

[19] 中华人民共和国国家知识产权局

[51] Int. Cl.

G06F 9/44 (2006.01)  
G06F 17/00 (2006.01)



# [12] 发明专利申请公布说明书

[21] 申请号 200780013290.1

[43] 公开日 2009年5月13日

[11] 公开号 CN 101432694A

[22] 申请日 2007.2.27

[21] 申请号 200780013290.1

[30] 优先权

[32] 2006.4.12 [33] US [31] 11/279,542

[86] 国际申请 PCT/US2007/005158 2007.2.27

[87] 国际公布 WO2007/130200 英 2007.11.15

[85] 进入国家阶段日期 2008.10.13

[71] 申请人 微软公司

地址 美国华盛顿州

[72] 发明人 V·米塔尔 W·W·希勒佩特

C·G·尚卡拉纳拉亚

G·R·普里克利

G·B·奇克多洛夫

[74] 专利代理机构 上海专利商标事务所有限公司

代理人 陈斌

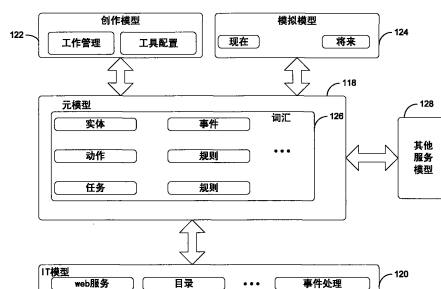
权利要求书4页 说明书17页 附图12页

[54] 发明名称

业务流程元模型

[57] 摘要

一种表示业务流程的元模型。定义用于标识业务流程的操作表达式。业务流程包括多个活动序列。收集与所定义的操作表达式相关联的元数据。根据一模式来格式化所收集的元数据。根据格式化的元数据和所定义的操作表达式将表示业务流程的元模型提供给用户以使得该用户能够使用所定义的操作表达式来与该业务流程交互。



1. 一种用于表示业务流程的方法，所述方法包括：  
定义用于标识所述业务流程的操作表达式，所述业务流程包括多个活动序列；  
收集与所定义的操作表达式相关联的元数据；  
根据一模式来格式化所收集的元数据；以及  
根据所述格式化的元数据和所定义的操作表达式将表示所述业务流程的元模型提供给用户以使得所述用户能够使用所定义的操作表达式来与所述业务流程交互。
2. 如权利要求 1 所述的方法，其特征在于，定义操作表达式包括定义用于展示所述业务流程的操作集合（116）以及供用户（108）操控的所包括的多个活动序列。
3. 如权利要求 1 所述的方法，其特征在于，定义操作表达式包括定义词汇集合（114），所述词汇集合为所述用户（108）和开发者（236）描述所述业务流程。
4. 如权利要求 1 所述的方法，其特征在于，还包括生成描述所述业务流程的状态的报告（312），所述状态根据所述操作表达式和所收集的元数据定义所述多个活动序列的进度。
5. 如权利要求 4 所述的方法，其特征在于，还包括响应于来自所述用户（108）的输入，基于所述操作表达式模拟所述多个活动序列的进度。
6. 如权利要求 1 所述的方法，其特征在于，还包括将所收集的元数据与包括在所述业务流程中的所述多个活动序列相关。
7. 如权利要求 1 所述的方法，其特征在于，提供包括发现与所述业务流

程相关联的一个或多个现有配置，所述现有配置定义用于与所述业务流程交互的现有操作。

8. 如权利要求 1 所述的方法，其特征在于，一个或多个计算机可读介质具有用于执行如权利要求 1 所述的方法的计算机可执行指令。

9. 一种用于描述业务流程的系统（100），所述系统包括：

用于收集元数据的接口；

用于存储所收集的元数据的存储器区域（106）；

处理器（104），它被配置成执行用于以下步骤的计算机可执行指令：

定义用于标识所述业务流程的操作表达式，所述业务流程包括多个活动序列；

根据所定义的操作表达式将所收集的元数据与所标识的业务流程相关联；

根据一模式（700）来格式化所收集的元数据；以及

按照所述格式化的元数据和所定义的操作表达式通过所述接口将表示所述业务流程的元模型（118）提供给用户（108）以使得所述用户（108）能够使用所定义的操作表达式来与所述业务流程交互。

10. 如权利要求 9 所述的系统（100），其特征在于，所述处理器（104）被配置成定义用于展示所述业务流程的操作集合（116）以及供所述用户（108）操控的所包括的多个活动序列，并且其中所述存储器区域（106）存储与所述操作集合（116）相关的信息。

11. 如权利要求 9 所述的系统（100），其特征在于，所述处理器（104）还被配置成生成描述所述业务流程的状态的报告（312），所述状态根据所述操作表达式和所收集的元数据定义所述多个活动序列的进度。

12. 如权利要求 11 所述的系统（100），其特征在于，所述处理器（104）还被配置成响应于来自所述用户（108）的输入，基于所述操作表达式模拟所

述多个活动序列的进度。

13. 如权利要求 9 所述的系统 (100)，其特征在于，所述处理器 (104) 还被配置成将所收集的元数据与包括在所述业务流程中的所述多个活动序列相关。

14. 如权利要求 9 所述的系统 (100)，其特征在于，所述处理器 (104) 还被配置成发现与所述业务流程相关联的一个或多个现有配置，所述现有配置定义用于与所述业务流程交互的现有操作。

15. 一种或多种具有用于表示业务流程的计算机可执行组件的计算机可读介质 (502)，所述计算机可执行组件包括：

用于描述所述业务流程的词汇组件 (504)；

用户收集对应于所描述的业务流程的元数据的数据组件 (506)；

用于向用户 (108) 提供所收集的元数据的接口组件 (508)；

用于报告对所描述的业务流程的分析的报告组件 (510)；

用于评估与所收集的元数据相关联的规则以使得所述用户 (108) 能够与所述业务流程交互的逻辑组件 (512)；

16. 如权利要求 15 所述的计算机可读介质 (502)，其特征在于，所述词汇组件 (504) 包括定义用于展示所述业务流程的操作集合 (116) 以及供用户 (108) 操控的所包括的多个活动序列。

17. 如权利要求 15 所述的计算机可读介质 (502)，其特征在于，所述词汇组件 (504) 包括定义词汇集合 (114)，所述词汇集合为所述用户 (108) 和开发者 (236) 描述所述业务流程。

18. 如权利要求 15 所述的计算机可读介质 (502)，其特征在于，所述报告组件 (510) 生成描述所述业务流程的状态的分析报告 (312)，所述状态

---

根据所述操作表达式和所收集的元数据定义所述多个活动序列的进度。

19. 如权利要求 18 所述的计算机可读介质 (502)，其特征在于，所述接口组件 (508) 从所述用户 (108) 接收输入，并且还包括模拟组件 (514)，其用于响应于由所述接口组件 (508) 接收到的输入基于所述操作表达式来模拟所述多个活动序列的进度。

20. 如权利要求 15 所述的计算机可读介质 (502)，其特征在于，还包括发现组件 (516)，其用于发现与所述业务流程相关联的一个或多个现有配置，所述现有配置定义用于与所述业务流程交互的现有操作。

---

## 业务流程元模型

### 背景

通常，被配置成处理商业交易的计算机软件或应用程序是从软件开发者的观点来设计的，这使得商业用户不能够充分使用该软件或应用程序。例如，在工作流或面向过程的应用程序中，商业用户可能希望定义涉及一系列过程步骤、步骤序列和被分配来执行这些过程步骤的角色（例如，个人）的工作流。商业用户在可满意地定义工作流之前常常需要向能够将这些概念翻译为编程代码的开发者表达工作流的概念。

开发者在设计这些应用程序时施加在该商业用户的观点之外的某些限制或条件是常见的。例如，在定义过程步骤时，开发者需要标识与过程步骤相关联的数据以及如何从源连接该数据以使得该数据能够在过程步骤期间使用。这可能涉及数据变换、详细数据连接控制以及数据流、输出数据等等。开发者还可能需要考虑异常，诸如当一数据片段不可用或一角色不能完成所分配的任务时。又一示例性条件可以是序列中的步骤可触发安全或认证机制，硬件配置或设置等等。

当今的实践和可用的软件工具的一个常见问题是业务层表示和开发者层表示之间的转换通过翻译—仅仅转换和维护两个不同表示之间共同且重叠的方面的过程—来实现。然而，该翻译过程丢失了从开发者的观点来看的流程设计的细节。在一抽象层（通常是开发者层）上提供的详细方面在到不同抽象层（通常是商业用户层）的转换期间被忽略。此外，在一抽象层上作出的改变常常破坏已在不同抽象层上作出的改进。由此，开发者和商业用户同样需要重新定义现有的配置仅仅为了恢复已经存在的东西。

由此，虽然商业用户能够使用在当前面向过程的应用程序中对他人/她可用的工具或功能来定义工作流，但是缺乏真正地标识在各抽象层上定义业务流程时可用的数据和操作并与其交互的能力。

### 概述

本发明的各实施例通过在业务流程设计阶段期间准确地捕捉商业意图而

克服了现有的方法的缺点。本发明的各方面提供了在不同抽象层之间变换流程设计的一致且高效的方法。通过定义一组词汇或一组通用操作表达式，本发明的各实施例桥接了所表达的商业意图与商业应用程序的实现和 IT 配置之间的间隙。

本发明的各替换实施例构造或建立业务流程元模型或接口，其不仅使商业用户能够在他/她的抽象处描述流程而且使信息技术 (IT) 个人或开发者能够按照相同的接口来展示实现。在本发明的又一示例中，IT 开发者可以使用由商业人士创建的业务流程的元模型来创建该业务流程的实现。或者，给定一现有的实现，本发明的各实施例可以使用该业务流程元模型来发现或展示该商业用户所需要的流程表达式的现有实现。

提供本概述以便以简化的形式介绍将在以下详细描述中进一步描述的一些概念。该概述不旨在标识所要求保护的的主题的关键特征或必要特征，也不旨在用于帮助确定所要求保护的的主题的范围。

其它特征的一部分将是显而易见的，一部分将在下文中指出。

#### 附图简述

图 1A 是示出根据本发明的实施例的用于表示业务流程的系统的示例性实施例的图示。

图 1B 是示出根据本发明的实施例的元模型的图示。

图 2A 和 2B 是示出根据本发明的实施例使用所定义的词汇或操作表达式来表示具有相关联的元数据的业务流程的框图。

图 3A 到 3D 是示出根据本发明的实施例的业务流程的不同观点的框图。

图 4 是示出根据本发明的实施例的表示业务流程的操作的示例性流程图。

图 5 是示出在其上可以存储本发明的各方面的示例性计算机可读介质的框图。

图 6 是示出可在其中实现本发明的合适的计算系统环境的一个示例的框图。

图 7 是示出根据本发明的实施例的具有与实体和动作相关联的模式的具体-动作模型的实现的图示。

相应的参考字符在全部附图中指示相应的部分。

### 详细描述

首先参考图 1A，一框图示出了根据本发明的实施例的用于表示业务流程的系统 100。系统 100 包括通用计算设备 102，其包括处理器 104 和存储器区域 106。在一个实施例中，计算设备 102 是如图 6 所示和所描述的计算机 130。系统 100 还包括用于从源收集数据或元数据的接口 118，该源可以是来自用户（例如，商业用户 108）的输入的形式或自动化过程的形式，诸如当数据被存储在存储器区域 106 或其他数据存储源中时与该数据附连或相关联的元数据。在一个实施例中，所收集的元数据被格式化为描述和定义该元数据的要求的统一模式。例如，该模式可定义该元数据应在某一字符长度内等等。在一个实施例中，使用系统 100 来实现图 1B 所描述的元模型 118 以表示从在业务流程设计阶段期间捕捉商业意图中得到的数据或元数据。

如图 1A 所示，一个或多个用户可访问系统 100 中的计算设备 102。在一个实施例中，系统 100 可以是分布式系统，其具有一个或多个服务器和客户机配置且计算设备 102 是这些服务器或客户机中的一个。例如，一个或多个商业用户  $A_{1-N}$  108 可访问计算设备 102，一个或多个商业用户  $B_{1-N}$  110 也可如此。与此同时，一个或多个开发者  $D_{1-N}$  112 负责为商业用户  $A_{1-N}$  108 和商业用户  $B_{1-N}$  110 维护、配置和设计操作。在一个示例中，商业用户  $A_{1-N}$  108 和商业用户  $B_{1-N}$  110 可以是相同商业企业的一部分或可与不同企业相关联。

现在参考图 1B，构建元模型 118 以便一方面与信息技术 (IT) 模型 120 进行通信并且另一方面与创作模型 122 和模拟模型 124 进行通信。例如，IT 模型 120 提供一个或多个 IT 实现，诸如 web 服务、目录服务、事件处理等等。创作模型 122 包括工作管理实现、软件工具配置等等。模拟模型 124（将在以下进一步描述）包括“现在 (as-is)”和“将来 (to-be)”实现。通过使用在元模型 118 中定义的一组词汇 126，本发明的各实施例帮助 IT 模型 120、创作模型 122 和模拟模型 124 以及用于在设计业务流程期间增强用户的商业意图的表示的其他软件模型、模块或组件之间的通信。

例如，通过使用词汇 126，设计 IT 模型 120 的 IT 开发者可使用相同的概



念来为其他模型定义功能或操作。与此同时，这些 IT 开发者可以发现并使用利用相同的一组词汇 126 来定义的现有的功能或操作（例如，在创作模型或模拟模型中）。类似地，创作模型 122 和模拟模型 124，以及其他服务模型 128 可以与其他软件模型、模块、或组件进行无缝通信以完整表示该用户的商业意图。

可以理解，虽然如图 1B 所示及以下进一步示出的在元模型 118 中仅定义了六个词汇术语，但是可以定义其他术语或构造以捕捉在业务流程中常用的概念而不背离本发明的范围。

图 2A 和 2B 是示出根据本发明的实施例使用所定义的词汇或操作表达式来表示具有相关联的元数据的业务流程的框图。在图 2A 中，一框图示出了商业用户的业务流程设计的概念或意图。例如，用户（例如，图 1 的用户 108）想要设计具有步骤 A302，之后是步骤 B304 和步骤 C306 的一组业务流程。

图 2B 示出了在业务流程（例如，购买订单）中定义任务的初始阶段。在 302 处，步骤 A 定义已被绑定到现有的实现的任务，导致没有关于不一致性的任何视觉线索。例如，用户  $A_1$  可以被分配来执行接收所有购买定单的任务。然而，在 304 处，步骤 B 定义还未被绑定到其实现的现有的操作的任务。由此，警告符号 308 指示这一问题。例如，定单批准任务被分配给用户  $A_1$  的主管，但是还不存在执行该任务的这一角色。在 306 处，步骤 C 表示复合任务。当用户点击‘+’符号时，将展开该任务的视觉表示以显示所包含的过程。例如，在批准定单的任务中，批准定单的个人需要检查库存或其他任务。该警告符号也指示该复合任务 306 中的某些元素还未被绑定到实现。

现在参考图 3A 和 3C，三张框图示出根据本发明的实施例的业务流程的不同的用户观点。图 3A 示出了用户 202（例如，商业用户 A108）的用户观点。在该示例中，用户 202 是对活动及其序列感兴趣的个人。换言之，用户 202 可能只对与诸如 204 中处理如购买定单 123 等购买定单等业务流程有关的任务和事件序列感兴趣。在该图示中，用户 202 将查看或访问到的是在视图 206 中所提供给用户 202 的东西：对于购买定单 123 所发生的任务列表和事件序列。例如，视图 206 描述了在 208 处购买定单 123 从诸如 ABC 公司等顾客发出。在 210 处，购买定单 123 接着由销售部门的主管接收。例如，主管的实例可以是

用户\_A<sub>1</sub>。在 212 处，购买定单 123 随后由销售部门的经理（例如，用户\_B<sub>1</sub>）批准。所示的下一个任务是在 214 处库存状态是“有库存”并且在 216 处购买定单 123 的送货状态是“未送货”。同样，视图 206 示出在 218 处购买定单 123 的发票是"\$5,000.00"。

另一方面，图 3B 示出了对于用户 202 的不同的视图，该用户可以是希望将活动链接到数据源、数据变换和数据流的人的。换言之，用户 220 希望将任务和事件绑定到实体。由此，呈现给用户 220 的购买定单 123 的视图 222 将与在图 3A 中呈现给用户 202 的视图 202 不同。例如，在该视图 222 中，对于由顾客（例如，ABC 公司）发出的相同定单，在 208 处，用户 220 可访问发出由 ABC 公司发送的定单的消息 224（例如，语音邮件消息、电子邮件（email）消息等）。对于在 210 处由主管（例如，用户\_A<sub>1</sub>）接收到的定单，视图 222 可显示该人的诸如电话号码、电子邮件帐户或其它联系信息等等主管的联系信息 226。对于在 212 处的定单批准，用户 220 除了可查看在 214 处的定单 123 的细节之外还可查看经理（例如，用户\_B<sub>1</sub>）的联系信息 228。在 216 处，视图 222 可显示送货员的数据存储以及在该送货员的数据存储所显示的状态的快照 232。在 218 处，发票可访问到其中\$5,000.00 与购买定单 123 相关联的会计部门的数据存储的链接 234。

图 3C 描述了从用户 236 的观点来看的视图，该用户可以是希望按照实体来描述实现的能力，以使应用程序能够在过程中使用，诸如展示实体和动作以供在任务序列中使用等的人。在该示例中，购买定单 123 的视图 252 可包括以下中的一个或多个，如下所述：

对于由顾客 ABC 公司发出的定单的相同的步骤 208，视图 252 可显示诸如块 238 所示的条件的一组条件。例如，用户 236 可能已经具有用于实现该步骤 208 的一组条件，诸如：

```
for 所有 ABC 公司定单
{ if 数量 > 500 单位
    销售部门的副主管来处理；
else
    任何用户都可处理； }
```

换言之，当用户 236 访问相同的购买定单 123 时，用户 236 查看更面向技术的不同的观点，而不是图 3A 或 3B 所描述的。

在 210 处，该定单由主管来接收，开发者还看见指示该主管（例如，用户 `_A1`）在或将在 2/27-3/3 期间度假的块 240。由此，用户 236 可实现将被发送到其他人员的自动响应事件以在没有主管的情况下处理定单。在 212 处，对于由经理（例如，用户 `_B1`）批准的定单，用户 236 可访问显示与以下步骤相关联的另一组限制的块 242：

```
for 所有 ABC 公司定单
{ if 数量 > 500 单位
    需要由销售部门的副主管来批准；
else
    需要由销售部门的经理来批准； }
```

在一个实施例中，用户 236 还可查看（未示出）实际购买定单 123 的细节以确定购买定单 123 是否确实具有少于 500 的单位以使得经理正确地处理该定单。否则，用户 236 可发出指示销售部门的副主管应该批准购买定单 123 的警告事件。

在 214 处，用户 236 可查看块 244 和 246 以进一步标识库存状态。在该示例中，用户 236 可将以下视作示例性库存状态验证实现的实例：

```
if ABC 公司的定单没有在 20 小时之内被批准
为每个单位支付$1 罚款；
库存状态检查：每 10 ms；
当前状态：50,000 库存；
下一次进货送货：2/22；
```

在该实施例中，用户 236 不仅可以查看与库存的条件和该特定顾客相关联的元数据，还可查看库存的当前状态以使得用户 236 在库存很少或下一次送货日期还未被标识时实现警告事件。

在 216 处，用户 236 查看显示以下送货状态的条件的块 248：

```
for ABC 公司定单
{ if 发票 > $10,000
```

使用当夜快递；

else

使用陆地送货； }

在 218 处，视图 252 可向开发者展示显示对于用户 236 的发票的另一个条件的又一个实例的块 250：

for 所有 ABC 公司定单

{ if 发票 > \$500

在 5 个工作日内发送发票；

else

在 30 日内发送发票； }

可以理解，图 3A 到 3C 所描述的步骤或条件/限制只是出于说明的目的并且因此并不限制本发明的范围。可包含可与业务流程相关联的其他特性、序列、描述、参数、条件等而不背离本发明的范围。

由此，取决于用户（例如，用户 202、用户 220、或用户 236）的观点，本发明的各实施例在适应不同用户的需求的同时一致地表示业务流程元模型。此外，用户 202、用户 220、或用户 236 可以在每个视图或观点中互换使用以使得每个用户都能够访问不同的视图或观点，倘若该用户具有访问或修改与该业务流程相关联的元数据的权限/许可。

在实现诸如发出购买定单等业务流程的这一普遍存在的表示时，本发明的各实施例定义公共词汇 114 或一组操作表达式，面向商业的人士和面向 IT 的个人可以使用该词汇或表达式来以一致的方式推理和表达业务流程。此外，通过该公共词汇或操作表达式，商业人士和 IT 个人可保存、展示、和注释对于其他组或观点有意义的信息。

现有的业务工作或面向过程的软件应用程序经常被用来管理不同实体以及组织角色（例如，人员或个人）如何与诸如在工作流程等中的各种任务交互。例如，典型的一系列购买定单履行工作流程可能发生：

1. 定单履行员工可负责从顾客接收定单；
2. 顾客区域的经理可负责批准定单；
3. 库存经理可负责标识所定购的产品的可用性；以及

#### 4. 送货部门经理处理定单的送货。

在许多实例中,诸如经理等相同角色中的任何个人都可执行该任务。然而,业务工作流程软件应用程序需要解决特定角色的特定个人必须执行特定任务的情况。例如,诸如"本.顾客.区域.经理"等表达式需要有权利担当对应于特定工作流程实例的角色的个人的标识。

与现有的实践或现有的业务流程应用程序不同,本发明的各实施例通过引入公共词汇或操作表达式来设计商业用户和 IT 开发者之间的与业务流程元模型的协作接口,而不是在业务流程模型的不同抽象层之间翻译。这是协作接口是因为它使从各种观点来看的用户能够用一致的,但仍灵活的接口来表达业务流程以使得用户可以完整地表达所需操作而不丢失技术实现的细节。由此,本发明的各方面描述了生产环境或接口,其中商业人士能够创建并改变流程模型以尝试某些事情而无需来自 IT 开发者的持续监督或协助以实现流程应用程序的改变。

本发明的各实施例定义了公共词汇或操作表达式,包括:

1. 事件:在任何过程或服务中的企业可对其做出响应的有意义的状态改变,其中事件承载“净荷”以按照实体来表达状态和/或改变。例如,事件可以是传入的购买定单请求、警告异常等等。

2. 实体:具有商业意义的一类信息;实体具有引用,其是通过 ID 或某些属性值来指向实体的实例的唯一方式。例如,实体可以是顾客 ID,其包括诸如顾客的姓名和联系信息等参数。

3. 动作:将要对实体实行或执行的操作。例如,由实体提供的诸如创建、读取、更新、删除(CRUD)或其他动作等操作。其他动作采用对实体的引用作为输入,但是没有对所引用的实体导致任何 CRUD 效果。

4. 任务:可分配给人或系统的具有明确定义的开始和结束状态的工作的逻辑单元。任务可被投影为实体。任务具有可包含对多个实体的引用的数据上下文(附加细节见表1)。例如,任务可以是批准购买定单。

5. 规则:通过参考在所述业务逻辑或流程的范围之内的一个或多个实体的状态来评估的商业决定的声明性表达式。规则可被应用于诸如事件、动作、角色以及任务等多个过程产物(artifact)。

6. 角色：能够执行哪些任务或采取哪些行动的个人。例如，角色定义包括对诸如人员与商业实体之间的关系以及 workflows 系统中的实体本身等信息的引用。可被表达成“本.顾客.区域.经理”的示例“该任务必须由该顾客所在区域的经理来批准”示出角色解析需要 workflow 所知的实体（即，“该顾客”）以及只有人员和商业系统知道的关系（即，该顾客居住在什么区域，并且谁是区域经理）。

作为另一示例，表 1 还描述了某些定义的词汇或操作表达式及其示例性操作：

词汇/操作表达式	操作	相关联的词汇	示例
实体	分解； 展示	实体；动 作	实体分解实体；实体 展示动作
事件	消费； 启动	模式；过 程	事件消费模式；事件 启动过程
动作	返回； 许可	实体；角 色	动作返回实体；动作 定义角色许可
任务	被分配 给；范围	角色；数 据上下文；动 作	任务被分配给角色； 任务在数据上下文的范 围之内
规则	应用	任务；数 据上下文	规则应用于任务（或 workflow）、动作、事件和 角色；规则应用于数据上 下文
角色	执行； 许可	任务；活 动	角色执行任务；角色 被准许执行特定活动
*活动	指向	实体	活动指向实体；活动 使用模式
模式	产生	事件	模式产生事件；模式 被展示为活动

表 1: 元数据接口中的定义的词汇或操作表达式以及交互; \*活动与商业活动监视相关联。

或者, 本发明的各实施例发现已经对用户可用的现有操作。例如, 为了得到关于给定顾客的顾客状态, 处理器 104 能够在操作库 116 中容易地确定在该操作库 116 中诸如‘得到顾客状态’等动作是否对实体‘顾客’可用。

在又一个实施例中, 当没有现有的操作可被标识为满足用户的需求时, 可能是不具有技术知识的商业用户的用户将由本发明的各实施例通过公共词汇或操作表达式来引导以便为 IT 开发者提供结构化的要求以设计出满足业务流程的需求的所请求的操作。尽管还未提供所请求的所述操作, 但是该商业用户可在商业层上继续并完成该流程设计。

如果开发者在不影响任何其他接口的情况下难以满足所请求的操作的需求, 则该开发者可以在其抽象层上的过程表示中标记其他设计组件上的改变对商业用户的影响。

在又一替换实施例中, 商业用户可以只通过利用现有的操作来修改或扩展业务流程设计。由此, 业务流程可在无需进一步的 IT 实现工作的情况下执行。例如, 处理器 104 可从用户 (例如, 图 1 的商业用户  $A_1$ ) 接收用于执行已经存储在操作库 116 中的操作的输入。

可以理解, 可以实现各种编程语言、例程、代码、或应用程序组件以构造业务流程元模型而不背离本发明的范围。例如, 在实现词汇或操作表达式“实体”时, 人们可以使用以 C#语言编写的应用程序组件来处理诸如用户帐户、任务等处理实体的所有执行。然而, 本发明的各实施例向用户提供与用户帐户或任务相关联的元数据, 以及与其相关联的数据或操作以使得具有不同观点的用户可以一致地与实体或数据交互。在又一替换实施例中, 可扩展词汇或操作表达式以基于顾客的组织方法需求来添加新行为或表达式。例如, 遵守如本发明的各实施例所描述的元模型模式的新构造可被添加到该元模型。同样, 可以发现现有的元数据, 只要其遵守本发明的各实施例所指示或描述的元模型模式。

图 3D 是示出诸如图 2A 所示的业务流程等业务流程的三种不同视图的框

图。具体地，图 3D 示出业务流程的运行时图像，其包括三个步骤：步骤 A、B、和 C。在该实施例中，所收集的元数据与该业务流程中所包括的多个活动序列相关。

例如，在作为聚合视图的视图 310 中显示该业务流程的当前状态。例如，从所收集的元数据来看，步骤 A 中有 13 个活动或实体的实例，而步骤 B 中有 4 个实例。在分离的视图 312（例如，由诸如双击等来自用户的输入来触发的深入视图）中显示如“在步骤 B 处的实例列表”所描绘的这四个运行实例的列表。在这一视图中，用户可通过选择实例中的一个来与所显示的元数据交互以查看进一步的细节。在视图 314 中，用户获得购买定单#324 的“实例视图”。该视图 314 在视觉上突出显示关于该特定实例的信息，诸如该实例所在的当前任务（此处：“步骤 B”）、任何先前、当前任务的开始时间和任何后续任务（例如，以虚线被示为还未完成的步骤 C）等等。

由此，本发明的各实施例在该流程的实例前进通过各种里程碑（图 3D 中被表示为“M1”、“M2”、或 “M3”）时将元数据与该序列中的活动相关。相关的元数据包括时间戳（例如，在 M1 处的 8:07 am、持续时间（例如，4 小时 1 分钟）等等）。

在一替换实施例中，可以生成显示业务流程的状态的报告。在另一个实施例中，处理器 104 可以在给定一组收集的元数据的情况下模拟业务流程的进度。例如，模拟的流程将确认实际业务流程将在给定所收集的元数据的情况下提供预期的结果。例如，模拟可通过“现在”过程以及“将来”过程来实现。“现在”过程根据对数据流向哪儿、在该数据上发生什么动作和该过程的最终结果是什么的理解来模拟结果。进行这一模拟多数是为了检查以保证该过程正在做所期望做的事情（满足 SLA、KPI 目标等）。模拟“将来”过程涉及改变过程中的步骤或过程中的数据以检查不同的最终结果。

图 4 是示出根据本发明的实施例的用于表示业务流程的示例性方法的流程图。在一个示例中，诸如图 5 中的计算机可读介质 502 等一个或多个计算机可读介质包括实现图 4 所示的方法的计算机可执行组件。

例如，在 402 处，词汇组件 504 可定义用于标识业务流程的操作表达式。业务流程包括多个活动序列。在 404 处，数据组件 506 收集与所定义的操作表



达式相关联的元数据。在一个实施例中，在 406 处根据一模式（未示出）来格式化所收集的元数据。在一个示例中，该模式包括描述包括实体、实体视图、动作信息等整个业务流程的可扩展标记语言（XML）格式的元数据。在这一示例中，操作表达式“动作“可使用如下的 XML 模式来描述：

1. ActionName(Type="Get", EntityReference)
2. ActionName(Type="Put", EntityReference, PutParameters)
3. ActionName(Type="Act", EntityReference, ActParameters)

图 7 也描述了示出根据本发明的实施例的基于元模型的模式 700 的实体-动作模型的示例性实现的示意图。

在 408 处，接口组件 508 向用户提供所收集的元数据。在另一个实施例中，词汇组件 504 描述业务流程以使得接口组件 508 能够向用户（例如，商业用户 A 108 或商业用户 B 110）提供所收集的元数据以使得该用户可操控、操作或访问与该业务流程相关联的元数据。在一个实施例中，报告组件 510 向用户报告对所描述的业务流程的分析。例如，报告组件 510 可报告业务流程的进度。

在一替换实施例中，逻辑组件 512 评估与所收集的元数据相关联的一组规则以使得该用户能够与该业务流程交互。在又一实施例中，模拟组件 514 响应于由接口组件接收到的输入基于操作表达式来模拟多个活动序列的进度。在再一实施例中，发现组件 516 发现与业务流程相关联的一个或多个现有的配置或操作。

图 6 示出了计算机 130 形式的通用计算设备的一个示例。在本发明的一个实施例中，诸如计算机 130 等计算机适于在此所示和描述的其它附图中使用。计算机 130 具有一个或多个处理器或处理单元 132 和系统存储器 134。在所示的实施例中，系统总线 136 将包括系统存储器 134 在内的各种系统组件耦合至处理器 132。总线 136 代表任何若干类型的总线结构中的一个或多个，包括存储器总线或存储器控制器、外围总线、加速图形端口以及使用任何各种总线体系结构中的任一种的处理器或局部总线。作为示例，而非限制，这样的体系结构包括工业标准体系结构（ISA）总线、微通道体系结构（MCA）总线、增强型 ISA（EISA）总线、视频电子技术标准协会（VESA）局部总线和外围部件互连（PCI）总线（也称为 Mezzanine 总线）。

计算机 130 一般具有至少某种形式的计算机可读介质。计算机可读介质，包括易失性与非易失性介质、可移动与不可移动介质，可以是任何可由计算机 130 访问的可用介质。作为示例而非限制，计算机可读介质包括计算机存储介质和通信介质。计算机存储介质包括以用于存储诸如计算机可读指令、数据结构、程序模块或其它数据的信息的任何方法或技术实现的易失性与非易失性、可移动与不可移动介质。例如，计算机存储介质包括 RAM、ROM、EEPROM、闪存或其它存储器技术、CD-ROM、数字多功能盘（DVD）或其它光盘存储、磁带盒、磁带、磁盘存储或其他磁存储设备、或可以用于存储所需信息并可由计算机访问的任何其它介质。通信介质一般以诸如载波或其它传输机制等已调制数据信号来体现计算机可读指令、数据结构、程序模块或其它数据，并且包括任何信息传递介质。本领域技术人员熟悉已调制数据信号，其一个或多个特征以将信息编码在该信号中的方式来设置与改变。诸如有线网络或直接线连接等有线介质，以及如声学、RF、红外线及其它无线介质等无线介质都是通信介质的示例。以上的任一种的组合也包括在计算机可读介质的范围之内。

系统存储器 134 包括可移动和/或不可移动、易失性和/或非易失性存储器形式的计算机存储介质。在所示的实施例中，系统存储器 134 包括只读存储器（ROM）138 和随机存取存储器（RAM）140。基本输入/输出系统 142（BIOS）包含有助于诸如启动时在计算机 130 中元件之间传递信息的基本例程，它通常被存储在 ROM 138 中。RAM 140 通常包含处理单元 132 可以立即访问和/或目前正在其上操作的数据和/或程序模块。作为示例而非局限，图 6 示出了操作系统 144、应用程序 146、其它程序模块 148 和程序数据 150。

计算机 130 也可以包括其它可移动/不可移动、易失性/非易失性计算机存储介质。例如，图 6 示出读写不可移动非易失性磁介质的硬盘驱动器 154。图 6 还示出读写可移动非易失性磁盘 158 的磁盘驱动器 156，以及读写如 CD-ROM 或其它光学介质等可移动非易失性光盘 162 的光盘驱动器 160。可以在示例性操作环境中使用的其它可移动/不可移动、易失性/非易失性计算机存储介质包括，但不限于，盒式磁带、闪存卡、数字多功能盘、数字录像带、固态 RAM、固态 ROM 等等。硬盘驱动器 154 和磁盘驱动器 156 以及光盘驱动器 160 一般通过如接口 166 等非易失性存储器接口连接到系统总线 136。

上面所述且在图 6 中所示的驱动器或其它大容量存储设备及其相关联的计算机存储介质为计算机 130 提供了对计算机可读指令、数据结构、程序模块和其它数据的存储。在图 6 中,例如,硬盘驱动器 154 被示为存储操作系统 170、应用程序 172、其它程序模块 174 和程序数据 176。注意,这些组件可以与操作系统 144、应用程序 146、其它程序模块 148 和程序数据 150 相同或不同。操作系统 170、应用程序 172、其它程序模块 174 和程序数据 176 在这里被标注了不同的标号是为了说明至少它们是不同的副本。

用户可通过如键盘 180 和定点设备 182 (例如鼠标、跟踪球、笔或触摸垫) 等输入设备或用户界面选择设备来向计算机 130 输入命令和信息。其它输入设备 (未示出) 可以包括话筒、操纵杆、游戏手柄、圆盘式卫星天线、扫描仪等。这些和其它输入设备通过耦合到系统总线 136 的用户输入接口 184 连接到处理单元 132, 但可通过其它接口和总线结构, 如并行端口、游戏端口或通用串行总线 (USB) 来连接。显示器 188 或其它类型的显示设备也经由接口, 如视频接口 190 连接到系统总线 136。除了显示器 188 以外, 计算机通常包括其它外围输出设备 (未示出), 如打印机和扬声器, 它们可通过输入外围接口 (未示出) 来连接。

计算机 130 可使用至一个或多个远程计算机, 如远程计算机 194 的逻辑连接在网络化环境中操作。远程计算机 194 可以是个人计算机、服务器、路由器、网络 PC、对等设备或其它常见的网络节点, 并且一般包括上面关于计算机 130 所述的许多或全部元件。图 6 中所示的逻辑连接包括局域网 (LAN) 196 和广域网 (WAN) 198, 但也可以包括其它网络。LAN 136 和/或 WAN 138 可以有网络、无线网络、它们的组合等等。这样的联网环境常见于办公室、企业范围计算机网络、内联网和全球计算机网络 (例如因特网)。

当在局域网环境中使用时, 计算机 130 通过网络接口或适配器 186 连接到 LAN 196。当在广域网络环境中使用时, 计算机 130 一般包括调制解调器 178 或通过如因特网等 WAN 198 建立通信的其它装置。调制解调器 178 可以是内置或外置的, 它经由用户输入接口 184 或其它合适的机制连接到系统总线 136。在网络化环境中, 相对于计算机 130 所描绘的程序模块或其部分可被存储在远程存储器存储设备 (未示出) 中。作为示例而非限制, 图 6 示出远程应用程序

192 驻留在存储器设备上。所示的网络连接是示例性的，并且可以使用在计算机之间建立通信链路的其他手段。

通常，计算机 130 的数据处理器通过在不同时间存储在计算机的各种计算机可读存储介质中的指令来编程。例如，程序和操作系统通常在软盘或 CD-ROM 中分发。从那里，它们被安装或加载到计算机的二级存储器中。在执行时，它们至少部分地被加载到计算机的主电子存储器中。当这些和其它各种类型的计算机可读存储介质包含用于实现下面结合微处理器或其它数据处理器描述的步骤的指令或程序时，在此所述的本发明的各方面包括这样的存储介质。此外，当计算机根据此处所述的方法和技术来编程时，本发明的各方面包括计算机本身。

出于说明的目的，程序和如操作系统等其它可执行程序组件在此被示为离散的框。然而，可以认识到，这样的程序和组件在不同时间驻留在计算机的不同存储组件中，并由计算机的数据处理器执行。

尽管结合包括计算机 130 的示例性计算系统环境进行了描述，但本发明的实施例可用于众多其它通用或专用计算系统环境或配置。计算系统环境不旨在对本发明的任何方面的使用范围或功能提出任何限制。而且，计算系统环境不应被解释为对在示例性操作环境中所示组件的任何一个或组合有任何依赖性 or 要求。适用于本发明各方面的公知的计算系统、环境和/或配置的示例包括，但不限于：个人计算机、服务器计算机、手持式或膝上型设备、多处理器系统、基于微处理器的系统、机顶盒、可编程消费电子产品、移动电话、网络 PC、小型机、大型机、包括上述系统或设备中的任一个的分布式计算机环境等。

本发明的各实施例可以在由一个或多个计算机或其他设备执行的诸如程序模块等计算机可执行指令的通用上下文中描述。一般而言，程序模块包括但不限于，执行特定的任务或实现特定的抽象数据类型的例程、程序、对象、组件和数据结构。本发明的各方面也可以在其中任务由通过通信网络链接的远程处理设备执行的分布式计算环境中实现。在分布式计算环境中，程序模块可以位于包括存储器存储设备在内的本地和远程计算机存储介质中。

软件体系结构的上下文中的接口包括软件模块、组件、代码部分、或其它计算机可执行指令的序列。该接口包括例如第一模块，该模块访问代表第一模

块执行计算任务的第二模块。在一个示例中，该第一和第二模块包括诸如由操作系统提供的应用程序编程接口（API）、组件对象模型（COM）接口（例如，用于对等应用程序通信）和可扩展标记语言元数据交换格式（XMI）接口（例如，用于 web 服务之间的通信）。

接口可以是紧耦合的同步实现，诸如在 Java 2 平台企业版（J2EE）、COM、或分布式 COM（DCOM）的示例中。另选地或另外地，接口可以是松耦合的异步实现，诸如在 web 服务中（例如，使用简单对象访问协议）。一般而言，接口包括以下特性的任何组合：紧耦合、松耦合、同步、和异步。此外，接口可遵循标准协议、专有协议、或标准和专有协议的任何组合。

此处所描述的接口都可以是单个接口的一部分，或可被实现为独立的接口或其中的任何组合。接口可以本地或远程地执行以提供功能。此外，接口可包括比此处所示出或描述的更多或更少的功能。

在操作中，计算机 130 执行诸如各附图（例如，图 4）所示的计算机可执行指令以实现本发明的各方面。

除非另有指定，否则此处所示和所述的本发明各实施例的操作的执行或进行的次序不是必需的。即，除非另有指明，否则各操作可按照任何次序执行，且本发明的实施例可以包括比本文所公开的更多或更少的操作。例如，构想了在另一操作之前、同时或之后执行或进行某一操作是在本发明各方面的范围内的。

本发明的各实施例可以用计算机可执行指令来实现。计算机可执行指令可以被组织为一个或多个计算机可执行组件或模块。本发明的各方面可以用任何数量的这些组件或模块及其任何组织来实现。例如，本发明的各方面不限于在各附图和本文中示出的特定的计算机可执行指令或者特定的组件或模块。本发明的其他实施例可以包括具有比在本文中示出和描述的更多或更少的功能的不同计算机可执行指令或组件。

当介绍本发明或其实施例的各方面的各元素时，冠词“一”、“一个”、“该”和“所述”指的是存在该元素的一个或多个。术语“包括”、“包含”、“具有”旨在是包括性的并且指的是可以有除所列元素之外的其它元素。

在不背离本发明各方面的范围的情况下，可对以上构造、产品和方法进行

---

各种改变，以上描述中所包含的以及在附图中所示出的所有一切旨在应被解释为说明性并且没有限制意义。

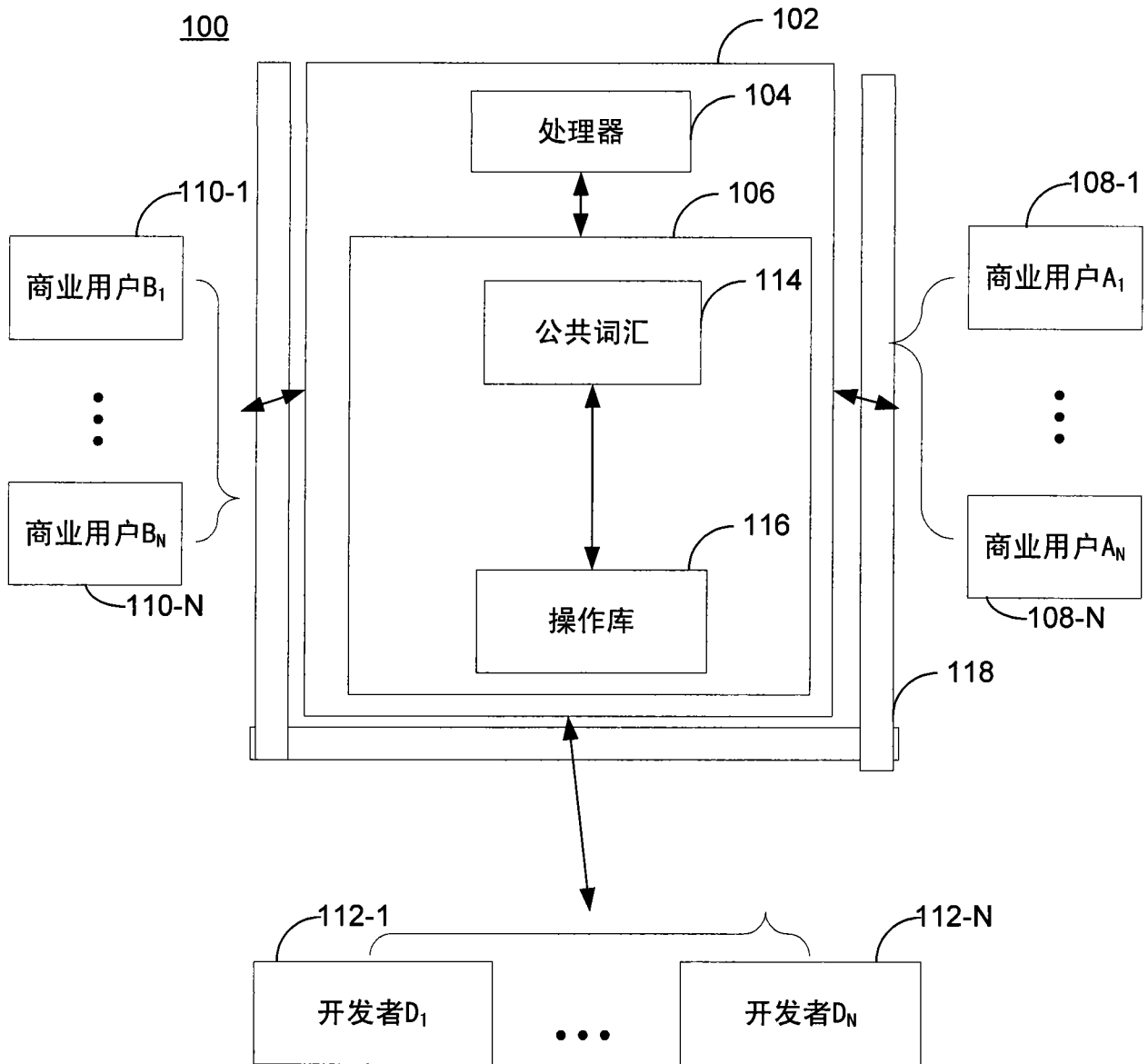


图 1A

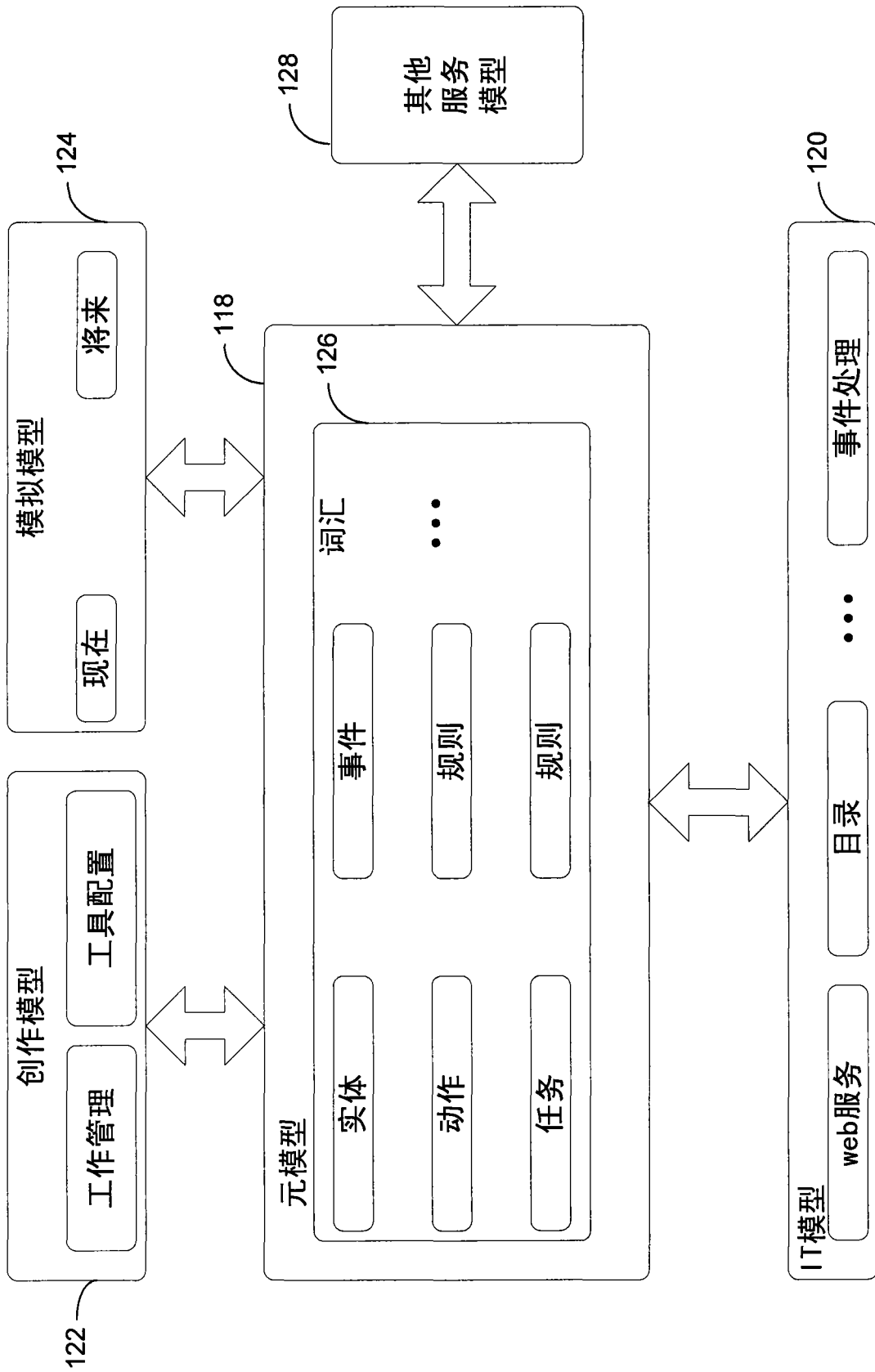


图 1B



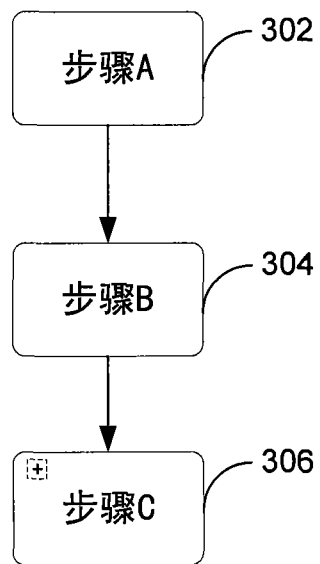


图 2A

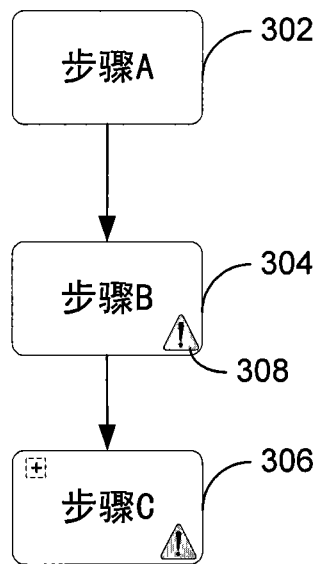


图 2B

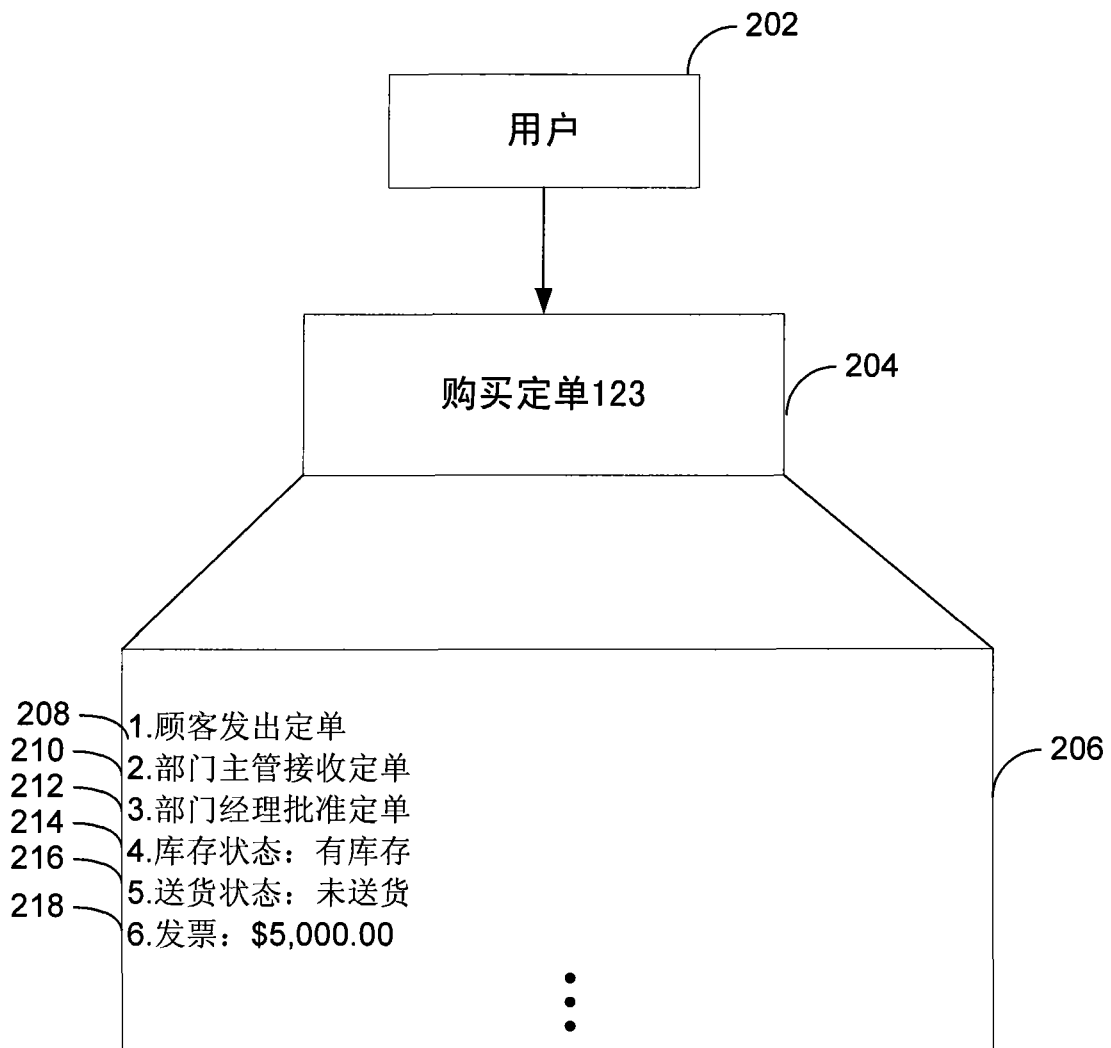


图 3A

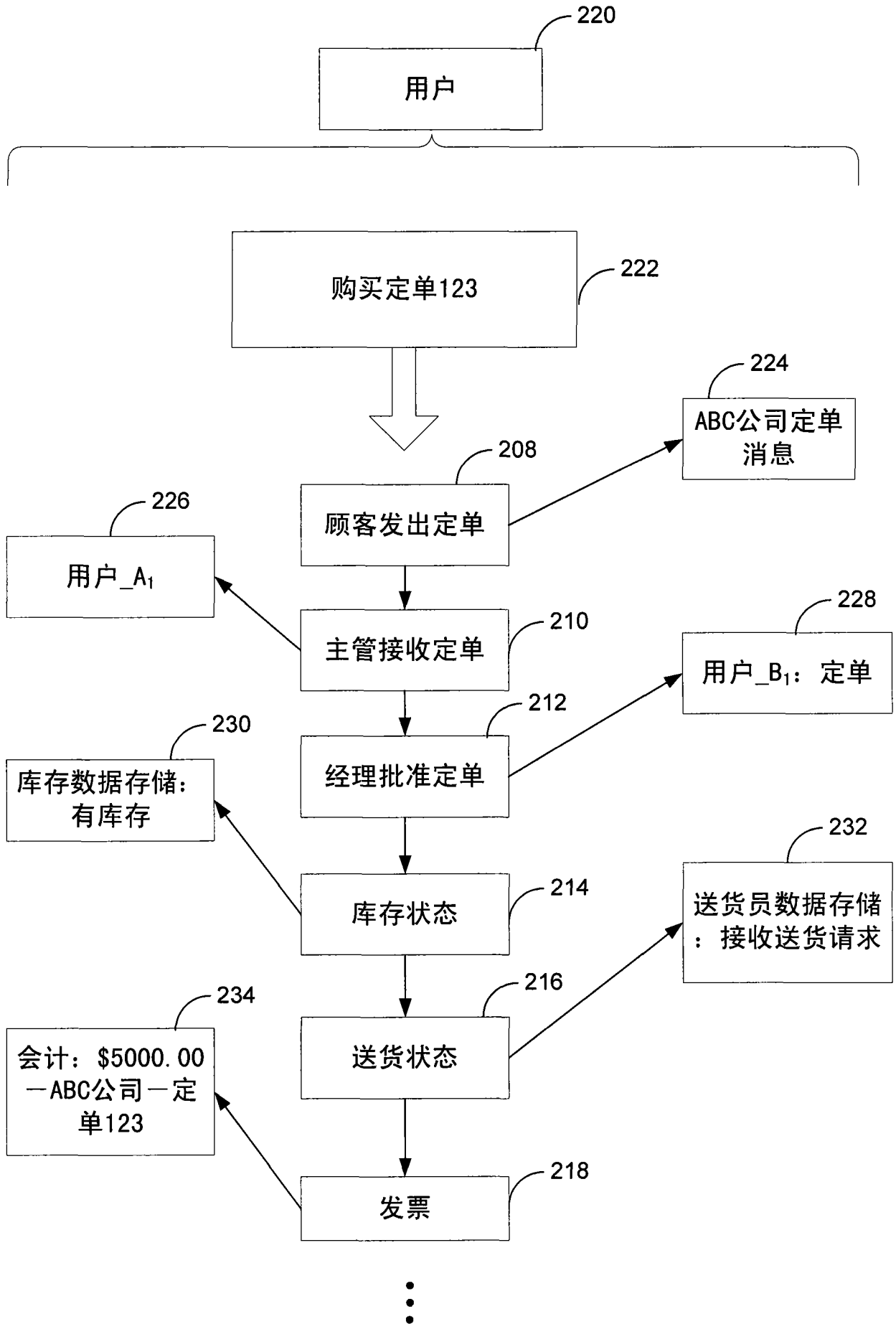
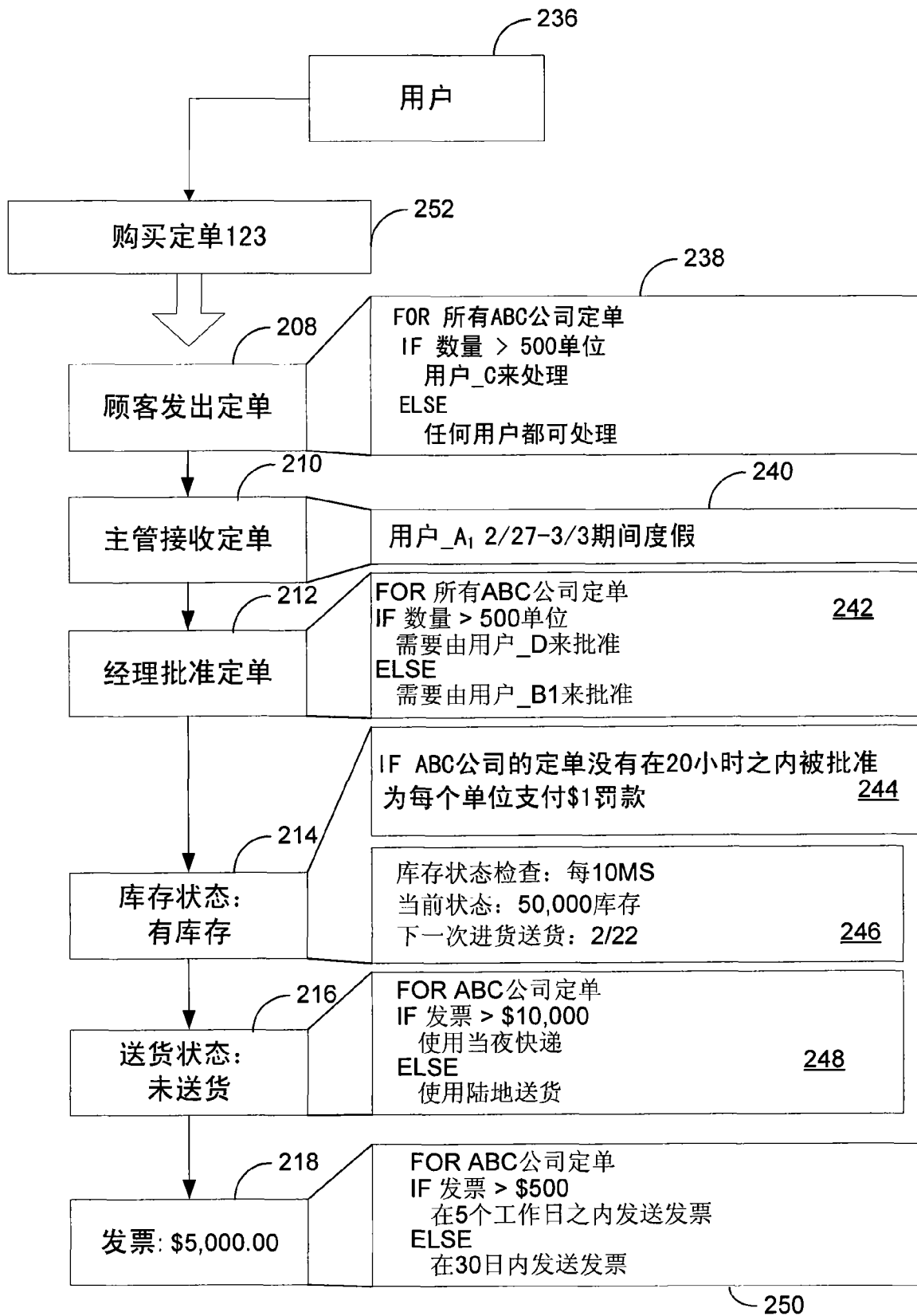


图 3B



⋮

图 30

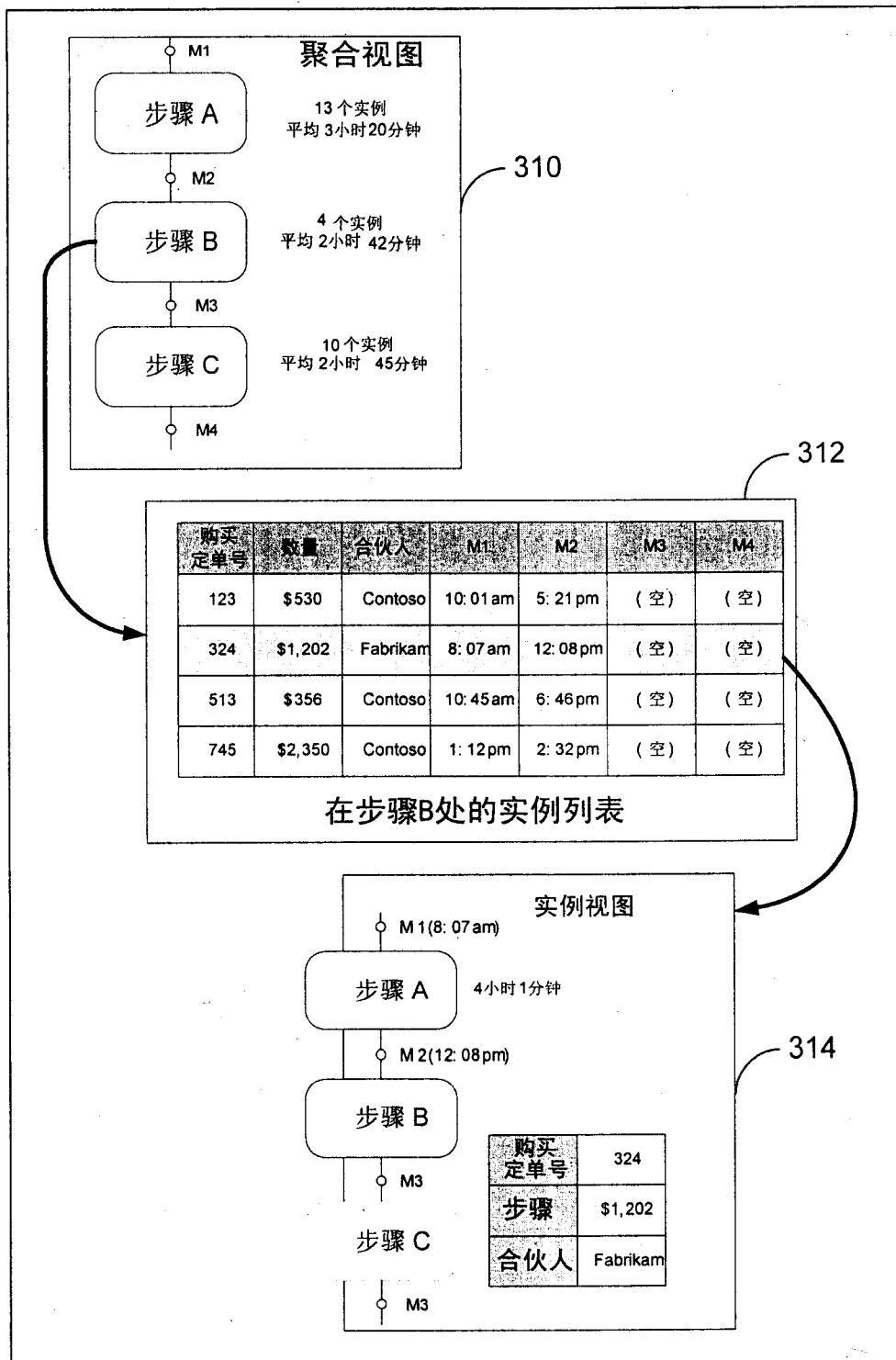


图 3D

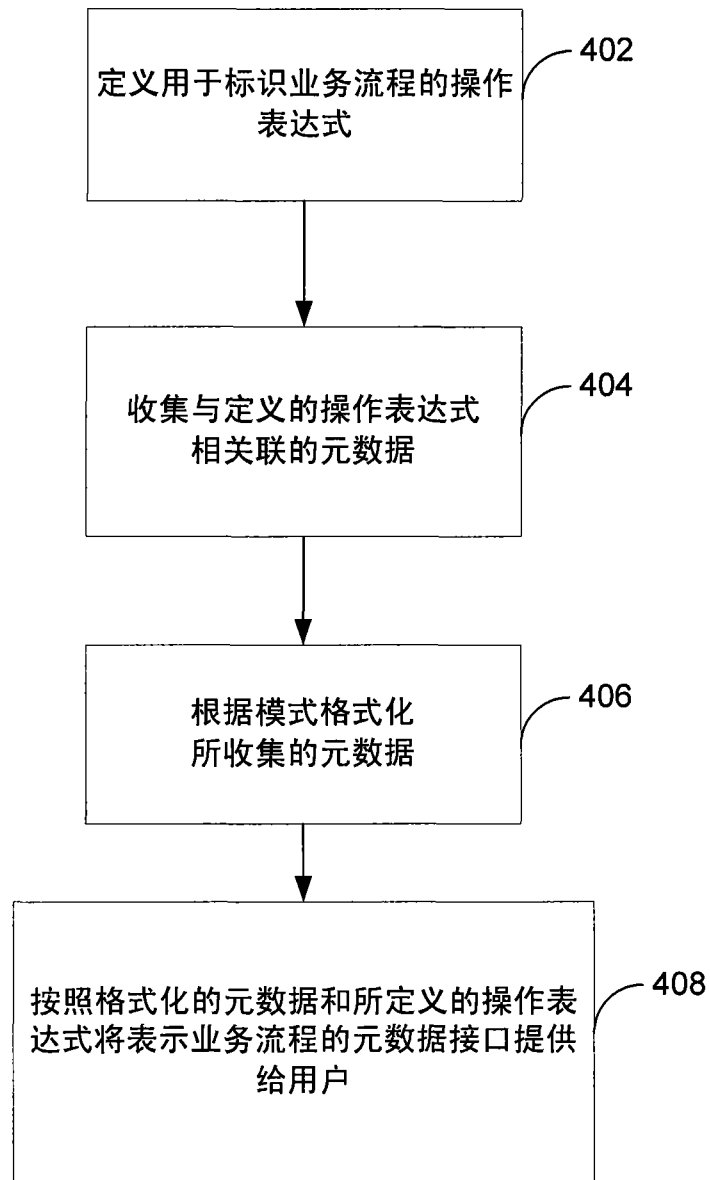


图 4

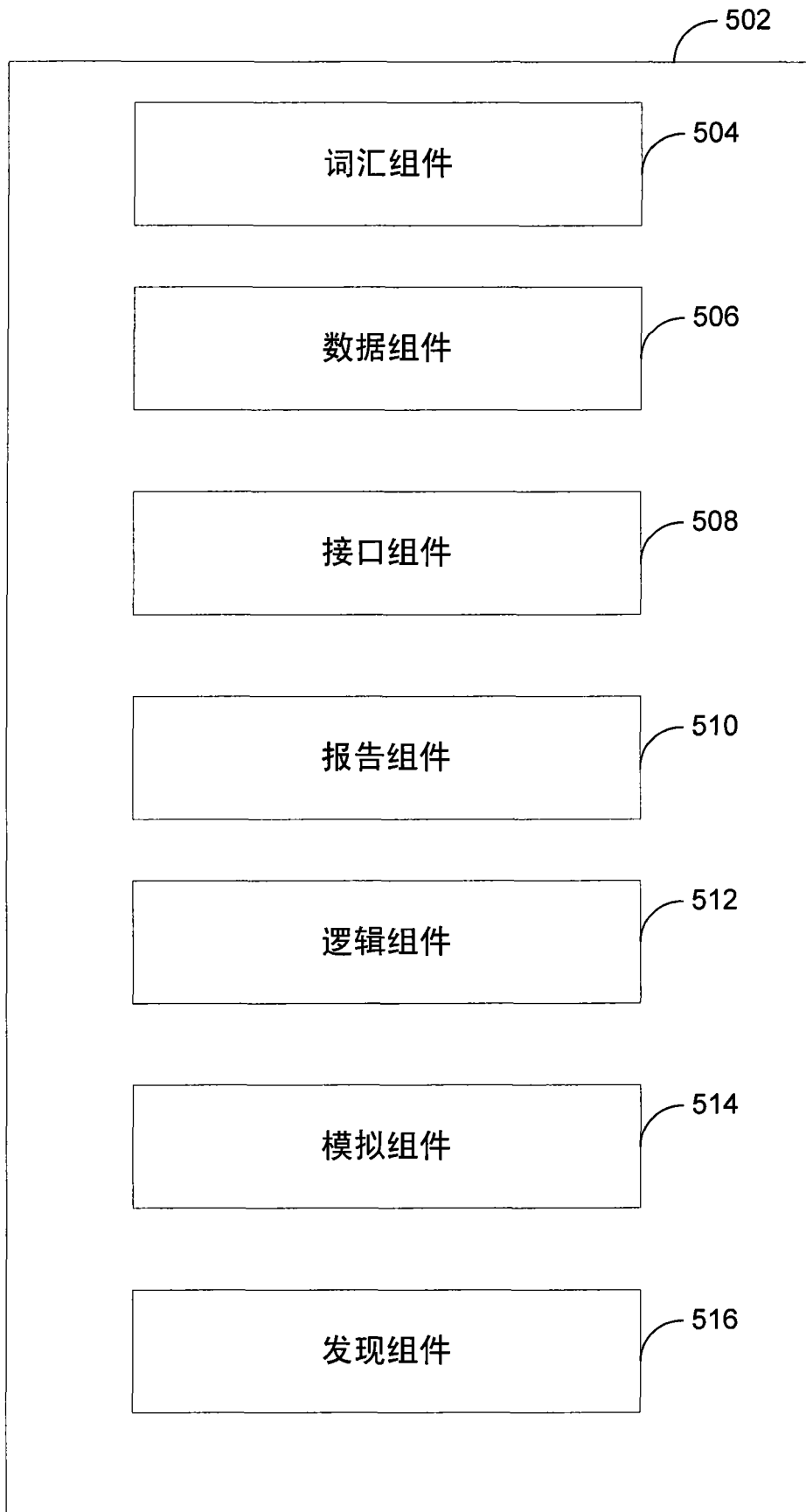


图 5



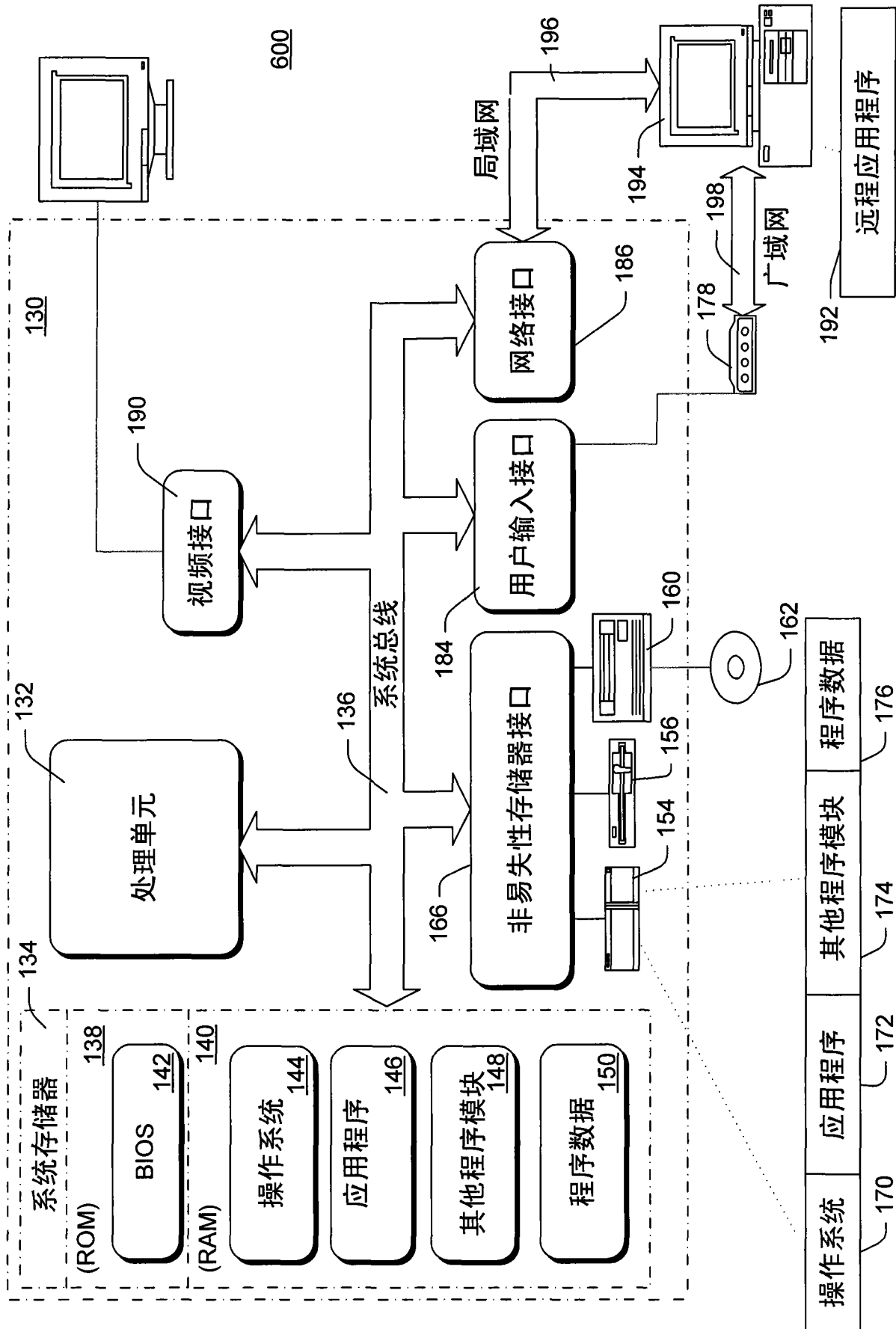
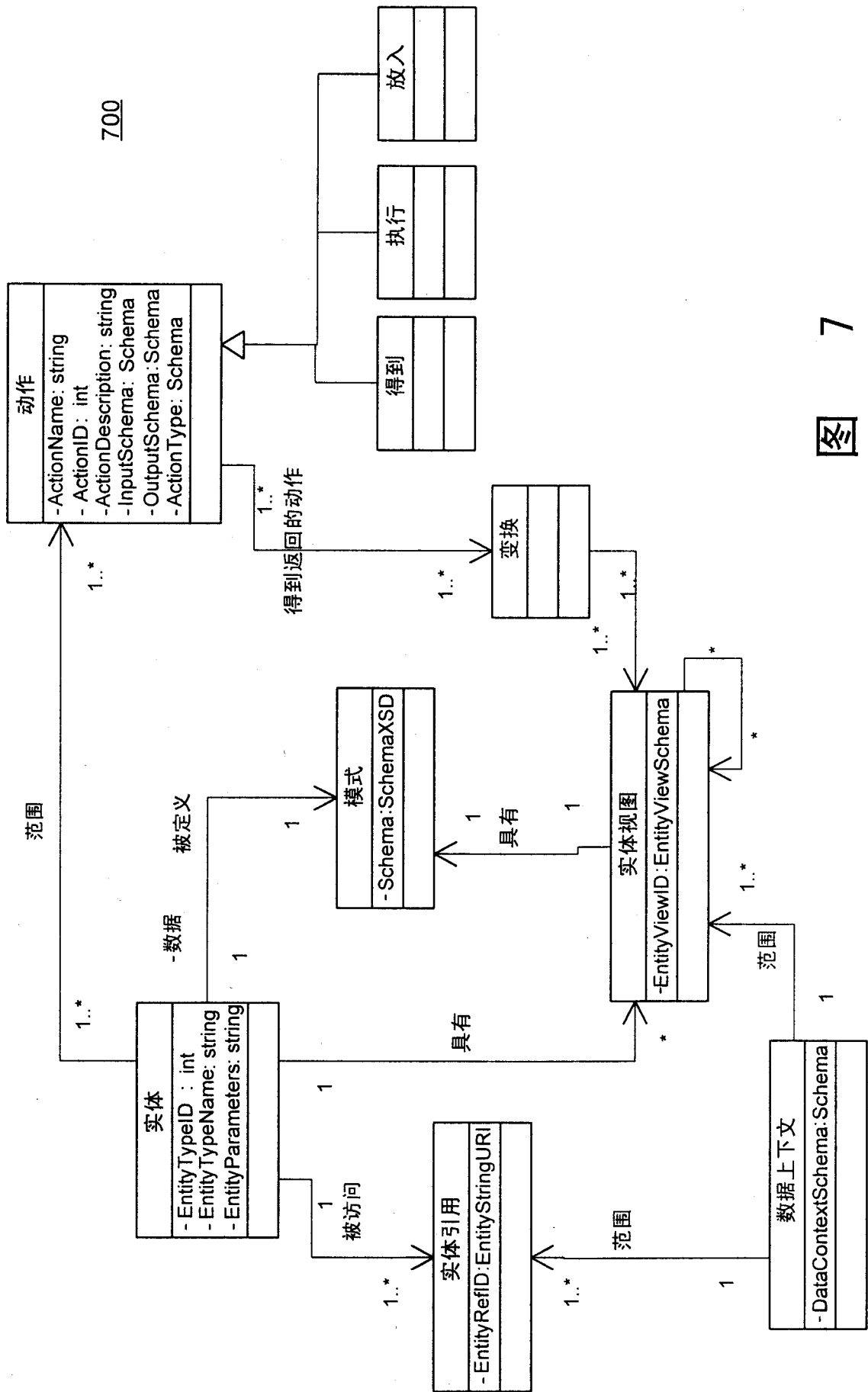


图 6



700

图 7