Employee).\
```

```
[0142] filter(ann.gender == "F").\
```

```
[0143] sort(ann.start_date).filter(position < 1)
```

[0144] 这个实例包括附加的 filter 处理。在得到该部门中的所有雇员之后,第一 filter 处理选择女性雇员。然后 sort 对女性雇员的列表进行逆序排序—从最老到最新的顺序排列雇员。在作用于已排序的女性雇员期间,第二 filter 处理丢弃除最老雇员以外的所有雇员。

[0145] 4.4 实例 4

[0146] 在某些情况,得到期望的信息的处理包括过滤掉在元素描述符的给定上下文中不能直接得到的数值。例如,假定一组雇员,期望可以得到这些雇员的管理者列表。诸如管理者性别的信息可以从这些管理者数据对象中得到。为了支持这种查询,可通过如上所述关键字 look\_ahead 来使用 filter 处理。以下示范性的元素描述符通过来自包括所有雇员的起始上下文的 look\_head 关键字来使用 filter 处理,从而得到其管理者是男性的雇员:

```
[0147] filter(look_ahead(walk(managed_byKEY, Employee).\
```

```
[0148] filter(ann.gender == "M")))
```

[0149] 使用 look\_ahead 关键字过滤该组雇员。通过 look\_ahead 的漫游描述了可以用于得到感兴趣的数据对象的一个或多个导航处理的序列。在这种情况下,漫游中的第一 walk 符合跟随雇员数据对象的 managed\_byKEY 注释规则。下一 walk 处理根据性别 (gender) 注释规则的数值过滤掉这些对象。如果雇员的管理者不是男性,则它们从结果组中被过滤掉。

[0150] 4.5 实例 5

[0151] 用于检索其性别不同于他们的管理者的所有雇员的元素描述符:

```
[0152] filter(look_ahead(walk(managed_byKEY, Employee).\
```

```
[0153] filter(ann.gender == "M"))).filter(ann.gender == "F")+
```

```
[0154] filter(look_ahead(walk(managed_byKEY, Employee).\
```

```
[0155] filter(ann.gender == "F"))).filter(ann.gender == "M")
```

[0156] 在这个实例中,“+”符号集合了多个漫游的结果。第一漫游与实例 4 类似,但是其管理者是男性的该组所有雇员进一步被过滤,以只返回该组中的女性雇员。第二漫游得到其管理者是女性的所有雇员,然后过滤该组,以只返回男性的雇员。

[0157] 4.6 实例 6

[0158] 在某些情况,可能希望看到循环执行的 walk 的结果。例如,可能希望不仅看到管

理者的直接报告,而且还有他的直接报告的直接报告等 adinfinitum。元素描述符可以使用 gather\_all 导航处理得到该信息,所述 gather\_all 导航处理为:

[0159] gather\_all(inv\_walk(managed\_byKEY, Employee))

[0160] 在这个实例中,漫游从一组雇员到由该组雇员管理的所有雇员。gather\_all 处理使该漫游循环执行,从而答案组是公司体系中起始雇员组“下面”的所有雇员组。

[0161] 5 实现方法

[0162] 可以使用在计算机上执行的软件来实现上述的数据对象管理技术。例如,该软件构成一个或多个计算机程序中的过程,所述计算机程序在一个或多个已编程或可编程计算机系统(其可以是各种结构例如分布式、客户机/服务器、或网格)上执行,每个计算机系统包括至少一个处理器、至少一个数据存储系统(包括易失性和非易失性存储器和/或存储元件)、至少一个输入设备或端口以及至少一个输出设备或端口。该软件可以构成较大程序的一个或多个模块,例如该较大程序提供与计算图形的设计和结构相关的其它服务。根据数据储存库中存储的数据模型,可以将图形的节点和组件实现为在计算机可读媒介中存储的数据结构或其它组织数据。

[0163] 该软件可以设置在由通用或专用可编程计算机读取的媒介(例如 CD-ROM)上,或者通过网络传送(编码为传播信号)到执行软件的计算机。所有函数可以在专用计算机上执行,或者使用诸如协处理器的专用硬件执行。可以以分布式的方式实现该软件,其中由软件指定的不同计算部分由不同的计算机来执行。每个这种计算机程序优选为存储到或者下载到通用或者专用可编程计算机可读的存储媒介或者设备(例如固态存储器或者媒介或者磁或光媒介),以在所述存储媒介或者设备被计算机系统读取时配置并运行该计算机,从而执行在此描述的程序。也可以认为本发明的系统可以被实现为配置有计算机程序的计算机可读存储媒介,其中这样配置的存储媒介使计算机系统以特定的和预定的方式运行,以执行在此描述的功能。

[0164] 应该理解上述说明书旨在说明,而并非限制本发明的保护范围,本发明的保护范围由所附权利要求书来限定。其它实施例也落在所附权利要求书的范围内。



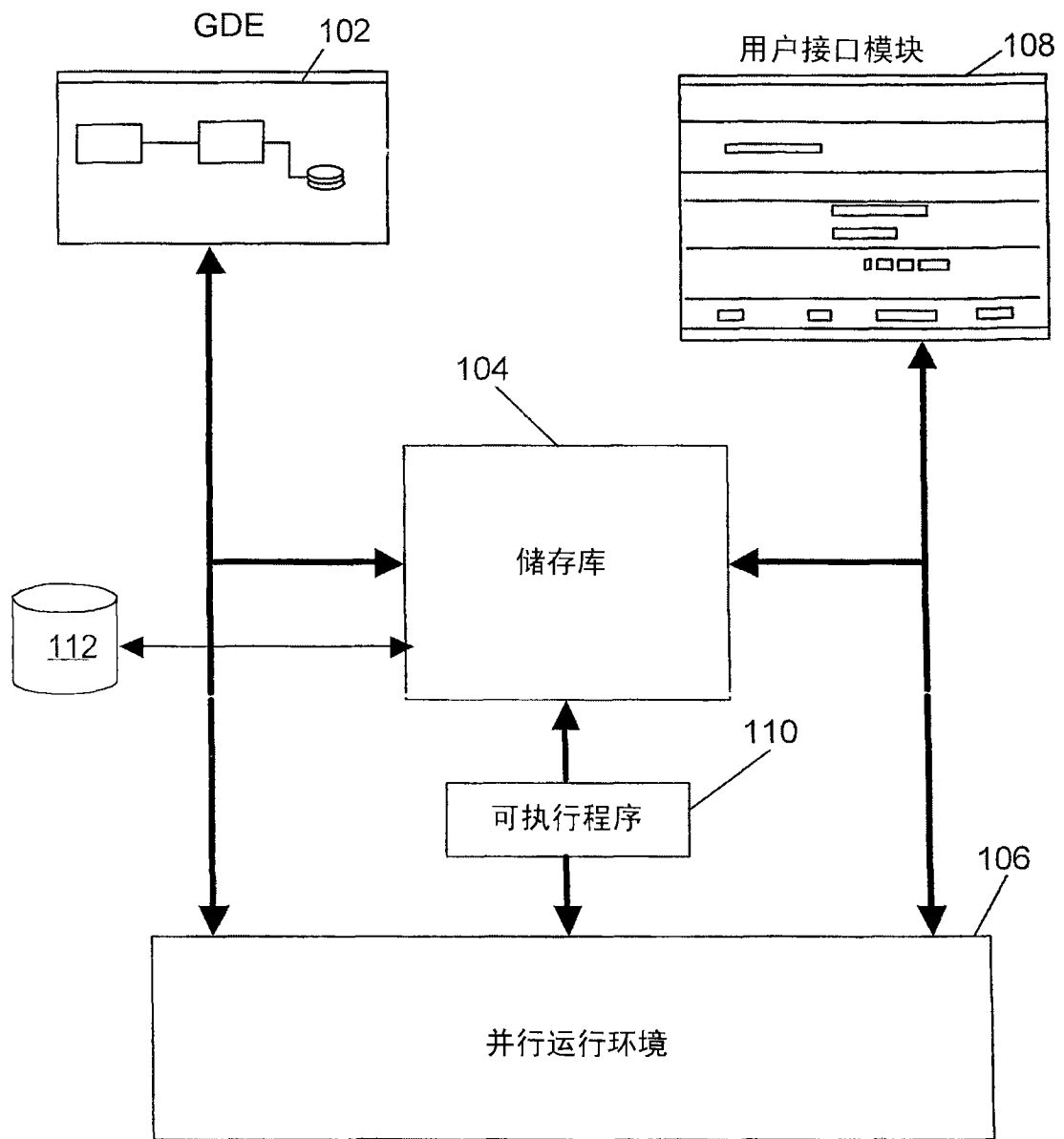


图 1A

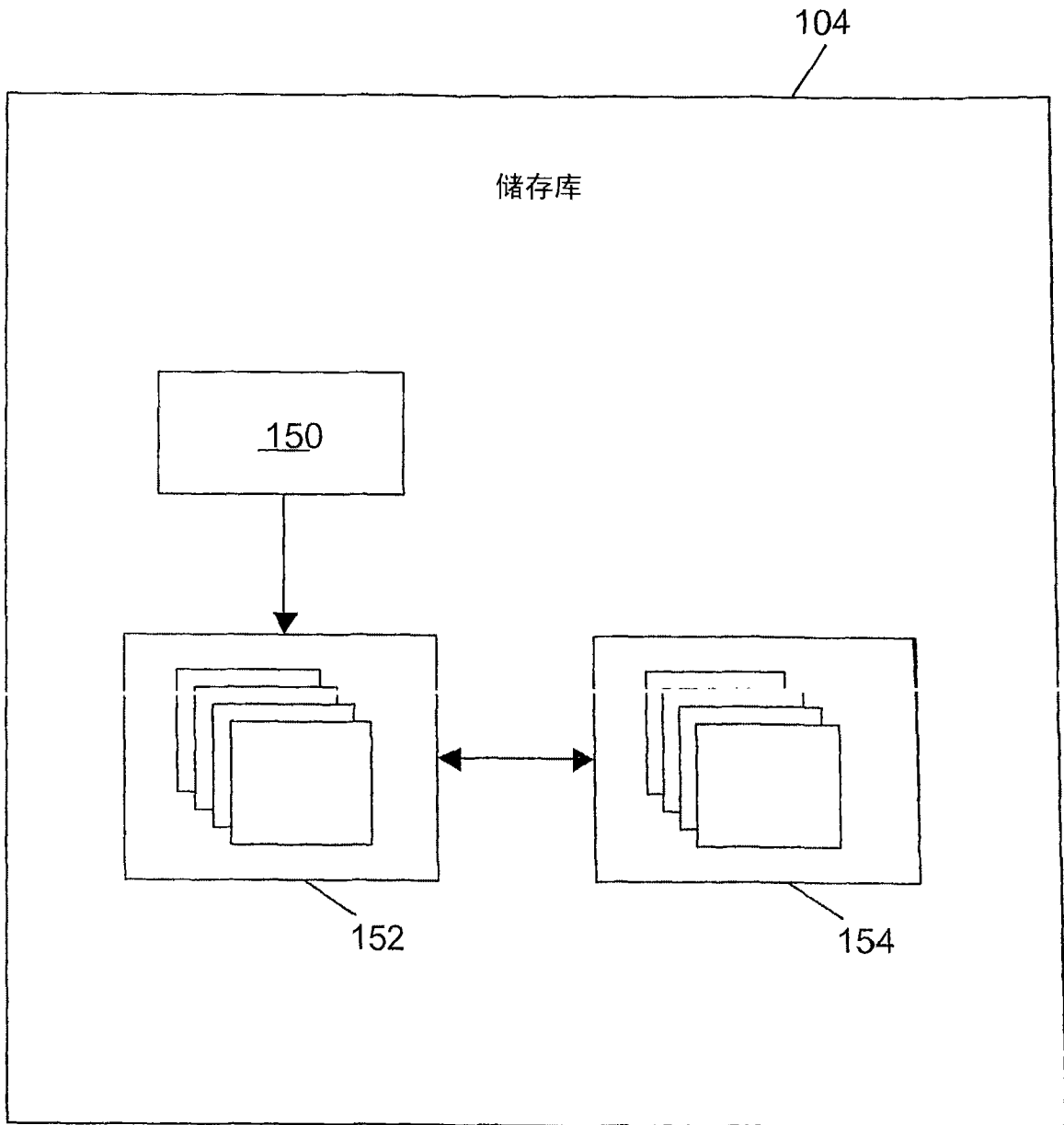


图 1B

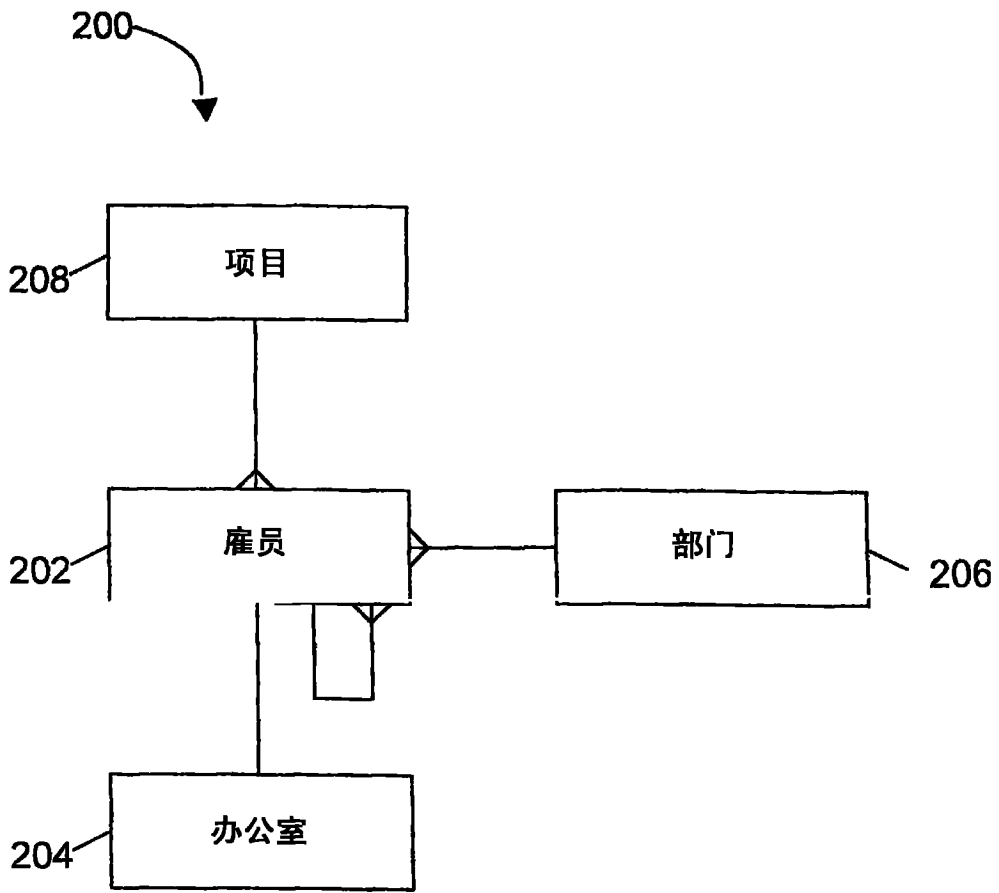


图 2

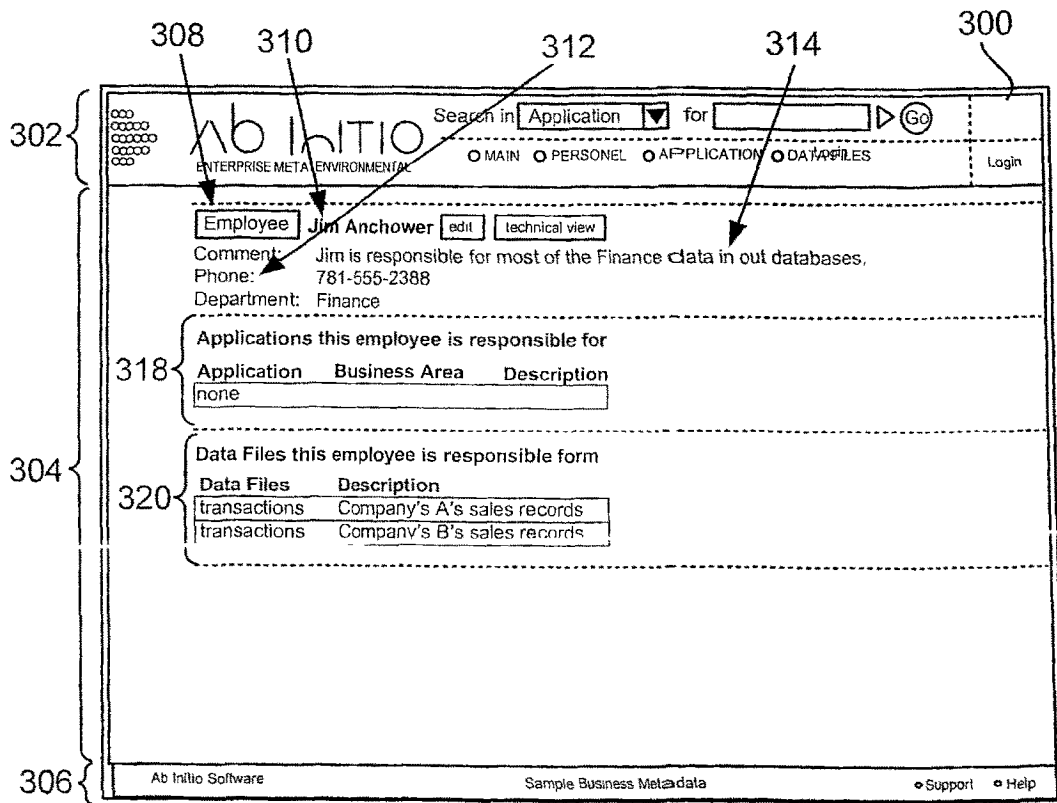


图 3