

[19]中华人民共和国国家知识产权局

[51]Int. Cl⁷

G06F 17/30

[12]发明专利申请公开说明书

[21]申请号 98804865.5

[43]公开日 2000年5月31日

[11]公开号 CN 1255215A

[22]申请日 1998.5.6 [21]申请号 98804865.5

[30]优先权

[32]1997.5.6 [33]US [31]08/852,099

[86]国际申请 PCT/NO98/00139 1998.5.6

[87]国际公布 WO98/50866 英 1998.11.12

[85]进入国家阶段日期 1999.11.5

[71]申请人 伯特斯泰普技术联合股份有限公司

地址 挪威奥斯陆

[72]发明人 奥拉夫·韦德

[74]专利代理机构 上海专利商标事务所

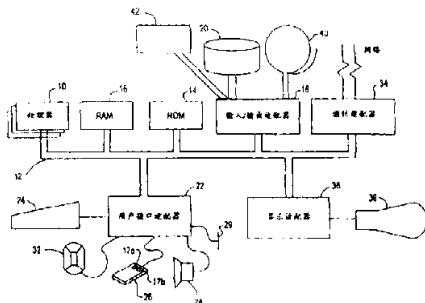
代理人 徐 泰

权利要求书 5 页 说明书 11 页 附图页数 9 页

[54]发明名称 在信息处理系统中存储和操作数据的系统
和方法

[57]摘要

本发明针对一种数据库和数据库管理系统,它设计来存储和操作任何类型的数据和数据类型的任何组合。基础数据结构是唯一柔性的,因此对于DBMS,数据库中的数据将以与DBMS的存取方法相符的类型出现。此外,本发明允许对于同一基础数据同时有不同的存取方法,使得它可以在一个容易管理的系统内与复杂数据一起工作。本发明的一个关键方面是数据的“原子化”。原子化是根据数据内容的定义以及根据它在数据库中的出现或场合,把数据内容的存储加以分开。在数据库中,原子是最基本的元素。数据内容存储在内容原子中,数据定义存储在类型原子中,而同一数据值/性质的每个场合由场合原子表示。当连接时,三个不同的原子类型形成一个分子。在数据库中,可以用把数个分子的场合原子连接在一起形成内部关系,并且再把内部关系连接在一起形成外部关系的方式来表示复杂数据。



专利文献出版社出版

权 利 要 求 书

1. 一种用于存储数据的存储器，所述数据在信息处理系统中通过程序执行而被存取，其特征在于，所述存储器包括：

存储在所述存储器中的一种数据结构，所述数据结构包括被所述程序使用的驻留在数据库中的信息，并且包括：

多个内容元素，其中，每个所述内容元素包含一个唯一的数据项；

多个类型元素，其中，每个所述类型元素存储类型数据，并且每个所述类型元素包括一个唯一的类型处理；以及

多个场合元素，其中，每个所述场合元素连接一个所述内容元素和一个所述类型元素。

2. 如权利要求 1 所述的存储器，其特征在于，每个所述数据项被唯一存储在仅仅一个所述内容元素中。

3. 如权利要求 1 所述的存储器，其特征在于，所述数据结构还包括一个或数个内部关系，其中，每个所述内部关系包括一个或数个连接在一起的所述场合元素。

4. 如权利要求 3 所述的存储器，其特征在于，所述数据结构还包括一个或数个外部关系，其中，每个所述外部关系包括一个或数个由连接手段连接在一起的所述内部关系。

5. 如权利要求 4 所述的存储器，其特征在于，所述连接手段包括在第一内部关系中的一个所述内容原子，所述第一内部关系包含一个指向在第二内部关系中的一个所述场合原子的指针。

6. 如权利要求 1 所述的存储器，其特征在于，所述数据结构还包括一个或数个历史链接，其中，每个所述历史链接把一个新的所述场合元素连至一个老的所述场合元素。

7. 如权利要求 1 所述的存储器，其特征在于，所述数据结构还包括数据词典，其中，所述类型元素存储在所述数据词典中。

8. 如权利要求 1 所述的存储器，其特征在于，所述数据结构还包括搜索结构，所述搜索结构包括：

一个或数个小结构，其中，每个所述小结构包含一个或数个连接在一起的所述内容元素；以及

对于每个所述小结构有一个矢量，该矢量包括一个或数个数据字节，其中，

所述矢量确定对于所要的所述内容元素要搜索哪个所述小结构.

9. 一种用于存储数据的计算机可读取的媒体，所述数据在信息处理系统中通过程序执行而被存取，其特征在于，所述计算机可读取的媒体包括：

存储在所述计算机可读取的媒体上的一种数据结构，所述数据结构包括被所述程序使用的驻留在数据库中的信息，并且包括：

多个内容元素，其中，每个所述内容元素包含一个唯一的数据项；

多个类型元素，其中，每个所述类型元素存储类型数据，并且每个所述类型元素包括一个唯一的类型处理；以及

多个场合元素，其中，每个所述场合元素连接一个所述内容元素和一个所述类型元素。

10. 如权利要求 9 所述的计算机可读取的媒体，其特征在于，每个所述数据项被唯一存储在仅仅一个所述内容元素中。

11. 如权利要求 9 所述的计算机可读取的媒体，其特征在于，所述数据结构还包括一个或数个内部关系，其中，每个所述内部关系包括一个或数个连接在一起的所述场合元素。

12. 如权利要求 9 所述的计算机可读取的媒体，其特征在于，所述数据结构还包括一个或数个外部关系，其中，每个所述外部关系包括一个或数个由连接手段连接在一起的所述内部关系。

13. 如权利要求 12 所述的计算机可读取的媒体，其特征在于，所述连接手段包括在第一内部关系中的一个所述内容原子，所述第一内部关系包含一个指向在第二内部关系中的一个所述场合原子的指针。

14. 如权利要求 9 所述的计算机可读取的媒体，其特征在于，所述数据结构还包括一个或数个历史链接，其中，每个所述历史链接把一个新的所述场合元素连至一个老的所述场合元素。

15. 如权利要求 9 所述的计算机可读取的媒体，其特征在于，所述数据结构还包括数据词典，其中，所述类型元素存储在所述数据词典中。

16. 如权利要求 9 所述的计算机可读取的媒体，其特征在于，所述数据结构还包括搜索结构，所述搜索结构包括：

一个或数个小结构，其中，每个所述小结构包含一个或数个连接在一起的所述内容元素；以及

对于每个所述小结构有一个矢量，该矢量包括一个或数个数据字节，其中，所述矢量确定对于所要的所述内容元素要搜索哪个所述小结构。

17. 一种信息处理系统，其特征在于，所述信息处理系统包括：
一个或数个处理器；
操作系统的一个或数个图象，所述图象用于控制所述处理器的运行；
在所述处理器中执行的一个或数个程序；
存储装置；以及
存储在所述存储装置中的一种数据结构，所述数据结构包括被所述程序使用的驻留在数据库中的信息，并且包括：
多个内容元素，其中，每个所述内容元素包含一个唯一的数据项；
多个类型元素，其中，每个所述类型元素存储类型数据，并且每个所述类型元素包括一个唯一的类型处理；以及
多个场合元素，其中，每个所述场合元素连接一个所述内容元素和一个所述类型元素。
18. 如权利要求 17 所述的信息处理系统，其特征在于，其特征在于，每个所述数据项被唯一存储在仅仅一个所述内容元素中。
19. 如权利要求 17 所述的信息处理系统，其特征在于，所述数据结构还包括一个或数个内部关系，其中，每个所述内部关系包括一个或数个连接在一起的所述场合元素。
20. 如权利要求 19 所述的信息处理系统，其特征在于，所述数据结构还包括一个或数个外部关系，其中，每个所述外部关系包括一个或数个由连接手段连接在一起的所述内部关系。
21. 如权利要求 20 所述的信息处理系统，其特征在于，所述连接手段包括在第一内部关系中的一个所述内容原子，所述第一内部关系包含一个指向在第二内部关系中的一个所述场合原子的指针。
22. 如权利要求 17 所述的信息处理系统，其特征在于，所述数据结构还包括一个或数个历史链接，其中，每个所述历史链接把一个新的所述场合元素连至一个老的所述场合元素。
23. 如权利要求 17 所述的信息处理系统，其特征在于，所述数据结构还包括数据词典，其中，所述类型元素存储在所述数据词典中。
24. 如权利要求 17 所述的信息处理系统，其特征在于，所述数据结构还包括搜索结构，所述搜索结构包括：
一个或数个小结构，其中，每个所述小结构包含一个或数个连接在一起的所述内容元素；以及

对于每个所述小结构有一个矢量，该矢量包括一个或数个数据字节，其中，所述矢量确定对于所要的所述内容元素要搜索哪个所述小结构。

25. 如权利要求 17 所述的信息处理系统，其特征在于，所述信息处理系统还包括用于将所述数据项添加至所述数据结构的手段，所述添加用的手段包括：

- 用于接收所述数据项的手段；
- 用于创建包含数据元素的新的所述内容元素的手段；
- 用于创建连至新的所述内容元素的新的所述场合元素的手段；以及
- 用于把新的所述内容元素和适当的所述类型元素连至新的所述场合元素的手段。

26. 如权利要求 25 所述的信息处理系统，其特征在于，所述信息处理系统还包括：

- 用于确定所述数据项是否已存在于所述数据结构的所述内容元素之中的手段；以及

用于把新的所述场合元素连至现有的所述内容元素的手段。

27. 如权利要求 17 所述的信息处理系统，其特征在于，所述信息处理系统还包括用于搜索所述数据结构以找出所要的所述数据项的手段，所述手段包括：

- 用于根据在所要的所述数据项中的一个或数个字节选择要搜索的小结构的手段；以及

用于搜索所述小结构以找出所要的所述内容元素的手段。

28. 如权利要求 17 所述的信息处理系统，其特征在于，所述信息处理系统还包括用于在所述数据结构中更新所述内容元素的手段，所述用于更新的手段包括：

- 用于创建新的所述内容元素的手段；
- 用于创建新的所述场合元素的手段；
- 用于把新的所述内容元素连至新的所述场合元素的手段；以及
- 用于把新的所述场合元素连至老的所述场合元素的手段。

29. 如权利要求 17 所述的信息处理系统，其特征在于，所述信息处理系统还包括用于在所述数据结构中更新所述内容元素的手段，所述用于更新的手段包括：

- 用于在所述内容元素中改变所述数据项的手段；以及
- 用于把经过改变的内容元素重新连至现有的场合元素的手段。

30. 一种处理数据库中的数据的方法，其特征在于，所述方法包括下述步骤：

创建多个内容元素，其中，每个所述内容元素包含一个唯一的数据项；

创建多个类型元素，其中，每个所述类型元素存储类型数据，并且每个所述类型元素包括一个唯一的类型处理；以及

创建多个场合元素，其中，每个所述场合元素连接一个所述内容元素和一个所述类型元素。

31. 如权利要求 30 所述的方法，其特征在于，每个所述数据项唯一存储在仅仅一个所述内容元素中。

32. 如权利要求 30 所述的方法，其特征在于，所述方法还包括把新的所述数据项添加至所述数据库的步骤，所述添加步骤包括下述步骤：

接收所述数据项；

创建包含数据元素的新的所述内容元素；

创建连至新的所述内容元素的新的所述场合元素；以及

把新的所述内容元素和适当的所述类型元素连至新的所述场合元素。

33. 如权利要求 32 所述的方法，其特征在于，所述方法还包括：

确定所述数据项是否已存在于所述数据库的所述内容元素之中；以及

把新的所述场合元素连至现有的所述内容元素。

34. 如权利要求 30 所述的方法，其特征在于，所述方法还包括搜索所述数据库，以找出所要的所述数据项的步骤，所述搜索包括下述步骤：

根据在所要的所述数据项中的一个或数个字节选择要搜索的小结构；以及

搜索所述小结构，以找出所要的所述内容元素。

35. 如权利要求 30 所述的方法，其特征在于，所述方法还包括更新所述数据库中的所述内容元素的步骤，所述更新步骤包括下述步骤：

创建新的所述内容元素；

创建新的所述场合元素；

把新的所述内容元素连至新的所述场合元素；以及

把新的所述场合元素连至老的所述场合元素。

36. 如权利要求 30 所述的方法，其特征在于，所述方法还包括更新所述数据库中的所述内容元素的步骤，所述更新步骤包括下述步骤：

改变所述内容元素中的所述数据项；以及

把经过改变的内容元素重新连至现有的场合元素。

说 明 书

在信息处理系统中存储和操作数据的系统和方法

发明领域

本发明涉及信息处理系统，本发明尤其涉及在信息处理系统中存储和操作数据的系统和方法。

发明背景

一个数据库包括一个或多个庞大的持久的数据组。一般，用户能够使用与数据库相关联的软件来更新和查询数据库。

数据库是由数据库管理系统(DBMS)存储的数据。DBMS是在数据库中控制数据的组织、存储和检索的一组软件程序。DBMS也控制数据库的安全和完整。一般，DBMS也提供交互查询功能，它允许用户对来自数据库的数据进行交互搜索和分析。

有数种现有技术可用于在数据库中组织数据。三种最常见类型的现有技术数据库是层次数据库、网络数据库和关系数据库。DBMS可以提供这些或其他类型的数据组织中的一种或数种。

在层次数据库中，把数据项目称为记录，并且存储在树形结构中。层次数据库以与典型的组织图类似的方式把记录连接在一起。这意味着每个记录只能被一个所有者记录所拥有。例如，一个部门记录可以“拥有”十五个雇员记录。然而，每个雇员记录只可以被一个所有者记录(在此情形中是被其部门记录)所拥有。这样使得使用层次数据库难于模拟真实世界情形。例如，一个雇员可以既是一个部门的一员，又是由来自数个部门的雇员组成的一个小组的一员。然而，层次数据库不允许同一个雇员记录既被部门记录拥有又被小组记录拥有。

网络数据库与层次数据库相类似，然而，可以自由互连数据记录，而无需把数据记录置于树形结构中。在网络数据库中，一个雇员记录能够既被一个部门记录拥有又被一个小组记录拥有。

层次数据库和网络数据库在搜索时都很费时并且难于改变。在层次数据库或网络数据库中改变数据结构一般需要关闭数据库再重建它。

另一种类型的现有技术数据库是关系数据库。在关系数据库中，把所有的数据存储在称之为关系的简单的表中。关系数据库把在层次数据库和网络数据

库中发现的记录之间的复杂的关系都去除了。在关系数据库中的记录的设计提供了一个用于匹配的公共字段(诸如雇员编号)。通常，为了加快搜索，对用于匹配的字段加以索引。

然而，关系数据库有几个缺点。对于许多数据结构(即，网络型数据结构)而言，关系数据库很复杂并且不自然。由于许多字段存储在多于一个的关系中，关系数据库是冗余的。虽然使用索引字段能够提高查询速度，但存储索引所需的空间有时会变得显著大于在数据库中存储数据所需的空间。索引的使用也是冗余的。当数据库有大量的数据要更新时，这种冗余性以及数据字段在多于一个关系中的冗余存储能够造成性能的降低。最后，由于必须频繁地重新组织数据以保持性能可以接受，关系数据库的行政管理费用很高。

由于现有技术数据库(诸如层次数据库、网络数据库和关系数据库)的许多缺点，一些软件制造商已经开始开发面向对象的数据库(OODB)。然而，现时存在的 OODB 使用传统的存储技术(即，关系的和其他的存储技术)来实际存储数据。当前的 OODB 和对象数据库管理系统(ODMS)对于老的数据库技术和老的数据库管理系统而言确实是面向对象的接口。当前的 OODB 和 ODMS 实际尝试(虽然不很成功)使用当今存在的技术(即，关系的和其他的技术)来存储面向对象的数据。

结果，希望有一个数据库以及用于组织数据的数据库管理系统和方法，从而使得它能被访问，好比它是任何一种可能的数据库组织，包括上述的那些数据库组织。如果该系统和方法能够同时支持多种不同的存取方法和存储/检索语法，则是所希望的。此外，如果该系统和方法允许快速而有效的搜索、动态方案演变而无需脱机重建数据库、以及自动产生历史，则是所希望的。

发明概要

因此，本发明致力于数据库以及数据库管理系统和方法，设计来存储和操作任何类型的数据(即，文本数据、数字数据、空间数据、图形数据、等等)和这些数据类型的任何组合。基础数据构造是唯一柔性的，因此，对于 DBMS，在数据库中的数据将以与 DBMS 的存取方法相符的类型出现(当然，虽然数据库的基础结构不变)。此外，本发明允许对于相同的基础数据同时有多种不同的存取方法，提供在一个容易管理的系统中用复杂的数据进行工作的能力。

本发明的一个关键的方面是数据的“原子化”(atomization)。原子化是根据数据的定义以及根据它在数据库中的出现或场合，把数据内容的存储加以分

解。在数据库中的最基本的元素是原子。数据内容存储在内容原子中，数据定义存储在类型原子中，而同一数据值/性质的每个场合用场合原子来表示。当连接时，三种不同的原子类型构成一个分子。可以在数据库中通过把来自数个分子的场合原子连在一起而形成内部关系，然后再把内部关系连接在一起而形成外部关系来表示复杂的数据。

本发明的一个优点是对于数据库中的每个场合原子可以自动地保持其单独的按时间顺序的历史。当变更一个分子的内容原子(即数据值)时，数据库将(按用户的选择)或者把场合原子重新连至新的内容原子，或者用新的内容产生新的场合，并且将老的场合接入历史链接。

本发明的另外一个优点是唯一的搜索结构，它允许对数据库进行快速而有效的搜索。对于每个内容原子，DBMS 使用数据中的‘n’个最高有效字节作为进入系统的搜索结构的一个矢量。DBMS 把搜索结构分为‘m’个分开的小结构，这里‘m’是‘n’个最高有效字节的范围。然后把实际的内容原子存储在其矢量的小结构中。

附图概述

通过下面对实现本发明的最佳方式的详细描述本发明的上述的和其他的特征和优点将变得更加显然。在下面的描述中，将要参照附图，在各个附图中，用相同的标号来识别相同的部分，其中：

- 图 1 是能够存储和操作本发明的原子数据库的信息处理系统的方框图；
- 图 2 描绘了内容原子、类型原子和场合原子连接在一起而形成一个分子；
- 图 3 描绘了数个分子连接在一起而形成一个内部关系；
- 图 4 描绘了两个内部关系连接在一起而形成一个外部关系；
- 图 5 说明使用历史连接来保持场合历史的方式；
- 图 6 描绘了本发明的数据库的内部成分；
- 图 7 描绘了有关本发明的数据库的内部结构的进一步的细节；
- 图 8 描绘了用于提高搜索性能的数据库的内部结构；
- 图 9 描绘了本发明的搜索结构
- 图 10 描绘了本发明的暴露存取方法；
- 图 11 是说明按照本发明的创建数据库的方法的流程图；
- 图 12 是说明把内容/场合原子添加至数据库的方法的流程图；
- 图 13 是说明搜索数据库的方法的流程图；

图 14 是说明更新内容/场合原子的方法的流程图。

较佳实施例的详细描述

可以在包括个人计算机、工作站、小型计算机和主计算机等许多硬件平台上实现本发明。本发明的方法的许多步骤可以在各种类型的并行处理器上有利地实现。现在参看图 1，下面将描述可以用来实现本发明的新颖方法的一种信息处理系统的典型结构。图 1 的计算机系统至少有一个处理器 10。处理器 10 通过系统总线 12 与随机存取存储器(RAM)16、只读存储器(ROM)14、输入/输出(I/O)适配器 18、用户接口适配器 22、通信适配器 34 以及显示适配器 36 互连，I/O 适配器 18 用于将诸如盘片装置 20、磁带驱动器 40 和打印机 42 等外围设备连至总线 12，用户接口适配器 22 用于将键盘 24、具有按钮 17a 和 17b 的鼠标器 26、扬声器 28、话筒 32 和/或其他用户接口装置(诸如触摸屏装置 29)连至总线 12，通信适配器 34 用于将信息处理系统连至数据处理网络，而显示适配器 36 用于将显示装置 38 连至总线 12。通信适配器 34 可以把图 1 所示的系统与成百上千个类似的系统或其他装置(诸如远程打印机、远程服务器或远程存储装置)相连。

设计本发明的系统和方法，以存储和操作任何类型的数据(即文本数据、数字数据、空间数据和图形数据)和数据类型的任何组合。基础数据结构是唯一柔性的。结果，本发明的数据库管理系统(DBMS)能够是关系 DBMS，对象 DBMS，层次 DBMS 或任何其他的 DBMS。此外，可以用每一种结构的存储/检索语法(例如对于关系 DBMS 的 SQL，对于对象 DBMS 的 OQL，等等)从数据库存取和检索数据。对于 DBMS，数据库中的数据将以与 DBMS 的存取方法相符的类型出现(当然，虽然数据库的基础结构不变)。此外，本发明允许对于相同的基础数据有多种同时的、不同的存取方法，提供在一个容易管理的系统中用复杂的数据进行工作的能力。

本发明的一个关键的方面是数据的“原子化”(atomization)。原子化是根据数据的定义(即数据类型以及与其他数据元素的关系)以及根据它在数据库中的出现或场合，把数据内容(例如，串“John Smith”)的存储加以分解。

原子(也称为“元素”)是数据库中最基本的元素。现在参看图 2，该图描绘了三种元素(或原子类型)之间的关系。数据内容(在此情形中是“John Smith”)存储在内容原子 50 中，其定义存储在类型原子 52 中，而同一数据值/性质的每个场合用场合原子 54 来表示。当连接时，三种不同的原子类型 50、

52、54 构成分子 56。类型原子 52 和场合原子 54 包括值和指针的数据结构。DBMS 管理这些原子以及它们之间的许多可能的关系。

类型原子 52 包含场合原子 54 的定义(在现有技术术语中称为字段定义)。此定义是具有多个元素(包括类型名字和基类型)的数据结构。类型名字可以是“名字”、“出生日期”、“病人”、“供给者”等等。基类型确定了数据如何在数据库中实际表示出来(即，名字 = 文本、出生日期 = 长整数，等等)。

内容原子 50 保持数据值(即，“John Smith”、“10091962”，等等)。每个数据值在数据库中只存储一次。同一数据值的多个场合由它们的场合原子来区分(如下面参照图 6 和 8 所描述的那样)。即使一个数据值对于数个类型原子来说是共同的，它实际上仍然只存储一次。例如，“John Smith”可以是类型原子“代理人”、“父母”、“供给者”的场合，但只产生一个内容原子，它以“John Smith”作为其数据值。内容原子 50 保持数据值和指向其场合原子 54 的指针(如下面参照图 6 和 8 所描述的那样)。

场合原子 54 起着在类型原子 52 和内容原子 50 之间的连接器的作用。每个场合原子 54 确定了一个组合的一次出现(即，类型原子“代理人” + 内容原子“John Smith”)，它代表了在数据库中的一个字段或对象的一个场合。

在图 2 中描绘了分子 56，该分子由 50、52 每种类型的一个原子通过一个场合原子 54 连接而成。不完整的分子(未示出)由一个场合原子 54 连至一个类型原子 52 或一个内容原子 50，而不同时连至两者而成。一个不完整的分子可以用来存储没有预定类型的数据。这样做允许有先存储数据，然后确定其类型的灵活性。一个不完整的分子还可以用来定义没有数据的场合或具有未定义数据的场合。这样做对于数据库的设计增添了在现有技术中得不到，但又是时常需要的灵活性。例如，假定 DBMS 从报纸广告的光学扫描接收到非结构数据。DBMS 不可能知道广告中的各个文本项的类型，这是因为对于每个广告来说，不同项目(电话、传真、地址、名字、描述、等等)的位置是不同的。因此，所有的项目在开始时没有类型，但由于它们的内容是已知的，用户可以搜索这些项目。在稍后时刻，可以对这些项目分派特定的类型。

现在参见图 3，该图描绘了内部关系 60。在此例中，内部关系 60 包括场合原子 54、62、64 和 66，它们用指针 68 连在一起。内部关系 60 能够用作传统记录中的数据字段的等价物，或者能够是更复杂的数据类型，诸如空间数据类型，对于“n”个坐标(coordinate)的每一个用一个原子，。

如图 3 所示，在内部关系 60 中，所有者是第一个场合原子，在这种情形下

是场合原子 54。成员是场合原子，包括所有者，它们属于内部关系。在图 3 所示的例子中，场合原子 62、64 和 66 连同所有者原子 54 是内部关系 60 的成员。

可以把内部关系连在一起而形成外部关系。可用的基类型之一是“指针”，这意味着在分子中的一个内容原子可以起着指向数据库中的特定位置的指针的作用，该特定位置通常是一个场合原子(例如，另一个内部关系中的一个场合原子)。这种能力允许在不同的内部关系之间连接，以形成无限复杂的数据结构。

现在参见图 4，该图中说明了外部关系的一例。在图 4 中，有两个所示的内部关系，1 号内部关系人物 80 和 2 号内部关系人物 82。1 号人物 80 包括三个场合原子 54、82 和 84。场合原子 54 连至内容原子 50 和类型原子 52。场合原子 82 连至内容原子 86 和类型原子 88。场合原子 84 连至内容原子 90 和类型原子 92。2 号人物 82 包括两个场合原子 94 和 96。场合原子 94 连至内容原子 98 和类型原子 100。场合原子 96 连至内容原子 102 和类型原子 104。

在图 4 描绘的例子中，1 号人物 80 是 2 号人物 82 的父母(注意类型原子 92 是“小孩”类型)。内容原子 90 包含场合原子 94 的地址(如虚线 106 所表示的)。通过使用附加的地址连接(它们类似于内容原子 90 的地址连接)，1 号人物 80 能够连至数个小孩。两个或多个内部关系的组合构成了一个外部关系，在此情形中，该外部关系代表整个家庭。

本发明的系统和方法能够对数据库中的每个场合原子自动地保持分开的按照日期顺序的历史。当变更一个分子的内容原子(即数据值)时，数据库将(按照用户的选择)或者把场合原子重新连至新的内容原子，或者用新的内容产生新的场合，并且把老的场合连入历史链接。结果，数据库能够对于数据库中的每个场合自动地保持分开的按照日期顺序的历史。

图 5 中示出场合历史形成的一例。类型原子 110 连至三个场合原子，它们是场合 1 112(它连至内容 1 114)、场合 2 116(它连至内容 2 118)、以及场合 3 120(它连至内容 3 122)。注意，历史连接 124 把场合 3 120 与场合 3 的两个较老的版本 126 和 130(连同它们的内容原子 128、132)相连。

本发明的数据库的主要内部成分示于图 6。图 6 中描绘的成分存储在图 1 的 RAM 16 中。图 6 中的实线表示了定义分子的连接，而虚线表示了定义内部关系的连接。如上所述，参见图 2，每个分子包括内容原子 150、场合原子 152

和类型原子 154。内部关系由定义场合原子 152 之间的连接而形成。下面参照图 9 将更详细地定义搜索结构 156。把类型原子 154 存储在词典 158(按照本发明，它本身是数据库)中，从而对于类型定义提供用户可扩展性(user-extensibility)。

图 7 中说明有关 DBMS 内部结构的进一步的细节。当产生每个类型原子 160 时，它包含一组特征。在所述的实施例中，这些特征包括基类型 162、类型名字 164、类型说明 166、类型处理 168、类型低值 170、类型高值 172 和类型计数(场合)174。给每个类型原子 160 分派一个唯一的类型处理 168，它是一个唯一地识别类型原子 160 的数字。类型处理 168 是包含在每个场合原子 176 中的信息的一部分，于是用类型原子 160 唯一地识别场合原子 176。

把内容原子作为数据库的整体部分加以保持。每个内容原子是分开的搜索结构的一部分，该结构从内容矢量出发(下面参照图 9 描述)。以两级分层结构把每个场合原子存储在内容原子的下方，如图 8 所描绘的那样。现在参见图 8，场合原子 200、202、204、206 位于分层结构的最低级，由类型处理来分组。在内容原子 208 下方的第一级，有一组“分支”(branch)210、212、214，每个分支包含一个类型处理。每个类型处理(它有一个或多个相应的场合原子)有一个分支(而如果一个特定的类型处理没有场合原子，就没有分支)。这些分支按升序连接。为搜索和提高性能，由 DBMS 在内部使用这种结构。

当插入或更新数据元素时，用户(一般，一个程序员)通过遍历场合之间的连接，在内部关系中定出正确的位置。接着，在内容/分支分层结构下找到位置，然后把新的场合(或变更的场合)存储和接入场合链接和内容结构两者之中。

本发明的系统和方法包括一个唯一的搜索结构，它允许对数据库进行快速和有效的搜索。对于每个内容原子，DBMS 使用数据中‘n’个最高有效的字节作为进入系统搜索结构的矢量。DBMS 把搜索结构分成‘m’个分开的小结构，这里‘m’是‘n’个最高有效字节的范围。然后把实际的内容原子存储在其矢量的小结构中。搜索结构类似于散列表(hash table)，只是不需要算法来确定其元素，而元素通常按照分类次序。为了提高性能，对于每个基类型有一个单独的搜索结构。

图 9 示出搜索的一例。现在参见图 9，假定用户希望定出特别的场合“John Smith”220。在所述的实施例中，最高有效字节(即“J”)用作进入搜索结构 222 的矢量。整个的内容(即“John Smith”)用于找出在搜索结构 222 中的正

确位置。在整个数据库中只有一个内容原子 224 包含“John Smith”的内容。一当定出这个内容原子 224，于是用户能够搜索连接至该内容原子的所有的场合原子，以找出所要的“John Smith”的场合。

本发明的方法和系统也包括了各种暴露存取方法，它们是存取内部成分的唯一方法。暴露功能提供了数据存储和检索的方法、确定和操作内部和外部关系和类型原子的方法以及完成各种系统功能的方法。图 10 说明暴露存取方法。

现在参见图 10，暴露存取方法可以是开发者(developer)存取方法 230(或许以类别库(class library)的形式)，以供应用程序开发者直接使用。用户也可以通过这个层/接口的其他的工具来访问系统，它们将系统表示为 SQL 存取方法 232、对象存取方法 234、网络存取方法 236、或者其他的数据库存取方法(诸如 TCP/IP 串行化 HTML238)。注意，在所有的工具中，原子和连接的基础结构保持相同，并且能够按照用户的需要同时由不同的方法来访问数据。还要注意，暴露存取方法以及数据库本身存储在图 1 的 RAM16 中。存取方法 230、232、234、236、238 通过使用应用程序员接口(API)240 连接至数据库。除了所示的存取方法之外，用户通过使用其他的应用(未示出)可以使用 API240 而创建新的数据库(如下面参见图 11 所讨论的那样)。

现在参图参见 11 至 14，将描述使用本发明的方法。图 11 说明创建新的数据库的方法。通过使用 API240，用户或用户应用程序首先创建类型原子的词典(步骤 300)。如上面参见图 6 所讨论的那样，按照本发明，词典也是数据库。接下来，创建场合原子和内容原子，以表示存储在数据可中的各种数据项(步骤 302)。注意，对于用户，一个场合原子和一个内容原子的组合将通常被视为单个数据项。最后，通过使用指针，把适当的场合原子、内容原子和类型原子连接在一起，以创建分子(步骤 304)。为了表示更复杂的数据类型可以把分子连接在一起，以形成内部关系(如上面参见图 3 所讨论的那样)，而可以把内部关系连接在一起，以形成外部关系(如上面参见图 4 所讨论的那样)。

现在参见图 12，说明把新的数据添加至数据库的方法。从用户接收数据(步骤 400)，并且搜索数据库(步骤 401)。下面将参照图 13 更充分地描述搜索。系统确定数据内容是否存在于数据库中(步骤 402)。如果不存在，则例示(instantiate)一个内容原子并且连接入搜索结构(步骤 403)。如果数据内容已经存在，则系统确定是否有一个以上的分支(步骤 404)。一个分支是从一个内容原子到一个或数个同一类型的场合原子的连接。如果内容原子只与一个类型原子相关联，则把一个新的场合存储在单个分支中(步骤 405)。如果内容原子与

一个以上的类型原子相关联，则系统搜寻正确的类型，并且把新的场合存储在正确的分支中(步骤 406)。

现在参见图 13，现在将描述搜索数据库的方法。从用户处接收数据(步骤 500)。用数据的开头的“n”个字节来选择适当的小结构，搜索在该小结构中开始(步骤 501)。然后搜索该小结构(步骤 502)。系统确定数据是否已经找到(步骤 503)。如果没有找到，则通知用户当前在数据库中不存在该数据(步骤 504)。如果找到了数据，则把它输出至用户(步骤 505)。

现在参见图 14，现在描述更新数据库中的数据项的方法。系统首先寻找打算改变的内容/场合对(步骤 600)。创建新的数据(步骤 601)。如果已经请求历史(步骤 602)，则在新创建的数据下方例示一个新的场合原子(步骤 603)，并且建立新场合以指向老场合(步骤 604)。如果不曾请求历史，则老场合变为新场合(步骤 605)，并且把它重新连接至新创建的数据(步骤 606)。

与现有技术相比，本发明的唯一的构造提供了许多优点。本发明提供了动态方案演变，无需脱机重建数据库，因而也免除了与脱机重建相关联的开销。本系统是可扩展的，支持新的、用户定义的数据类型。数据库只存储数据代码变换器(codex)一次(例如，串“John Smith”实际上只存储一次而不管在数据库中出现多少个 John Smith)。这个特征减小了数据库的大小，并且当与本发明的搜索结构组合时(参见图 9 的描述如上)，提供了速度很高的检索和更新操作。本发明的 DBMS 无需各别地规定和保持索引，因为它对于所有的数据元素保持其自己的内部搜索结构。因此，系统更加稳健而几乎不需要系统管理。

本发明自动保持用户指派的数据元素的历史，大大减少了在现有技术中为提供这一能力而需花费的编程工作量。在现有技术系统中保持历史需要大量的编程和数据库设计。在关系数据库中，这是通过对每个需要历史的表确定各自的历史表而完成的。本发明的系统和方法在场合原子层上自动提供这种功能，而不用由开发者或数据库设计者所作的各自的外部定义。本发明的编程接口包括用历史原子倒转场合变更的功能。把用于事务倒转的编程减至最少，而能够容易地确定由系统起始的重新运行(rollback)。

本发明还支持对象继承(inheritance)、封装(encapsulation)和多形态(polymorphism)。类型原子的基类型可以随时间改变(通过过载)，而自动保持完整的历史，无需把老的数据值转变为新添加的定义。例如，类型原子“温度”可以定义为 INTEGER(固定数)，但是还可以有类型名称为“温度”的类型原子，其基类型为 TEXT(公式)和 BLOB(小应用程序(applet))。结果，能够以多种格式

接受和存储数据值。能够使用这种功能以提供继承和多形态。作为另一个例子，能够把类型名称 SPEED 定义为 INTEGER(例如，基本速度 = 65mph)，描写速度的 TEXT(例如“很快”、“极慢”、“比光快”)和 BLOB(具有使速度杆(speed bar)动作的小应用程序)。注意，通过不同的用户/开发者接口也能够把多形态补充入数据库。

本发明的还有一个优点是它的通用的对象 - 关系 - 层次能力。每个内部关系可以包括分子的任何组合(因此包括数据类型、场合数目、等等的任何组合)，而每个内部关系可以随意改变。因此本发明无需与静态和非柔性的记录和数据库结构一起工作。复杂的数据结构能够作为数据库“天生的”部分而实现，而不需要作复杂的操作和变换以适合数据库结构。

能够定义数据结构以匹配任何类型的数据库(关系数据库、层次数据库、等等)，并且能够实现任何的存取方法(用户接口/语法，诸如 SOL)。即使原子和分子的基础系统保持不变，DBMS 有效地采取任何所要的 DBMS 的形式。

本发明还支持第一标准形式数据库，其做法是确保任何数据值只存储一次(在内容原子中)，而绝不重复。这个特征提供了建造第一标准形式数据库的基础。

本发明还允许动态方案演变。一个类型原子的任何(或全部)属性可以随意改变。例如，可以改变“湿度”的类型原子，由此改变所有的“湿度”的场合的描述而不影响相关的内容原子和任何的场合原子。与传统的 DBMS 相比，这是一个显著的优点，对于传统的 DBMS 而言，当进行改变记录布局、添加或去除字段、等等操作时，需要重建 DBMS。可以从内部关系(记录)中添加分子(字段)、改变其格式、或者将其擦除，而不影响数据库的其他部分。无需进行数据库重建、卸载/装载、等等操作，这就降低了成本和减少了数据管理的复杂性，还避免了停机时间。

本发明也不需要索引字段。而由基于内容原子的整体搜索结构提供迅速查找特定数据值的能力。本发明的搜索结构(如上面参照图 9 所描述的那样)要比索引更稳健，后者一般被保持在分开的文件中。于是本发明的搜索结构减少了数据库设计和管理的复杂性。

与现有技术的系统相比，本发明有几个特性和存储方面的优点。可以断开在重新连接场合原子而不触及相关联的类型或内容原子。例如，在一个步骤中“John Smith”这个场合原子可以与类型原子“顾客”断开而与类型原子“供给者”重新连接。在传统的 DBMS 中，这个操作需要删去一个记录并且创建另

一个记录，这是一个慢很多的过程。

每个数据值/性质(即“John Smith”)在数据库内将只在一个内容原子中出现一次。这个特征节省了存储空间。此外，把所有的场合的数据内容“John Smith”改为“Tom Smith”只需作一次处理，这就极大地提高了系统的性能。由于查找“John Smith”只需要识别一个值，然后它把数据内容“John Smith”提供给所有的场合，因此提高了性能。此外，在本发明中，记录中的空字段(即具有零值的字段)不存在，于是节省了空间并且提高了性能。

能够把复杂的数据结构作为数据库的“天生的”部分来实现，而不需要作复杂的操作和变换来适合数据结构。这种能力提高了性能，因为它对于每次读写避免了变换过程，因此也节省了存储空间。因为是在场合原子层而不是在记录或对象层存储历史信息，因此在本发明还能提高性能和节省存储空间。

虽然已经以某种程度的特殊性描述了本发明，但应该理解，熟悉本领域的人可以改变其元素而不偏离本发明的精神和范围。本发明只由权利要求书及其等价物限定。

说 明 书 附 图

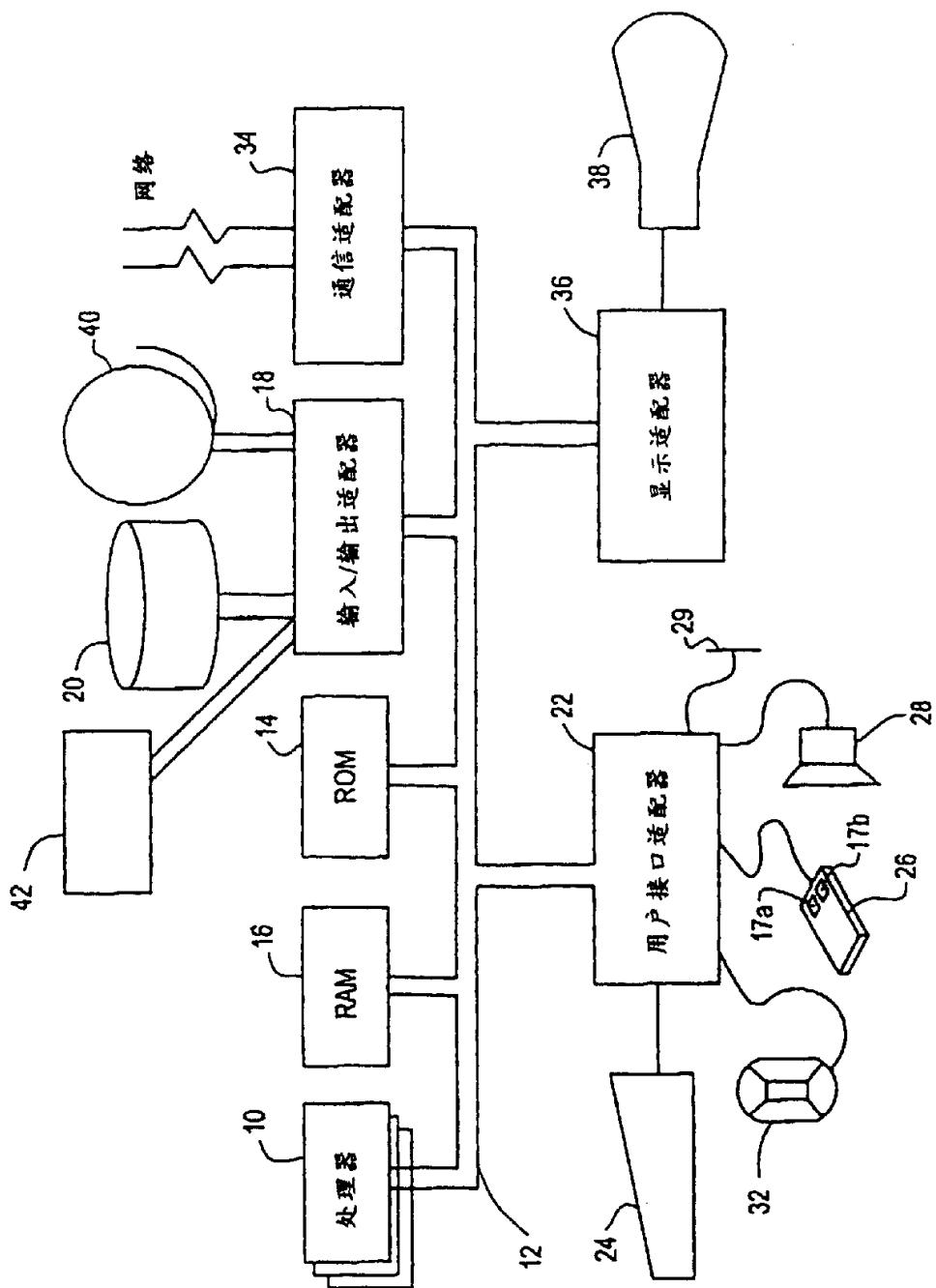
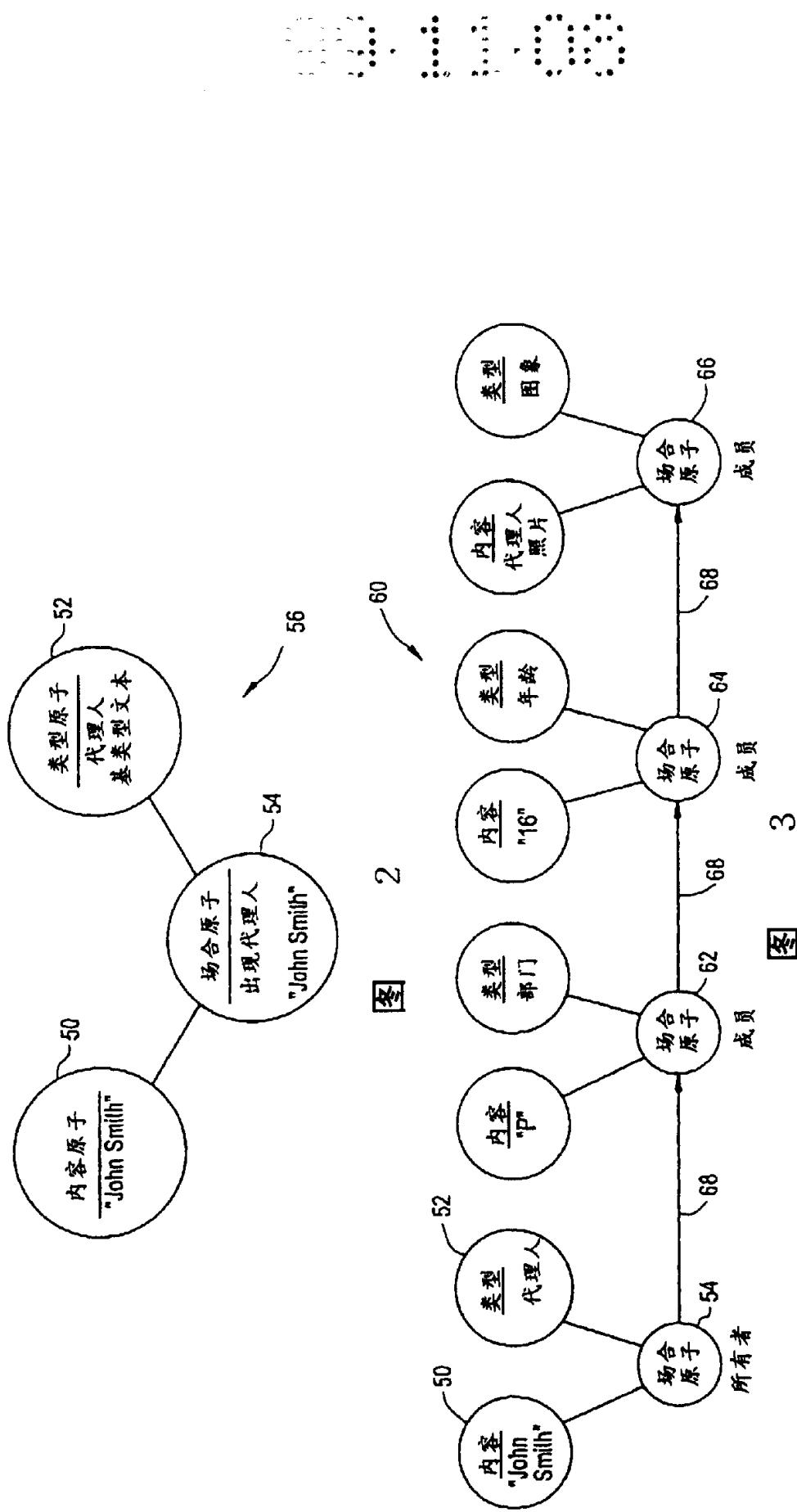


图 1



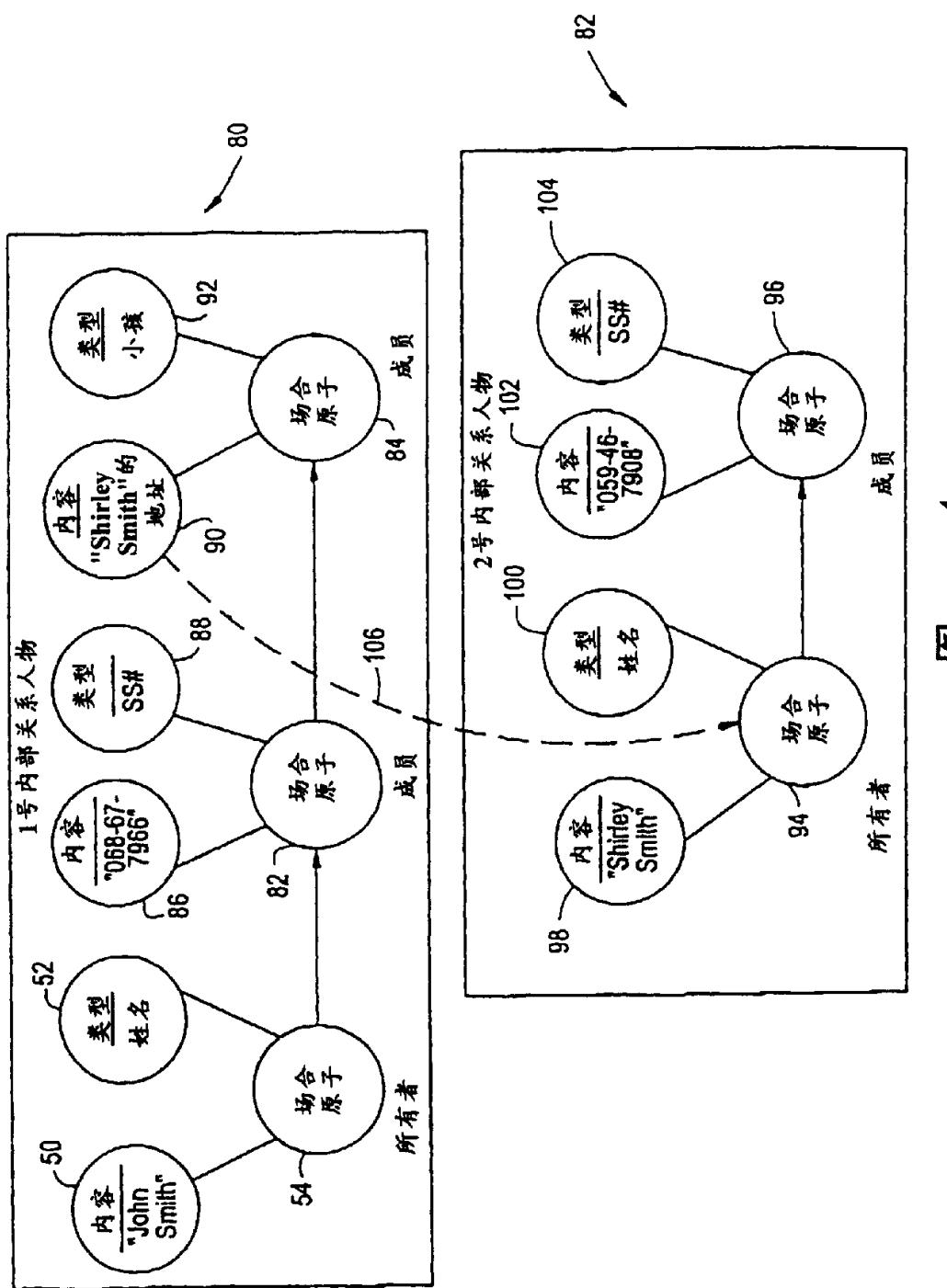
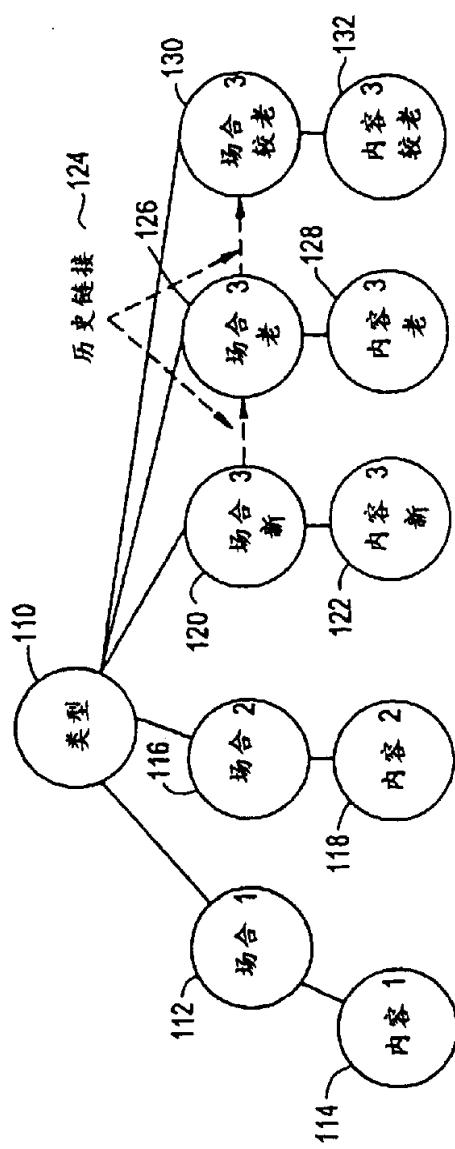


图 4

图 5



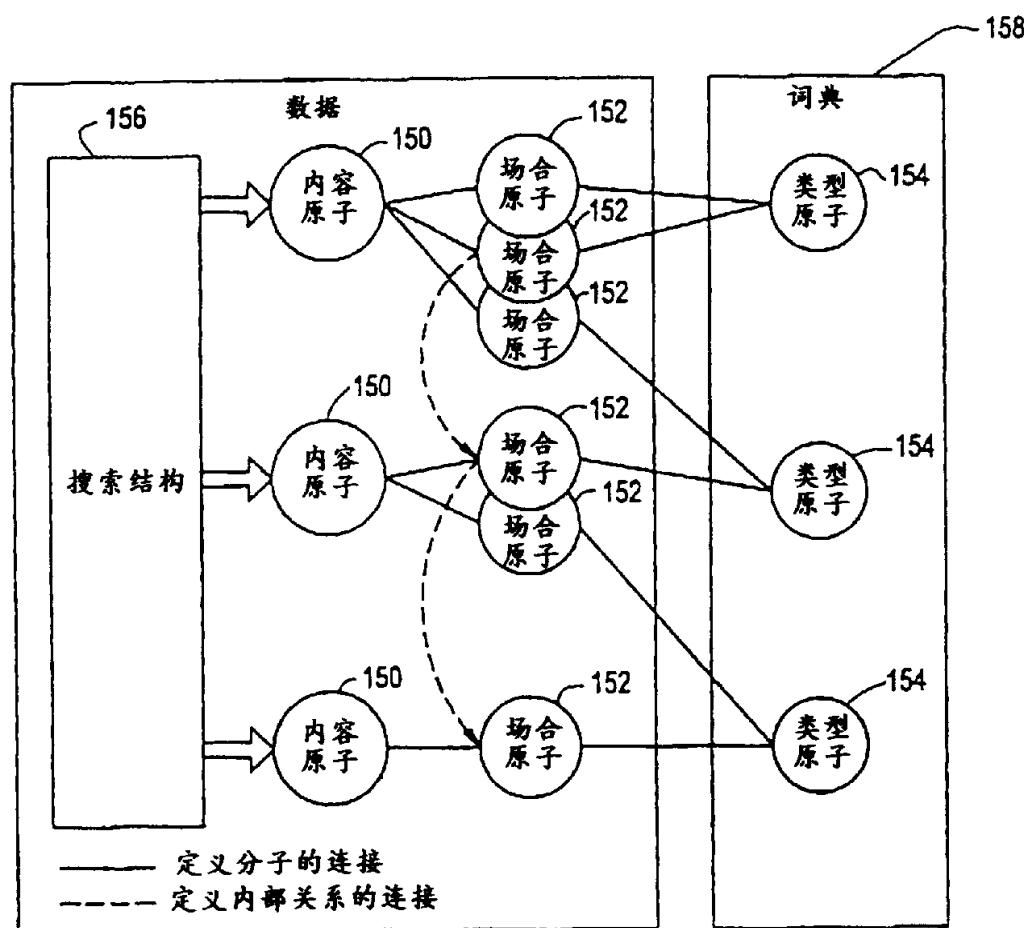


图 6

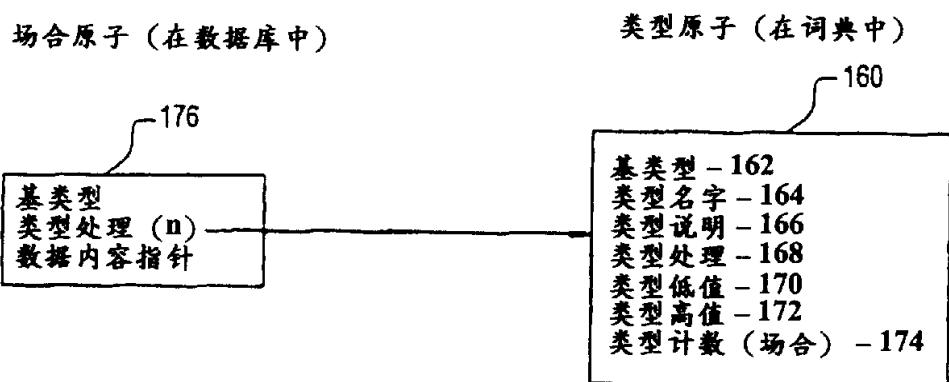


图 7

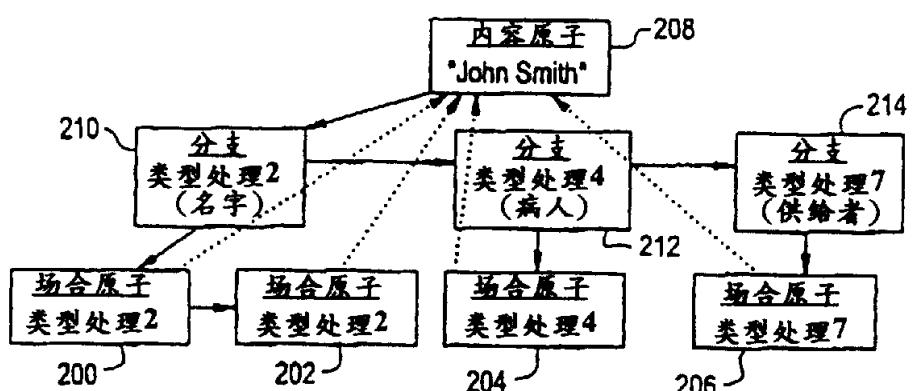


图 8

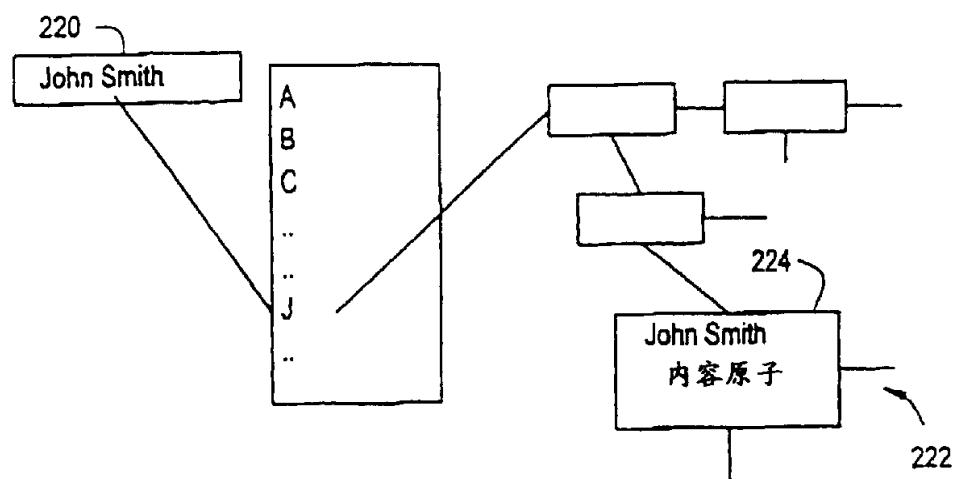


图 9

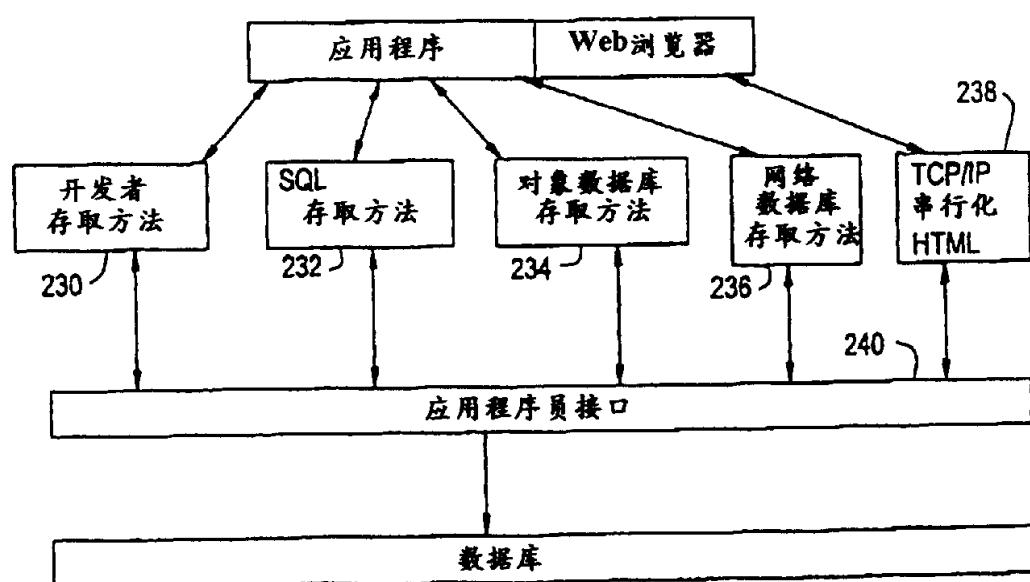


图 10

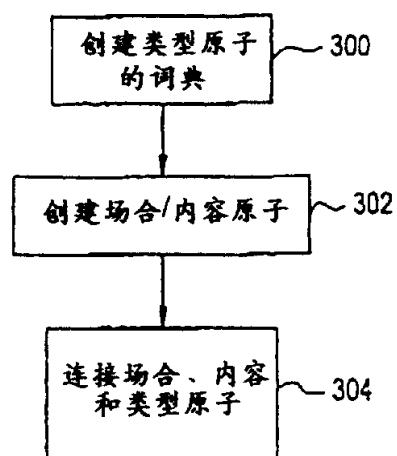


图 1.1

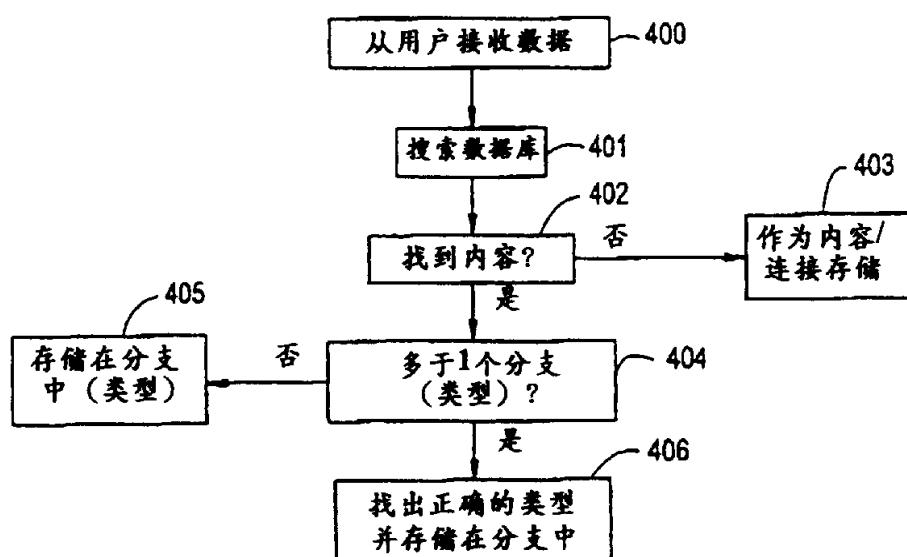


图 1.2

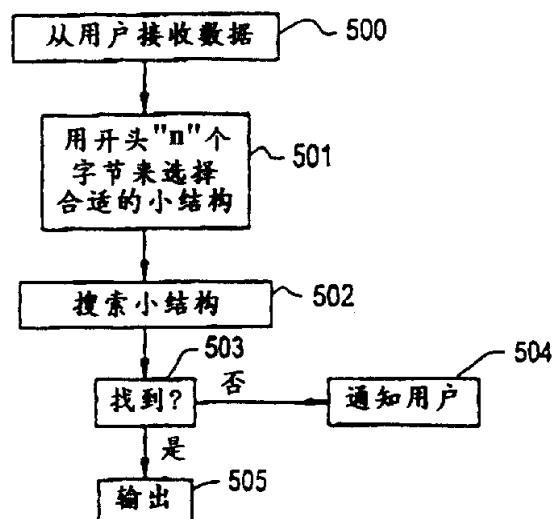


图 1 3

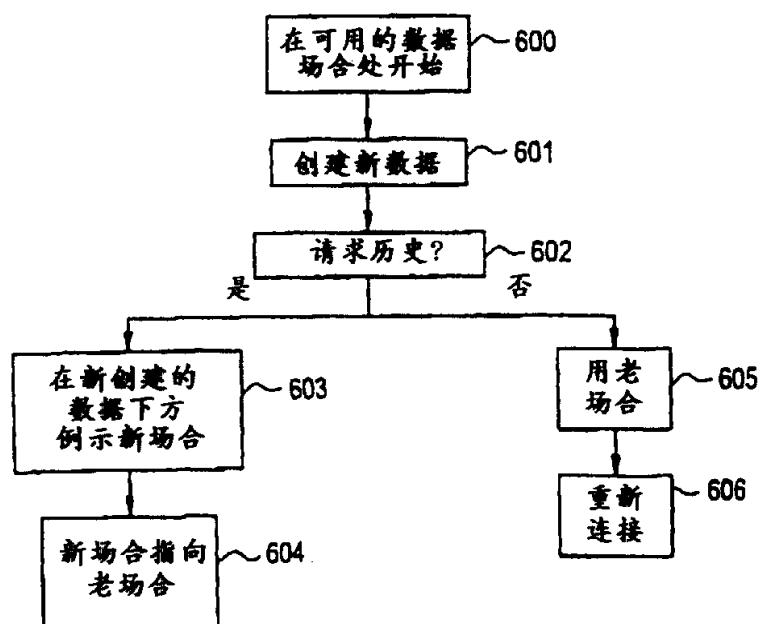


图 1 4