

[19] 中华人民共和国国家知识产权局

[51] Int. Cl.

G06F 17/00 (2006.01)

G06F 15/16 (2006.01)



[12] 发明专利申请公布说明书

[21] 申请号 200580029034.2

[43] 公开日 2007 年 9 月 26 日

[11] 公开号 CN 101044472A

[22] 申请日 2005.8.31

[21] 申请号 200580029034.2

[30] 优先权

[32] 2004.8.31 [33] US [31] 60/606,407

[86] 国际申请 PCT/US2005/031097 2005.8.31

[87] 国际公布 WO2006/026702 英 2006.3.9

[85] 进入国家阶段日期 2007.2.27

[71] 申请人 国际商业机器公司

地址 美国纽约

[72] 发明人 拉塞尔·G·安德森

穆哈梅德·伯兹亚内

文森特·A·马斯特罗

罗伯特·C·韦伯三世

[74] 专利代理机构 中国国际贸易促进委员会专利商
标事务所

代理人 李 颖

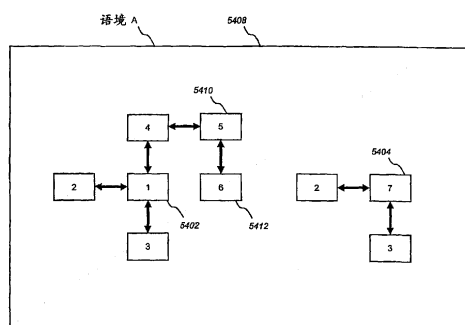
权利要求书 4 页 说明书 22 页 附图 17 页

[54] 发明名称

数据系统中的语义识别方法和系统

[57] 摘要

提供涉及语义标识符、转换引擎和中心或数据库的抽象层次性质的方法和系统，所述语义标识符便于根据一个项目与其它项目的关系，唯一地识别该项目，而不需要保存另外的数据；所述转换引擎可把数据、元数据、语义标识符和其它项目从一种格式、语言和/或数据模型转换成另一种格式、语言和/或数据模型；所述中心或数据库的抽象层次性质便于区分一个项目的多个实例或形式。



1、一种数据集成方法，包括：

提供用于根据项目与另一项目的关系来识别所述项目的语义标识符；

获得数据模型的映射，以使得能够确定数据模型中的项目的语义标识符；

使映射与数据集成功能相关联，其中数据集成功能的执行基于所述映射和所述语义标识符至少之一。

2、按照权利要求 1 所述的方法，其中项目包括对象、数据项、数据、列、行、表、数据库、实例、属性、元数据、概念、题目、主题、标识符、语义标识符、RFID 标签、卖主、供应者、顾客、个人、团队、组织、用户、网络、系统、设备、家庭、商店、产品、产品线、产品特征、产品规格、产品属性、价格、成本、材料单、装运数据、税款数据、课程、教育规划、位置、地图、部门、组织、有机体、过程、规则、法律、收费制度、物品、服务和服务提供中的一个或多个。

3、按照权利要求 1 所述的方法，其中关系涉及项目在关系分层结构中的位置。

4、按照权利要求 1 所述的方法，其中语义标识符是项目的唯一标识符。

5、按照权利要求 1 所述的方法，其中语义标识符以比该项目与其它项目的全部关系少、但足以确保标识符唯一的足够数目的关系为基础。

6、按照权利要求 1 所述的方法，其中语义标识符以确保标识符唯一的最少数目的关系为基础。

7、按照权利要求 1 所述的方法，其中语义标识符是项目的与语境相关的标识符。

8、按照权利要求 1 所述的方法，其中按照原子格式保存语义标识符。

9、按照权利要求 1 所述的方法，其中按照原子格式把语义标识符保存在数据储存库中。

10、按照权利要求 1 所述的方法，其中语义标识符是动态的。

11、按照权利要求 1 所述的方法，其中语义标识符依据语境而变化。

12、一种进行数据集成过程的方法，包括：

使模型和数据集关联起来；和

形成从数据集中选择项目的选择命令，其中选择命令的形式以项目的从所述模型确定的区别特性为基础。

13、按照权利要求 12 所述的方法，其中针对使用选择命令/查询的过程在运行时执行该选择命令/查询的形成。

14、按照权利要求 12 所述的方法，其中在设计使用选择命令/查询的过程时执行该选择命令/查询的形成。

15、一种进行数据集成过程的方法，包括：

使模型和数据集关联起来；和

形成查询该数据集的查询，其中所述查询的形式以项目的从所述模型确定的区别特性为基础。

16、按照权利要求 15 所述的方法，其中针对使用选择命令/查询的过程在运行时执行该选择命令/查询的形成。

17、按照权利要求 15 所述的方法，其中在设计使用选择命令/查询的过程时执行该选择命令/查询的形成。

18、一种数据集成系统，包括：

根据项目与另一项目的关系来识别所述项目的语义标识符；

使得能够确定数据模型中的项目的语义标识符的数据模型的映射；

使映射与数据集成功能相关联的装置，其中数据集成功能的执行基于所述映射和所述语义标识符至少之一。

19、按照权利要求 18 所述的系统，其中所述关系涉及项目在关系分层结构中的位置。

20、按照权利要求 18 所述的系统，其中语义标识符是项目的唯一标识符。

21、按照权利要求 18 所述的系统，其中语义标识符以比该项目与其它项目的全部关系少、但足以确保标识符唯一的足够数目的关系为基础。

22、按照权利要求 18 所述的系统，其中语义标识符以确保标识符唯一的最少数目的关系为基础。

23、按照权利要求 18 所述的系统，其中语义标识符是项目的与语境相关的标识符。

24、按照权利要求 18 所述的系统，其中按照原子格式保存语义标识符。

25、按照权利要求 18 所述的系统，其中按照原子格式把语义标识符保存在数据储存库中。

26、按照权利要求 18 所述的系统，其中语义标识符是动态的。

27、按照权利要求 18 所述的系统，其中语义标识符依据语境而变化。

28、按照权利要求 18 所述的系统，其中通过捕获与和第二项目具有直接关系的第一项目的直接关系，语义标识符递归地捕获与第二项目的间接关系。

29、按照权利要求 18 所述的系统，其中在串中捕获语义标识符，并且其中如果唯一标识符并不需要所有元素，则截断所述串。

30、按照权利要求 18 所述的系统，其中数据集成功能是转换操作。

31、按照权利要求 30 所述的系统，其中转换操作修改语义标识符的格式、语义标识符的语言和语义标识符的数据模型中的一个或多个。

32、按照权利要求 30 所述的系统，其中转换操作的映射能够跟踪在初始语义语境和转换后的语义语境之间来回执行的操作中被转换的数据。

33、按照权利要求 30 所述的系统，其中作为面向服务的体系结构中的服务来提供转换操作。

34、按照权利要求 18 所述的系统，还包括根据实体的区别特性，有选择地过滤逻辑实体的实例的过滤器。

35、按照权利要求 34 所述的系统，其中所述区别特性得自于所述映射和所述语义标识符至少之一。

数据系统中的语义识别方法和系统

相关申请

本申请要求于 2004 年 8 月 31 日提交的题为“Methods and Systems for Semantic Identification in Data Systems”的美国临时申请 NO.60/606,407 的优先权。

技术领域

本发明涉及信息技术的领域，更具体地说，涉及数据集成系统的领域。

背景技术

计算机应用程序的出现使许多商业过程更快、更高效；但是，使用不同数据结构、通信协议、语言和平台的不同计算机应用程序的激增已导致典型商业企业的信息技术基础结构极其复杂。典型企业内的不同商业过程可能使用完全不同的计算机应用程序，每种计算机应用程序是为特定的商业过程开发和优化的，而不是为整个企业开发和优化的。例如，企业可能具有一种用于跟随应付账款的特殊计算机应用程序，和一种完全不同的用于记录客户关联人的计算机应用程序。事实上，即使相同的商业过程也可能使用一种以上的计算机应用程序，例如当企业维持集中式客户关联数据库，但是雇员把他们自己的关联人信息保持在例如个人信息管理器中时。

虽然专用计算机应用程序带来定制的解决方案的优点，不过专用计算机应用程序的激增会导致效率低，例如在整个企业内多次重复输入和处理相同的数据，或者当企业执行可受益于与一个过程相关的数据的另一过程时，企业不能利用该数据。例如，如果应付账款过程与供应链和定货过程分离，那么企业可能接受和供应来自其信用会导致

企业拒绝其定货的客户的定货。企业可受益于跨越各种计算机应用程序对其所有数据的一致访问的其它例子有很多。

许多公司已认识到并致力于满足跨越商业企业中的不同应用程序，共享数据的需要。从而，作为一种处理来自不同来源的数据的基于消息的策略，出现了企业应用程序集成或者说 EAI。随着计算机应用程序在复杂性和数量方面的增加，EAI 遇到许多挑战，从需要处理不同的协议，到需要应付不断增大的数据量和事务数量，和不断增长的对更快速的数据集成的强烈愿望。已采取了对 EAI 的各种方法，包括最小公分母方法，原子方法和桥式方法。但是，EAI 以单个应用程序之间的通信为基础。一个明显的缺陷在于，随着平台和应用程序的线性增加，EAI 解决方案的复杂性几何增大。

虽然数据集成系统提供解决企业的需要的有用工具，不过这样的系统通常被部署为客户解决方案。它们具有很长的部署周期，并且要求高级的技术训练，以适应企业结构和信息要求方面的变化。存在对数据集成系统的需要，所述数据集成系统允许在不断变化的企业环境中使用、重新使用和修改功能。这样的一种工具是便于根据一个项目与其它项目的关系唯一地识别该项目，而不需要保存另外的数据的语义标识符。转换（translation）引擎是另一种这样的工具，它可把数据、元数据、语义标识符和其它项目从一种格式、语言和/或数据模型变换成另一种格式、语言和/或数据模型。最后，中心（hub）或数据库的抽象层次的性质便于区分某一项目的多个实例或形式。

发明内容

一个项目可存在一个语义标识符。所述项目可以是对象、数据项、数据、列、行、表、数据库、实例、属性、元数据、概念、题目、主题、语义标识符、其它标识符、RFID 标签、卖主、供应者、顾客、个人、团队、组织、用户、网络、系统、设备、家庭、商店、产品、产品线、产品特征、产品规格、产品属性、价格、成本、材料单、装运数据、税款数据、课程、教育规划、位置、地图、部门、组织、有

机体、过程、规则、法律、收费制度 (rating system)、物品、服务和服务提供或者其它项目或概念。一个项目可与一个数据集成作业和/或数据集成平台相关。语义标识符可根据项目与一个或多个其它项目的关系识别该项目。关系可以是缺乏关系。关系可基于语义。关系可涉及该项目在关系分层结构中的位置。

语义标识符可以是项目的唯一标识符。某一项目的唯一语义标识符可能考虑比该项目与其它项目的所有关系少的关系。创建一个以确保唯一性的最少数量的关系为基础的语义标识符是有利的。创建一个项目的唯一语义标识符所需的关系的数目可根据语境而变化。语义标识符可以是取决于语境的。语义标识符可以是动态的。

可用以串结构或格式保存、维持、记录、处理和/或解释的语法保存、维持、记录、处理和/或解释语义标识符。语法和/或串结构或格式是可解析的。语法和/或串结构或格式可被截断、修改、缩短、解析或重新排序。截断、修改、缩短或重新排序语法和/或串，仍然保留唯一标识符是可能的。

语义标识符可与语义语境相关联，所述语义语境比如是企业方法中的步骤，数据库中的数据，行或列中的数据，表中的行或列，数据库中的行或列，表中的数据，数据库中的表，数据库中的元数据，中心或储存库的项目，数据库中的项目，表中的项目，列中的项目，行中的项目，机构中的人，通信的发送者或接收者，网络上的用户，网络上的系统，网络上的设备，家庭中的成员，商店中的物品，菜单上的菜，产品线上的产品，产品提供中的产品，教育或训练计划中的课程或步骤，地图上的位置，物品的位置，机构的部门，团体中的个人，规则体系中的规则，一套服务中的服务，企业的组织层次中的实体，供应链中的实体，市场中的消费者，购买决策的购买者，商品或服务的价格，商品或服务的成本，产品或系统的组件，方法的步骤和/或小组的成员。

在一个实施例中，数据库可具有含有一列的表。该列的唯一语义标识符可以是“数据库名称的表名称的列名称”。可利用下述语法：列

名称: : 表名称: : 数据库名称保存、维持、记录、处理和/或解释该唯一语义标识符。所述语法和/或任何相关串可被解析,不必要的元素可被除去。例如,如果只存在一个数据库,那么下述语法仍然产生该列的唯一标识符:列名称: : 表名称。创建唯一语义标识符并不需要数据库关系。在另一例子中,数据库可能只有一个表,从而下述语法是该列的唯一标识符:列名称: : 数据库名称。创建唯一标识符不需要表关系。使用较短的语法和/或串可减少处理时间和提高效率。

转换引擎可对一个或多个语义标识符,数据库,包括语义标识符的数据库,信息系统,包括语义标识符的信息系统或者其它项目进行转换操作。转换操作可转换或者以其它方式修改语义标识符的格式、语言和/或数据模型。转换操作可涉及从一个或多个数据工具、语言、格式和/或数据模型到至少一种其它的数据工具、语言、格式和/或数据模型的转换或映射,或者从至少一种其它的数据工具、语言、格式和/或数据模型到一个或多个数据工具、语言、格式和/或数据模型的转换或映射。转换操作可涉及往来于 DataStage 7, QualityStage, Business Object, IBM-DB2 Cube Views, UML 1.1, UML 1.3, ERStudio, ProfileStage, PowerDesigner(另外支持 Packages and Extended Attributes)和/或 MicroStrategy 的转换或映射。转换引擎和/或转换操作可包含在元代理中。转换引擎,转换操作的映射或者转换操作能够跟踪在初始语义语境和转换后的语义语境之间来回执行的转换操作中转换的数据。可分批地、实时地和/或连续地进行、执行和/或实施转换操作。可以服务的形式,例如作为面向服务的体系结构的一部分的形式提供服务或者使之可用。

一旦对于语义标识符,数据库,包括一个或多个语义标识符的数据库,信息系统,包括一个或多个语义标识符的信息系统或者其它项目来说,存在转换操作,那么它可被转换成或者转换自,映射到,链接到任意其它的语义标识符,数据库,包括一个或多个语义标识符的数据库,信息系统,包括一个或多个语义标识符的信息系统或者共享至少一个转换操作的其它项目,和任意其它的语义标识符,数据库,

包括一个或多个语义标识符的数据库，信息系统，包括一个或多个语义标识符的信息系统或者共享至少一个转换操作的其它项目一起使用或者与之相关联。

项目可以多种形式或实例存在，例如物理建模活动和/或逻辑建模活动。项目，包括任何相关数据或元数据可以多种形式或实例存在于数据库和/或中心中。为了区分项目的各种形式或实例，可以使用任何辨别特征，例如抽象层次，分层结构中的位置，与另一项目的关系，项目的一个或多个区别属性，其中发现该项目的语境，发现该项目的物理位置等等。

在一个实施例中，名为“雇员”的表可被带入中心中。中心控制者在中心中具有“雇员”的两种形式中实例；一种对应于物理建模活动，另一种对应于逻辑建模活动。中心数据收集的抽象层次性质便于区分物理模型和逻辑模型实例或形式。

当响应查询进行转换操作时，转换引擎可从中心或数据库夺取、装入或者获得所有项目。它随后根据区别特征，比如抽象层次、分级结构中的位置、与另一项目的关系、项目的属性、物理位置等，过滤、选择、保存、转换、修改或者以其它方式作用于项目。在备选实施例中，当响应查询进行转换操作时，转换引擎可在中心或数据库过滤、选择、保存、转换、修改或者以其它方式作用于项目，包括任何数据和/或元数据，只夺取、装入或获得相关抽象层次的，或者具有相关属性、位置、关系、地点等的那些项目。可在运行时或者设计时进行所述过滤、选择、存储、转换、修改或者其它操作，并且可以分批地、实时地或者连续地进行所述过滤、选择、存储、转换、修改或者其它操作。在实施例中，过滤、选择、存储、转换、修改或者其它操作可以在开发时、设计时或者运行时由转换引擎和/或系统获得的信息或输入为基础，例如数据模型、数据模型的映射，标识符的语法的辨别特征等。所述信息可被实时地动态更新。从而，在一个优选实施例中，系统可根据数据库的已知映射，修改用于从数据库选择数据的选择命令，以便选择逻辑项目和忽略物理项目，反之亦然。

在一些情况下，在整个过程中，过滤、选择或其它操作越接近中心或数据库，操作就越高效和越快。转换引擎可对查询本身进行转换操作，产生可被直接发送给中心或数据库的修改的查询或选择命令。修改的查询或选择命令可以采取直接与中心或者数据库兼容的格式。

在其它方面，计算机程序产品可包括计算机可用介质，所述计算机可用介质包括计算机可读程序代码，其中当在一个或多个计算机上执行时，所述计算机可读程序代码使一个或多个计算机执行上述任意一种或多种方法。

这里使用的“国际商用机器”或者“IBM”指的是 Armonk, New York 的国际商用机器公司。

这里使用的“数据源”或“数据目标”意图具有与这些术语相符的最宽的可能含义，包括数据库、多个数据库、储存库信息管理器、队列、消息服务、储存库、数据设施、数据存储装置、数据提供者、网站、服务器、计算机、计算机存储装置、CD、DVD、移动存储装置、中央存储装置、硬盘、多调谐数据存储装置、RAM、ROM、闪速存储器、存储卡、临时存储装置、永久存储装置、磁带、本地连接的计算装置、远程连接的计算装置、无线装置、有线装置、移动装置、中央装置、web 浏览器、客户端、膝上型计算机、个人数字助手(“PDA”)、电话机、蜂窝电话机、移动电话机、信息平台、分析装置、处理装置、商业企业系统或者处理数据的其它装置或者提供的保存数据或其它信息的其它装置，以及保持在任意上述系统中使用的结构化或非结构化数据，或者任何流化、消息化、事件驱动的或者其它来源的数据的文件或文件类型，和上面所述的任意组合，除非另外指出特定的含义或者短语的上下方要求另外的含义。存储机构是任意逻辑或物理装置、资源、或者能够充当数据源或数据目标的装置。

“Enterprise Java Bean(EJB)”包括 J2EE 平台的服务器方组件体系结构。EJB 支持快速、简化地开发分布式、事务性、安全并且可移植的 Java 应用程序。EJB 支持允许消息的同时消费的容器体系结构，并且提供对分布式事务的支持，从而利用 J2EE 体系结构的数据库更

新、消息处理和与企业系统的连接能够参与相同的事务语境。

“JMS”意味着 Java 消息服务,Java 消息服务是基于 Java 的 J2EE 企业体系结构的一种企业消息服务。“JCA”意味着下面更详细说明的 J2EE 平台的 J2EE 连接器体系结构。应认识到虽然 EJB、JMS 和 JCA 是当代的分布式事务环境中的常用软件工具,不过提供类似功能的任何平台、系统或体系结构可和这里描述的数据集成系统一起采用。

这里使用的“实时”包括接近商业事务或业务的持续时间的周期,并且包括与离线,例如在每夜的批处理操作中发生的过程或服务相对的在业务操作或业务中发生的过程或服务。根据商业过程的持续时间,实时可包括几秒,几分之一秒,几分钟,几小时或者甚至几天。

这里使用的“商业过程”、“商业逻辑”和“商业事务”包括可由企业执行的任何方法、服务、操作、过程或事务,包括(但不限于)销售、行销、合同履行、库存管理、定价、产品设计、职业服务、金融服务、行政管理、财政、保险业、分析、订合同、信息技术服务、数据存储、数据挖掘、信息的传递、物品的发送、调度、通信、投资、交易、供货、推销、广告、出价、工程、制造、供应链管理、人力资源管理、数据处理、数据集成、工作流程管理、软件生产、硬件生产、新产品的开发、研究、开发、策略功能、质量控制和保险、包装、后勤、客户关系管理、处理折扣和返还、客户支持、产品维护、电话销售、企业传播、投资人关系和其它许多。

这里使用的“面向服务的体系结构”(SOA)包括构成商业企业的基础结构的一部分的服务。在 SOA 中,服务可变成应用程序开发和部署的构件块,便于快速应用程序开发和避免冗余代码。每种服务可具体体现被约束在周围环境,比如服务的数据输入的源或者服务的数据输出的目标上的一组商业逻辑或商业规则。在下面的说明中提供 SOA 的各种实例。

这里使用的“元数据”包括产生正被处理的数据的语境的数据,和正被处理的数据有关的数据,与相关信息的语境有关的信息,与数据

的起源有关的信息，与数据的位置有关的信息，与数据的含意有关的信息，与数据的寿命有关的信息，与数据的标题有关的信息，与数据的单位有关的信息，与数据的字段有关的信息和/或与涉及数据的语境的任何其它信息有关的信息。

这里使用的“WSDL”或者“Web 服务描述语言”包括把网络服务(通常 web 服务)描述成作用于包含面向文档或者面向过程信息的消息的一组端点。操作和消息被抽象描述，随后被绑定到具体的网络协议和消息格式上，从而定义一个端点。相关的具体端点被组合成抽象端点(服务)。WSDL 是可扩展的，从而允许端点和它们的消息的描述，而不管什么消息格式或网络协议被用于通信。

这里使用的“元代理”包括调用转换引擎或者其它装置对数据或元数据进行转换操作或者其它操作的系统或方法。转换操作或其它操作可涉及数据或元数据从一种或多种格式、语言和/或数据模型到一种或多种格式、语言和/或数据模型的转换。

附图说明

图 1 是具有多个商业过程的商业企业的示意图，每个商业过程可包括多个不同的计算机应用程序和数据源。

图 2 是表示跨越商业企业的多个商业过程的数据集成的示意图。

图 3 是表示为商业企业的多个数据源提供数据集成的体系结构的示意图。

图 4 表示与其它项目有关的一个项目。

图 5 表示与其它项目有关的一个项目。

图 6A 表示在某一语境中的一个项目。

图 6B 表示在某一语境中的一个项目。

图 7 表示某些串。

图 8 表示一个项目和对应的串。

图 9 表示一个串及其某些变异。

图 10 表示作用于某些串的转换引擎。

图 11 表示可以多种形式或实例存在的一个项目。

图 12 表示可以多种形式或实例存在于中心或数据库中的一个项目。

图 13 表示在不同抽象层次下的中心中的项目。

图 14 表示在数据库或中心夺取所有项目的转换过程。

图 15 表示在数据库或中心过滤项目的转换过程。

图 16 表示翻译查询的转换过程。

具体实施方式

在下面的说明中，相同的附图标记指的是相同的部件，除非另外明确说明。

这里公开的发明可以采取纯硬件实施例，纯软件实施例或包含硬件部件和软件部件的实施例的形式。在一个优选实施例中，用软件实现本发明，所述软件包括(但不限于)固件、驻留软件、微代码等。

此外，本发明可以采取可从计算机可用或计算机可读介质存取的计算机程序产品的形式，所述计算机可用或计算机可读介质提供供计算机或者任何指令执行系统之用或者与之结合使用的程序代码。对于本说明来说，计算机可用或计算机可读介质可以是能够包含、保存、传递、传播或传送供指令执行系统、设备或装置之用或者与之结合使用的程序的任意设备。

所述介质可以是电子、磁、光、电磁、红外或半导体系统(或设备或装置)或者传播介质。计算机可读介质的例子包括半导体或固体存储器、磁带、可拆卸的计算机磁盘、随机存取存储器(RAM)、只读存储器(ROM)、硬磁盘和光盘。目前的光盘例子包括光盘-只读存储器(CD-ROM)、光盘-读/写存储器(CD-R/W)和 DVD。

适合于保存和/或执行程序代码的数据处理系统将包括至少一个通过总线直接或间接地与存储元件耦合的处理器。存储元件可包括在程序代码的实际执行期间采用的本地存储器，大容量存储器和提供至少一些程序代码的临时存储，以减少执行期间，必须从大容量存储器

取回代码的次数。

输入/输出或者说 I/O 装置(包括但不限于键盘、显示器、指示装置等)可直接地或者通过居中的 I/O 控制器与数据处理系统耦接。

网络适配器也可与系统耦接,从而使数据处理系统能够通过居中的专用或公共网络,与其它数据处理系统或远程打印机或存储装置耦接。调制解调器、线缆调制解调器和以太网卡只是目前可用的各类网络适配器中的一些。

图 1 表示便于商业企业的各种数据的集成的平台 100。该平台包括多个商业过程,每个商业过程可包括多个不同的计算机应用程序和数据源。该平台可包括几个数据源 102,数据源 102 可以是如上所述的那些数据源。这些数据源可包括来自各种物理位置的各种数据类型。例如,数据源可包括来自诸如 Sybase、Microsoft、Informix、Oracle、Inlomover、EMC、Trillium、First Logic、Siebel、PeopleSoft、IBM、Apache 或 Netscape 之类提供商的系统。数据源 102 可包括使用数据产品或者标准,比如 IMS、DB2、ADABAS、VSAM、MD 系列、UDB、XML、复合平面文件、或者 FTP 文件的系统。数据源 102 可包括由诸如 Microsoft Outlook、Microsoft Word、Microsoft Excel、Microsoft Access 之类应用程序创建或使用的文件,以及诸如 ASCII、CSV、GIF、TIF、PNG、PDP 之类标准格式的文件。数据源 102 可来自于不同的位置,或者它们可位于中心。从数据源 102 供给的数据可以不同的格式到来,并且具有可能相互兼容也可能相互不兼容的不同格式。

数据目标在后面说明。一般来说,这些数据目标可以是上述任意数据耦接 102。命名方面的这种差别一般表示在数据集成过程中,数据系统是提供数据还是接收数据。但是,应认识到这种区别并不意图转达数据源和数据目标之间能力的差别(除非另外明确指出),因为在常规的数据集成系统中,数据源可接收数据,数据目标可提供数据。

图 1 中图解说明的平台还包括数据集成系统 104。数据集成系统可简化作为数据集成系统接收查询或检索命令的结果,从数据源 102 的数据收集。数据集成系统 104 可向一个或多个数据源 102 发送命令,

以致数据源向数据集成系统 104 提供数据。由于接收的数据可以是包括变化的元数据在内的各种格式，因此数据集成系统可重新配置接收的数据，以致接收的数据随后可被组合以便集中处理。下面更详细地说明可由数据集成系统 104 执行的功能。

平台 100 还包括几个检索系统 108。检索系统 108 可包括用于进一步处理来自数据集成系统 104 的数据的数据库或处理平台。例如，数据集成系统 104 可净化、组合、变换或以其它方式处理它从数据源 102 接收的数据，以致检索系统 108 能够使用处理的数据来产生对企业有用的报告 110。报告 110 可用于报告数据关联、回答复杂查询、回答简单查询、或者形成对企业或用户有用的其它报告，并且可包括原始数据、表格、图表、图形或者来自检索系统 108 的数据的任何其它表现。

平台 100 还可包括数据库或数据库管理系统 112。数据库 112 可用于临时地、永久地或者长期地保存数据。例如，数据集成系统 104 可从一个或多个数据源 102 收集数据，并把数据变换成相互兼容的形式，或者适合于相互组合的形式。一旦数据被变换，那么数据集成系统 104 可按照分解形式、组合形式或者其它形式保存在数据库 112 中以便稍后检索。

图 2 是表示跨越商业企业的多个实体和商业过程的数据集成的示意图。在图解说明的实施例，数据集成系统 104 简化用户接口系统 202 和数据源 10 之间的信息流动。数据集成系统 104 可接收来自接口系统 202 的查询，其中所述查询使驻留在一个或多个数据源 102 中的数据的提取和可能的变换成为必需。接口系统 202 可包括与数据集成系统 104 通信的任意装置或程序，比如在膝上型计算机或桌面计算机上工作的 web 浏览器、蜂窝电话机、个人数字助手(“PDA”)、连网平台及与之连接的装置、或者可能与数据集成系统 104 面接的任何其它装置或系统。

例如，用户可能正在操作一个 PDA，并且通过 WiFi 或无线接入协议/无线置标语言(“WAP/WML”)接口向数据集成系统发出信息请

求。数据集成系统 104 可接收该请求，并产生任何请求的查询，以便从网站或者其它数据源 102，比如 FTP 文件站点访问信息。来自数据源 102 的数据可被提取并被变换成与发出请求的接口系统(在本例中，PDA)兼容的格式，随后被传送给接口系统 202 供用户查看和处理。在另一实施例中，数据可能先前已从数据源被提取，并被保存在独立的数据库 112 中，数据库 112 可以是数据仓库或者数据集成系统 104 使用的其它数据设施。数据可能已按照变换后的情形或者以其初始状态被保存在数据库 112 中。例如，数据可按照变换后的情形被保存，以致来自许多数据源 102 的数据可在另一变换过程中被组合。例如，来自 PDA 的查询可被传送给数据集成系统 104，数据集成系统 104 可从数据库 112 提取信息。在所述提取之后，数据集成系统 104 可把数据变换成与 PDA 兼容的组合格式，之后传送给 PDA。

图 3 是表示向商业企业提供多个数据源 102 的数据集成的体系结构的示意图。数据集成系统 104 的一个实施例可包括从数据源提取数据并分析源数据的列值和表格结构，以及执行其它过程的发现数据阶段 302。发现数据阶段 302 还可产生关于数据目标的表格结构、关系和关键字的推荐。更高级的剖析(profiling)和审计功能可包括数据范围确认、计算的准确性、if-then 评估的准确性等。发现数据阶段 302 可使数据归一化，例如通过消除源数据中的冗余相关性和其它异常。发现数据阶段 302 可提供另外的功能，比如深挖(drill down)数据源 102 中的例外以便进一步分析，或者能够实现主机数据的直接剖析。在 IBM 的 WebSphere ProfileStage 产品中可找到发现数据阶段 302 的商业实施例的一个非限制性例子。

数据集成系统 104 还可包括数据准备阶段 304，在数据准备阶段 304，数据被准备、标准化、匹配或者以其它方式处理，从而产生稍后将被变换的质量数据。数据准备阶段 304 可执行类属数据质量功能，比如调和和不一致性或者检查数据内的正确匹配(包括一对一匹配，一对多匹配，和去除重复数据)。数据准备阶段 304 还可提供专用数据增强功能。例如，数据准备阶段 304 可确保地址符合改进的国际通信用跨

国邮政索引。数据准备阶段 304 可使位置数据符合空间信息管理用跨国地理编码标准。数据准备阶段可修改或增加地址,以确保地址信息根据政府核准的美国地址更正,取得美国邮政服务邮费率折扣的资格。类似的分析和数据修改可被提供给加拿大和澳大利亚邮政系统,加拿大和澳大利亚邮政系统为收信地址正确的邮件提供折扣率。在 IBM 的 WebSphere QualityStage 产品中可找到数据准备阶段 304 的商业实施例的一个非限制性例子。

数据集成系统还可包括一个变换、富集和递送变换后数据的数据变换阶段 308。数据变换阶段 308 可进行过渡服务,比如数据的重新组织和重新形成,并根据系统用户的商业规则和算法进行计算。数据变换阶段 308 还可把目标数据组织成称为数据集市(datamart)或者立方体的子集,以便在某些分析语境中更调谐地处理数据。数据变换阶段 308 可采用桥、翻译器或者其它接口(如下概述)来跨越数据集成系统 104 使用的各种数据源和数据目标的各种软件和硬件体系结构。数据变换阶段 308 可包括图形用户接口,命令行接口,或者这些接口的一些组合来设计跨平台 100 的数据集成作业。在 IBM 的 WebSphere DataStage 产品中可找到数据变换阶段 308 的商业实施例的一个非限制性例子。

可利用并行执行系统 310,或者按照串行或者组合的方式执行数据集成系统 104 的阶段 302、304、308,以优化系统 104 的性能。

数据集成系统 104 还可包括管理与数据源 102 相关的元数据的元数据管理系统 312。一般来说,元数据管理系统 312 可跨越数据集成环境中的所有工具,提供元数据的互换、集成、管理和分析。例如,元数据管理系统 312 可提供不同的来源,例如 IBM 的 WebSphere ODBC MetaBroker、CA ERwin,IBM 的 WebSphere ProfileStage,IBM 的 WebSphere DataStage,IBM 的 WebSphere QualityStage,IBM DB2 Cube Views 和 Cognos Profilestage 中的数据的数据的常见、普遍的可访问视图。元数据管理系统 312 还可为数据结构的变化提供用于分析数据沿袭和影响的分析工具。元数据管理系统 312 还可被用于为数据集成系

统内的数据准备数据定义、算法和商业环境的商业数据术语表，所述术语表可被公布以便在整个企业内使用。在 IBM 的 WebSphere MetaStage 产品中可找到数据管理系统 312 的商业实施例的一个非限制性例子。

参见图 4，可关于各种语境和分层结构说明与企业相关的项目，以便捕获项目的语义语境。从而，图 4 描述项目的一个语义标识符。项目可以是对象、类、属性、数据项、数据模型、元数据模型、模型、定义、身份、结构、语言、映射、关系、实例或其它项目或概念，包括另一语义标识符。语义标识符可根据项目的属性，项目的物理位置，项目与一个或多个其它项目的关系，例如在分层结构中的关系等来识别项目。在一些情况下，关系可被定义为某一特定关系的不存在。关系可涉及项目在关系分层结构中的位置。例如，在图 4 中，可根据项目 1 5202 和与之关联的其它项目的关系，识别项目 1 5202。项目 1 5202 可被识别成与项目 2 5204、项目 3 5204 和项目 4 5210 直接相关，与项目 5 5212 间接相关，并通过项目 5 5212 和项目 5 5210 与项目 6 5214 间接相关。项目 1 还可被识别成与项目 2 5204、项目 3 5204 和项目 4 5210 直接相关。在实施例中，项目 1 5202 和项目 5 5212 及项目 6 5214 之间的间接关系可被记录在项目 1 5202 到项目 4 5210 的关系中。除了静态标识符之外，这种串联或递归的识别还允许动态标识符。例如，如果项目 4 5210 和项目 6 5214 之间的关系改变，那么通过项目 4 5210 的并入，项目 1 5202 的包含项目 2 5204、项目 3 5204 和项目 4 5210 的语义标识符会体现这种变化，不需要为了说明项目 6 5214 方面的变化而被更新，好像项目 6 5214 被直接包括在语义标识符中那样。

图 5 表示了语义标识符的一个更具体例子。Jim 可被识别成居住在 111 Anyroad, Anytown, Anystate USA，电话号码为 555-555-5555 并且社会保险号为 013-65-8067 的 Jim。另一方面，可按照 Jim 与其它人的关系识别他。如图 5 中所示，Jim 可被识别成 Betty 的儿子，Larryt 和 Jeff 的兄弟，Jessica 的父亲，以及 Frank 的侄子。

语义标识符可以是一个项目的唯一标识符。在图 5 的例子中，如

果在世界上只存在一个为 Betty 的儿子, Larryt 和 Jeff 的兄弟, Jessica 的父亲, 以及 Frank 的侄子的 Jim, 那么该语义标识符是 Jim 的唯一标识符。某一项目的唯一语义标识符可能考虑比该项目与其它项目的所有关系少的关系。在图 5 的例子中, 如果世界上只有一个 Jim 是 Betty 的儿子, Larry 的兄弟和 Jessica 的父亲, 那么这些关系的单独存在就足以产生个唯一的语义标识符。不需要考虑 Jim 与 Jeff 和 Frank 的关系。创建一个以确保唯一性的最少数量的关系为基础的语义标识符是有利的。例如, 如果语义标识符将被保存在数据库 112 中或者将由数据集成系统 104 处理, 那么不太复杂的语义标识符需要较少的空间, 并且便于更快速地处理。

创建某一项目的唯一语义标识符所需的关系的数目可能根据语境发生变化。图 6A 描述关心的两个项目: 项目 1 5402 和项目 7 5404。在语境 A 5408 中, 可依据项目 1 5402 与项目 5 5410 和项目 6 5412 的关系, 把项目 1 5402 和项目 7 5404 区分开。即, 在语境 A 中, 项目 1 5402 的唯一语义标识符可以是直接与项目 2、3 和 4 相关的语义标识符, 通过项目 4 间接与项目 5 5410 相关的语义标识符, 通过项目 5 5410 和项目 4 间接与项目 6 5412 相关的语义标识符。在语境 A 中, 项目 7 5494 的唯一标识符可以是只与项目 2 和 3 直接相关的语义标识符。图 6B 表示在不同语境 B, 语境 B 5414 中的项目 1 5402。为了在语境 B 5414 中唯一地识别项目 1 5402, 可以考虑项目 1 5402 与项目 4 的直接关系, 与项目 6 的直接关系的不存在, 或者与项目 5 的间接关系中的任意一个或多个。在语境 B 5414 中, 项目 1 5402 在语义上可被唯一地识别成与项目 2 和 3 直接相关, 但是不与项目 6 直接相关。从而, 在语境 A 5408 和语境 B 5414 之间, 项目 1 的唯一标识符不同。从而, 在这里描述的数据集成方法和系统的实施例中, 项目的语义标识符, 比如与数据集成作业或者数据集成平台相关的项目可具备该项目的语境相关标识符。在实施例中, 可以原子格式把这样的语境相关标识符保存在数据储存库中。

在其它实施例中, 语境 A 5408 和 B 5414 可以是两个不同的进口、

映射、运行版本 **run version**、模型、元代理(**metabroker**)模型、实例、工具、视图、对象、类、项目、关系、属性或者上述任意的任意组合。匹配或比较装置可比较在不同进口、运行版本、模型、元代理模型、实例、工具和/或项目中，一个项目的身份的语法，并根据所述比较确定或帮助确定要采取或者要避免采取什么行动。例如，匹配引擎可比较进口实例 A 使用的模型和元代理 B 使用的模型。根据该比较，可确定元代理 B 可在没有变换或修改的情况下访问进口实例 A 的数据和元数据，比较装置可指令元代理 B 继续前进。在另一例子中，可比较工具 A 5408 和工具 B 5414，可确定进行交叉工具对象合并，其中每个工具能够访问和使用另一工具的对象。在实施例中，比较装置可触发翻译装置帮助交叉工具对象合并，例如建立桥，元代理，中心(**hub**)等，以便转换任何需要转换的对象，所述转换比如是以在每个相应工具中特定项目的身份的处理的不同语法为基础的转换，或者以所述比较确定的工具之间的其它差别为基础的转换。

在实施例中，可用以串结构或格式保存、维持、记录、处理和/或解释的语法保存、维持、记录、处理和/或解释语义标识符。图 7 描述语法和用该语法构成的对应串的一个例子。语法 5502 可以是列名称：：表名称：：数据库名称。该语法可与识别数据库中的一个表的某一系列的语法标识符相关。以该语法构成的串 5504 可以是年龄：：雇员：：雇员数据库。该串可与识别特定雇员数据库中某一雇员的年龄的语义标识符相关。在图 6B 的例子中，对应于语境 B 5414 中项目 1 5402 的语义标识符的串可以是：与项目 2 的直接关系：：与项目 3 的直接关系：：与项目 4 的间接关系。语义标识符和对应的串还可体现项目 1 5402 和项目 6 之间直接关系的不存在。

在图 8 中，项目 9 5602 的串格式的语义标识符可以是：直接到项目 2：：直接到项目 3：：直接到项目 4：：间接到项目 5 5604。串能够被解析。语法和/或串可被截短、修改和/或语法和/或串的元素可被重新排序。在图 9 中，串 5702 是串 5604 的截断，串 5704 是串 504 的截断和修改和/或重新排序，串 5708 是串 5606 的个性和/或重新排

序。所述截断、修改和/或重新排序可由转换引擎进行。当就语义标识符的唯一性来说，不需要包括在语法和/或串中的所有关系时，截短语法和/或串是有益的。假定在串 5604 的指定语境中，所有项目都与项目 3 直接相关；例如，项目 3 是其中保存所有项目的数据库。串 5604 可被截短，从而产生串 5702，省略涉及项目 3 的关系，同时仍然是一个唯一的标识符。截短语法和/或串可降低存储要求，并提高处理效率。改变语法和/或串中的关系的顺序也有益于减少数据集成过程的处理时间。如果首先处理不太常见的关系，那么系统很可能只需要访问和处理与某一项目相关的较少关系就能够识别该项目。例如，如果很少的项目与项目 3 相关，更少的项目与项目 4 相关，并且许多项目与项目 2 相关，那么取决于语境，与串 5604 相比，串 5708 可能便于在较短的时间中识别项目 9。在该语境中唯一地识别项目 9 可能只需要串 5708 的前两个元素，而需要串 5604 的前三个元素。

转换引擎可对一个或多个语义标识符，数据库 112，包括语义标识符的数据库 112，信息系统，包括语义标识符的信息系统或者其它项目进行转换操作。图 10 描述作用于体现为串 5804 的语义标识符，以及作用于体现为位于数据库 5808 中的串的的语义标识符的转换引擎 5802。转换操作可转换或者以其它方式修改语义标识符的格式、语言和/或数据模型。转换操作可涉及从一个或多数据工具、语言、格式和/或数据模型到至少一种其它的数据工具、语言、格式和/或数据模型的转换或映射，或者从至少一种其它的数据工具、语言、格式和/或数据模型到一个或多数据工具、语言、格式和/或数据模型的转换或映射。例如，转换操作可涉及往来于已知的数据集成工具，或者已知的数据集成工具之间的转换或映射，所述已知的数据集成工具例如是 IBM 的 WebSphere DataStage 7，IBM 的 WebSphere QualityStage，Business Object 工具，IBM-DB2 Cube Views，UML 1.1，UML 1.3，ERStudio，IBM 的 WebSphere ProfileStage，PowerDesigner(另外支持 Packages and Extended Attributes)和/或 MicroStrategy 工具。转换引擎和/或转换操作可随意地包含在元代理中。可分批地，实时地和/

或连续地进行、执行和/或实施转换操作。可以服务的形式，例如作为面向服务的体系结构的一部分的形式提供服务或者使之可用。SOA 可以是商业企业的企业计算系统的基础结构的一部分。在 SOA 中，服务变成应用程序开发和部署的构件块，允许快速应用程序开发和避免冗余代码。每个服务具体体现周围的环境，例如服务的数据输入的来源或者服务的数据输出的目标看不见的一组商业逻辑或者商业规则。从而，服务可随同各种应用程序被重新使用，只要在服务和应用程序之间建立了正确的输入和输出。面向服务的体系结构允许保护服务免受环境变化的影响，以致即使周围的计算机环境被改变，该体系结构也能够正常工作。从而，服务不需要被记录成基础结构变化的结果，这可节省时间和工作。SOA 可用于一个 web 服务，可涉及三个实例，一个服务提供者，一个服务请求者和一个服务注册器（registry）。注册器可以是公共注册器或者专用注册器。服务请求者可搜索注册器寻找适当的服务。一旦发现适当的服务，服务请求者可接收调用该服务所必需的代码，例如 Web 服务描述语言（“WSDL”）代码。WSDL 是通常用于描述 web 服务的编程语言。服务请求者随后例如通过恰当格式（例如关于 web 服务消息的简单对象访问协议（“SOAP”）格式）的消息，与服务提供者连接，以便调用该服务。SOAP 协议是在 web 服务中传送数据的优选协议。SOAP 协议定义 web 服务客户端和 web 服务服务器之间消息的交换格式。SOAP 协议使用可扩展置标语言（“XML”）方案，XML 是在 web 服务中常见的用于标记数据的类似语言规范，不过也可使用其它置标语言。

一旦对于语义标识符，数据库 112，包括一个或多个语义标识符的数据库 112，信息系统，包括一个或多个语义标识符的信息系统或者其它项目来说，存在转换操作，那么它可被转换成或者转换自，映射到，链接到任意其它的语义标识符，数据库 112，包括一个或多个语义标识符的数据库 112，信息系统，包括一个或多个语义标识符的信息系统或者共享至少一个转换操作的其它项目，和任意其它的语义标识符，数据库 112，包括一个或多个语义标识符的数据库 112，信息

系统，包括一个或多个语义标识符的信息系统或者共享至少一个转换操作的其它项目一起使用或者与之相关联。在实施例中，例如通过把原子数据储存库用作转换操作的中心，除了其它之外，转换操作的映射跟踪在初始语义语境和转换后的语义语境之间来回执行的转换操作中转换的数据。根据语境，数据的适当标识符可变化，例如在语义语境变化的情况下，通过改变或者截短语法和/或串，从而能够实现更有效的存储或更快的处理，或者通过改变用于形成唯一标识符的关系。从而，在使用数据项的各种语境中，动态标识符能够结合可回溯转换的优点与快速处理、有效的数据处理和有效操作的优点。

指定的项目，比如在模型中具有身份的项目可以多种形式或实例存在，例如物理实例和逻辑建模实例。图 11 描述一个项目，即雇员信息表 5902。但是，概念或实体“雇员”可以多种不同的形式存在于企业内。例如，雇员表 5902 可以保存和雇员相关的值的物理表的形式存在于物理数据存储装置中。另一方面，实体雇员也可被表现为逻辑实例，例如逻辑建模活动 5908 中代表雇员的图标或文本，或者各种其它的形式或实例。即，相同的项目(包括任何相关的数据或元数据)可跨视图、模型、结构或数据集成环境，以多种形式或实例存在于数据库、数据储存库、模型、中心等中。图 12 描述在数据库 6002 中呈一种形式或者单一实例，和/或在数据库 6004 或中心 6008 中呈一种以上的形式或实例的雇员表 5902。

为了区分项目的各种形式或实例，可以使用任何辨别特征，例如抽象层次，项目的物理性质，项目在分层结构中的位置，项目在数据库中的位置，其中发现项目的语境，项目的语法，项目与其它项目的关系，项目的属性，项目的类别或者其它特性。例如，参见图 5，根据年龄、性别、发色、IQ、政治面貌和/或过去三个月中看医生的次数区分项目，或者说这种情况下区别个人。例如，如果年龄被选为产品区别特征，那么 Jessica 是唯一一个小于 10 岁的，Betty 是唯一一个介于 57 岁和 67 岁之间的，Jim 是唯一一个 37 岁的。在另一例子中，项目的不同形式或实例可存在于不同的抽象层次或者不同的语境中。例

如，雇员表可以多种形式或实例存在于中心 6102 中，例如物理雇员表 5904(比如用于把和关于雇员的数据有关的值保存在数据库中)，和逻辑雇员模型 5908(比如将用在和雇员相关的过程的视图中)。

区分识别的特定项目的不同实例能够实现各种其它方法和过程。例如，在一个实施例中，项目，比如名为“雇员”的表可被带到中心。中心收集者在中心中具有“雇员”的两种形式中实例；一种对应于物理数据库实例，另一种对应于逻辑建模活动。辨别特征，比如中心中起因于该项目的项目特性可以供区分物理实例和逻辑模型实例或形式之用。在实施例中，辨别特征可被称为抽象层次，以便区别逻辑抽象层次和物理抽象层次。在其它情况下，中心可把其它特征与项目关联起来，比如不同形式的标识符、关系、类、属性、物理位置、逻辑位置、模型等。

如图 14 中所述，当进行操作时，例如选择将被装入数据库中的数据，转换数据，产生查询等时，系统，比如转换引擎 6204 能够从中心 208 或数据库 6210 夺取、装入或者获得所有项目。它可根据任意辨别特征选择或过滤 6204 项目。例如，它可选择或滤出具有物理抽象层次，和其它项目具有特殊关系，具有逻辑抽象层次，在规定的日期和时间之前创建的，或者具有任何其它辨别特征的那些实例或形式。从而，这里描述的方法和系统根据任何辨别特征，有选择地处理相同项目或实体的实例。

如图 15 中所示，当进行响应查询 6202 的数据集成操作，比如转换操作时，转换引擎 6204 可在中心 6208 或数据库 6210 过滤或选择项目，包括任何数据和/或元数据，只夺取、装入或获得相关抽象层次的那些项目。例如，它可滤出或选出具有逻辑抽象层次的那些实例或形式，只保留具有物理抽象层次的那些实例或形式。可在运行时或者设计时进行所述过滤或选择，可以分批地、实时地或者连续地进行所述过滤或选择。在实施例中，可以面向服务的体系结构中的 RTI 服务的形式提供这样的过滤或选择方法。

过滤或选择可以在开发时、设计时或者运行时由转换引擎和/或

系统获得的信息为基础，例如数据模型的映射，元数据模型的映射，辨别特征，项目与其它项目的关系，项目的属性，或者标识符的语法。在实施例中，所述信息可被实时地动态更新。

在整个过程中，过滤或选择越接近中心或数据库，操作就越高效和越快。如图 16 中所示，转换引擎 6204 可对查询 6202 本身进行转换操作，产生修改的查询 6402，修改的查询 6402 可被直接发送给中心 6208 或数据库 6210 以便进一步处理。例如，可用直接与中心 6208 或者数据库 6210 的本地格式兼容的格式呈递修改的查询 6402。例如，通过用数据库 6210 的本地格式呈递查询，系统能够提高对该查询的处理效率。类似地，查询 6402 可被过滤，或者可产生诸如选择命令之类的命令，以保持逻辑建模实体而不是物理实体，在这种情况下，可按照适合于逻辑建模活动，而不是适合于数据库的格式(例如图形用户接口)呈递查询 6402。当然，不仅查询，而且其它消息和操作都可按照抽象层次被过滤，使得能够跨数据集成平台跟踪相同的实体，并按照特定的数据集成活动的适当操作环境处理相同的实体。

这里描述的方法和系统可被用于捕获语义语境，以及关于和企业相关的各种项目处理数据集成任务，所述各种项目比如是对象、数据项、数据、列、行、表、数据库、实例、属性、元数据、概念、题目、主题、语义标识符、其它标识符、RFID 标签、卖主、供应者、顾客、个人、团队、组织、用户、网络、系统、设备、家庭、商店、产品、产品线、产品特征、产品规格、产品属性、价格、成本、材料单、装运数据、税款数据、课程、教育规划、位置、地图、部门、组织、有机体、过程、规则、法律、收费制度、物品、服务和/或服务提供。

这里描述的方法和系统可用在各种语义语境中，例如企业方法中的步骤，数据库中的数据，行或列中的数据，表中的行或列，数据库中的行或列，表中的数据，数据库中的表，数据库中的元数据，中心或储存库的项目，数据库中的项目，表中的项目，列中的项目，行中的项目，机构中的人，通信的发送者或接收者，网络上的用户，网络上的系统，网络上的设备，家庭中的成员，商店中的物品，菜单上的

菜，产品线上的产品，产品提供中的产品，教育或训练计划中的课程或步骤，地图上的位置，物品的位置，机构的部门，团体中的个人，规则体系中的规则，一套服务中的服务，企业的组织层次中的实体，供应链中的实体，市场中的消费者，购买决策的购买者，商品或服务的价格，商品或服务的成本，产品或系统的组件，方法的步骤，小组的成员或者其它许多。

虽然关于一些优选实施例说明了本发明，不过应明白本领域的技术人员会认识到其它实施例，并且所述其它实施例在本公开的范围內。

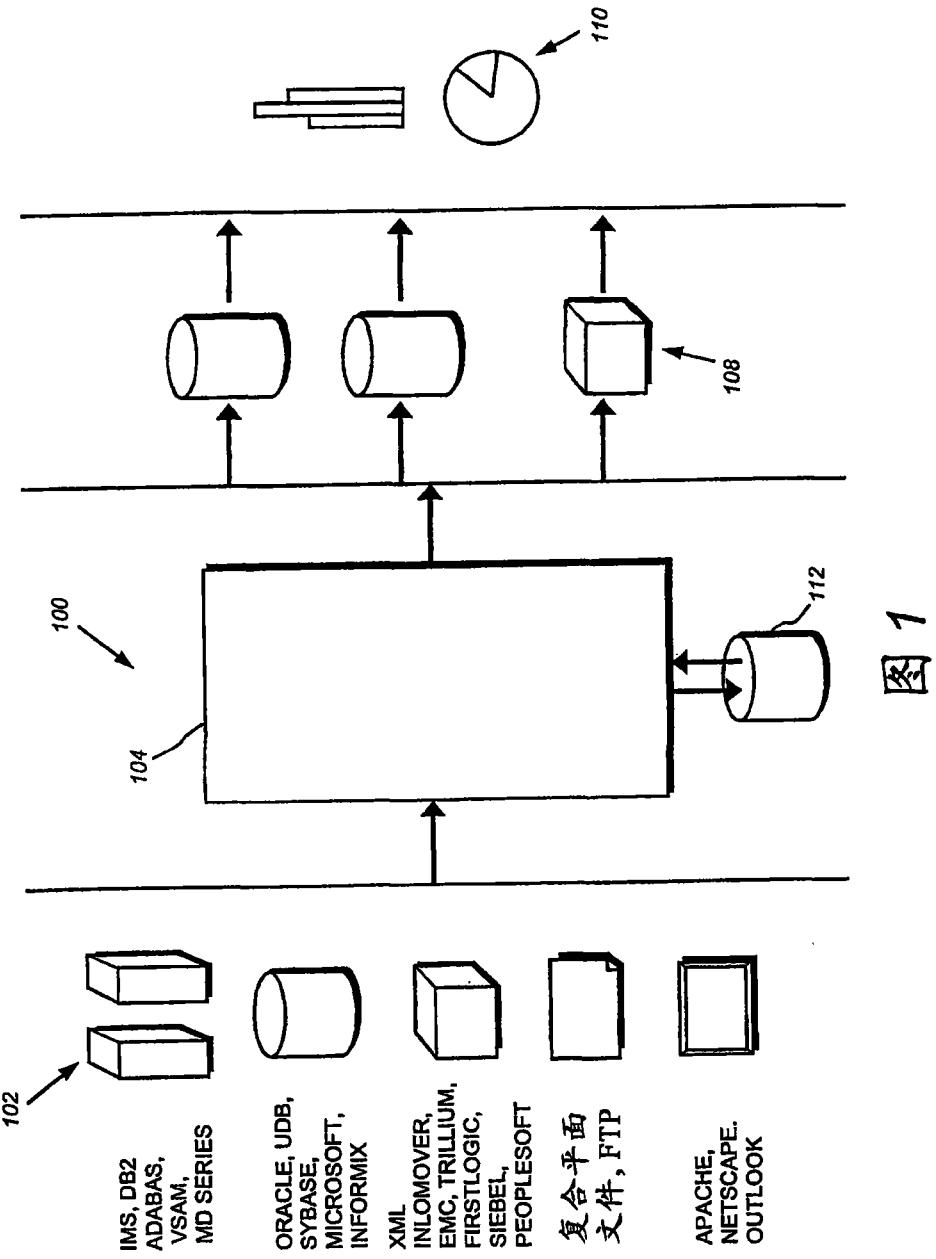


图1

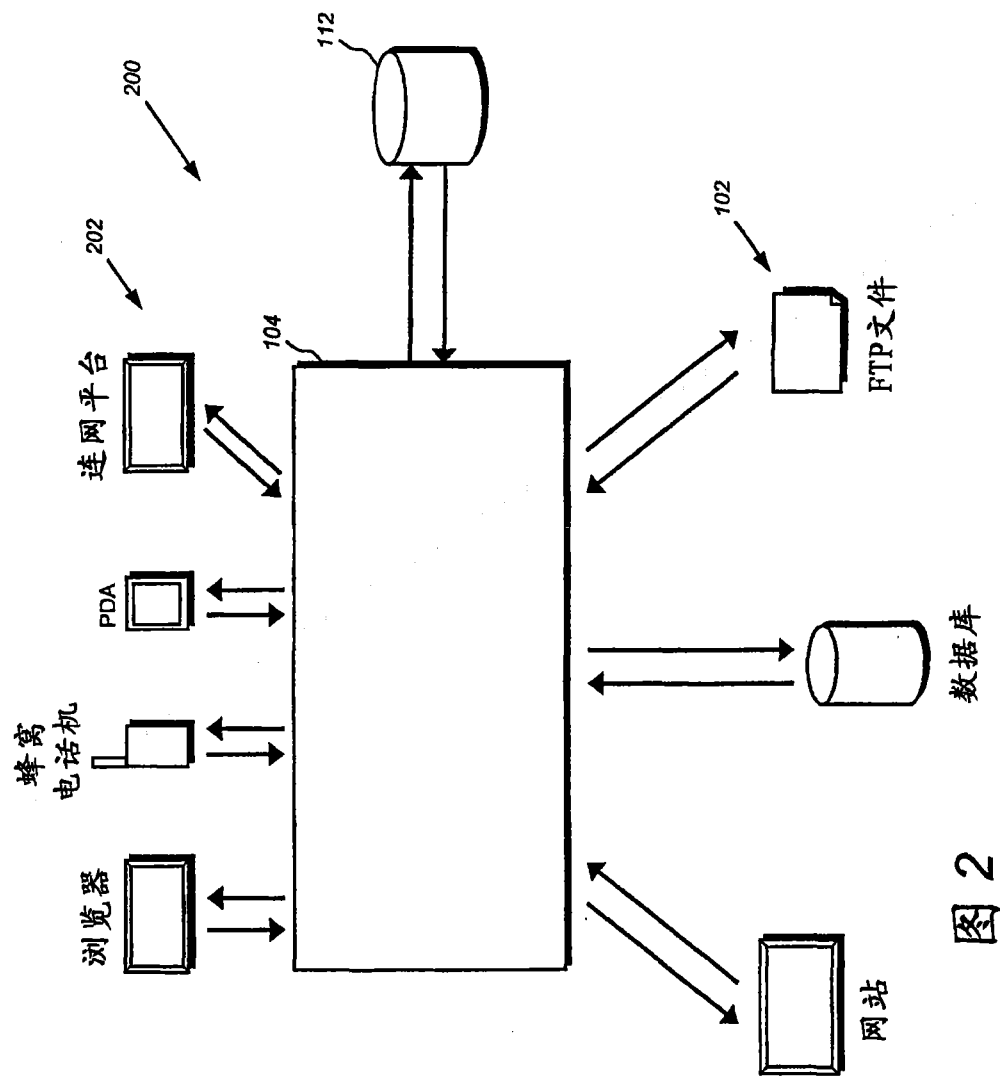


图 2

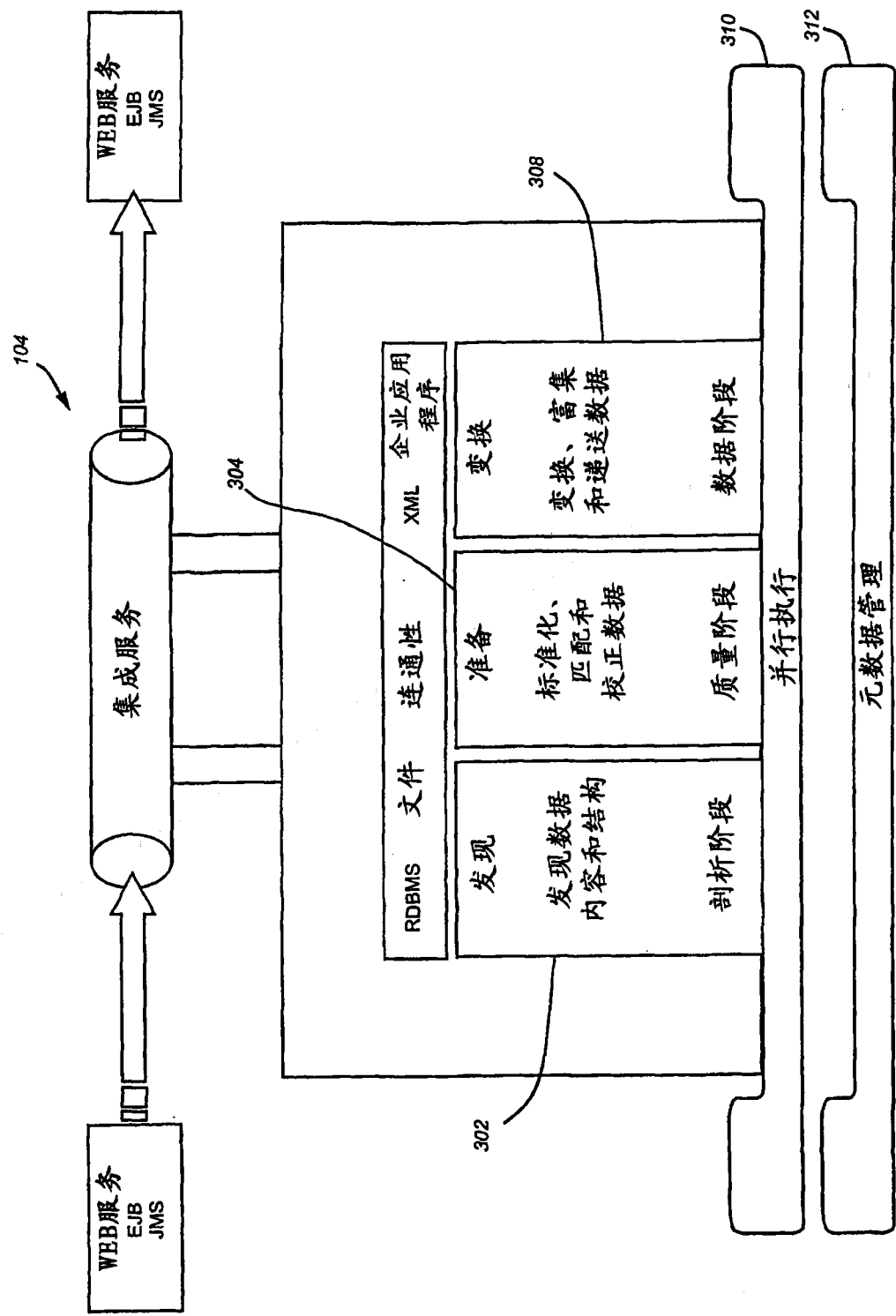


图 3

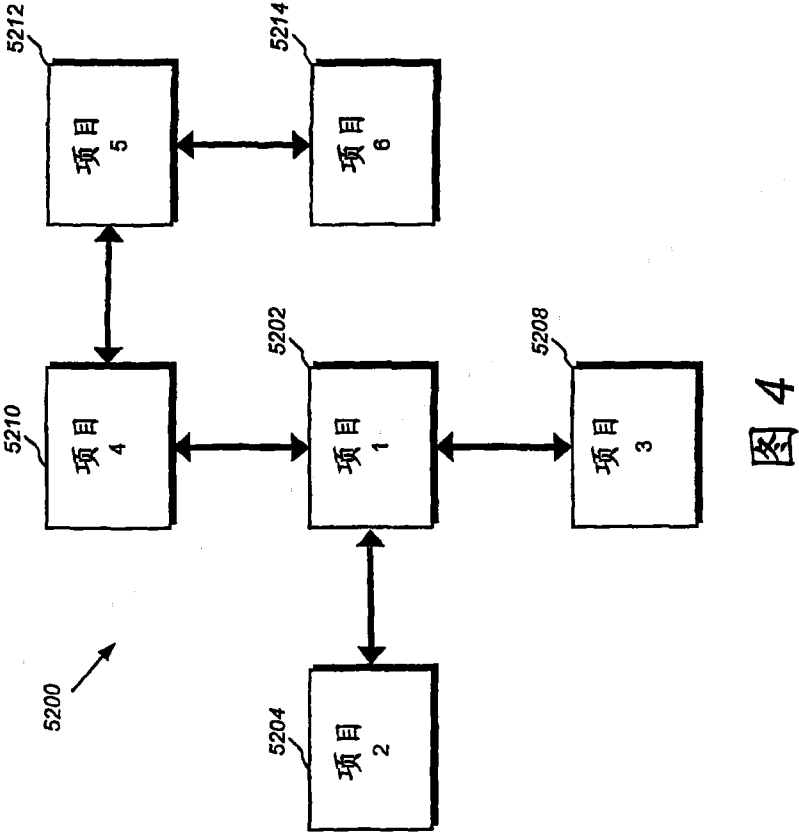
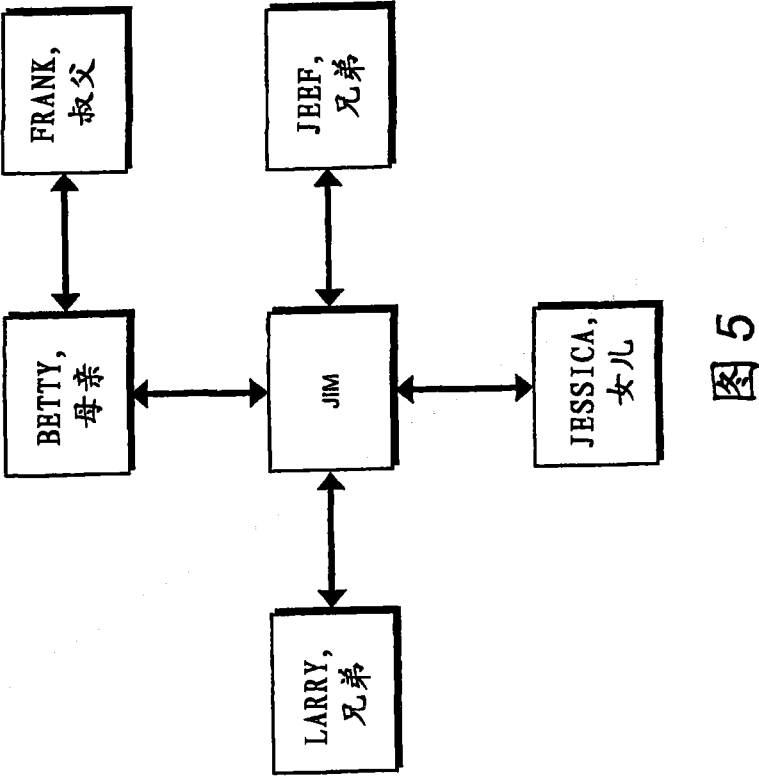


图 4



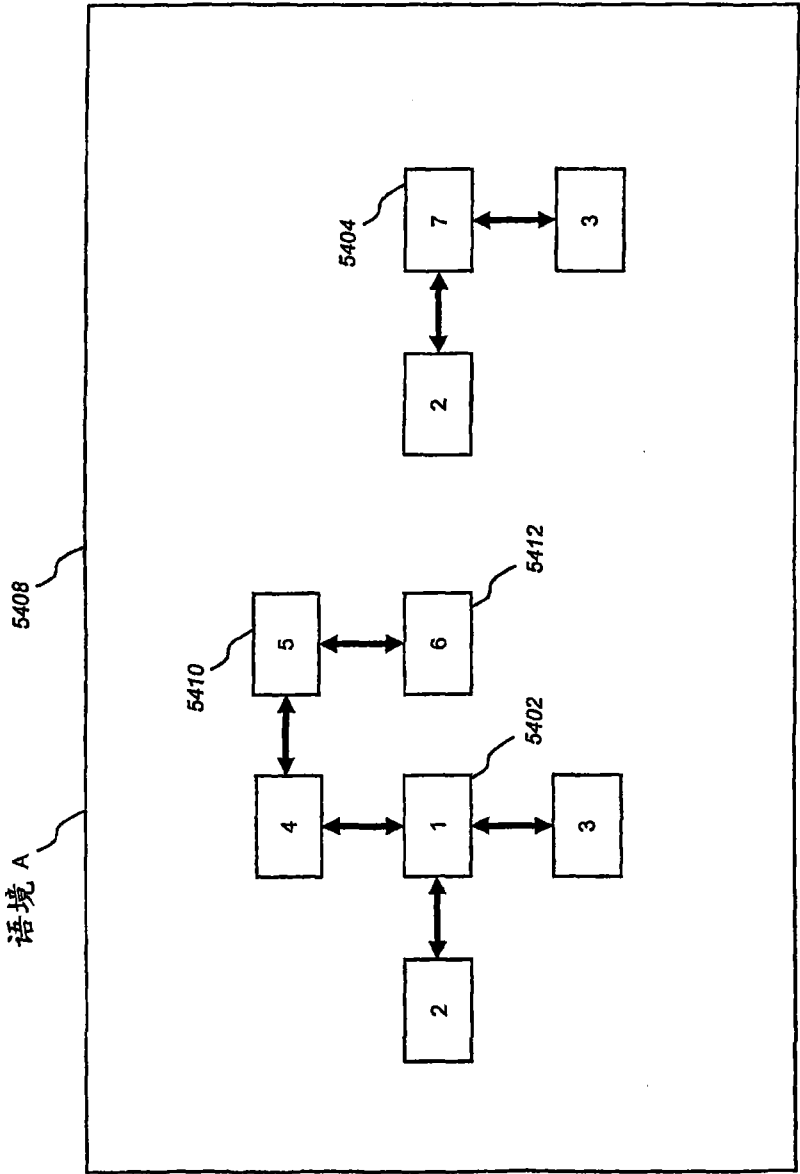


图 6A

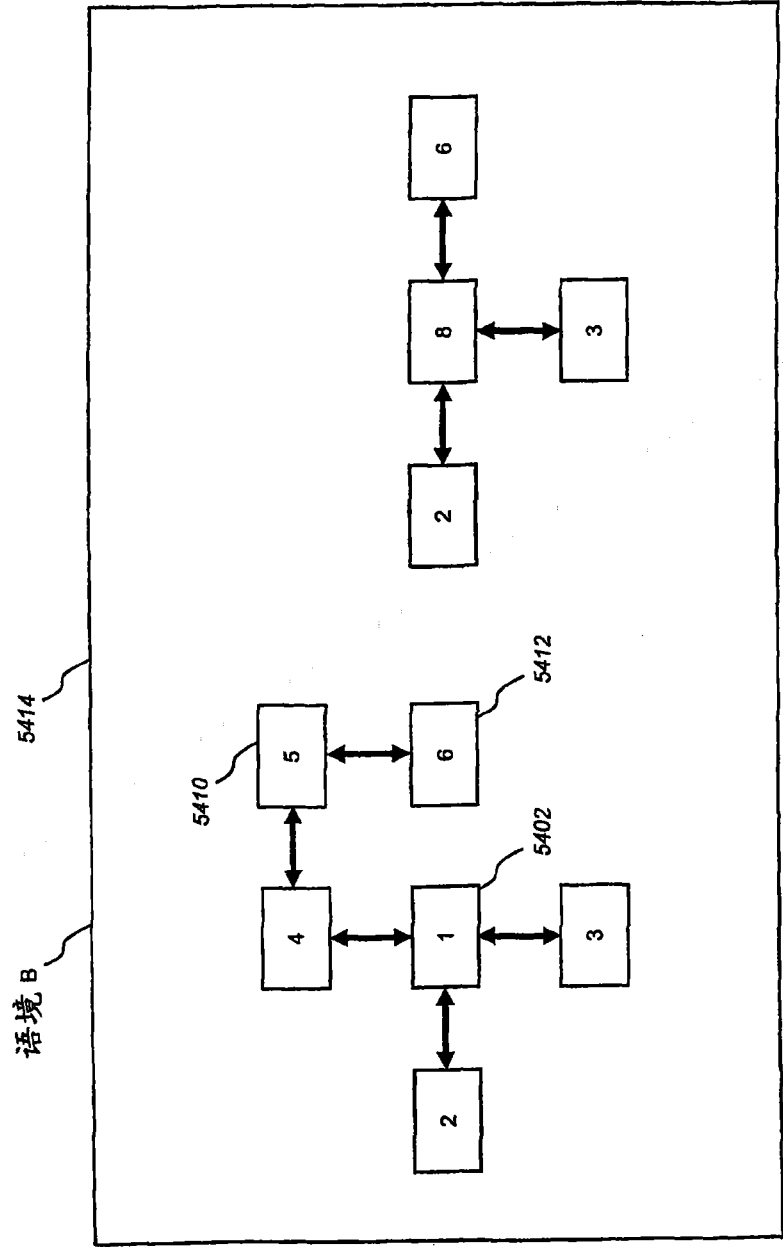


图 6B

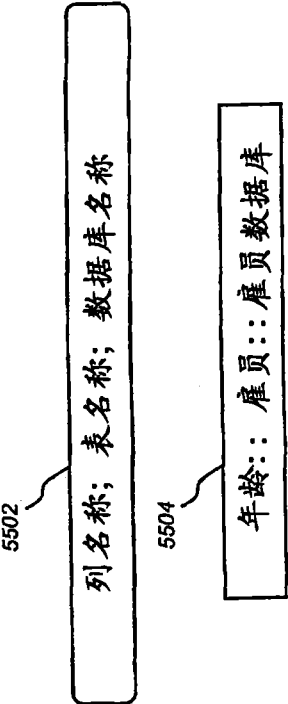


图 7

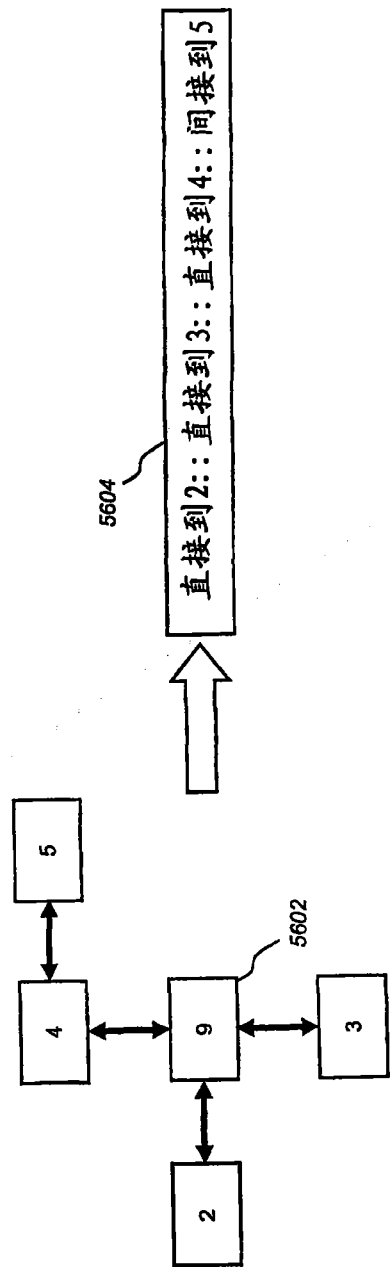


图 8

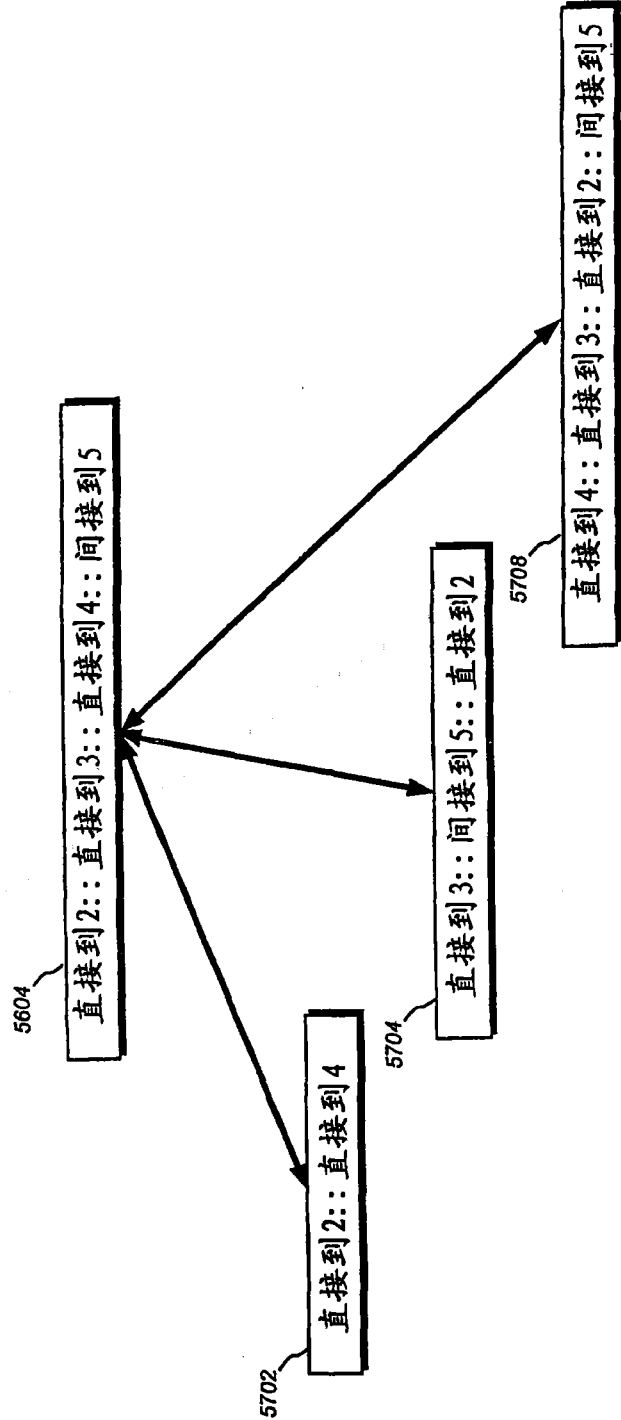


图 9

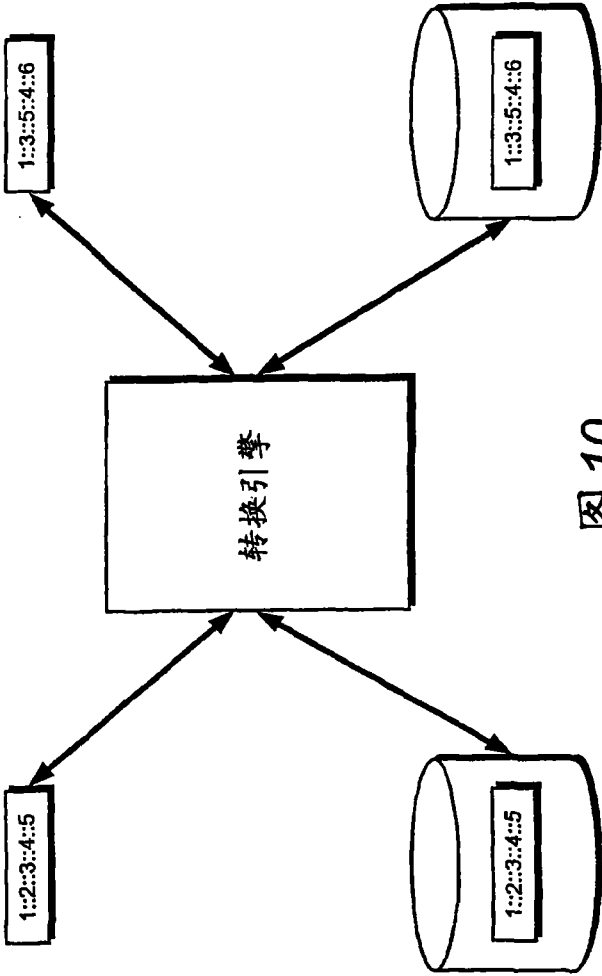


图10

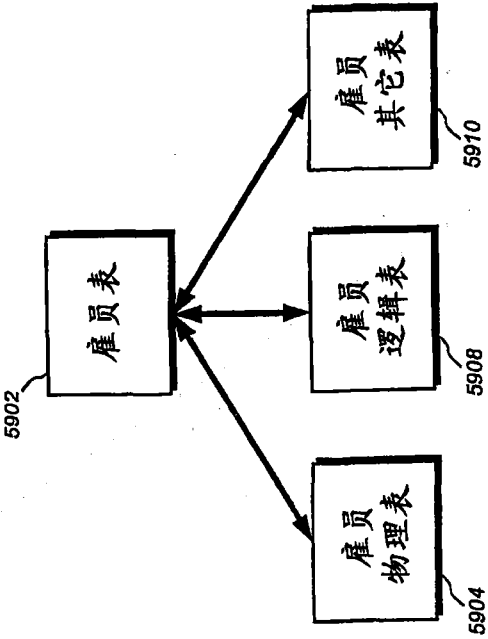


图 11

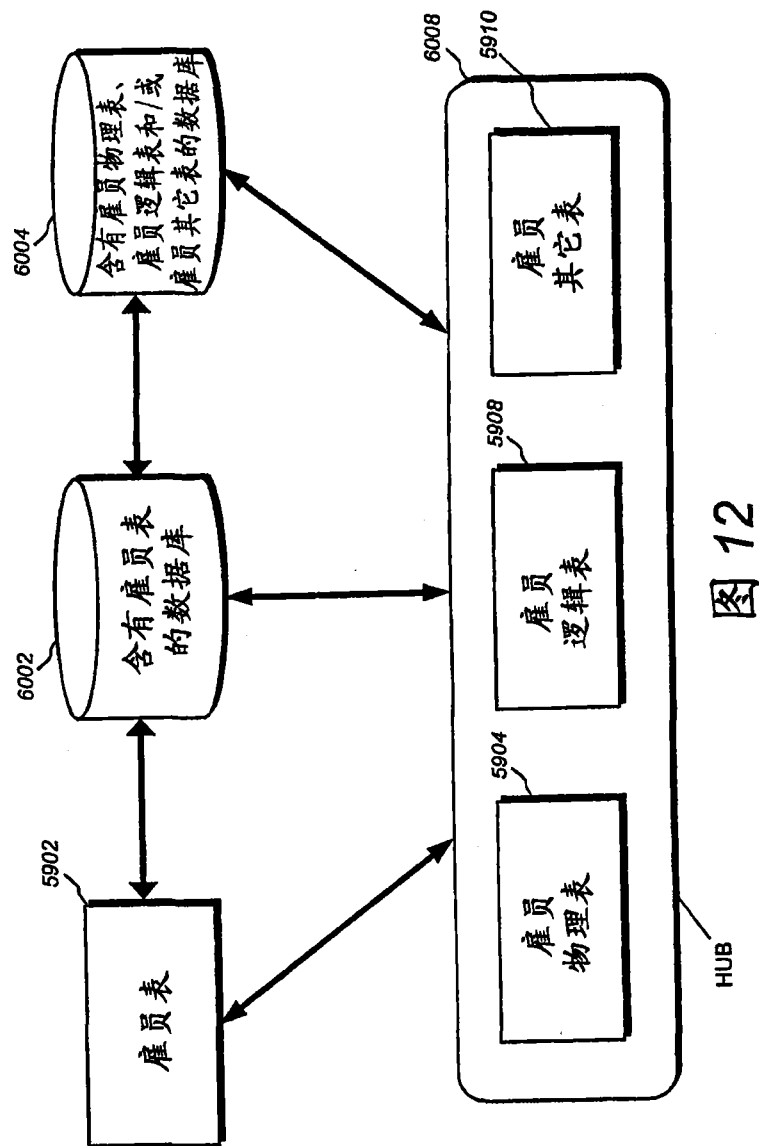


图 12

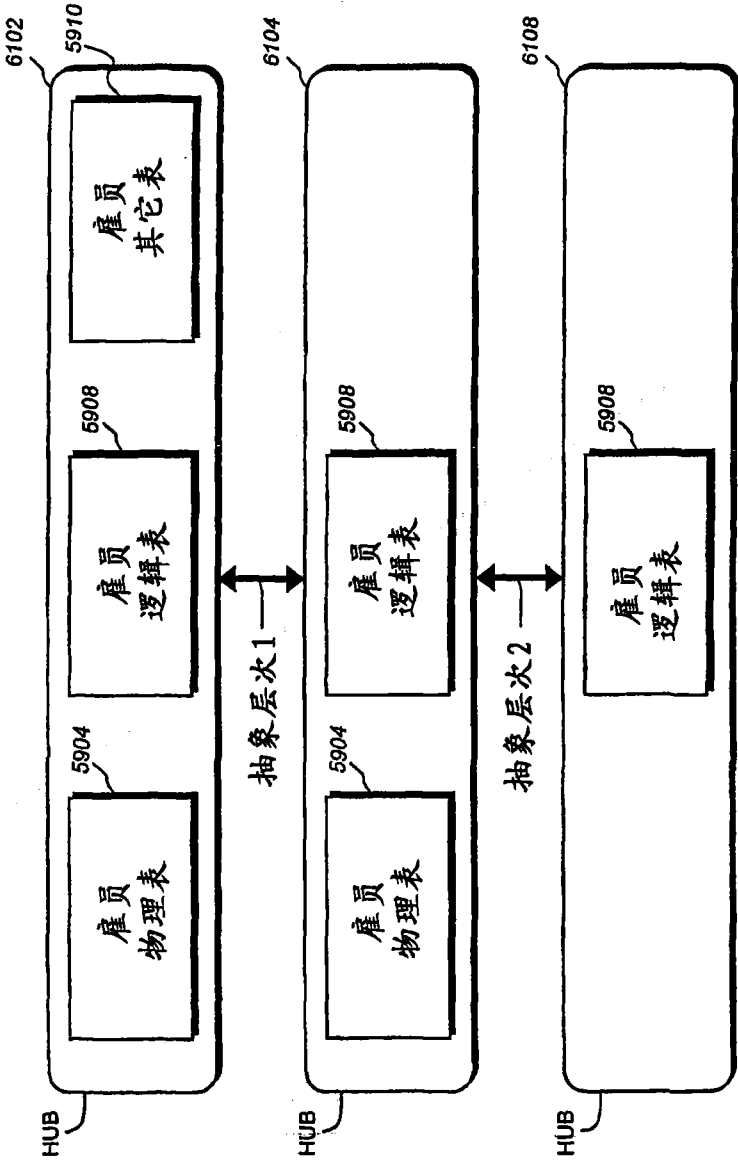
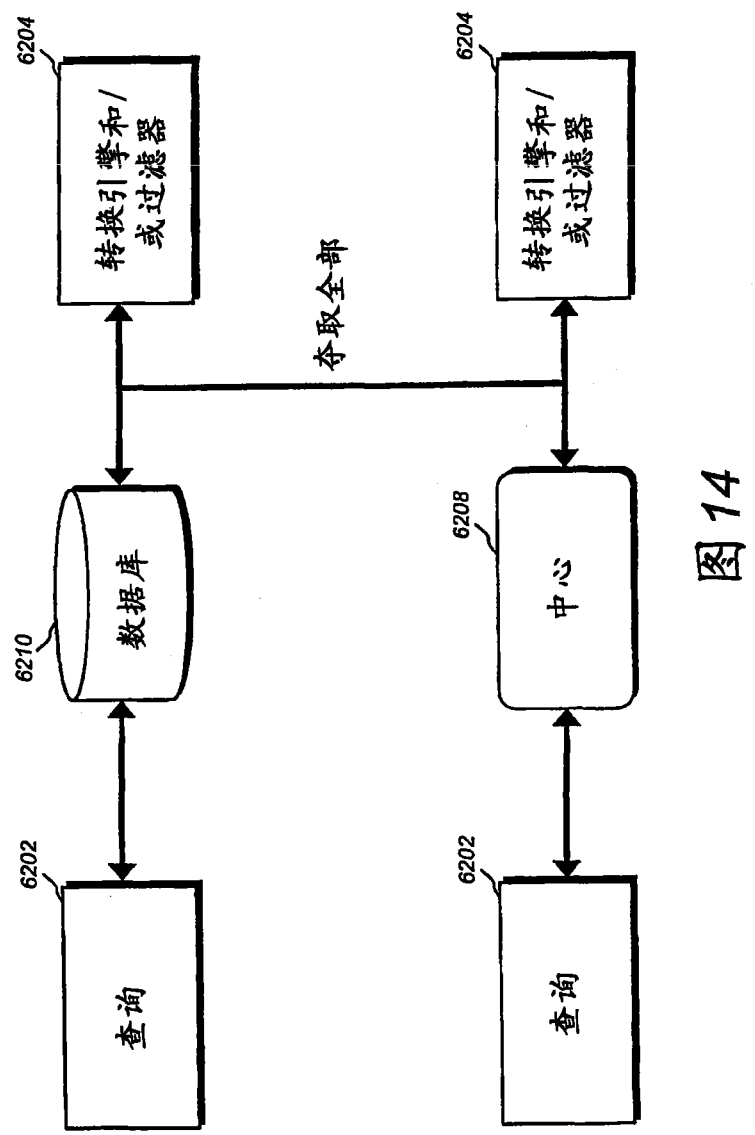
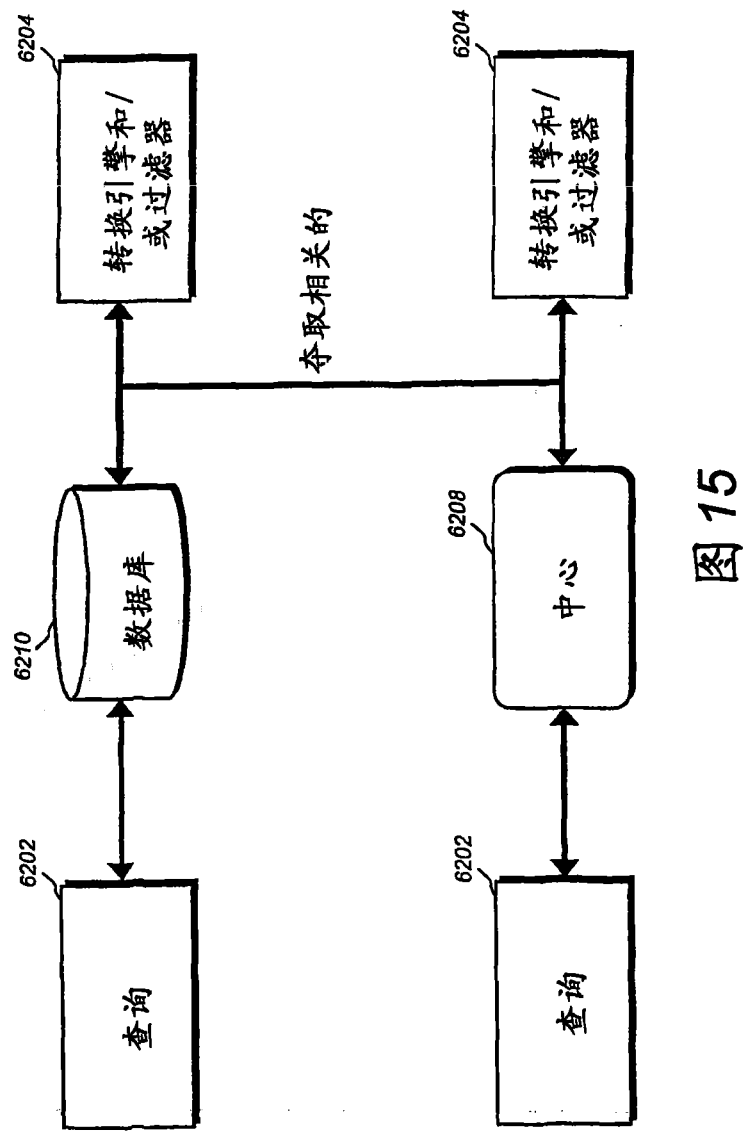


图13





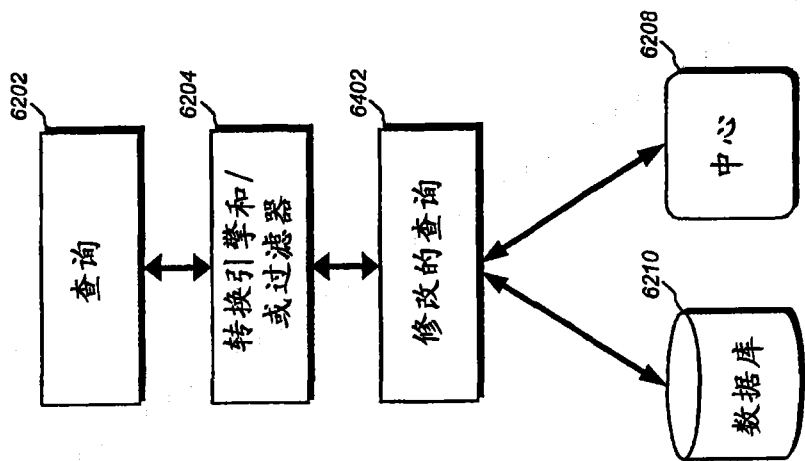


图16