



(12) 发明专利

(10) 授权公告号 CN 107077477 B

(45) 授权公告日 2021.01.01

(21) 申请号 201580049531.2

(22) 申请日 2015.09.14

(65) 同一申请的已公布的文献号  
申请公布号 CN 107077477 A

(43) 申请公布日 2017.08.18

(30) 优先权数据  
14/486,638 2014.09.15 US

(85) PCT国际申请进入国家阶段日  
2017.03.14

(86) PCT国际申请的申请数据  
PCT/US2015/049876 2015.09.14

(87) PCT国际申请的公布数据  
W02016/044114 EN 2016.03.24

(73) 专利权人 微软技术许可有限责任公司  
地址 美国华盛顿州

(72) 发明人 L·诺维克 O·纳诺  
I·J·G·多斯桑托斯 L·巴萨德  
A·莱兹曼 D·巴尼克

(74) 专利代理机构 北京市金杜律师事务所  
11256

代理人 王茂华

(51) Int.Cl.  
G06F 16/2458 (2019.01)

(56) 对比文件  
CN 103036736 A, 2013.04.10  
CN 102597992 A, 2012.07.18  
WO 2014113273 A2, 2014.07.24  
Mohamed Ali et al. "the extensibility framework in microsoft StreamInsight".  
《2011 IEEE 27th International Conference on Data Engineering》. 2011,  
torsten grabs et al. "Introducing Microsoft StreamInsight".  
《URL: http://www.microsoft.com/sqlserver/2008/en/us/R2-complex-event.aspx》. 2009,

曲家朋. "实时数据流动态模式发现与跟踪方法".  
《中国优秀硕士学位论文全文数据库 信息科技辑》. 2011,

审查员 张思洋

权利要求书3页 说明书17页 附图4页

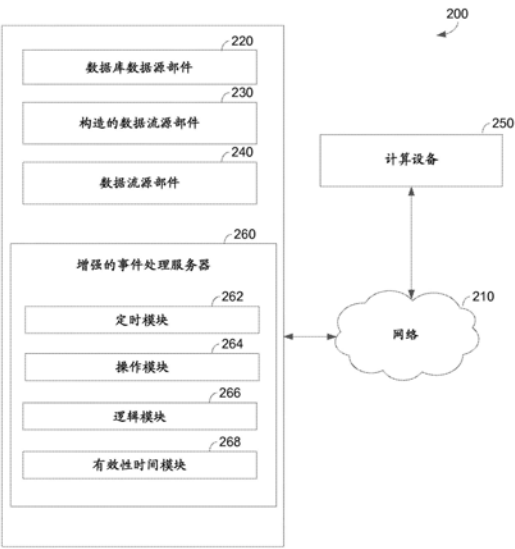
(54) 发明名称

用于增强的事件处理的构造的数据流

(57) 摘要

本文中描述的实施例涉及用于增强的事件处理的方法和系统。增强的事件处理请求被接收，增强的处理请求包括来自数据库数据源的输入数据的指示。在增强的事件处理请求中指示的输入数据使用构造的数据流而被访问。构造的数据流提供对数据库数据源的数据库数据随时间的快照的集合的访问。来自构造的数据流源的数据可以根据所选择的值和所选择的时间而被访问。构造的数据流数据包括数据库数据不可获得的时间元素，该时间元素有助于使用构造的数据流数据为数据库数据执行增强的事件处理。针对增强的事件处理请求的输出基于使用构造的数

据流数据访问的输入数据而被生成。



1. 一种用于增强的事件处理的计算机实现的方法,所述方法包括:

接收增强的事件处理请求,其中所述增强的事件处理请求包括来自数据库数据源的输入数据的指示;

使用构造的数据流来访问所述输入数据的指示中的所述输入数据,其中所述构造的数据流提供对所述数据库数据源的数据库数据的随时间的快照的集合的访问,其中使用构造的数据流来访问所述输入数据,所述构造的数据流操作为用于所述数据库数据源的数据流模拟代理,所述构造的数据流从所述快照的集合被构造,使得所述构造的数据流包括时间元素,所述时间元素指示对应于所述构造的数据流的事件的时间;以及

基于使用所述构造的数据流被访问的所述输入数据来生成针对所述增强的事件处理请求的输出。

2. 根据权利要求1所述的方法,其中来自所述数据库数据源的所述输入数据不与针对所述输入数据的时间元素相关联,所述数据库数据源在关系数据库表中维持数据库数据。

3. 根据权利要求1所述的方法,其中所述增强的事件处理请求中的所述输入数据对应于所述快照的集合中经由所述构造的数据流访问的选定快照中的选定值和选定时间。

4. 根据权利要求1所述的方法,其中所述构造的数据流的所述时间元素从所述构造的数据流的事件发生并且作为所述数据库数据中的所述事件的快照的一部分被存储的所述时间被确定。

5. 根据权利要求1所述的方法,其中所述构造的数据流使用随时间改变的所述数据库数据来提供可重复计算,并且其中所述构造的数据流根据所选择的值和所选择的时间而被访问。

6. 根据权利要求1所述的方法,其中所述增强的事件处理请求还包括来自数据流的输入数据的第二指示。

7. 根据权利要求6所述的方法,还包括:

使所述构造的数据流的第一时间元素与所述数据流的第二时间元素相关;以及

基于所述构造的数据流的所述输入数据以及所述数据流的所述输入数据来生成针对所述增强的事件处理请求的输出。

8. 根据权利要求1所述的方法,其中所述输出与有效性时间元素相关联,所述有效性时间元素指示基于所述构造的数据流的所述输出准确地反映所述数据库数据中的数据的所述时间。

9. 一种或多种计算机存储介质,其上实施有计算机可执行指令,所述计算机可执行指令在由一个或多个处理器执行时引起所述一个或多个处理器执行用于增强的事件处理的方法,所述方法包括:

标识数据库数据源;

对所述数据库数据源的数据库数据的至少部分采样,以生成所述数据库数据的一个或多个快照;

基于所述数据库数据的所述一个或多个快照来生成构造的数据流,其中所述构造的数据流与时间元素相关联;以及

基于增强的事件处理请求来提供对所述构造的数据流的访问;

其中所述增强的事件处理请求与使用所述构造的数据流可访问的所述数据库数据源

的数据库中的输入数据的指示相关联。

10. 根据权利要求9所述的介质,其中对所述数据库数据采样基于周期性采样机制,其基于预定时间间隔生成一个或多个快照。

11. 根据权利要求9所述的介质,其中对所述数据库数据采样基于差异采样机制,其生成仅包括对所述数据库数据的改变的一个或多个快照。

12. 根据权利要求9所述的介质,其中对所述数据库数据采样基于通知机制,其指示对所述数据库数据的改变何时发生,使得快照基于对所述数据库数据的改变的指示而被生成。

13. 根据权利要求12所述的介质,其中所述通知机制还包括批处理特征,其中所述批处理特征确定预定数目的改变或者确定在触发执行采样的通知之前的预定时间段。

14. 根据权利要求12所述的介质,其中所述通知机制基于与所述数据库数据的基于操作的功能相关联的管理时间元素,其中所述管理时间元素没有被记录在所述数据库中。

15. 根据权利要求14所述的介质,其中所述管理时间元素指示与用于生成具有所述时间元素的所述构造的数据流的快照相关联的时间。

16. 一种用于执行增强的事件处理的计算机系统,所述计算机系统包括:

数据库数据源部件,其被配置用于:

提供对数据库数据的访问;

构造的数据流源部件,其被配置用于:

标识数据库数据源;

对所述数据库数据源的数据库数据的至少部分采样,以生成所述数据库数据的一个或多个快照;

基于所述数据库数据的所述一个或多个快照来生成构造的数据流,其中所述构造的数据流与时间元素相关联;以及

基于增强的事件处理请求来提供对所述构造的数据流的访问,其中所述增强的事件处理请求与使用所述构造的数据流可访问的所述数据库数据源的所述数据库中的输入数据的指示相关联;

数据流源,其被配置用于:

提供对数据流的访问;以及

增强的事件处理服务器,其被配置用于:

接收所述增强的事件处理请求,其中所述增强的事件处理请求包括来自所述数据库数据源的输入数据的指示以及来自所述数据流源的输入数据的第二指示;

使用构造的数据流源来访问所述输入数据的指示中的所述输入数据,其中所述构造的数据流源提供对所述数据库数据源的数据库数据的随时间的快照的集合的访问,其中使用包括第一时间元素的所述构造的数据流来访问所述输入数据;

使用所述数据流源来访问所述输入数据的第二指示中的输入数据;

使所述构造的数据流的所述第一时间元素与所述数据流的第二时间元素相关联;以及

基于所述构造的数据流的所述输入数据和所述数据流的所述输入数据来生成针对所述增强的事件处理请求的输出。

17. 根据权利要求16所述的系统,其中所述增强的事件处理服务器包括逻辑模块,其中

所述逻辑模块被配置用于：

标识用于基于所述增强的事件处理请求来生成输出的逻辑，其中所述逻辑选自以下中的至少一项：

最新快照逻辑；以及

插值逻辑；以及

提供所述逻辑，使得所述输出基于所述逻辑而被生成，其中所述逻辑与所述构造的数据流后向兼容。

18. 根据权利要求16所述的系统，其中所述增强的事件处理服务器包括操作模块，其中所述操作模块被配置用于：

接收具有数据库中的输入数据的指示的所述增强的事件处理请求，所述输入数据不与时间元素相关联；以及

使用构造的数据流来访问所述输入数据，其中所述构造的数据流包括与时间元素相关联的所述输入数据。

19. 根据权利要求16所述的系统，其中所述增强的事件处理服务器包括有效性时间模块，其中所述有效性时间模块被配置用于：

提供有效性时间元素，所述有效性时间元素指示基于所述构造的数据流的所述输出准确地反映所述数据库数据中的数据的时间。

20. 根据权利要求19所述的系统，其中所述有效性时间元素被加速以推进增强的事件处理请求的处理。

## 用于增强的事件处理的构造的数据流

### 背景技术

[0001] 事件处理是指跟踪和分析信息流以从它们导出结论。信息流可以是具体地关于事件的。事件处理还包括组合来自多个数据源的数据以推断暗示更复杂的情况的事件或模式的事件处理。因此，事件处理可以支持标识有意义的事件并且尽可能快地对它们做出响应。

### 发明内容

[0002] 提供本发明内容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概念的选择。本发明内容不旨在标识所要求保护的主题的关键特征或必要特征，也不旨在单独用于帮助确定所要求保护的主题的范围。

[0003] 本文中描述的实施例涉及用于增强的事件处理的方法、系统和计算机存储介质。增强的事件处理请求可以被接收，增强的处理请求包括来自数据库数据源的输入数据的指示，输入数据要用于导出输出数据。增强的事件处理请求中指示的输入数据使用构造的数据流源而不是数据库数据源而被访问。构造的数据流数据源提供对数据库数据源的数据库数据的随时间的快照的集合的访问。来自构造的数据流源的数据可以根据所选择的值和所选择的时间而被访问。构造的数据流数据包括时间元素。构造的数据流数据包括在数据库数据处不可获得的时间元素，该时间元素有助于使用构造的数据流数据来为数据库数据执行增强的事件处理。针对增强的事件处理请求的输出基于使用构造的数据流数据访问的输入数据而被生成。

[0004] 增强的事件处理还可以包括组合数据流源和构造的数据流源，构造的数据流源用作数据库数据源的代理。增强的事件处理请求可以包括来自数据库数据源的输入数据的第一指示和来自数据流源的输入的第二指示。在这一点上，在输入数据的第一指示中指示的输入数据可以经由构造的数据流源而被访问，并且在第二指示中指示的输入数据可以经由数据流源而被访问。输入数据使用构造的数据流的第一时间元素和数据流的第二时间元素而被组合并且使其相关，以生成针对增强的事件处理请求的导出的输出。在实施例中，有效性时间机制提供用于指示针对增强的事件处理请求的不矛盾的输出的例程和过程。具体地，有效性时间元素指示如下时间，在该时间处，基于一个或多个数据源的输出准确地反映在该时间处在数据源中的数据。

### 附图说明

[0005] 下面参考附图详细描述本发明，在附图中：

[0006] 图1是适于在实现本文中描述的实施例时使用的示例性计算环境的框图；

[0007] 图2是其中可以采用本文中描述的实施例的示例性操作环境的框图；

[0008] 图3是示出根据本文中描述的实施例的用于执行增强的事件处理的示例性方法的流程图；

[0009] 图4是示出根据本文中描述的实施例的用于执行增强的事件处理的示例性方法的流程图；以及

[0010] 图5是示出根据本文中描述的实施例的用于执行增强的事件处理的示例性方法的流程图。

### 具体实施方式

[0011] 在本文中具体描述本发明的实施例的主题以满足法定要求。然而，描述本身并不旨在限制本专利的范围。相反，发明人预期所要求保护的主体还可以以其它方式实施，以便结合其他现有或未来技术包括不同的步骤或者与本文档中描述的步骤类似的步骤组合。此外，尽管术语“步骤”和/或“框”在本文中可以用于指示所采用的方法的不同元素，但是这些术语不应当被解释为暗示本文中公开的各种步骤中或各种步骤之间的任何特定顺序，除非和除了当明确地描述个体步骤的顺序时。

[0012] 为了本公开的目的，词语“包含(including)”具有与词语“包括(comprising)”相同的广义含义。此外，除非另有相反指示，否则诸如“一(a)”和“一个(an)”的词语包括复数以及单数。因此，例如，在存在一个或多个特征的情况下满足“一个特征”的要求。而且，术语“或”包括连词、转折连词和两者(a或b因此包括a或b、以及a和b)。

[0013] 为了下面的详细讨论的目的，参考在数据中心中提供机架内硬件库存并且作为示例在引导之前提供硬件库存来描述实施例。然而，提供工作流可以用于为各种不同类型的硬件服务提供新的机架内硬件库存，并且数据中心引导仅仅是示例性的。预期所描述的技术可以扩展到其它实现上下文。

[0014] 事件处理是指跟踪和分析信息流以从它们导出结论。信息流可以是具体地关于事件的。事件处理还包括组合来自多个数据源的数据以推断暗示更复杂的情况的事件或模式的事件处理。因此，事件处理可以支持标识有意义的事件(例如，机会或威胁)并且尽可能快地对它们做出响应。事件通常可以包括销售线索、订单或客户服务呼叫、新项目、文本消息、社交媒体发布、股票市场馈送、流量报告、天气预报、物联网数据或其他类型的数据。

[0015] 事件处理数据源可以包括数据流和数据库中的数字化数据。在示例性场景中，处理数据流可以包括基于所标识的标准过滤或聚合事件。可以基于不同的事件执行其他类型的处理。事件处理还可以基于分析来自不同数据流源的多个输入数据集。

[0016] 事件处理数据源还可以是数据库。数据库是有组织的数据集合。数据通常被组织为以支持处理要求的方式对现实的各方面建模。例如，以支持知道客户是谁以及客户住在哪里(例如，客户名称、客户地址)的方式对客户事务建模。数据库管理系统在与用户和其他应用交互以捕获和分析数据的软件应用中维持数据库表(例如，关系数据库表)。

[0017] 数据库数据可以具体地包括参考数据，参考数据包括相关或被分类的全局数据的集合。参考数据可以是与被存储在数据库表中的事件相关联的数据。参考数据可以在信任的公共域处或者从高级商业内容提供商处可获得。参考数据还可以包括要由其他数据字段使用的可允许值的集合。

[0018] 可以使用声明性框架来执行与数据库表的交互，声明性框架利用与数据库表本质上相关联的逻辑。声明性框架可以最小化描述控制流(即，个体语句、指令或函数调用被执行的顺序)。在这一点上，可以通过简单地指示什么数据需要来自数据库表而不是如何从表获得数据来支持与数据库表的交互。作为示例，SQL(结构化查询语言)可以用于管理关系数据库中保存的数据。

[0019] 包括参考数据的数据库数据不对流事件建模。例如,数据库数据不被配置为时间数据(例如,诸如时间的时间元素),因此不能对数据库数据执行依赖于数据流的事件的时间元素的数据流操作。因此,基于数据库数据或者数据库数据和数据流数据的组合的事件处理操作可以有利地利用不同的框架来单独地或与数据流源组合地增强与数据库数据源相关联的事件处理请求。在没有定义数据库数据、数据流及其组合的增强的事件处理的框架的情况下,不能对数据库数据执行可以对数据流执行的操作,因此限制了从这两个数据源推断事件和模式的能力。

[0020] 备选解决方案可以包括主动地跟踪针对数据库中的数据条目的时间戳。例如,针对数据库中的每个数据条目的开始时间和结束时间。然而,这样的实现因为需要被存储的附加量的数据而通常变得非常昂贵并且进而降低了关系数据库的效率。此外,备选解决方案不是简单的成套解决方案,并且缺乏向后兼容性,因为针对使用数据库数据的事件处理的逻辑不被配置成在数据在数据库表中被捕获的状态下与数据库数据一起工作。有时,数据库表可能必须完成移植,以便实现使用数据库数据的事件处理。

[0021] 因此,本发明的实施例提供了用于基于构造的数据流源的增强的事件处理的简单且有效的方法和系统。增强的事件处理请求可以被生成和传送以导出输出数据。增强的事件处理请求指示要从数据库数据源访问的输入数据。增强的事件处理提供使用构造的数据流源来访问输入数据。构造的数据流源提供对随时间取得的数据库数据的快照的集合的访问,使得来自构造的数据流源的数据可以根据所选择的值和所选择的时间而被访问。

[0022] 输入数据作为具有时间元素的构造的数据流被访问。构造的数据流数据包括在数据库数据源处不可获得的时间元素,该时间元素有助于使用构造的数据流来为数据库数据执行增强的事件处理。因此,可以使用构造的数据流数据代替地执行不能使用数据库数据执行的数据流操作。针对增强的事件处理请求的输出基于构造的数据流数据而被生成。构造的数据流可以被用于基于所选择的值和所选择的时间来执行查找请求。在这一点上,构造的数据流可以被用于基于查询来拉取数据库数据。

[0023] 增强的事件处理还可以包括组合至少一个数据流源和至少一个构造的数据流源,至少一个构造的数据流源用作针对数据库数据源的代理。在这一点上,增强的事件处理请求可以包括来自数据库数据源的输入数据(例如,参考数据)的第一指示和来自数据流源的输入数据(例如,事件数据)的第二指示。在输入数据的第一指示中被指示的输入数据可以经由构造的数据流源而被访问,并且在第二指示中被指示的输入数据可以经由数据流源而被访问。使用构造的数据流数据的第一时间元素和数据流数据的第二时间元素来组合输入数据并且使其相关,以生成针对增强的事件处理请求的导出的输出。在执行事件处理操作时,数据流和构造的数据流可以更容易地相关。

[0024] 使用构造的数据流的事件处理被配置成保证可重复性。可重复性或可重复的计算是指稍后使用相同的数据源重新执行事件处理分析并且生成相同的结果。例如,在第一时间对实时数据流执行的查询当在第一时间之后的第二时间被执行时产生完全相同的结果。使用构造的数据流和数据流,组合来自两个流的输入数据的查询产生相同的结果,即使从其生成构造的数据流的数据库数据已经被改变。在操作中,时间元素(例如,应用时间)被标识以与数据库数据相对应从而生成构造的数据流。构造的数据流包括数据库数据的采样快照,使得具有时间元素的构造的数据流可以被访问以执行被输入的查询。在具有可重复性

的情况下,事件处理可以包括在针对相同输入数据被执行的查询之间进行比较。查询可以被比较以评估和优化查询的性质,构造查询的备选版本以生成改进的查询,比较查询的变体之间的结果,或者运行查询的备选版本以获得更多信息。

[0025] 增强的事件处理框架(“增强的框架”)还包括构造的数据流和附加的模块/部件和机制(例如,例程和过程),附加的模块/部件和机制提供用于实现基于数据流和作为构造的数据流的数据库数据的事件处理的支持系统。在实施例中,有效性时间机制提供用于指示针对增强的事件处理请求的不矛盾的输出的例程和过程。具体地,有效性时间元素指示如下时间,在该时间处,基于一个或多个数据源的输出准确地反映在该时间处在数据源中的数据。在这一点上,本文中描述的实施例可以有利地在维持时间状态查询的有效性和性能的同时允许历史查询。

[0026] 贯穿本说明书使用各种术语。虽然贯穿本说明书提供了关于各种术语的更多细节,但是下面包括一些术语的一般定义以提供对本文中公开的想法的更清楚的理解:

[0027] 增强的事件处理请求是指具有一个或多个输入的请求,一个或多个输入至少包括数据库数据,该数据库数据不与时间元素相关联而是使用具有时间元素的构造的数据流而被访问。增强的事件处理还可以包括用于生成输出的、针对被组合的数据库数据和数据流两者的输入。

[0028] 数据库数据是指有组织的数据集合。数据库数据可以具体地包括参考数据,参考数据包括有相关或被分类的全局数据的集合。数据库数据可能不具有与数据库中的数据相关联的时间元素。本文中使用的的时间元素可以是指发生与数据库表中的条目相关联的事件的特定时间。

[0029] 数据流是指数据流中的数字化数据。事件处理可以基于分析来自不同数据流源的多个输入数据集。因此,事件处理可以支持标识有意义的事件(例如,机会或威胁)并且尽可能快地对它们做出响应。事件通常可以包括销售线索、订单或客户服务呼叫、新项目、文本消息、社交媒体发布、股票市场馈送、流量报告、天气预报、物联网数据或其他类型的数据。

[0030] 构造的数据流是指从数据库数据生成的、作为随时间的快照的集合的数据。构造的数据流与时间元素相关联,因此构造的数据流源可以根据所选择的值和所选择的时间而被访问。构造的数据流可以作为具有时间元素的数据流而被访问,并且还可以使用查找机制而被访问,该查找机制基于查询从构造的数据流提取数据。

[0031] 时间元素是指发生事件的特定时间。时间元素可以与被输入到数据库中的事件相关联,该时间元素没有被保留在该数据库中。时间元素可以与数据流事件相关联。时间元素还可以与构造的数据流事件相关联,该构造的数据流事件从发生事件时被导出并且被保存为数据库数据的快照的一部分。构造的数据流还可以与多个时间元素相关联,使得构造的数据流支持双时间连接,如下面更详细地讨论的。

[0032] 因此,在本文中描述的第一实施例中,提供了一种用于增强的事件处理的计算机实现的方法。该方法包括接收增强的事件处理请求。增强的事件处理请求包括来自数据库数据源的输入数据的指示。该方法还包括使用构造的数据流源来访问在输入数据的指示中指示的输入数据,其中构造的数据流源提供对随时间的快照的集合的访问。输入数据使用包括时间元素的构造的数据流源而被访问。该方法还包括基于使用构造的数据流访问的输入数据来生成针对增强的事件处理请求的输出。



[0033] 在本文中描述的第二实施例中,提供了一种存储计算机可使用指令的一个或多个计算机存储介质,计算机可使用指令在由一个或多个计算设备使用时引起一个或多个计算设备执行用于增强的事件处理的方法。该方法包括标识数据库数据源。该方法还包括对数据库数据源的数据库数据的至少部分采样,以生成数据库数据的一个或多个快照。该方法还包括基于数据库数据的一个或多个快照来生成构造的数据流。构造的数据流与时间元素相关联。该方法还包括基于增强的事件处理请求来提供对构造的数据流的访问;其中增强的处理事件请求与使用构造的数据流可访问的数据库数据源的数据库中的输入数据的指示相关联。

[0034] 在本文中描述的第三实施例中,提供了一种用于执行增强的事件处理的计算机系统。该系统包括被配置用于提供对数据库数据的访问的数据库数据源部件。该系统包括构造的数据流源部件,其被配置用于:标识数据库数据源;对数据库数据源的数据库数据的至少部分采样,以生成数据库数据的一个或多个快照;基于数据库数据的一个或多个快照生成构造的数据流,其中构造的数据流与时间元素相关联;以及基于增强的事件处理请求来提供对构造的数据流的访问。增强的处理事件请求至少与使用构造的数据流可访问的数据库数据源的数据库中的输入数据的指示相关联。该系统还包括被配置用于提供对数据流的访问的数据流源。

[0035] 该系统包括事件处理服务器,其被配置用于:接收增强的事件处理请求,增强的事件处理请求包括来自数据库数据源的输入数据的指示和来自数据流的输入数据的第二指示;使用构造的数据流源访问输入数据的指示中的输入数据,构造的数据流源提供对数据库数据源的数据库数据的随时间的快照的集合的访问,输入数据使用包括第一时间元素的构造的数据流而被访问;使用数据流源访问输入数据的第二指示中的输入数据;使构造的数据流的第一时间元素与数据流的第二时间元素相关;以及基于构造的数据流的输入数据和数据流的输入数据来生成针对增强的事件处理请求的输出。

[0036] 在简要描述本发明的实施例的概述之后,下面描述可以实现本发明的实施例的示例性操作环境,以便为本发明的各个方面提供一般上下文。具体来说,首先参考图1,用于实现本发明的实施例的示例性操作环境被示出并且一般地被指定为计算设备100。计算设备100仅仅是合适的计算环境的一个示例,而非意图暗示对于本发明的使用范围或功能的任何限制。计算设备100也不应当被解释为具有与所示的部件中的任一个或组合相关的任何依赖性要求。

[0037] 本发明可以在由计算机或其他机器(诸如个人数字助理或其他手持设备)执行的计算机代码或机器可使用指令的一般上下文中描述,计算机代码或机器可使用指令包括计算机可执行指令,诸如程序模块。通常,包括例程、程序、对象、组件、数据结构等的程序模块是指执行特定任务或实现特定抽象数据类型的代码。本发明可以在各种系统配置中被实践,包括手持设备、消费电子产品、通用计算机、更专业的计算设备等。本发明还可以在分布式计算环境中被实践,其中任务由通过通信网络链接的远程处理设备来执行。

[0038] 参考图1,计算设备100包括直接或间接耦合以下设备的总线110:存储器112、一个或多个处理器114、一个或多个呈现部件116、输入/输出端口118、输入/输出部件120和说明性电源122。总线110表示可以是一个或多个总线(例如地址总线、数据总线或其组合)的总线。虽然图1的各个框为了清楚起见用线条示出,但实际上,界定各种部件不是那么清楚,并

且隐喻地,线条将更准确地是灰色和模糊的。例如,可以将诸如显示设备的呈现部件视为I/O部件。此外,处理器具有存储器。认识到,这是本领域的本质,并且重申图1的图仅仅说明可以结合本发明的一个或多个实施例使用的示例性计算设备。在诸如“工作站”、“服务器”、“膝上型计算机”、“手持设备”等的类型之间不进行区分,因为所有这些类型都在图1的范围内并且是指“计算设备”。

[0039] 计算设备100通常包括各种计算机可读介质。计算机可读介质可以是任何可用介质,该可用介质可以由计算设备100访问并且包括易失性和非易失性介质、可移除和不可移除介质。作为示例而非限制,计算机可读介质可以包括计算机存储介质和通信介质。

[0040] 计算机存储介质包括以用于存储信息的任何方法或技术实现的易失性和非易失性、可移除和不可移除介质,该信息诸如计算机可读指令、数据结构、程序模块或其它数据。计算机存储介质包括但不限于RAM、ROM、EEPROM、闪存或其他存储器技术、CD-ROM、数字通用盘(DVD)或其他光盘存储装置、磁带盒、磁带、磁盘存储装置或其他磁性存储设备、或者可以用于存储所需信息并且可以由计算设备100访问的任何其他介质。计算机存储介质排除信号本身。

[0041] 通信介质通常在诸如载波或其它传输机制的已调制数据信号中实施计算机可读指令、数据结构、程序模块或其他数据,并且包括任何信息传递介质。术语“已调制数据信号”表示使其特性中的一个或多个特性以在信号中编码信息的方式设置或改变的信号。作为示例而非限制,通信介质包括有线介质(诸如有线网络或直接有线连接)以及无线介质(诸如声学、RF、红外和其他无线介质)。任何上述的组合也应当被包括在计算机可读介质的范围内。

[0042] 存储器112包括易失性和/或非易失性存储器形式的计算机存储介质。存储器可以是可移除的、不可移除的或其组合。示例性硬件设备包括固态存储器、硬盘驱动器、光盘驱动器等。计算设备100包括从诸如存储器112或I/O部件120的各种实体读取数据的一个或多个处理器。一个或多个呈现部件116向用户或其他设备呈现数据指示。示例性呈现部件包括显示设备、扬声器、打印部件、振动部件等。

[0043] I/O端口118允许计算设备100逻辑地耦合到包括I/O部件120的其它设备,I/O部件120中的一些可以是内置的。说明性部件包括麦克风、操纵杆、游戏板、碟形卫星天线、扫描仪、打印机、无线设备等。

[0044] 现在转到图2,图2是描绘适用于在实现本文中描述的实施例时使用的示例性增强的事件处理系统200的框图。通常,系统200示出可以用于执行增强的事件处理等的操作环境。增强的事件处理可以与指示至少来自数据库数据源的输入数据的事件处理请求相关联。

[0045] 在未示出的其他部件中,系统200通常可以包括数据库数据源部件220、构造的数据流源部件230、数据流源部件240、计算设备250和增强的事件处理服务器260,所有这些部件经由网络270彼此通信。网络210可以包括但不限于一个或多个局域网(LAN)和/或广域网(WAN)。这样的网络环境在办公室、企业范围的计算机网络、内联网和因特网中很常见。

[0046] 在一些实施例中,所示部件和/或模块中的一个或多个可以被实现为独立的应用。在另外的实施例中,所示部件和/或模块中的一个或多个可以经由计算设备实现为基于因特网的服务和/或增强的事件处理服务器260内的模块。本领域普通技术人员将理解,图2所

示的部件和模块在性质和数目上是示例性的,而不应当被解释为限制。可以采用任何数目的部件和/或模块来实现本文中描述的功能。例如,在本发明的实施例的范围内,可以在系统200中采用任何数目的计算设备、数据库数据源部件、构造的数据流源部件、数据流源部件、计算设备、增强的事件处理服务器和网络。每个可以包括单个设备/接口或者在操作环境中协作的多个设备/接口。例如,布置在分布式环境中的多个设备和/或模块可以共同地提供本文中描述的增强的事件处理服务器260功能。如本文中使用的短语“应用”或“服务”可以广泛地是指计算设备和/或多个计算设备(诸如数据中心中的多个计算设备)之上运行的或访问其内的存储位置的任何软件或软件的部分。

[0047] 应当理解,本文中描述的这种和其他布置仅作为示例阐述。除了或代替所示的布置和元件,可以使用其他的布置和元件(例如,机器、接口、功能、顺序和/或功能分组),并且可以将一些元件一起省略。此外,本文中描述的元件中的很多元件是可以被实现为分立的或分布式部件、或者与其他部件结合来实现、以及以任何合适的组合和位置来实现的功能实体。本文中描述为由一个或多个实体执行的各种功能可以由硬件、固件和/或软件来执行。例如,各种功能(包括以下关于系统200描述的功能)可以由执行存储在存储器中的指令的处理器来执行。

[0048] 现在转向可以被包括在系统200中的各种部件,数据库数据源部件220负责存储和提供对数据库数据的访问。数据库数据源部件可以在具有关系数据库表的数据库中存储数据。数据库数据可以不具有与数据库中的数据相关联的时间元素。如本文中使用的元素可以是指发生与数据库表中的条目相关联的事件的特定时间。例如,对于客户,第一事务可以捕获客户的第一地址。客户可以在第二时间执行第二事务,并且然后使用客户的新的第二地址更新第一地址。因此,数据库更新客户地址而不考虑时间元素。数据库表包括基于被记录的新信息而随时间改变的记录,而不考虑与先前条目或新条目相关联的时间。在这一点上,对数据库数据部件的数据请求(例如,查询)在数据随时间改变时不可重复。可重复计算是指稍后使用相同的数据源重新执行事件处理分析并且生成相同的结果。

[0049] 另外,数据库数据部件可以支持不包括时间元素的操作。针对客户地址信息的查询提供在查询被执行时的客户的客户地址信息。不支持具体地在第一事务时或在第二事务期间找出客户地址信息的操作,因为数据库表仅记录与事务相关联的最新的客户地址信息。

[0050] 在示例性实施例中,数据库数据源部件220可以提供对支持管理时间元素的数据库的访问。管理时间元素是指用于数据库数据的基于操作的功能(例如,审计、维护和提供)的时间元素(例如,时间)。在具有管理时间元素的数据库中,管理时间元素并非意图被记录在数据库表中。然而,管理时间元素可以用作向一个或多个外部部件通知数据库数据的改变的通知机制的一部分。通知机制还可以实现批处理特征,其中批处理特征可以确定预定数目的改变或者确定在触发执行数据库数据的采样的通知之前的预定时间段。因此,即使管理时间元素的预期性质可能是管理时间元素是短暂的并且未被记录,时间元素仍然可以用于生成构造的数据流。可以设想,构造的数据流源部件可以具有订阅机制,订阅机制用于接收数据库数据的改变的通知并且使用创建差异快照的采样来生成构造的数据流,如下面更详细地讨论的。

[0051] 进一步设想,数据库数据源部件220可以在标识增强的事件处理逻辑之前已经被

开发, 增强的事件处理逻辑用于仅使用事件处理或者按照与数据流源集成的方式处理数据库数据部件中的数据。在这一点上, 数据库数据源部件220可以不包括与用于增强的事件处理的所选择的逻辑具体地相关联的任何特征。换句话说, 本文中描述的实施例中的增强的事件处理逻辑被配置成与传统系统一起工作或者与独立于增强的事件处理逻辑而开发的数据库向后兼容, 如下面更详细地讨论的。

[0052] 构造的数据流源部件230负责管理和提供对构造的数据流的访问。构造的数据流被生成, 使得其可以基于与数据库数据相关联的值和时间而被访问。在这一点上, 构造的数据流根据值和时间元素。基于数据的值和时间元素, 构造的数据流与数据库数据相关联。可以设想, 关键性质或性质集合可以用于保证存在针对要被构造成数据流的数据库数据的条目。

[0053] 构造的数据流在给定的持续时间内保存快照, 以便使能可重复性。当将数据流与数据库数据连接时, 构造的数据流快照可以由数据流 (例如, 实时数据流) 的时间元素 (例如, 应用时间的时间戳) 定义。数据库数据不管理具有时间概念而具有管理时间元素 (例如, 数据库系统的提交时间) 作为与被采样以生成快照的数据库数据相关联的时间的关系表。在这一点上, 数据库数据源为了支持以提交时间作为同步时间进行快照, 数据库数据可以作为事件处理系统中的流式数据源可访问。

[0054] 在实施例中, 数据库数据源可以利用时间增量策略来实现, 该策略即使在没有快照修改的情况下也将在增强的事件处理系统侧使时间自动地前进。该策略还可以用于从数据库数据源驱动基于拉取的更新。当可选的时间戳可以被提供时, 数据库数据源可以允许初始化模式 (类似于概述 (rundown))。在该模式下, 时间戳可以不需要具有任何特定的顺序。此操作只能在使用数据库数据之前执行一次。

[0055] 在下面的代码中被指示作为Reference Data或refData的构造的数据流可以支持允许使用构造的数据流运行操作的应用编程接口 (API)。API可以是其中用户可以执行动作的CRUD (创建、读取、更新和删除) 接口。下面的代码中的示例性场景可以与应用服务相关联, 应用服务可以被配置成从各种设备收集记录的错误信息以执行近实时分析。每个设备可以由设备标识符 (例如, Machine\_ID) 来标识。可以设想, 设备标识符可以被映射到对应设备名称和细节 (例如, MachineName和MachineDetails)。从设备标识符到对应设备名称的映射可以被存储为将多个设备标识符映射到设备名称的数据库表中的元数据。API可以具体地支持键值, 该键值存储实现时间元素 (例如, 作为系统时间的事务时间) 的表模式。通过下面的比较, 具有显式时间的API可以用于构造的数据流的初始化。

[0056]

```
var refData = new ReferenceData<int, MachineDetails>(new Uri("http://..."));

refData.Add(1, new MachineDetails(machineId:1, machineName:"x", domain:"A"));
refData.Add(2, new MachineDetails(2, "y", "A"));
refData.Update(1, new MachineDetails(1, "x2", "A"));
refData.Delete(2);
```

[0057]

```
var refData2 = new ReferenceData<int, MachineDetails, long>(new Uri("http://myRefData"));
refData2.Add(1, new MachineDetails(machineId: 1, machineName: "x", domain: "A"), 1000);
refData2.Add(2, new MachineDetails(2, "y", "A"), 5000);
```

[0058]

```
refData2.Update(1, new MachineDetails(1, "x2", "A"), 3000);
refData2.Delete(2, 5000);
refData2.Add(2, new MachineDetails(2, "yz", "ABC"), 4000);
refData2.Add(2, new MachineDetails(2, "y", "A"));
```

[0059] 不以任何特定顺序获取时间信息,并且针对数据库数据源的时间相关信息的调用可以与将时间设置为当前系统时间的正常API调用混合。

[0060] 构造的数据流API可以实现步骤功能,因为当在t1和t2之间不存在改变时,时间t2处的值是t1处的值。API可以用于读取给定时间的特定键的值:

[0061]

```
refData2.MoveTimeTo(4500);
MachineDetails value = refData2.ReadAt(key: 2, time: 3500);           // (2, "y", "A")
value = refData2.ReadAt(key: 2, time: 4200);                         // 空
```

[0062] 可以设想,在构造的数据流中的最后一次应用时间之后读取构造的数据流可以被禁止,因为它将导致不可再现的结果。在应用时间的情况下,标点 (punctuation) 机制可以被用于强制时间前进,向前移动应用时间,或者关闭初始化阶段并且切换到系统时间。在系统时间的情况下,本文中的实施例可以禁止在当前时间之后读取,如下面更详细地讨论的。

[0063] API还可以用于读取给定时间的所有值:

[0064]

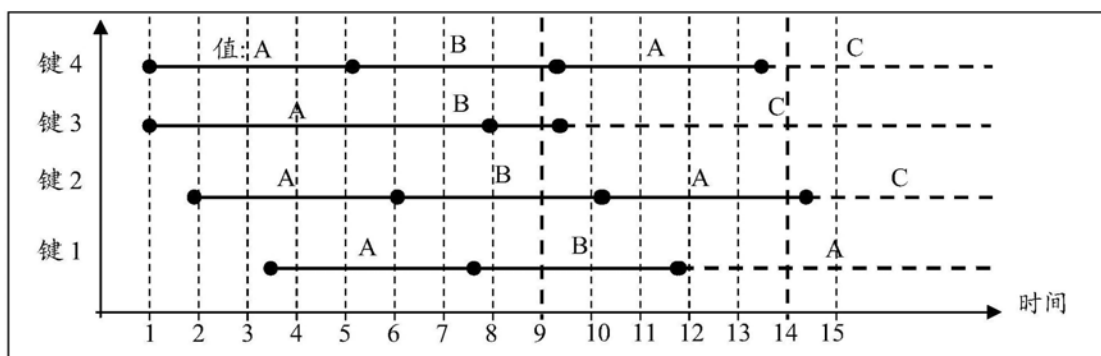
```
Dictionary<int, MachineDetails> values = refData2.ReadAt(2500);      // {{1, (1, "x", "A")},
                                                                    // {2, (2, "y", "A")}}
```

[0065] 结果是所有本地已知的值。取决于分片/分区,这可以是构造的数据流中的参考数据的子集。

[0066] 可以实现流式传输机制以对构造的数据流进行流式传输。API的基本读取功能(例如,ReadAt)可能不足以支持构造的数据流的时间元素,因为有必要知道改变何时发生。流式传输机制利用构造的数据流来创建对应边缘和间隔事件。流式传输机制包括为每个改变创建边缘事件的能力:每个改变导致具有前值的边缘结束事件和具有新值的边缘开始事件。流式传输机制还包括知道当构造的数据流必须从特定时间(即,不是从时间开始)被流式传输时的初始状态的能力。

[0067] 作为示例:

[0068]



[0069] 表-构造的流数据

[0070] 从9到14的对应于该构造的流数据的流式传输改变应当导致：

[0071] 1) 在时间9的初始“状态”

[0072] • 时间戳:5.2, 键:4, 值:B

[0073] • 时间戳:6.1, 键:2, 值:B

[0074] • 时间戳:7.5, 键:1, 值:B

[0075] • 时间戳:7.9, 键:3, 值:B

[0076] 2) 9到14之间的改变

[0077] • 时间戳:9.2, 键:4, 值:A

[0078] • 时间戳:9.3, 键:3, 值:C

[0079] • 时间戳:10.1, 键:2, 值:A

[0080] • 时间戳:11.8, 键:1, 值:A

[0081] • 时间戳:13.5, 键:4, 值:C

[0082] 然而,从该改变流创建边缘/间隔事件流将需要跟踪每个键(最后时间戳和最后值),以便创建与前值相对应的边缘结束事件。将改变作为一对值(旧的和新的)来存储和流式传输将使得能够更容易:

[0083] 1) 在时间9的初始“状态”

[0084] (可以用于每个改变事件创建一个边缘开始事件)

[0085] • 时间戳:5.2, 键:4, 值:B, 旧的:空

[0086] • 时间戳:6.1, 键:2, 值:B, 旧的:空

[0087] • 时间戳:7.5, 键:1, 值:B, 旧的:空

[0088] • 时间戳:7.9, 键:3, 值:B, 旧的:空

[0089] 2) 9到14之间的改变

[0090] (可以用于每个改变事件创建一个边缘结束事件和一个边缘开始事件)

[0091] • 时间戳:9.2, 键:4, 值:A, 旧的: {值:B, 自:5.2}

[0092] • 时间戳:9.3, 键:3, 值:C, 旧的: {值:B, 自:7.9}

[0093] • 时间戳:10.1, 键:2, 值:A, 旧的: {值:B, 自:6.1}

[0094] • 时间戳:11.8, 键:1, 值:A, 旧的: {值:B, 自:7.5}

[0095] • 时间戳:13.5, 键:4, 值:C, 旧的: {值:A, 自:9.2}

[0096] 该流足以创建对应于构造的数据流的边缘事件。这将在内部用于将构造的数据流暴露为可以由增强的事件处理系统使用的源。

[0097]

```
IObservable<Change<int, MachineDetails, long>> observable =  
refData.StreamChanges(fromTime: 0, toTime: 5000);  
  
observable.ForEachAsync(change => Console.WriteLine(string.Format(  
    "time: {0}, key: {1}, oldValue: {2}, newValue: {3}",  
    change.Time, change.Key,  
    change.OldValue!=null ? change.OldValue.Domain + "\\" +  
change.OldValue.MachineName : "null",  
    change.NewValue!= null ? change.NewValue.Domain + "\\" +  
change.NewValue.MachineName : "null"))
```

[0098]

```
));
```

[0099] 可以设想,当从t1到t2并且接下来从t2到t3流式传输参考数据时,可以忽略t2处的初始状态。要么有一种方法不请求它或有必要去重复所得到的边缘开始事件。

[0100] 构造的流数据还可以支持操作者将构造的数据流与点事件流(例如,数据流)连接。作为示例,代码如下

[0101]

```
var queryDef = ReferenceData.From(refData)  
    .TemporalJoin(EventStream.From(stream),  
        (ref, str) => ref.MachineId == ref.Id,  
        (ref, str) => new { Id = str.Id, State = str.State, Name = ref.MachineName } )  
    .To(sink);  
  
var query = ctx.StartQuery("q1", queryDef);
```

[0102] 连接的参数是:1)流,2)连接谓词,3)结果选择器(即投影)。当流和参考数据已经根据机器id被重新分组时,连接谓词可以被忽略或用“(ref, str) => true”替换。

[0103] 可以设想,可以使用构造的数据流来执行双时间连接。例如,当数据库中的参考数据使用流式传输机制作为构造的数据流被消耗时,事件的时间元素(例如,时间戳)可以在时间连接中使用,并且当数据库中的参考数据使用查找机制作为构造的数据流被消耗时,可以使用不同的时间线(例如,事件的有效载荷中的另一时间)作为连接条件。

[0104] 构造的数据流部件230被配置成使用具有在增强的事件处理中使用的数据库数据的标识的数据库数据源来生成构造的数据流。还设想,预期的增强的事件处理请求可以预先确定和指示在生成构造的数据流时使用什么数据库数据。在构造构造的数据流时使用的数据库数据可以是针对数据库数据进行的预期的增强的事件处理请求的因素。

[0105] 增强的事件处理请求包括针对数据库数据部件220中的数据的请求,该数据库数据部件220中的数据使用构造的数据流源部件230的构造的数据流而实际被访问。构造的数据流可以用作数据库数据部件的代理,同时模拟数据流特征和功能。可以设想,构造的数据流可以被定义为增强的事件处理系统中的数据类型。数据类型可以被定义为确定如何组织

构造的数据流的模式。

[0106] 在操作中,构造的数据流可以基于采样机制而被生成,该采样机制支持对数据库数据的离散实例采样以创建数据库数据的快照。可以对整个数据库或者仅对数据库数据的标识部分采样。还可以具体地基于预期的增强的事件处理请求来进行采样,使得与增强的事件处理请求相关联的值被保存在数据库的快照中。

[0107] 在一个实施例中,构造的数据流源部件230可以实现周期性采样机制。周期性采样机制支持以预定或指定的时间间隔执行采样。作为示例,可以每五分钟对数据库数据采样。在另一示例性实施例中,构造的数据流可以基于差异采样机制而被生成,该差异采样机制支持采样以生成数据库数据的差异快照。差异快照指示数据库数据的改变,而不是数据库数据的整个快照。可以设想,用于生成不同快照的差异采样可以被实现以降低存储数据库数据的整个快照的成本。作为示例,缓慢改变的数据集对于差异采样创建差异快照可以是理想的,因为改变不频繁,因此,差异快照捕获可能曾经已经发生的任何改变。构造的数据流的快照支持可重复性,因为针对具有快照的构造的数据流的查询提供相同的答案,而对数据库数据的查询可以随着数据库中的数据随时间改变而提供不同的答案。

[0108] 构造的数据流源部件230可以实现订阅机制,该订阅机制接收数据库数据的改变的通知。改变的通知可以基于在数据库数据源部件220处实现的通知机制,该数据库数据源部件220使用管理时间元素来使数据库数据中的改变与改变的时间相关联。当通知被接收时,构造的数据流数据部件可以对数据库数据采样并且使采样的数据库数据与管理时间元素中的时间相关联。采样可以是如上所讨论的全采样或差异采样。在这一点上,采样不是基于周期性机制,而是基于数据库数据的改变的通知。此外,改进了来自增强的事件处理请求的输出的正确性,因为不像周期性采样机制,构造的数据流数据准确地反映了数据库数据的改变。

[0109] 构造的数据流有利地是灵活的,因为它可以支持与构造的数据流相关联的不同类型的增强的事件处理逻辑,并且可以以向后兼容的方式这样做。构造的数据流可以利用采样来生成数据库数据的快照。采样可以基于周期性采样机制、差异采样机制。构造的数据流提供具有可重复性的增强的事件处理,而不必改变数据库数据或构造的数据流的历史信息或结构。本文中描述的用于生成针对增强的事件处理请求的输出的逻辑可以用不同的采样方法以及甚至用对采样方法的改变来维持。对构造的数据流的请求有利地与数据库数据的快照的构造数据流以任一方式起作用。

[0110] 继续参考图2,数据流源240负责管理和提供对数据流的访问。数据流或信息流可以具体地是关于事件的。数据流数据与指示何时发生与数据相关联的事件的时间元素相关联。数据流可以经由监听器应用主动地被监听。数据流也可以针对短的时间段(例如5分钟)或长的时间段(例如,一年到几年)暂时地被存储,并且好象数据正在被主动地流式传输一样被重放。作为示例,事件处理请求可以用于将新的流式传输数据与来自一年前的流式传输数据比较,以从其比较得出结论。在这一点上,数据流可以被组合以执行跟踪和分析从而导出输出。此外,数据流可以被包括在具有数据库数据的增强的事件处理请求中,使得数据流与构造的数据流组合以生成针对增强的事件处理请求的输出。

[0111] 例如,计算设备250可以包括以上参考图1描述的任何类型的计算设备100。计算设备250可以提供对本文中描述的不同部件的访问。具体地,如本文中另外描述的,计算设备



250可以访问增强的事件处理服务器260以执行由增强的事件处理服务器支持的一个或多个操作。

[0112] 增强的事件处理服务器260负责管理如本文中描述的增强的事件处理。增强的事件处理服务器260根据所选择的值和时间来访问作为快照的集合的构造的数据流。增强的事件处理服务器260可以接收增强的事件处理请求。可以设想,增强的事件处理请求可以经由计算设备250或者使用被编程为生成针对系统事件处理的请求的请求服务而被接收。增强的事件处理请求可以基于增强的事件处理请求与一个或多个数据库数据源相关联还是与数据库数据源和数据流源二者相关联而被处理。在这一点上,增强的事件处理服务器260可以使用特定逻辑来使数据流和数据库数据相关,以导出针对数据的输出。增强的事件处理请求可以基于跟踪的数据的输入来请求输出,以便分析数据以生成推断。输出可以是数据流中的数据和构造的流数据二者导出的结论。

[0113] 继续参考图2,增强的事件处理服务器260可以包括若干模块,包括定时模块262、操作模块264、逻辑模块266和有效性时间模块268。定时模块可以实现用于处理事件的定时机制。具体地,定时机制具体地标识与事件相关联的时间元素和到期元素,其中时间元素指示事件发生的时间,并且到期元素指示事件何时停止可用。在这一点上,使用定时机制的定时模块262可以基于事件有多快到达来支持事件处理,以基于与接收的输入相关联的定时来快速地生成输出

[0114] 操作模块264负责管理可以针对数据流、数据库数据、构造的数据流及其任何组合执行的操作。具体地,事件处理请求可以包括来自数据库数据的输入数据的指示,或者可以包括组合请求,该组合请求包括针对数据库数据和数据流的输入数据的指示。构造的数据流可以用作数据库数据的代理,并且因此允许使用与构造的数据流和数据库数据相关联的事件来执行不同类型的操作。

[0115] 增强的事件处理请求可以包括否则将经由对数据库数据的直接访问不可能的时间元素。在这一点上,操作模块264可以支持声明性的、发出增强的事件处理数据请求的查询机制。声明性查询机制最小化或消除了操作应当如何工作并且支持操作应当实现的更多操作。例如,用户可以输入简单连接操作,以与连接两个传统数据库表相同的方式,连接数据流与使用构造的数据流的、来自数据库表的参考数据。

[0116] 作为示例,假定对应于由不同仪器完成的化学分析的事件流:

[0117] UseCase1\_stream

[0118]

Assay_time	Chem_id	Instrument_id
上午8:10	1	1
上午8:20	1	2
上午8:30	2	1
上午8:40	3	2

[0119] 为了知道已经进行了给定类型的多少测试(例如,H1AC),需要用名称(映射id——名称在参考数据中)来注释事件。因此,假定以下快照从上午8:00到上午9:00有效。

[0120]

Chem_Id	名称	描述
1	H1AC	用于糖尿病的血红蛋白 A1c 测试
2	Xyz	Xyz 测试
3	Abc	Abc 测试

[0121] 为了用名称替换chem\_id,查询可以是:

[0122] 1 SELECT System.Timestamp AS Assay\_time,stream.Chem\_id,map.Name AS Chem\_name,

[0123] 2 stream.Instrument\_id

[0124] 3 FROM Timestamp(UseCase1\_stream,Assay\_time) stream

[0125] 4 JOIN UseCase1\_mapping map

[0126] 5 ON stream.Chem\_id=map.Chem\_Id

[0127] 具有名称的所得到的流将是:

[0128]

Assay_time	Chem_id	Chem_name	Instrument_id
上午8:10	1	H1AC	1
上午8:20	1	H1AC	2
上午8:30	2	Xyz	1
上午8:40	3	Abc	2

[0129] 有利地,生成增强的事件处理请求与现有的声明性查询机制一致,并且维持用于从所标识的输入生成输出的直观方法。因此,增强的事件处理特征对于向操作部件提交查询或操作的用户是透明的。操作模块264还可以至少部分地基于如下面更详细地讨论的有效性时间机制来执行增强的事件处理请求。

[0130] 逻辑模块266负责管理基于构造的数据流的结构的事件处理的特定方法。如上所述,构造的数据流可以基于采样,该采样使用周期性采样或差异采样来生成数据库数据的快照。采样方法通知逻辑模块266的逻辑以用于基于输入来生成输出。特定类型的数据还影响可以用于生成输出的逻辑。

[0131] 在一个实施例中,逻辑模块266可以实现所选择的逻辑,使得经由操作模块264接收的操作使用所选择的逻辑而被执行。最新快照逻辑利用包括最新快照的快照的集合来执行与增强的事件处理请求相关联的操作。在这一点上,数据库数据在构造的数据流中的给定快照中可获得,并且操作使用直到最新数据快照的快照的集合而被执行。

[0132] 在另一示例性实施例中,逻辑模块266可以实现如下插值逻辑,其支持在给定来自构造的数据流的两个快照的值的条件下近似构造的数据流的值。插值逻辑可以为特定类型的可变数据类型而不是其他类型(例如,温度而不是客户地址)生成中间值。在操作中,采样机制(例如,周期性的或差异性的)可以包括在第一时间(例如,下午5:00)的第一值(例如70°F)和在第二时间(例如,下午5:05)的第二值(例如,73°F)。如果增强的事件处理请求查询在中间时间(例如,下午5:03)的温度,则插值逻辑被配置成近似中间值(例如,70°F和73°F)

作为潜在实际值。可以设想,插值机制可以显式地指示所得到的输出基于插值而不是来自构造的数据流的实际输入数据。

[0133] 有利地,逻辑模块266可以利用向后兼容的构造的数据流来实现改变逻辑选择(例如,从最新快照到插值逻辑或反之亦然),以支持逻辑的任何改变。换句话说,构造的数据流的性质支持从构造的数据流生成输出数据的逻辑的改变,而不重构构造的数据流本身。作为示例,可以在明确定义的时间点基于不同的逻辑来访问具有参考数据的数据库数据,该参考数据作为数据流或使用查找机制而被访问。具体地,参考数据可以是每分钟准时得到的温度读数。构造的流数据根据所得到的最后温度读数值(例如,最新数据快照逻辑)。因此,产生阶梯函数。在逻辑在不同的时间点(例如,指定的逻辑改变时间)被改变为线性插值逻辑的事件中,构造的数据流查询可以与直到指定的逻辑改变时间的最新快照逻辑相关联,并且然后线性插值逻辑可以用于在指定的逻辑改变时间之后的值。这样的语句不影响过去积累的参考数据,因此不需要任何重新计算,因此构造的数据流是向后兼容的。

[0134] 继续参考图2,有效性时间模块268负责指示针对增强的事件处理请求的不矛盾的输出。具体地,有效性时间模块268实现有效性时间机制,该有效性时间机制指示针对与构造的数据流相关联的输出的有效性时间元素。有效性时间机制提供用于指示针对增强的事件处理请求的不矛盾的输出的例程和过程。具体地,有效性时间元素指示如下时间,在该时间处,基于一个或多个数据源的输出在执行时准确地反映数据源中的数据。构造的数据流的性质使得它可以准确地反映数据库数据,仅直到在本文中讨论的一个或多个机制被用于更新构造的数据流之前的某一时间。构造的数据流反映数据库中的最新数据的能力根据采样,并且具体地根据所利用的采样机制。因此,请求有效,仅直到构造的数据流数据的最后快照的时间(例如,有效性时间元素)。甚至对于数据流数据,其他限制(例如,等待时间)可以使得延迟最新信息。

[0135] 可以设想,有效性时间模块268可以将针对操作的有效性时间传达给操作模块,使得操作模块可以在批准请求的有效性之前等待所有事件数据(实际数据流或构造的数据流)到达。还设想,给定使用有效性模块268指示的有效性时间,操作模块264可以防止运行在请求中具有时间的请求,其输出将不与有效性时间元素相关联。在实施例中,操作模块264可以运行请求并且提供输出,但是将输出与最近的有效性时间元素相关联,使得知道请求有效仅直到有效性时间元素的时间。

[0136] 在实施例中,有效性时间可以使用有效性时间模块268而被加速。围绕事件处理系统的环境(例如,系统故障或等待时间)可以延迟对用于更新构造的数据流的数据库数据的访问。为了继续事件处理,当用于生成数据库数据的快照的采样尚未被执行时,有效性时间模块可以指示与构造的数据流事件相关联的新的有效性时间元素,以支持有效性时间元素。因此,有效性时间模块268可以牺牲构造的数据流的准确性,以便推进增强的事件处理请求。

[0137] 现在参考图3,示出了示出用于执行增强的事件处理的示例性方法300的流程图。尽管术语“步骤”和/或“框”在本文中可以用于指示所采用的方法的不同元素,但是这些术语不应当被解释为暗示本文中公开的各种步骤中或各种步骤之间的任何特定顺序,除非和除了当明确地描述个体步骤的顺序时。

[0138] 在框310,增强的事件处理请求被接收。增强的事件处理请求包括来自数据库数据

源的输入数据的指示。来自数据库数据源的输入数据不与针对输入数据的时间元素相关联。在框320,输入数据的指示中的输入数据使用构造的数据流源而被访问。构造的数据流源提供对随时间的快照的集合的访问。快照是数据库数据源的数据库数据。输入数据使用包括时间元素的构造的数据流而被访问。时间元素指示与输入数据相关联的事件发生的时间。构造的数据流的时间元素从如下时间被确定,在该时间处,构造的数据流的事件发生并其作为数据库数据中的事件快照的一部分被存储。在框330,针对增强的事件处理请求的输出基于使用构造的数据流访问的输入数据而被生成。输出与有效性时间元素相关联,有效性时间元素指示如下时间,在该时间处,基于构造的数据流的输出准确地反映数据库数据中的数据。

[0139] 现在转到图4,示出了用于执行增强的事件处理的示例性方法400的流程图。在框410,增强的事件处理请求被接收。增强的事件处理请求包括来自数据库数据源的输入数据的指示和来自数据流源的输入数据的第二指示。在框420,使用构造的数据流源来访问输入数据的指示中的输入数据。构造的数据流源提供对数据库数据源的数据库数据的随时间的快照的集合的访问。输入数据使用包括第一时间元素的构造的数据流而被访问。

[0140] 在框430,输入数据的第二指示中的输入数据使用数据流源被访问。在框440,使构造的数据流的第一时间元素与数据流的第二时间元素相关。在框450,针对增强的事件处理请求的输出基于构造的数据流的输入数据和数据流的输入数据而被生成。

[0141] 现在转到图5,示出了用于执行增强的事件处理的示例性方法500的流程图。在框510,数据库数据源被标识。数据库数据源与数据库数据相关联。数据库数据包括参考数据。在框520,对数据库数据源的数据库数据的至少部分采样,以生成数据库数据的一个或多个快照。对数据库数据采样基于周期性采样机制或者差异采样机制,周期性采样机制基于预定的时间间隔生成来一个或多个快照,差异采样机制生成仅包括对数据库数据的改变的一个或多个快照。

[0142] 采样还可以基于通知机制,通知机制指示何时已经发生对数据库数据的改变,使得快照在对数据库数据的改变的指示时被生成。通知机制可以包括批处理特征,批处理特征确定预定数目的改变或者确定在触发执行采样的通知之前的预定时间段。通知系统可以基于与数据库数据的基于操作的功能相关联的管理时间元素,管理时间元素没有被记录在数据库中。管理时间元素指示与快照相关联的时间,该快照用于生成具有时间元素的构造的数据流。

[0143] 在框530,构造的数据流基于数据库数据的一个或多个快照而被生成,其中构造的数据流与时间元素相关联。在框540,对构造的数据流的访问基于增强的事件处理请求而被提供;增强的处理事件请求与在使用构造的数据流可访问的数据库数据源的数据库中的输入数据的指示相关联。

[0144] 已经关于特定实施例描述了本文中给出的实施例,这些特定实施例在所有方面都旨在是说明性的而不是限制性的。在不脱离本发明范围的情况下,替选实施例对于本发明所属领域的普通技术人员而言将变得显而易见。

[0145] 从前述内容可以看出,本发明适于实现上述所有的目标和目的、以及其它显而易见的且结构所固有的优点。

[0146] 应当理解,某些特征和子组合是有用的,并且可以在不参考其它特征或子组合的

情况下使用。这是权利要求的范围所设想的并且在权利要求的范围内。

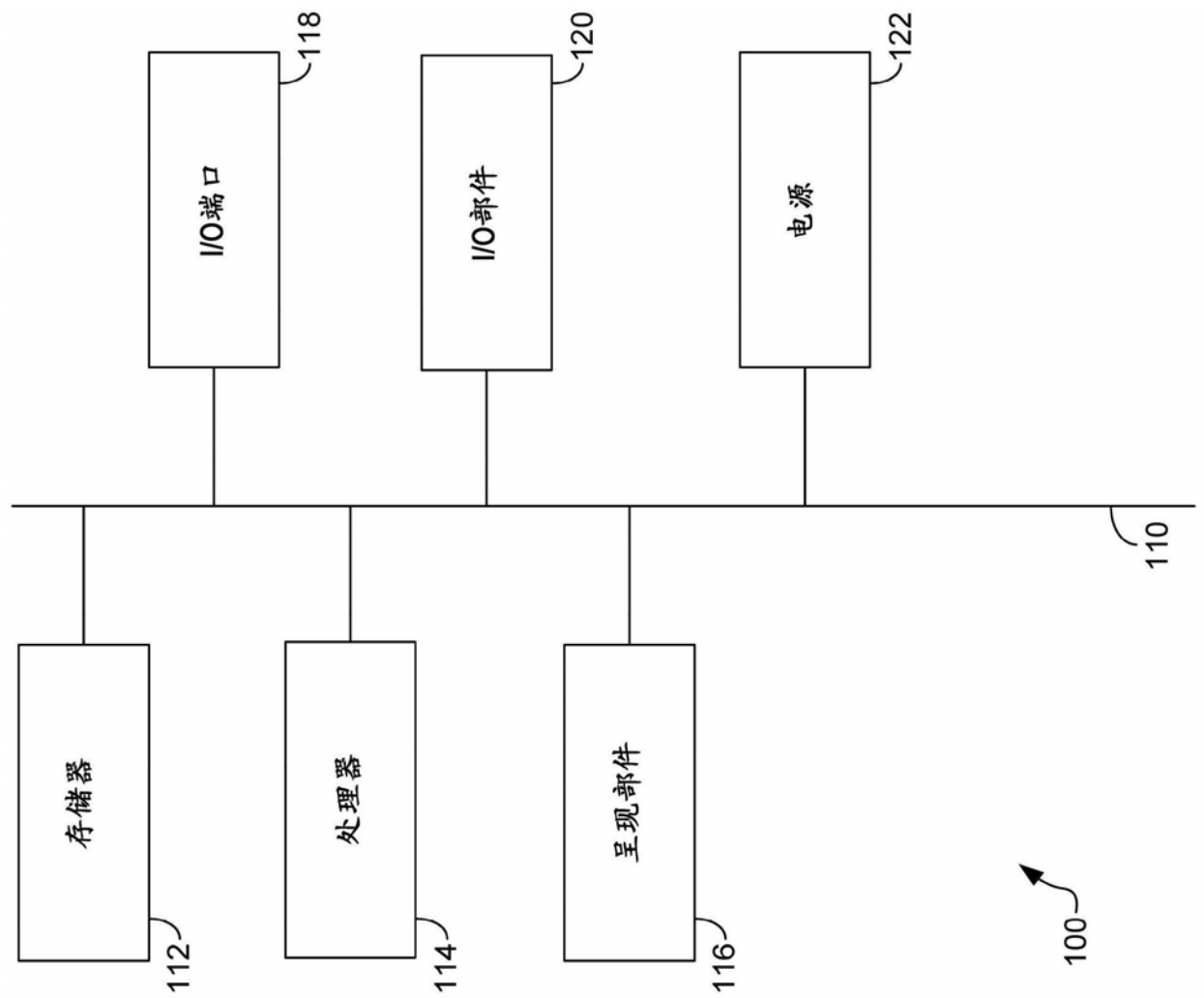


图1

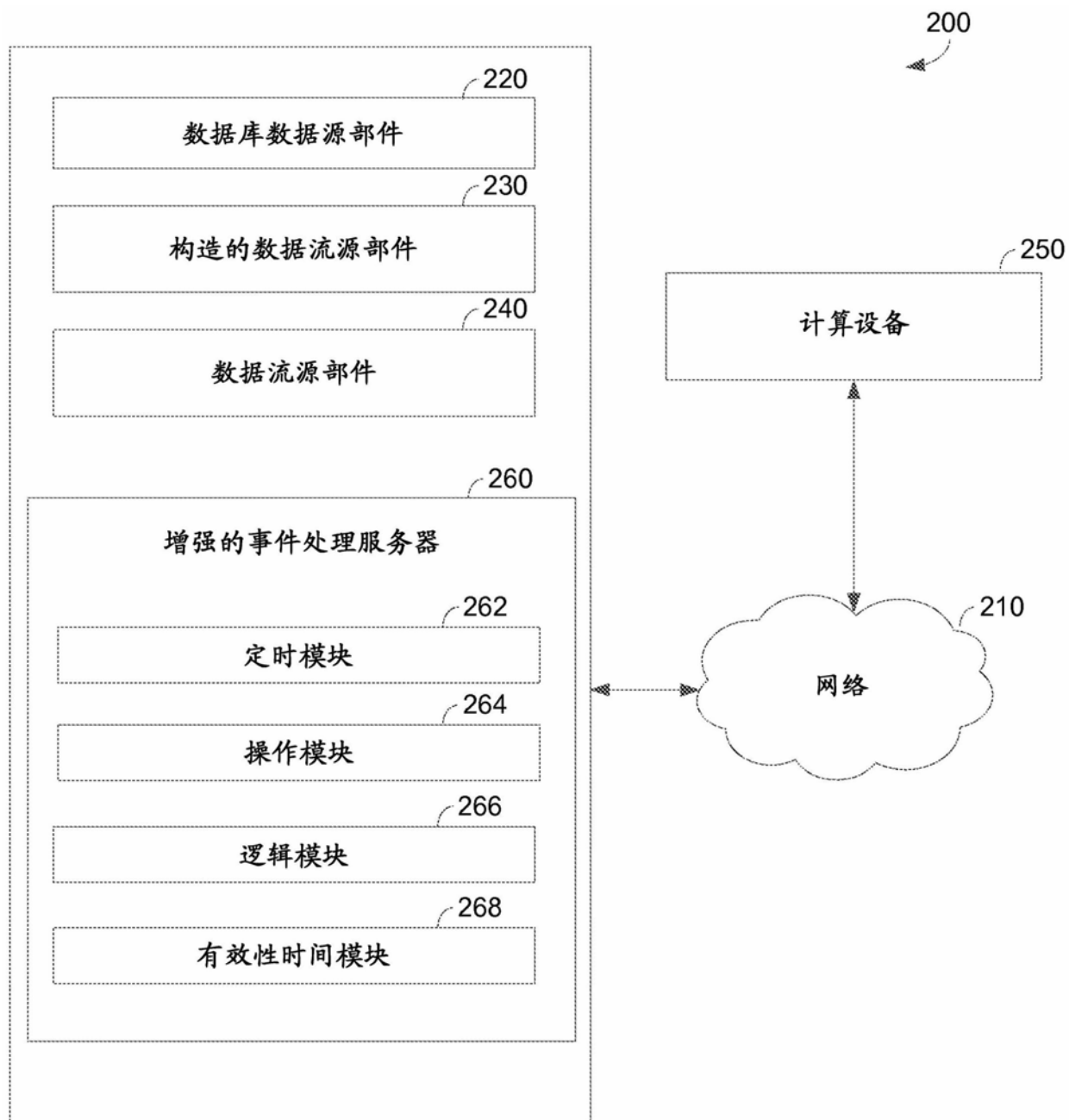


图2

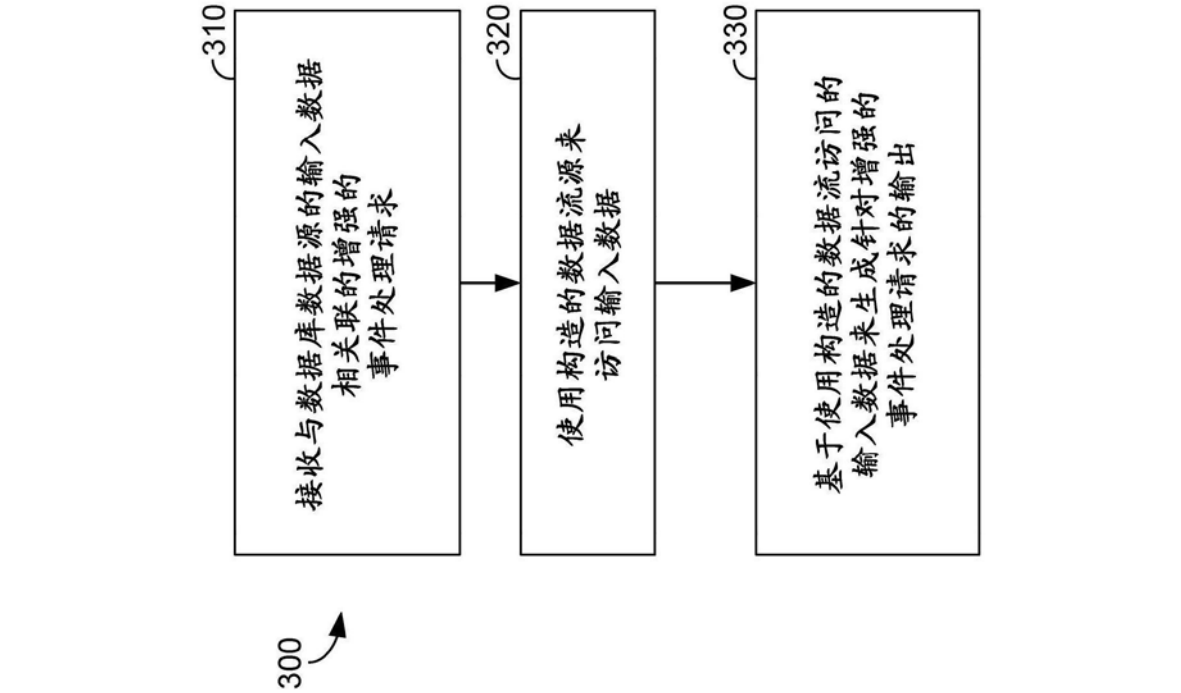


图3

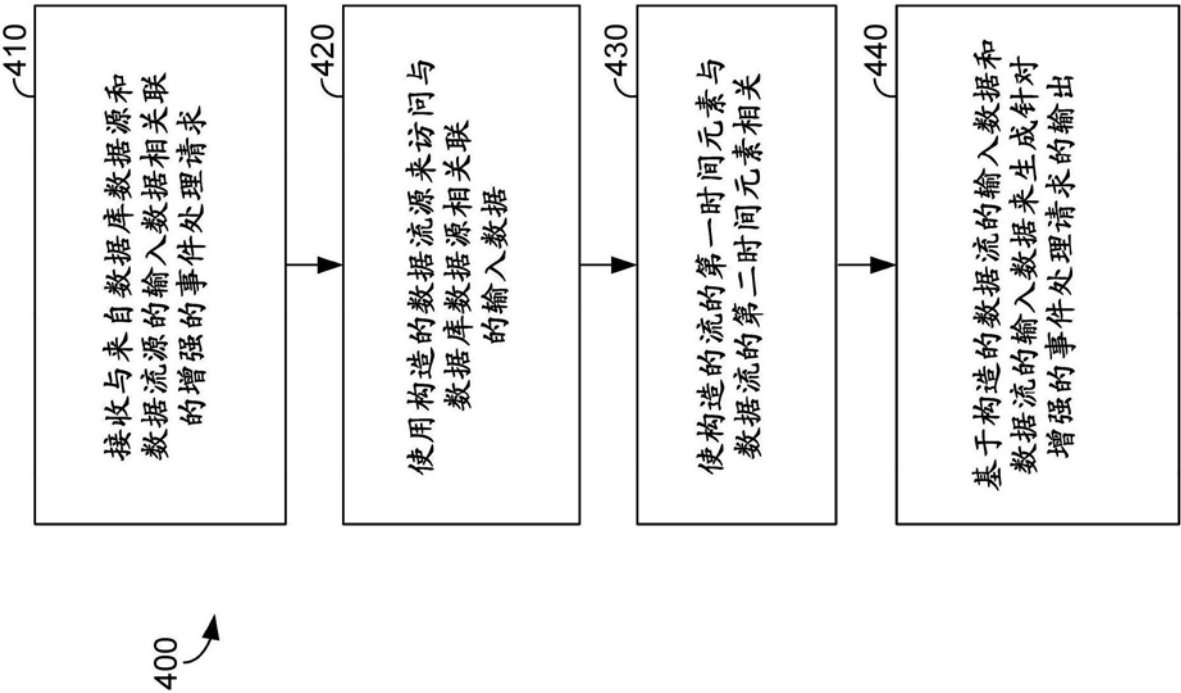


图4



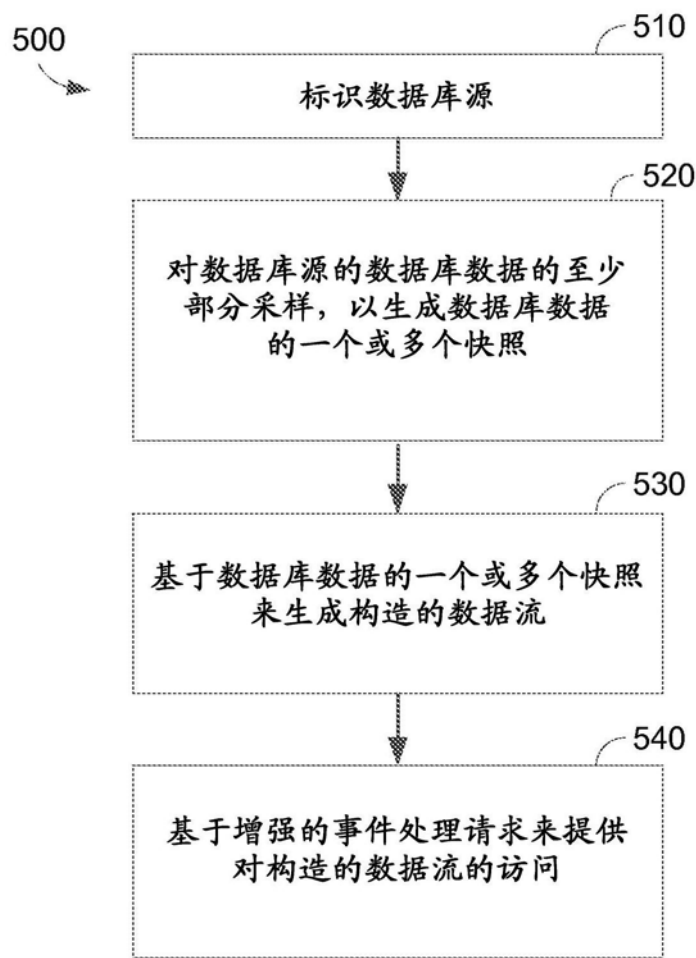


图5