



## (12) 发明专利

(10) 授权公告号 CN 104919445 B

(45) 授权公告日 2022.01.11

(21) 申请号 201380068894.1

(22) 申请日 2013.12.09

(65) 同一申请的已公布的文献号  
申请公布号 CN 104919445 A

(43) 申请公布日 2015.09.16

(30) 优先权数据  
61/735,451 2012.12.10 US  
61/751,814 2013.01.11 US  
13/958,037 2013.08.02 US

(85) PCT国际申请进入国家阶段日  
2015.06.29

(86) PCT国际申请的申请数据  
PCT/US2013/073899 2013.12.09

(87) PCT国际申请的公布数据  
W02014/093232 EN 2014.06.19

(73) 专利权人 起元技术有限责任公司  
地址 美国马萨诸塞州

(72) 发明人 S.斯塔德 J.S.沃利三世  
A.韦斯曼

(74) 专利代理机构 北京林达刘知识产权代理事  
务所(普通合伙) 11277  
代理人 刘新宇

(51) Int.Cl.  
G06F 16/23 (2019.01)  
G06F 16/2453 (2019.01)  
G06F 16/25 (2019.01)

(56) 对比文件  
US 2012/0005144 A1,2012.01.05  
CN 101702942 A,2010.05.05  
US 7215637 B1,2007.05.08

审查员 魏留强

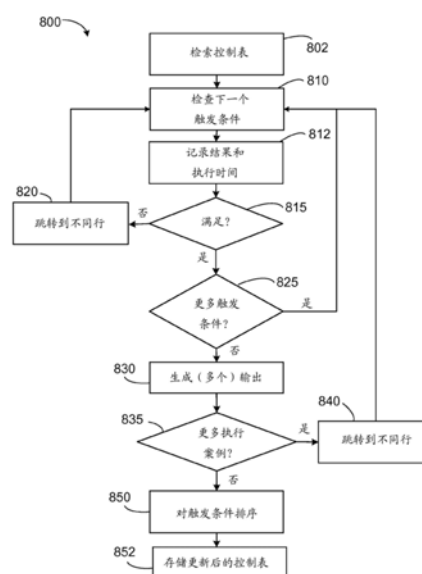
权利要求书6页 说明书19页 附图11页

### (54) 发明名称

变换生成系统

### (57) 摘要

本说明书描述了基于规则集生成变换的相关技术。一般地,在本说明书中描述的一个方面可以在包括接收包括执行案例的规则集的方法中实施,其中规则集中的至少一个执行案例包括一个或多个触发条件以及在一个或多个触发条件全部满足时将要生成的输出的规范。该方法还可以包括生成控制结构,其包括对应于规则集中的一个或多个执行案例的行序列。每一行可以包括一个或多个触发条件的序列以及指定相应执行案例的输出的信息。对于触发条件中的至少一个,当触发条件不通过时,控制结构可以引导处理过程跳过行序列中的至少一行。



1. 一种由一个或多个数据处理设备执行的用于对变换数据的规则集编码的方法,包括:

接收包括执行案例序列的规则集,执行案例序列中的至少一个执行案例包括一个或多个触发条件以及在一个或多个触发条件全部满足时将生成的输出的规范;

至少部分地基于所述规则集生成控制结构,所述控制结构包括与规则集中的一个或多个执行案例相对应的行的序列,

其中,每一行包括:

一个或多个触发条件的序列;以及

信息,所述信息用于指定与该行所对应的执行案例的输出,

其中,生成的控制结构被配置为在由至少一个处理器执行时,当某一行的触发条件中的一个不通过时,将处理过程从该行引导到不同的行,并且对于控制结构的第一行中的触发条件中的至少一个,当所述第一行的所述触发条件中的至少一个不通过时,引导处理过程至少跳过行序列中的第二行,所述第二行与包括所述第一行中的至少一个不通过的该触发条件的执行案例相对应;以及

存储或发送所述控制结构。

2. 如权利要求1所述的方法,还包括:

接收输入数据;

以使用控制结构确定的顺序,针对输入数据检查触发条件;以及

基于由控制结构指定的输出,存储或发送数据。

3. 如权利要求1所述的方法,其中所述行中的至少一行省略针对相应执行案例的触发条件,其中所省略的触发条件在执行案例序列中所述相应执行案例之前的执行案例中出现。

4. 如权利要求1所述的方法,其中所述行中的触发条件的序列是代码部分的序列,所述代码部分各自引导处理过程到规则集的唯一触发条件的列表中的触发条件。

5. 如权利要求1所述的方法,其中指定行中的输出的所述信息是代码部分,其引导处理过程到规则集的唯一输出的列表中的输出表达式。

6. 如权利要求1所述的方法,还包括:

基于在处理数据期间当序列中的触发条件不通过时处理将要被引导到的不同的行,对行的触发条件的序列排序。

7. 如权利要求1所述的方法,还包括:

基于触发条件的执行时间对行的触发条件的序列排序。

8. 如权利要求7所述的方法,还包括:

接收输入数据;

以使用控制结构确定的顺序,针对输入数据检查触发条件;

基于在输入数据下执行触发条件使用的时间,更新唯一触发条件的列表中的触发条件的执行时间;以及

基于更新后的执行时间,对控制结构中的行的触发条件的指针排序。

9. 如权利要求1所述的方法,还包括:

基于触发条件的不通过率,对行的触发条件的序列排序。

10. 如权利要求9所述的方法,还包括:

接收输入数据;

以使用控制结构确定的顺序,针对输入数据检查触发条件;

基于输入数据中的记录是否满足触发条件,更新唯一触发条件的列表中的触发条件的不通过率;以及

基于更新后的不通过率,对控制结构中的行的触发条件的指针排序。

11. 如权利要求1所述的方法,其中,所述控制结构的行还包括代码部分,在行的所有触发条件都满足时,所述代码部分引导处理过程到控制结构的接下来将要处理的不同的行。

12. 如权利要求1所述的方法,其中,通过图形用户界面指定规则集。

13. 如权利要求1所述的方法,其中,所述规则集中的执行案例的至少两个触发条件被合并,并通过控制结构中的单个触发条件来表示。

14. 如权利要求1所述的方法,其中,所述规则集中的不同的执行案例的至少两个输出被合并,并通过控制结构的行中的单个输出表达式来表示。

15. 如权利要求1所述的方法,其中,所述控制结构是节点对应于控制结构的行中的触发条件和输出表达式的有向无环图。

16. 一种存储用于对变换数据的规则集编码的软件的计算机可读介质,所述软件包括指令,用于使得计算系统:

接收包括执行案例序列的规则集,执行案例序列中的至少一个执行案例包括一个或多个触发条件以及在一个或多个触发条件全部满足时将生成的输出的规范;

至少部分地基于所述规则集生成控制结构,所述控制结构包括与规则集中的一个或多个执行案例相对应的行的序列,

其中,每一行包括:

一个或多个触发条件的序列;以及

信息,所述信息用于指定与该行所对应的执行案例的输出,

其中,生成的控制结构被配置为在由至少一个处理器执行时,在未来变换输入数据的处理期间,当某一行的触发条件中的一个不通过时,将处理过程从该行引导到不同的行,并且对于控制结构的第一行中的触发条件中的至少一个,当所述第一行的所述触发条件中的至少一个不通过时,引导处理过程至少跳过行序列中的第二行,所述第二行与包括所述第一行中的至少一个不通过的该触发条件的执行案例相对应;以及

存储或发送所述控制结构。

17. 如权利要求16所述的介质,包括指令用于使得计算系统:

接收输入数据;

以使用控制结构确定的顺序,针对输入数据检查触发条件;以及

基于由控制结构指定的输出存储或发送数据。

18. 如权利要求16所述的介质,其中,所述行中的至少一行省略相应执行案例的触发条件,其中所省略的触发条件在执行案例序列中相应执行案例之前的执行案例中出现。

19. 如权利要求16所述的介质,其中,所述行中的触发条件的序列是代码部分序列,所述代码部分各自引导处理过程到规则集的唯一触发条件的列表中的触发条件。

20. 如权利要求16所述的介质,其中,指定行中的输出的所述信息是代码部分,其引导

处理过程到规则集的唯一输出的列表的输出表达式。

21. 如权利要求16所述的介质,包括指令用于使得计算系统:

基于在处理数据期间当序列中的触发条件不通过时处理将要被引导到的不同的行,对行的触发条件的序列排序。

22. 如权利要求16所述的介质,包括指令用于使得计算系统:

基于触发条件的执行时间,对行的触发条件的序列排序。

23. 如权利要求22所述的介质,包括指令用于使得计算系统:

接收输入数据;

以使用控制结构确定的顺序,针对输入数据检查触发条件;

基于利用输入数据执行触发条件使用的时间,更新唯一触发条件的列表中的触发条件的执行时间;以及

基于更新后的执行时间,对控制结构中的行的触发条件的指针排序。

24. 如权利要求16所述的介质,包括指令用于使得计算系统:

基于触发条件的不通过率,对行的触发条件的序列排序。

25. 如权利要求24所述的介质,包括指令用于使得计算系统:

接收输入数据;

以使用控制结构确定的顺序,针对输入数据检查触发条件;

基于输入数据中的记录是否满足触发条件,更新唯一触发条件的列表中的触发条件的不通过率;以及

基于更新后的不通过率,对控制结构中的行的触发条件的指针排序。

26. 如权利要求16所述的介质,其中,所述控制结构的行还包括代码部分,在行的所有的触发条件都满足时,其引导处理过程到控制结构的接下来将要处理的不同的行。

27. 如权利要求16所述的介质,其中,通过图形用户界面指定规则集。

28. 如权利要求16所述的介质,其中,规则集中的执行案例的至少两个触发条件被合并,并通过控制结构中的单个触发条件来表示。

29. 如权利要求16所述的介质,其中,规则集中的不同执行案例的至少两个输出被合并,并通过控制结构的行中的单个输出表达式来表示。

30. 如权利要求16所述的介质,其中,所述控制结构是节点对应于控制结构中的行的触发条件和输出表达式的有向无环图。

31. 一种对用于变换数据的规则集进行编码的计算系统,所述计算系统包括:

输入装置或端口,被配置为接收包括执行案例序列的规则集,执行案例序列中的至少一个执行案例包括一个或多个触发条件以及在一个或多个触发条件全部满足时将要生成的输出的规范;以及

至少一个处理器,被配置为执行操作,所述操作包括至少部分地基于所述规则集生成控制结构,所述控制结构包括与规则集中一个或多个执行案例相对应的行的序列,其中,每一行包括:

一个或多个触发条件的序列;以及

信息,所述信息用于指定与该行所对应的执行案例的输出,

其中,所生成的控制结构被配置为在由所述至少一个处理器执行时,在未来变换输入

数据的处理期间,在某一行的触发条件中的一个不通过时,将处理过程从该行引导到不同的行,以及对于控制结构的第一行中的触发条件中的至少一个,当所述第一行的所述触发条件中的至少一个不通过时,引导处理过程至少跳过行序列中第二行,所述第二行与包括所述第一行中的至少一个不通过的该触发条件的执行案例相对应;以及

输出装置或端口,被配置为发送控制结构。

32. 如权利要求31所述的系统,包括:

输入装置或端口,被配置为接收输入数据;

至少一个处理器,被配置为执行操作,所述操作包括以使用控制结构确定的顺序,针对输入数据检查触发条件;以及

输出装置或端口,被配置为基于由控制结构所指定的输出发送数据。

33. 如权利要求31所述的系统,其中,所述行中的至少一行省略相应执行案例的触发条件,其中所省略的触发条件在执行案例序列中相应执行案例之前的执行案例中出现。

34. 如权利要求31所述的系统,其中,所述行中的触发条件的序列是代码部分序列,所述代码部分各自引导处理过程到规则集的唯一触发条件的列表的触发条件。

35. 如权利要求31所述的系统,其中,指定行中的输出的所述信息是引导处理过程到规则集的唯一输出的列表的输出表达式的代码部分。

36. 如权利要求31所述的系统,其中,所述操作还包括:

基于在处理数据期间当序列中的触发条件不通过时处理将要被引导到的不同的行,对行的触发条件的序列排序。

37. 如权利要求31所述的系统,其中,所述操作还包括:

基于触发条件的执行时间对行的触发条件的序列排序。

38. 如权利要求37所述的系统,其中,所述操作还包括:

接收输入数据;

以使用控制结构确定的顺序,针对输入数据检查触发条件;

基于在输入数据下执行触发条件使用的时间,更新唯一触发条件的列表中的触发条件的执行时间;以及

基于更新后的执行时间,对控制结构中的行的触发条件的指针排序。

39. 如权利要求31所述的系统,其中,所述操作还包括:

基于触发条件的不通过率,对行的触发条件的序列排序。

40. 如权利要求39所述的系统,其中,所述操作还包括:

接收输入数据;

以使用控制结构确定的顺序,针对输入数据检查触发条件;

基于输入数据中的记录是否满足触发条件,更新唯一触发条件的列表中的触发条件的不通过率;以及

基于更新后的不通过率,对控制结构中的行的触发条件的指针排序。

41. 如权利要求31所述的系统,其中,所述控制结构的行还包括代码部分,当行的所有的触发条件都满足时其将处理引导到控制结构的接下来将要处理的不同的行。

42. 如权利要求31所述的系统,其中,通过图形用户界面指定规则集。

43. 如权利要求31所述的系统,其中,所述规则集中的执行案例的至少两个触发条件被

合并,并通过控制结构中的单个触发条件来表示。

44.如权利要求31所述的系统,其中,所述规则集中的不同的执行案例的至少两个输出被合并,并通过控制结构中的单个输出表达式表示。

45.如权利要求31所述的系统,其中,所述控制结构是节点对应于控制结构的行中的触发条件和输出表达式的有向无环图。

46.一种对用于变换数据的规则集进行编码的计算系统,该计算系统包括:

输入装置或端口,被配置为接收包括执行案例序列的规则集,执行案例序列中的至少一个执行案例包括一个或多个触发条件以及对在所述一个或多个触发条件全部满足时将要生成的输出的规范;以及

用于至少部分地基于所述规则集生成控制结构的装置,所述控制结构包括与规则集中的一个或多个执行案例相对应的行的序列,

其中,每一行包括:

一个或多个触发条件的序列;以及

信息,所述信息用于指定与该行所对应的执行案例的输出,

其中,所生成的控制结构被配置为在由至少一个处理器执行时,在未来变换输入数据的处理期间,在某一行的触发条件中的一个不通过时,将处理过程从该行引导到不同的行,以及对于控制结构的第一行中的触发条件中的至少一个,当所述第一行的触发条件中的至少一个不通过时,引导处理过程至少跳过行序列中的第二行,所述第二行与包括所述第一行中的至少一个不通过的该触发条件的执行案例相对应;以及

数据存储系统,被配置为存储控制结构。

47.如权利要求46所述的系统,包括:

输入装置或端口,被配置为接收输入数据;

至少一个处理器,被配置为执行操作,所述操作包括以使用控制结构确定的顺序,针对输入数据检查触发条件;以及

输出装置或端口,被配置为基于由所述控制结构所指定的输出,发送数据。

48.如权利要求46所述的系统,其中,所述行中的至少一行省略相应执行案例的触发条件,其中所省略的触发条件在执行案例序列中相应执行案例之前的执行案例中出现。

49.如权利要求46所述的系统,其中,所述行中的触发条件的序列是代码部分序列,所述代码部分各自引导处理过程到规则集的唯一触发条件的列表中的触发条件。

50.如权利要求46所述的系统,其中,指定行中的输出的所述信息是引导处理过程到规则集的唯一输出的列表中的输出表达式的代码部分。

51.如权利要求46所述的系统,包括:

用于基于在处理数据期间当序列中的触发条件不通过时处理将要被引导到的不同的行来对触发条件的序列排序的装置。

52.如权利要求46所述的系统,包括:

用于基于触发条件的执行时间对行的触发条件的序列排序的装置。

53.如权利要求52所述的系统,包括:

输入装置或端口,被配置为接收输入数据;

至少一个处理器,被配置为执行操作,所述操作包括:

以使用控制结构确定的顺序,针对输入数据检查触发条件;以及  
基于利用该输入数据执行触发条件使用的时间,更新唯一触发条件的列表中的触发条件的执行时间;以及

用于基于更新后的执行时间对控制结构中的行的触发条件的指针排序的装置。

54. 如权利要求46所述的系统,包括:

用于基于触发条件的不通过率,对行的触发条件的序列排序的装置。

55. 如权利要求54所述的系统,还包括:

输入装置或端口,被配置为接收输入数据;

至少一个处理器,被配置为执行操作,所述操作包括:

以使用控制结构确定的顺序,针对输入数据检查触发条件;以及

基于输入数据中的记录是否满足触发条件,更新唯一触发条件的列表中的触发条件的不通过率;以及

用于基于更新后的不通过率对控制结构中的行的触发条件的指针排序的装置。

56. 如权利要求46所述的系统,其中,所述控制结构的行还包括当行的所有的触发条件都满足时将处理引导到控制结构的接下来将要处理的不同的行的代码部分。

57. 如权利要求46所述的系统,其中,通过图形用户界面指定规则集。

58. 如权利要求46所述的系统,其中,所述规则集中的执行案例的至少两个触发条件被合并,并通过控制结构中的单个触发条件来表示。

59. 如权利要求46所述的系统,其中,所述规则集中的不同执行案例的至少两个输出被合并,并通过控制结构中的行的单个输出表达式表示。

60. 如权利要求46所述的系统,其中,所述控制结构是节点对应于控制结构的行中的触发条件和输出表达式的有向无环图。

## 变换生成系统

[0001] 相关申请的交叉引用

[0002] 本申请要求提交于2013年8月2日的美国专利申请序列号13/958,037的优先权,其要求提交于2013年1月11日的美国临时申请序列号61/751,814以及提交于2012年12月10日的美国临时申请序列号61/735,451的优先权,该申请的全部内容通过引用并入本文。

### 技术领域

[0003] 本说明书涉及基于规则集为数据生成变换的系统。

### 背景技术

[0004] 复杂的计算经常可以通过有向图表示为数据流(所谓的“数据流图”),其中计算组件与图的顶点关联并且组件之间的数据流对应于图的链接(弧、边)。所述组件可以包括在一个或多个输入端口接收数据、处理数据、以及从一个或多个输出端口提供数据的数据处理组件,以及作为数据流的源或汇点的数据集组件。在美国专利5,966,072“Executing computations expressed as graphs”中描述了实现这样基于图的计算的系统。

### 发明内容

[0005] 在一般的方面1中,一种由一个或多个数据处理设备执行的方法,用于对用于变换数据的规则集编码,包括:接收包括执行案例序列的规则集,执行案例序列中的至少一个执行案例包括一个或多个触发条件,以及对在一个或多个触发条件全部满足时将要生成的输出的规范;生成控制结构,其包括对应于规则集中一个或多个执行案例的行序列,每一行包括:一个或多个触发条件的序列以及指定相应执行案例的输出的信息,其中所生成的控制结构被配置为,在未来变换输入数据的处理期间,当触发条件中的一个不通过时,引导处理过程到不同的行继续,以及其中所生成的控制结构被配置为,对于控制结构中触发条件中的至少一个,当触发条件中的至少一个不通过时,控制结构将引导处理过程跳过行序列中的至少一行;以及存储或发送控制结构。

[0006] 根据方面1的方面2,还包括:接收输入数据;以使用控制结构确定的顺序,针对输入数据检查触发条件;并基于由控制结构所指定的输出存储或发送数据。

[0007] 根据方面1至方面2中的任一方面的方面3,其中,行中的至少一行省略相应执行案例的触发条件,其中所省略的触发条件在执行案例序列中的相应执行案例之前的执行案例中出现。

[0008] 根据方面1至方面3中的任一方面的方面4,其中,行中的触发条件序列是代码部分序列,所述代码部分各自引导处理过程到规则集的唯一触发条件的列表的触发条件。

[0009] 根据方面1至方面4中的任一方面的方面5,其中,所述规定行中的输出的信息是引导处理过程到规则集唯一输出的列表的输出表达式的代码部分。

[0010] 根据方面1至方面5中的任一方面的方面6,还包括:基于在数据的处理期间当序列中的触发条件不通过时处理将被引导到的不同的行,对行的触发条件序列排序。



[0011] 根据方面1至方面6中的任一方面的方面7,还包括:基于触发条件的执行时间对行的触发条件序列排序。

[0012] 根据方面1到方面7中的任一方面的方面8,,还包括:接收输入数据;以使用控制结构确定的顺序,针对输入数据检查触发条件;基于在该输入数据下执行触发条件使用的时间,更新唯一触发条件的列表中的触发条件的执行时间;以及基于更新后的执行时间,对控制结构中的行的触发条件指针排序。

[0013] 根据方面1至方面8中的任一方面的方面9,还包括:基于触发条件的不通过率,对行的触发条件序列排序。

[0014] 根据方面1至方面9中的任一方面的方面10,还包括:接收输入数据;以使用控制结构确定的顺序,针对输入数据检查触发条件;基于输入数据中的记录是否满足触发条件,更新唯一触发条件的列表中的触发条件的不通过率;以及基于更新后的不通过率,对控制结构中的行的触发条件指针排序。

[0015] 根据方面1至方面10中的任一方面的方面11,其中,控制结构的行还包括代码部分,当所有的触发条件都满足时,其将处理引导到控制结构的接下来将要处理的不同的行。

[0016] 根据方面1至方面11中的任一方面的方面12,其中,通过图形用户界面指定规则集。

[0017] 根据方面1至方面12中的任一方面的方面13,其中,规则集中的执行案例的至少两个触发条件被合并,并通过控制结构中的单个触发条件表示。

[0018] 根据方面1至方面13中的任一方面的方面14,其中,规则集中的不同的执行案例的至少两个输出被合并,并通过控制结构的行中的单个输出表达式表示。

[0019] 根据方面1至方面14中的任一方面的方面15,其中,控制结构是节点对应于控制结构的行中的触发条件和输出表达式的有向无环图。

[0020] 在一般的方面16中,一种包括数据处理设备以及耦合于该数据处理设备的存储器的系统。存储器具有指令存储于其上,当数据处理设备执行指令时使得数据处理设备执行操作,包括:接收包括执行案例序列的规则集,执行案例序列中的至少一个执行案例包括一个或多个触发条件,以及对在一个或多个触发条件全部满足时生成的输出的规范。所述操作还可以包括:生成包括对应于规则集中一个或多个执行案例的行序列的控制结构,每一行包括一个或多个触发条件的序列以及指定相应执行案例的输出的信息,其中生成的控制结构被配置为,在未来变换输入数据的处理期间,当触发条件中的一个不通过时,引导处理过程到不同的行继续,并且其中所生成的控制结构被配置为,对于控制结构中触发条件中的至少一个,当触发条件中的至少一个不通过时,则控制结构将引导处理过程跳过行序列中的至少一行。所述操作还可以包括存储或发送控制结构。

[0021] 根据方面16的方面17,操作还包括:接收输入数据;以使用控制结构确定的顺序,针对输入数据检查触发条件;并基于控制结构指定的输出存储或发送数据。

[0022] 根据方面16至方面17中任一方面的方面18,其中,行中的至少一行省略相应执行案例的触发条件,其中所省略的触发条件在执行案例序列中相应执行案例之前的执行案例中出现。

[0023] 根据方面16至方面18中的任一方面的方面19,其中,行的触发条件序列是代码部分序列,所述代码部分各自引导处理过程到规则集的唯一触发条件的列表中的触发条件。

[0024] 根据方面16至方面19中的任一方面的方面20,其中指定行中的输出的信息是将处理引导到规则集中唯一输出的列表的输出表达式的代码部分。

[0025] 根据方面16至方面20中的任一方面的方面21,所述操作还包括:基于在数据的处理期间,当序列中的触发条件不通过时处理将被引导到的不同的行,对行的触发条件序列排序。

[0026] 根据方面16至方面21中的任一方面的方面22,操作还包括:基于触发条件的执行时间,对行中的触发条件序列排序。

[0027] 根据方面16至方面22中的任一方面的方面23,操作还包括:接收输入数据;以使用控制结构确定的顺序,针对输入数据检查触发条件;基于在该输入数据下执行触发条件使用的时间,更新唯一触发条件的列表中的触发条件的执行时间;以及基于更新后的执行时间对控制结构中的行的触发条件的指针排序。

[0028] 根据方面16至方面23中的任一方面的方面24,操作还包括:基于触发条件的不通过率,对行的触发条件序列排序。

[0029] 根据方面16至方面24中的任一方面的方面25,操作还包括:接收输入数据;以使用控制结构确定的顺序,针对输入数据检查触发条件;基于输入数据中的记录是否满足触发条件,更新唯一触发条件的列表中的触发条件的不通过率;以及基于更新后的不通过率对控制结构的行的触发条件指针排序。

[0030] 根据方面16至方面25中的任一方面的方面26,其中,控制结构的行还包括代码部分,当所有的触发条件都满足时,其将处理引导到接下来处理的控制结构的不同的行。

[0031] 根据方面16至方面26中的任一方面的方面27,其中,通过图形用户界面指定规则集。

[0032] 根据方面16至方面27中的任一方面的方面28,其中,规则集中的执行案例的至少两个触发条件被合并,并通过控制结构中的单个触发条件表示。

[0033] 根据方面16至方面28中的任一方面的方面29,其中,规则集中的不同的执行案例的至少两个输出被合并,并通过控制结构的行中的单个输出表达式表示。

[0034] 根据方面16至方面29中的任一方面的方面30,其中,控制结构是节点对应于控制结构的行中的触发条件和输出表达式的有向无环图。

[0035] 在一般的方面31中,一种存储包括处理设备可以执行的指令的软件的计算机可读存储介质,在执行指令的时候,使得处理设备执行操作,包括:接收包括执行案例序列的规则集,在执行案例序列中的至少一个执行案例包括一个或多个触发条件以及对在一个或多个触发条件全部满足时将要生成的输出的规范。所述操作还可以包括:生成控制结构,其包括对应于规则集中一个或多个执行案例的行序列,每一行包括一个或多个触发条件和指定相应执行案例的输出的信息,其中生成的控制结构被配置为,在未来变换输入数据的处理期间,当触发条件中的一个不通过时,引导处理过程到不同的行继续,以及其中所生成的控制结构被配置为,对于控制结构中触发条件中的至少一个,当触发条件中的至少一个不通过时,控制结构将引导处理过程跳过行序列中的至少一行。这些操作还可以包括存储或发送控制结构。

[0036] 根据方面31的方面32,操作还包括:接收输入数据;以使用控制结构确定的顺序,针对输入数据检查触发条件;并基于控制结构所指定的输出存储或发送数据。

[0037] 根据方面31至方面32中的任一方面的方面33,其中,至少一行省略相应执行案例的触发条件,所省略的触发条件在执行案例序列中相应执行案例之前的执行案例中出现。

[0038] 根据方面31至方面33中的任一方面的方面34,其中,行中的触发条件序列是代码部分序列,所述代码部分各自将引导处理过程到规则集的唯一触发条件的列表中的触发条件。

[0039] 根据方面31至方面34中的任一方面的方面35,其中,指定行中的输出的信息是将处理引导到规则集中的唯一输出的列表的输出表达式的代码部分。

[0040] 根据方面31至方面35中的任一方面的方面36,操作还包括:基于在数据的处理期间,当序列中的触发条件不通过时处理将被引导到的不同的行,对行的触发条件序列排序。

[0041] 根据方面31至方面36中任一方面的方面37,操作还包括:基于触发条件的执行时间,对行中的触发条件序列排序。

[0042] 根据方面31至方面37中的任一方面的方面38,操作还包括:接收输入数据;以使用控制结构确定的顺序,针对输入数据检查触发条件;基于在该输入数据下执行触发条件使用的时间,更新唯一触发条件的列表中的触发条件的执行时间;以及基于更新后的执行时间对控制结构的行的触发条件指针排序。

[0043] 根据方面31至方面38中的任一方面的方面39,所述操作还包括:基于触发条件的不通过率,对行的触发条件序列排序。

[0044] 根据方面31至方面39中的任一方面的方面40,所述操作还包括:接收输入数据;以使用控制结构确定的顺序,针对输入数据检查触发条件;基于输入数据中的记录是否满足触发条件,更新唯一触发条件的列表中的触发条件的不通过率;以及基于更新后的不通过率对控制结构的行的触发条件的指针排序。

[0045] 根据方面31至方面40中的任一方面的方面41,其中控制结构的行还包括代码部分,当所有的触发条件都满足时,其将处理引导到接下来处理的控制结构的不同的行。

[0046] 根据方面31至方面41中的任一方面的方面42,其中,通过图形用户界面指定规则集。

[0047] 根据方面31至方面42中的任一方面的方面43,其中,规则集中的执行案例的至少两个触发条件被合并,并通过控制结构中的单个触发条件表示。

[0048] 根据方面31至方面43中的任一方面的方面44,其中,规则集中的不同的执行案例的至少两个输出被合并,并通过控制结构的行中的单个输出表达式表示。

[0049] 根据方面31至方面44中的任一方面的方面45,其中,控制结构是节点对应于控制结构中的触发条件和输出表达式的有向无环图。

[0050] 根据方面1至方面15中的任一方面的方面46,其中,所述控制结构是在多个处理设备并行执行的变换的一部分。

[0051] 根据方面16至方面30中的任一方面的方面47,其中,所述控制结构是在多个处理设备上并行执行的变换的一部分。

[0052] 根据方面31至方面45中的任一方面的方面48,其中,所述控制结构是在多个处理设备上并行执行的变换的一部分。

[0053] 在一个方面中,一般地,基于规则集生成变换的方法包括:接收包括执行案例序列的规则集,执行案例序列中的至少一个执行案例包括一个或多个触发条件,以及对在一个

或多个触发条件全部满足时将生成的输出的规范。所述方法还可以包括：生成包括对应于规则集中一个或多个执行案例的行序列的控制结构，每一行包括一个或多个触发条件和指定相应执行案例的输出的信息，其中生成的控制结构被配置为，在未来变换输入数据的处理期间，当触发条件中的一个不通过时，引导处理过程到不同的行继续，以及其中所生成的控制结构被配置为，对于控制结构中触发条件中的至少一个，当触发条件中的至少一个不通过时，控制结构将引导处理过程跳过行序列中的至少一行。该方法可以还包括存储或发送控制结构。

[0054] 一般地，在本说明书中描述的主题的一个方面可以实施在包括数据处理设备以及耦合到该数据处理设备的存储器的系统中。存储器具有指令存储于其上，当数据处理设备执行指令时使得数据处理设备执行操作，包括：接收包括执行案例序列的规则集，执行案例序列中的至少一个执行案例包括一个或多个触发条件，以及对在一个或多个触发条件全部满足时将生成的输出的规范。操作还可以包括：生成控制结构，其包括对应于规则集中一个或多个执行案例的行序列，每一行包括一个或多个触发条件以及指定相应执行案例的输出的信息，其中生成的控制结构被配置为，在未来变换输入数据的处理期间，当触发条件中的一个不通过时，引导处理过程到不同的行继续，以及其中所生成的控制结构被配置为，对于控制结构中触发条件中的至少一个，当触发条件中的至少一个不通过时，控制结构将引导处理过程跳过行序列中的至少一行。所述操作还可以包括存储或发送控制结构。

[0055] 一般地，本说明书中描述的主题的一个方面可以实施在存储包括可由处理设备执行的指令的软件的计算机可读存储介质中，当数据处理设备执行指令时使得数据处理设备执行操作，包括：接收包括执行案例序列的规则集，执行案例序列中的至少一个执行案例包括一个或多个触发条件，以及对在一个或多个触发条件全部满足时将生成的输出的规范。所述操作还可以包括：生成包括对应于规则集中一个或多个执行案例的行序列的控制结构，每一行包括一个或多个触发条件和指定用于相应执行案例下的输出的信息，其中生成的控制结构被配置为，在未来变换输入数据的处理期间，当触发条件中的一个不通过时，引导处理过程到不同的行继续，其中所生成的控制结构被配置为，对于控制结构中所述触发条件中的至少一个，当触发条件中的至少一个不通过时，控制结构将引导处理过程跳过行序列中的至少一行。所述操作还可以包括存储或发送控制结构。

[0056] 一般地，本说明书中描述的主题的一个方面可以实施在包括被配置为接收包括执行案例序列的规则集的输入设备或端口的系统中，执行案例序列中的至少一个执行案例包括一个或多个触发条件，以及对在一个或多个触发条件全部满足时将生成的输出的规范。该系统可以包括用于生成包括对应于规则集中一个或多个执行案例的行序列的控制结构的装置，每一行包括：一个或多个触发条件序列以及指定相应执行案例的输出的信息，其中生成的控制结构被配置为，在未来变换输入数据的处理期间，当触发条件中的一个不通过时，引导处理过程到不同的行继续，以及其中所生成的控制结构被配置为，对于控制结构中的触发条件中的至少一个，当触发条件中的至少一个不通过时，控制结构将引导处理过程跳过行序列中的至少一行。该系统可以包括被配置为存储控制结构的数据存储系统。

[0057] 一般地，本说明书中描述的主题的一个方面可以实施在包括配置成接收包括执行案例序列的规则集的输入设备或端口的系统中，执行案例序列中的至少一个执行案例包括一个或多个触发条件，以及对在一个或多个触发条件全部满足时将生成的输出的规范。该

系统可以包括执行操作的至少一个处理器,操作包括生成控制结构,其包括对应于规则集中一个或多个执行案例的行序列,每一行包括一个或多个触发条件和指定相应执行案例的输出的信息,其中生成的控制结构被配置为,在未来变换输入数据的处理期间,当触发条件中的一个不通过时,引导处理过程到不同的行继续,以及其中生成的控制结构被配置为,对于控制结构中触发条件中的至少一个,当触发条件中的至少一个不通过时,控制结构将引导处理过程跳过行序列中的至少一行。该系统可包括被配置为发送控制结构的输出设备或端口。

[0058] 方面可以包括以下特征中的一个或多个。可以接收输入数据,以及可以以使用控制结构确定的顺序,针对输入数据检查触发条件。可以存储或传输基于控制结构指定的输出的数据。行中的至少一个可以省略对应的执行案例的触发条件,其中所省略的触发条件在执行案例序列中相应执行案例之前的执行案例中出现。行中的触发条件序列可以是代码部分序列,所述代码部分各自将引导处理过程到规则集的唯一触发条件的列表中的触发条件。指定行中的输出的信息是将处理引导到规则集中的唯一输出的列表的输出表达式的代码部分。基于在数据的处理期间,序列中的触发条件不通过时处理将被引导到的不同的行,对行的触发条件序列排序。基于触发条件的执行时间,对行中的触发条件序列排序。可以基于在输入数据下执行触发条件使用的时间,对在唯一触发条件的列表中的触发条件的执行时间进行更新。可以基于更新后的执行时间,对控制结构中的行的触发条件的指针进行排序。可以基于触发条件的不通过率,对控制结构中的行的触发条件序列进行排序。可以基于输入数据中的记录是否满足触发条件,对唯一触发条件的列表中的触发条件的不通过率进行更新。可以基于更新后的不通过率,对控制结构的行的触发条件的指针进行排序。控制结构的行可以包括代码部分,当满足所有的触发条件时,其将处理引导到接下来处理的控制结构的不同行。该规则集可以通过图形用户界面指定。规则集中的执行案例的至少两个触发条件可以被合并,并通过控制结构中的单个触发条件表示。规则集中的不同的执行案例的至少两个输出可以被合并,并通过控制结构的行中的单个输出表达式表示。控制结构是节点对应于控制结构中触发条件和输出表达式的有向无环图。

[0059] 方面可以包括以下优点中的一个或多个。

[0060] 一些实现可以缩减基于规则集的变换的记录平均的处理时间。一些实现可以缩减基于规则集的变换的编译和启动时间。一些实现可以降低基于规则集的变换在编译过程中的内存使用。一些实现可以提供对表示物理实体——如飞机、汽车、计算机、建筑或其他基础设施——的数据的更有效的处理。一些实现可以降低用户处理海量数据时的认知负担,如:用户更容易地指定对海量数据(例如,数百万甚至数十亿的记录)的处理,从而使用户更容易地理解数据的处理,并因此施加用户的特定应用的领域知识,而无需担心寻找用于规则集规范的有效结构。

[0061] 本发明的其它特征和优点将从下面的描述以及权利要求书中变得明显。

## 附图说明

[0062] 图1是示例数据流图的原理图。

[0063] 图2A-2B是用于基于电子表格的规则条目的示例图形用户界面的图示。

[0064] 图3A示出了规则集的唯一触发条件的列表的示例。

- [0065] 图3B示出了规则集的唯一输出的列表的示例。
- [0066] 图4A-4B示出了基于表示为有向无环图的规则集的变换的示例控制结构。
- [0067] 图5示出了基于表示为显示表的规则集的变换的示例控制结构。
- [0068] 图6是用于执行基于图的计算的系统的框图。
- [0069] 图7是用于生成和执行基于规则集的变换的示例过程的流程图。
- [0070] 图8是用于执行基于规则集的变换的示例过程的流程图。

## 具体实施方式

[0071] 基于图的计算可以用于处理大型数据集。例如，信用卡发行商可以使用基于图的计算处理以百万计的信用卡的交易数据，以发行奖励点数并选择分支机构的产品用于奖励点数赎回的供应的展示。在另一个例子中，航空公司可以使用基于图的计算更新以百万计的航空公司乘客的飞行常客的里程账户。在另一个例子中，银行可以使用基于图的计算处理各种来源的消费者数据，并为特定消费者生成达到依赖于可用数据的最大限额贷款的贷款批准。在另一个例子中，航空公司可以使用基于图的计算跟踪和控制飞机机队的维护。在另一个例子中，汽车租赁公司可以使用基于图的计算跟踪和控制汽车车队的维护。在另一个例子中，在线服务提供商可以使用基于图的计算，以跟踪和控制的一个或多个网络服务器集群的维护和/或负载均衡。在另一个例子中，城市可以使用基于图的计算来基于道路交通数据控制道路交通信号灯。在另一个例子中，无线网络运营商可以使用基于图的计算控制个人无线通信装置（例如，智能电话或平板电脑设备）的无线网络接入和带宽分配。这些只是基于图的计算的应用的几个说明性的例子，并且许多其它应用是可能的。

[0072] 基于图的计算可以包括一个或多个对应于施加到输入数据中的集合中的记录（或其他数据元素）的变换的组件。一般地，变换处理输入记录以生成（例如，创建或更新）一个或多个输出记录。例如，信用卡发行商可以处理与信用卡账户记录相关的交易记录，对拟议的信用卡交易生成交易的批准或拒绝记录。变换如何处理输入记录以生成输出数据记录的细节可能是复杂的，并且关于特定的应用可能依赖于大量的领域知识。使得几乎没有软件开发或编码知识的用户（例如，操作员或开发者）能够基于其特定应用的知识容易地配置变换可能是有用的。基于通过易于理解的界面指定的规则生成高效变换的系统可能有助于降低此类用户的认知负担。

[0073] 在一些实现中，可以使得用户能够通过基于电子表格的图形用户界面（GUI）配置可能复杂的变换。例如，用户可以通过为每一个执行案例在基于电子表格的图形用户界面（GUI）中创建行，来指定包括执行案例的规则集。每一个执行案例可以包括称为触发条件的一个或多个条件，可以针对输入数据和一个或多个输出对其进行测试。当输入数据满足执行案例的所有触发条件时，可以生成该执行案例的一个或多个输出。例如，触发条件和输出可以对应于基于电子表格的GUI的列，该GUI使缺乏软件编码技能但具有领域知识并且擅长电子表格的用户能够指定规则集。在某些实现中，可以使用除基于电子表格的GUI之外的格式指定一个规则集（例如，以数据操纵语言（DML）格式）。

[0074] 一旦用户已经指定规则集，就可以基于规则集生成一个或多个能够处理输入数据记录的变换。该规则集可以包括在多个执行案例中出现的触发条件以及和/或输出。可以用这样方式生成变换：它利用规则集的冗余来减少编译变换并将其施加到输入数据记录时所

需的内存和处理时间。例如,当一个触发条件在多个执行案例中出现时,如果输入数据不符合该触发条件,则变换可以跳过检查剩余的具有该触发条件的执行案例。利用规则集中的冗余可以使得能够对大型输入数据集执行变换同时使用更少的内存和处理器周期。变换可以部分编码为控制变换的执行流的控制结构。控制结构可以包括称为行的触发条件的逻辑分组,以及分别对应于来自规则集的执行案例的输出。在变换的执行期间,当触发条件的评估通过时,评估行中的触发条件序列中的下一个触发条件。如果当用输入数据进行评估时行的最后一个触发条件通过,则生成至少部分是由行指定的相应的输出。如果用输入数据评估时触发条件不通过(例如,触发条件不满足),则控制结构可以引导处理过程到对应于不同的执行案例的不同的行。例如,在触发条件不通过时,通过跳转到对不在控制结构中的行序列的下一行的触发条件开始评估,执行流可以跳过若干执行案例。在一些实现中,控制结构引用唯一触发条件和唯一输出的列表,因此即使它们在整个规则集中多次出现,这些触发条件和输出也仅需在变换的编码中存储一次。

[0075] 在一些实现中,可以从针对相应的(多个)变换的控制结构中省略在规则集的规范中出现的触发条件和对触发条件的引用。例如,当相同的触发条件在规则集的别处出现且其将必然在对当前触发条件的实例进行评估之前被评估时,可以从控制结构的行中省略该触发条件。从行中省略触发条件可以降低用于相应变换的内存消耗和处理时间。

[0076] 在一些实现中,可以基于触发条件的参数对控制结构中的、或由控制结构的行引用的触发条件排序。可以执行排序以降低(多个)变换施加到的记录的平均处理时间。例如,可以基于触发条件的执行时间、评估时如果触发条件不通过则将被跳过的行数(例如,跳转的大小)、或先前已使用(多个)变换处理过的输入数据的不通过频率,对触发条件的序列进行排序。在一些实现中,可以对控制结构中的行的序列进行排序,以增加触发条件的跳转大小(例如,通过将具有共同触发条件的执行案例分组在一起)。在一些实现中,改变行中的触发条件序列和/或行的序列可以减少对具有类似于先前处理的数据的统计数据的大型数据集执行相应的变换所需的处理时间。

[0077] 在本说明书中,术语“规则集”是指一条或多条规则的集合,其中每一条规则由一个或多个执行案例构成。规则中的执行案例可能有一个排序,其可以确定在处理输入数据期间执行案例的评估顺序。“执行案例”是与一个或多个输出相关联的一个或多个触发条件的集合。“触发条件”是连同执行案例的任何其它的触发条件一起用于确定执行案例是否将引发的条件。当确定输入数据满足执行案例的所有触发条件时,则生成执行案例“引发”,意为执行案例的一个或多个输出。从它是基于指定为规则集的一部分的恒定参数而生成的意义上说,输出可以是静态的;或者从它是部分基于输入数据值和/或中间结果值而生成的意义上说,输出可以是动态的。规则可以是单发的(single-fire)或是多发的(multi-fire)。在单发规则中,只有第一个满足所有触发条件的执行案例引发。在多发规则中,检查所有构成规则的执行案例,并且所有的其各自的触发条件都满足的所有执行案例都生成输出。规则集中的多个规则可以顺序施加于数据流。例如,基于规则集中的第一条规则的变换可以从多个输入数据源获取记录并生成输出记录。这些输出记录又可以作为输入记录被传递给基于规则集的第二条规则的第二变换,以生成第二输出记录集。

[0078] 图1显示了一个或多个变换的示例数据流图100的原理图。数据通过数据流图100的一系列数据处理组件,数据流图100处理从一个或多个数据源到一个或多个数据汇点

的数据流。数据流图中的各种数据处理组件的任一个都可以通过在单个处理设备上的运行来实现,或者多个处理组件可以由在单个处理设备上的运行的一个或多个进程来实现。在一些实现中,输入数据记录可以在到达时被连续地处理(例如,响应于信用卡交易请求)。在一些实现中,数据可以在标识将要由数据流图100处理的输入数据记录集批量进行处理。

[0079] 数据流图100对批量数据的处理可以由用户输入或诸如定时器到期的某些其他事件启动。当开始批量数据的处理时,从一个或多个输入数据源读取输入数据记录。例如,可以从存储在诸如由数据存储组件110表示的计算机可读存储设备上的一个或多个文件中读取输入数据。也可以从诸如数据存储组件112表示的服务器上运行的数据库中读取输入数据记录。连接组件120从多个数据源顺序地读取数据(例如,记录)并将输入数据排列成离散的工作单元序列。工作单元可以表示诸如基于输入数据以预定格式存储的记录,或者可以表示诸如将要处理的交易。工作单元(例如,记录)顺序传递给数据流图中的下一个组件。

[0080] 示例数据流图100还包括变换组件130和140。由变换组件130执行的变换基于单发规则。也就是说,对每个工作单元(例如,来自连接过程的输入数据记录),只有一个执行案例会引发。变换组件130生成传递到下一个数据处理组件——在这种情况下,也即变换组件140——的输出记录。

[0081] 变换组件140执行的变换基于多发规则。由变换组件140生成的输出数据记录可以包括对应于每一个由输入工作单元引发的执行案例的输出值的列表。

[0082] 例如,变换组件130可以处理来自连接过程120的对应于来自多种数据源的信用卡交易记录的输入数据记录。变换组件130可以生成反映分配给信用卡作为记录中反映的交易的结果的奖励点数量的输出记录。在本例中,变换组件140可以处理分配给信用卡账户的奖励点数,以生成将要展示给相应的信用卡账户持有人的一个或多个产品供应。

[0083] 变换组件130和变换组件140可以组在一起成为更大的组件150,其对应于既包括对应于变换组件130的单发规则,又包括对应于变换组件140的多发规则的规则集。

[0084] 在工作单元经过数据流图的数据处理组件的时候,与每个工作单元关联的结果输出记录在被转移到数据汇点170之前传递给数据队列160并在其中累积。例如,数据汇点170可以是存储工作单元或基于工作单元的一些累积的输出的数据存储单元,或者数据汇点170可以是工作单元发布到其的队列,或是一些其它类型的用于接收最终结果的汇点。在一些实现中,当批次中的所有工作单元的结果已经被转移到数据汇点170的时候,批量处理结束。此时,可以终止数据流图中的组件。

[0085] 图2A是用于基于电子表格的规则条目的示例GUI200的图示。GUI 200已经由用户配置为指定基于与信用卡账户相关联的一个或多个记录中可用的交易数据,为信用卡账户确定“奖励点数”值的单发规则。GUI 200包括五行,指定了规则的五条规则案例。用户使用GUI 200为规则的执行案例指定触发条件和输出,并显示来自基于规则的变换的测试运行的元数据,用户可以使用该条规则以便于变换的调试或调谐。例如,图形用户界面200可以通过图6中的应用专家环境622呈现给用户。在一些实现中,通过GUI 200指定的规则集可以通过图6的执行环境604的网络接口接收。

[0086] 第一列204指定施加到反映信用卡账户平均每月费用的输入数据记录中的变量的触发条件。第三行第一列中的向下箭头206指示该第三执行案例的第一触发条件与第二执行案例的第一触发条件的触发条件是相同的。在一些实现中,用户可以手动选择向下箭头



图标以在基于电子表格的GUI 200的一个或多个单元格中插入向下箭头,从而指定向下箭头穿过的执行案例的触发条件和向下箭头的起点的正上方的触发条件的是相同的。在一些实现中,用户可以手动地在基于电子表格的GUI 200的相邻单元格中输入相同的触发条件,GUI 200可以自动识别触发条件是相同的,并生成向下箭头以指示此重复。

[0087] GUI 200的第二列208指定施加到从反映信用卡账户已激活的年数的信用卡账号输入数据记录生成的变量的触发条件。列208还包括两个向下箭头210,每一个指示一对匹配的触发条件。

[0088] GUI 200的第三列214指定当执行案例的所有的触发条件都被评估并发现满足时的生成的输出。列214还包括向下箭头216,指示前两个执行案例具有相同的输出。因为用户已通过图形用户界面200指定这是单发规则218,所以执行案例可以一次评估一个直到发现一个可以引发(例如,输入数据的工作单元满足执行案例的所有触发条件)。当执行案例引发时,将生成该执行案例指定的输出,并且基于该规则的变换将完成工作单元的处理。这时,变换可以开始处理数据流的下一个工作单元或终止。

[0089] GUI 200的最后一行将所有的触发条件单元格设置为关键字“任何”230,指示没有相应的触发条件,或等效地,指示对这些触发条件的评估总是为真。因为第五行没有触发条件并且最后被评估,所以指定第五执行案例的相应输出为缺省输出。

[0090] 第四列220显示基于规则的变换的跟踪测试运行的元数据,用户可以使用该规则以便于规则的调试或调谐。一般地,基于规则集可以生成对于至少三种不同的操作模式的不同的变换:生产模式、记录测试模式以及文件测试模式。生产模式变换,实现了规则集的逻辑并将其施加于输入数据,几乎没有任何额外的代码。对记录测试模式变换编码以使得可以步进执行对单独的工作单元(例如代表帐户、交易、飞机、汽车、计算机、移动设备、建筑物的输入数据记录等)的变换。除了编码规则集的基本逻辑外,记录测试模式变换可以包括生成详细的日志信息(例如,反映每一个输入字段、输出、触发条件的结果(真或假)、查找关键字、查找字段、哪一个执行案例引发以及一些中间参数的值)的代码。对文件测试模式变换编码,以使能施加该变换到大批量的测试数据并且审查为许多工作单元总结测试结果的日志。除了编码规则集的基本逻辑,文件测试模式变换还可以包括生成日志消息(例如,反映每个执行案例被引发的次数和/或每一个触发条件评估、通过、和/或不通过的次数)的代码。在本例中,列220为显示对于每一个执行案例在对数以千计的工作单元(例如,信用卡账户记录)进行处理的文件测试模式下运行期间该执行案例被引发的次数的计数。

[0091] 图2B是基于电子表格的规则条目的示例GUI 250的示意图。GUI 250已经由用户配置为指定多发规则,其基于使用基于在图2A中的示例GUI 200中指定的规则变换确定的“奖励点数”值以及与信用卡帐户相关联的其他数据,生成产品供应列表以呈现给信用卡账户持有人。GUI 250包括五行指定该规则的五个规则案例。用户使用GUI 250为规则的执行案例指定触发条件和输出,并且显示变换的测试运行的元数据,该变换基于可以由用户使用以便于调试或调谐变换的规则。例如,图形用户界面250可以通过图6中的应用专家环境622呈现给用户。在一些实现中,通过图形用户界面250指定的规则集可以通过图6的执行环境604的网络接口接收。

[0092] 第一列254指定施加到反映信用卡帐户“奖励点数”的变量的触发条件。这些“奖励点数”的值可以由基于图2A的图形用户界面200中指定的规则的变换在数据流中设定并写

入记录。贯穿第一列的第二行和第三行的向下箭头256指示第二执行案例和第三执行案例的第一触发条件和第一执行案例的第一触发条件的触发条件是相同的。

[0093] GUI 200的第二列和第三列258指定施加于从与信用卡账户相关联的,反映信用卡类型和信用卡帐户持有人的居住国人口的输入数据记录生成的其他变量的触发条件。列258还包括向下的箭头,每一个表示一对匹配的触发条件。

[0094] GUI 250的第四列264指定在对执行案例的所有触发条件进行评估并发现满足时生成的输出。因为用户已通过GUI 250指定这是多发规则268,所以可以评估所有执行案例。当执行案例引发时,将生成为执行案例指定的输出,并可以追加到当前工作单元的输出值列表。当已经评估了所有的执行案例时,变换就可以开始处理数据流中的下一个工作单元或终止。

[0095] 第五列270基于可以由用户使用以便于调试或调谐变换的规则,显示来自变换的跟踪测试运行的元数据。在本例中,列270为每一个执行案例显示该执行案例在文件测试模式变换执行期间引发的次数的计数,该变换期间对数千工作单元(例如,信用卡帐户的记录)进行了处理。

[0096] 图3A示出了规则集的唯一触发条件的列表300的示例。在一些实现中,当基于规则集生成一个或多个变换时,变换生成过程的一部分是生成唯一触发条件的列表,其可由用于变换的控制结构引用以节省内存,否则该内存可能用于为每一个它们出现于其中的执行案例存储触发条件的重复副本。在一些实现中,部分规则,包括列表300,可以基于规则规范生成变换,所述规则规范可以包括将指定规则的应用专家使用的数据名称转换为在变换编码中所使用的技术后端名称。例如,可以基于变量名的固定关键字或其他映射执行转换操作。在本例中,规则集包括通过在图2A的GUI 200中指定的规则(规则1)以及通过图2B的GUI 250指定的规则(规则2)。

[0097] 唯一触发条件的列表可以包括在规则集中一次或多次出现的每一个触发条件的单个副本。在本例中,将每一个触发条件编码为数据操纵语言(DML)的表达式。图3A的第一列310示出了触发条件的DML编码。触发条件的其他编码格式是可能的(例如,C、C++、Java或Cobol代码)。

[0098] 在一些实现中,唯一触发条件的列表还可以包括便于反向查找变换中触发条件的出现的使用指针列表。例如,图3A的第二列320示出了包括规则1和规则2的规则集的使用指针列表。每一个指针是三元组数,标识rule\_id(规则1或规则2)、ROW\_ID(例如,对应到特定的执行案例)以及COLUMN\_ID(例如,对应于执行案例中的触发条件序列位置)。在本例中,前五个唯一触发条件出现在规则1中,后七个唯一触发条件出现在规则2中。在一些实现中,唯一触发条件的列表不包括使用指针。

[0099] 在一些实现中,列表300可以包括用于缓存列表300中的表达式指定的复杂计算的结果的数据结构,因此可以重复使用结果,并且在变换执行期间发现输入数据是相同的时候,可以避免基于同样的输入重新计算该结果。

[0100] 唯一触发条件的列表300可以以种类繁多的格式或数据结构(例如,像链表或索引数组)存储。在本例中,唯一触发条件的列表300存储为索引数组(indexed array),以便于基于通过变换控制结构中的索引对触发条件的引用查找触发条件。

[0101] 图3B示出了规则集的唯一输出的列表的示例。在一些实现中,当基于规则集生成

一个或多个变换时,变换生成过程的一部分生成唯一输出的列表,其可以由用于变换的控制结构引用以节省内存,否则该内存可能用于为每一个它们出现于其中的执行案例,存储重复的输出副本。在本例中,规则集包括规则1和规则2。

[0102] 唯一输出的列表可以包括在规则集中一次或多次出现的每一个输出的单个副本。在本例中,每一个输出编码为DML表达式。图3B的第一列360示出了输出的DML编码。输出的其他编码格式是可能的(例如,C、C++、Java或Cobol代码)。

[0103] 在一些实现中,唯一输出的列表还可以包括便于反向查询变换中输出的出现的使用指针的列表。例如,图3B的第二列370示出了包括规则1和规则2的规则集的使用指针列表。在本例中,前四个唯一输出在规则1中出现,以及后五个唯一输出在规则2中出现。在一些实现中,唯一输出的列表中不包括使用指针。

[0104] 唯一输出的列表350可以存储于种类繁多的格式或数据结构(例如,作为一个链表或索引数组)中。在本例中,唯一输出的列表350存储于索引数组中,其由唯一触发条件的列表300共同索引(例如,在不相交的索引值区间中)。

[0105] 变换生成过程的一部分是生成控制结构,其控制变换的执行流,并且可以引用唯一触发条件和/或唯一输出的列表。在本说明书中,术语“控制结构”是指种类繁多的编码格式,并且不限于双索引二维数组(dual indexed two dimensional array)。例如,图4A和4B中的有向无环图以及图5中示出的双向链表是控制变换执行流的控制结构的例子。控制结构具有对应于执行案例的行。在变换控制结构的上下文中,术语“行”是指一个或多个触发条件以及一个或多个输出的逻辑分组,其中,当已经确定满足行中的所有的触发条件时,行的(多个)输出被执行。在此上下文中,术语“行”并不限于所显示的表格的水平子集。

[0106] 图4A-4B示出了基于规则集的变换的表示为有向无环图的示例控制结构400。在本例中,变换所基于的规则集包括规则1和规则2。在本例中,变换可以在数据流图100中的组件150中实现,或等效地,变换可以实现为在组件130中实现的基于规则1的第一变换以及在组件140中实现的基于规则2的第二变换的串联。在图4A和图4B中,每个节点由如相对图3A中描述的使用指针(rule\_id,ROW\_ID,COLUMN\_ID)标记。

[0107] 控制结构400中的节点对应于触发条件或输出。对应于触发条件的节点具有从节点引出的两条边。如果节点对应的触发条件在施加到输入数据时确定为真,则跟随两条边中的一条。如果节点相对应的触发条件在施加到输入数据时确定为假,则跟随两条边中的第二条。对应于输出的节点具有从节点引出的一条边,在生成相应的输出后总是跟随这条边。控制结构400的行可以包括一个或多个触发条件节点序列,由从前一个触发条件节点引出的“真”边依次连接。针对行的序列中的最后一个触发条件节点可通过它的“真”边连接到针对该行的一个或多个输出节点中的第一个。一行中的输出节点引出的边可以连接到该行的另外的输出节点。一行中的最后一个输出节点引出的边可以连接到对应于不同的执行案例或规则的另一行的节点,或者可以将执行流引导到对工作单元的变换处理的终点448。类似地,从触发条件节点引出的“假”边可以连接到对应于不同的执行案例或规则的另一行的节点,或者可以将执行流引导到对工作单元的变换处理的终点448。

[0108] 控制结构400对变换的执行流的起点402编码。变换通过评估第一执行案例的第一触发条件404启动。

[0109] 在一些情况下,“假”边可能会使得执行流跳转到与当前行不相邻的行,以此跳过

对一些执行案例的评估。在处理工作单元(例如,对应于可表示账户、交易、飞机、汽车、计算机、移动设备、建筑物等的输入数据记录)时,这样跳过行可以降低变换的复杂度并减少处理时间。

[0110] 在本例中,一些节点,包括触发条件节点420,都没有边连接到它们。缺乏边连接的节点反映了一个事实,即在变换所基于的规则集的逻辑下,相应的触发条件或输出将永远不需要处理。其结果是,可以从控制结构400中省略这些无连接的节点以及相应的触发条件。在图4B的控制结构450中示出了对不必要触发条件的省略。

[0111] 在一些情况下,节点可以合并并通过控制结构中的单个节点表示。例如,可以通过创建单个触发条件节点——其引导对由节点454引用的触发条件和由节点456引用的触发条件进行逻辑AND评估——来合并节点454和节点456。这是可能的,因为如果两个触发条件都为真,则接下来处理输出节点458,并且如果两个触发条件中的一个为假,则接下来处理节点460。以这种方式,可以合并规则集中对于执行案例的两个触发条件,并由控制结构450中的单个触发条件表示。如果合并的节点对应于这些触发条件的唯一实例,则触发条件也可以在唯一触发条件的列表的单个条目中合并。类似地,可以合并输出节点470和节点472,因为它们相应的输出总是一起生成。在本例中,通过合并输出节点470和节点472,它们相应的行可以合并成单个行,其对应于变换所基于的规则集上的两个执行案例。这些节点合并技术通过降低必须生成的和执行的控制流的代码量,可以降低存储器的使用以及变换的处理时间。

[0112] 图5示出了基于表示为所显示的表的规则集的变换的示例控制结构500。控制结构500可以存储为诸如双重链表(例如,行的链表,其中每行包括行的触发条件和输出链表)。在本例中,控制结构500控制基于包括规则1和规则2的规则集的变换的执行流。在本例中,控制结构500引用唯一触发条件的列表300中的触发条件以及唯一输出的列表350中的输出。

[0113] 图5中的第一列510描绘控制结构500的行的索引(例如,执行案例号)。可以使用这个索引引用控制结构中的行,以便于在执行流中跳转从而避免不必要的处理。第二列520描绘了每一行的触发条件序列。例如,控制结构500可以包括针对触发条件的代码部分,其引用唯一触发条件的列表300中的触发条件,并在触发条件被评估之后基于结果(例如,通过或不通过/真或假)进一步将引导变换的处理。如果触发条件不通过,则将变换处理引导到不同的行,其可能是序列中远离当前行的多行之外的行,或者是控制结构500中的行。例如,第三行中的代码部分522引导处理过程以评估唯一触发条件的列表300中的第三触发条件,并且如果触发条件不通过,则代码部分522将引导处理过程到控制结构500的第五行,因此跳过第四行。

[0114] 如果针对控制结构中的列出行的所有触发条件都通过,则引导处理过程到该行的输出。图5的第四列530描绘了对唯一输出的列表350的输出的引用。最后,在执行对于控制结构500中的行的输出之后,可以引导处理过程以继续处理控制结构500的另一行或结束对当前工作项的处理。图5的第三列540描绘了对控制结构500中的其他行的引用,所述其它行被控制结构的行索引(例如,执行案例号)引用。

[0115] 图6展示了可以使用变换生成技术的示例数据处理系统600。系统600包括数据源602,其可以包括数据的一个或多个诸如存储设备或到在线数据流的连接之类的数据源,每

一个数据源都可以以多种存储格式的任何一种存储数据(例如,数据库表、电子表格文件、平面文本文件或大型机使用的本机格式)。执行环境604包括变换生成模块606和执行模块612。执行环境604可以托管在一台或多台在诸如UNIX操作系统之类的合适的操作系统控制之下的通用计算机之上。例如,执行环境604可以包括多节点并行计算环境,其包括使用多个中央处理单元(CPU)的计算机系统的配置,无论是本地的(例如,诸如SMP计算机的多处理器系统)、或本地分布(例如,耦合为群集或MPP的多个处理器),或远程的、或远程分布(例如,多个通过局域网络(LAN)和/或广域网(WAN)耦合的处理器),或它们的任何组合。

[0116] 变换生成模块606接收包括执行案例序列的规则集,执行案例的至少一个包括一个或多个触发条件以及对在一个或多个触发条件都满足时将生成的输出的规范。规则集还可以包括执行案例,其包括一个或多个输出但缺乏触发条件,或者等同地,具有总是评估为真的触发条件(例如,在示例规则1中的默认的执行案例)。例如,规则集可以由通过应用专家环境622访问该执行环境604的用户626指定。在一些实现中,应用专家环境包括运行在远程计算设备上的客户端软件,其提供给用户626用于指定规则集的GUI。例如,应用专家环境622可以使得用户能够在基于电子表格的GUI中指定规则集,如相对图2A和图2B所述。在一些实现中(未示出),开发环境618和应用程序专家环境622可以被合并,并且可由编辑数据流图以及对于在那些数据流图中实例化的变换的规则集规范的单个用户或用户组访问。

[0117] 变换生成模块606基于收到的规则集生成一个或多个变换。可以生成控制一个或多个变换的执行流的控制结构。控制结构可以包括对应于规则集中的执行案例的行。每一行可以包括一个或多个触发条件序列,以及指定对于执行案例的输出的信息。在处理变换数据期间当触发条件不通过时,一些触发条件可引导处理过程在当前行之下的多行的不同行继续,因此跳过对一些执行案例的评估,以缩减变换所需的处理时间。

[0118] 控制结构可以连同任何其它对所生成的(多个)变换进行编码的数据一起存储或发送。例如,可以在数据存储系统616中存储控制结构的所生成的变换。在一些实现中(未示出),该变换生成模块606可以被实现为应用程序专家环境622的一部分,并且编码所生成的(多个)变换的数据——包括控制结构,可以从运行应用专家环境622的远程计算设备发送到执行环境604。

[0119] 执行模块612使用由变换生成模块606生成的一种或多种变换以处理输入的数据记录,并生成供传输或存储的输出数据记录。一旦已经生成了基于图的计算的一个或多个变换,则包括(多个)变换的计算可以由执行模块612施加到输入数据。执行模块612从数据源602中读取数据,并生成可以存储在执行环境604可访问的数据存储系统616中的输出数据记录614。例如,数据存储系统616可以包括数据库服务器和/或运行版本控制应用的服务器。

[0120] 在一些实现中,执行模块还记录在输入数据记录处理期间评估的触发条件的执行时间和/或结果。例如,变换生成模块606可以使用这些日志,通过改变控制结构的触发条件的顺序更新变换。

[0121] 提供数据源602的存储设备可以相对执行环境604是本地的,例如,存储在连接到运行执行环境604的计算机的存储介质上(例如,硬盘驱动器608),或者可以相对执行环境604是远程的,例如,托管在通过远程连接与运行执行环境604的计算机通信的远程系统(例如,主机610)上。

[0122] 数据存储系统616还可以由开发环境618访问,其中,开发者620能够创建和管理基于图的计算,其可以包括对应于变换的组件。这些变换的操作可以由用户(例如,用户626)配置,用户可能有该基于图的计算将施加到的应用的专门知识。在一些实现中,开发环境618是用于开发数据流图应用的系统,数据流图包括通过顶点(表示组件或数据集)之间的有向链接(表示工作元素的流)连接的顶点。例如,这样的环境在题为“Managing parameters for graph-based applications”的美国公开号2007/0011668中描述得更加详细,以引用方式将其并入本文。用于执行这样的基于图的计算的系统在美国专利5,566,072“EXECUTING COMPUTATIONS EXPRESSED AS GRAPHS”中描述,通过引用将其并入本文。根据本系统制成的数据流图提供用于将信息传入或移出由图组件表示的单个进程、用于在进程之间传递信息、以及用于规定进程的运行顺序的方法。该系统包括选择进程间通信方法的算法(例如,根据图的链接的通信路径可以使用TCP/IP或UNIX域套接字,或使用在进程之间传递数据的共享内存)。

[0123] 执行模块612可以从包括不同形式的数据库系统的多种类型的系统中接收数据。数据可以组织为具有用于各自字段(也称为“属性”或“列”)的值的记录,可能包括空值。当第一次从数据源读取数据时,执行模块612通常以关于该数据源中的记录的一些初始格式信息开始。在某些情形下,数据源的记录结构可能是最初未知的,并且可以在分析数据源之后确定。关于记录的初始信息可以包括表示不同值的比特位数、记录中字段的顺序以及比特表示的值的类型(例如,字符串、有符号/无符号整数)。

[0124] 图7示出了示例变换生成和执行过程700的流程图。例如,过程700的可以由图6中的执行环境604执行。

[0125] 当接收到规则集时702,可以启动700。规则集可以包括执行案例序列。规则集中的执行案例可以包括一个或多个触发条件以及对在执行案例的一个或多个触发条件都满足时将要生成的输出的规范。在一些实现中,通过用户界面(例如,文本文件编辑器、基于电子表格的GUI、或一些其他类型的GUI)接收规则集,用户界面包括与接收规则集的处理装置本地连接的硬件(例如,计算机显示器以及键盘和/或鼠标)。例如,规则集可以通过图6的执行环境604的用户接口来接收。在一些实现中,由服务器通过网络接口从远程处理设备接收规则集。例如,可以通过执行环境604的网络接口从运行应用专家环境622的远程处理设备接收规则集。

[0126] 基于接收到的规则集,生成包括控制结构的变换704。在变换施加到输入数据时,控制结构可以用于控制变换的执行流。控制结构可以引用生成的变换的其它部分(例如,唯一触发条件的列表和/或唯一输出的列表)。控制结构可以用多种格式编码。控制结构格式的例子包括编译为用于执行环境中的一个或多个处理器的可执行文件、包括可由计算机语言解释器或编译器在运行时编译的文本的文本文件、包括可被解释或编译的代码部分的文本记录的双索引(二维)数组、图4A和图4B的有向无环图以及图5中示出的双向链表,等等。

[0127] 所生成的控制结构可以包括对应于规则集中一个或多个执行案例的行。控制结构中的行可以包括一个或多个触发条件的序列以及指定执行案例的输出的信息。控制结构的行可以是对应于规则集中一个或多个触发条件以及执行案例的一个或多个输出的执行控制流代码的逻辑分组。在一些实现中,控制结构的行包括代码部分序列(例如,存储为链表),其引导变换的执行,以使得处理设备检查触发条件或生成输出。基于当前触发条件的

评估结果,触发条件的代码部分也可以引导变换执行到对应于不同的触发条件或输出的另一代码部分。当触发条件在变换数据处理期间不通过时,控制结构中的触发条件可以引导处理过程以在行序列中距离当前行多行之外的不同的行继续。以这种方式,可以跳过一些行的触发条件和/或输出的执行,以减少工作单元(例如,输入数据记录)的处理时间。

[0128] 在一些实现中,可以基于当序列中的触发条件在数据处理期间不通过时处理将被导向到其的不同行的行号,对行的触发条件序列进行排序。例如,在变换生成期间可以通过以下方式对行的触发条件序列进行排序:通过将在不通过条件下会在控制结构中导致大的跳转的触发条件放在行的触发条件序列中更前的位置,同时将在不通过条件下会在控制结构中导致小的跳转的触发条件放在行的触发条件序列中靠后的位置。

[0129] 在一些实现中,控制结构的行通过以下方式生成704:省略也在执行案例序列中紧接着对应的执行案例之前的执行案例中出现的相应的执行案例的触发条件。这一省略可以降低变换的编译和/或执行的内存需求。

[0130] 在一些实现中,作为变换的一部分,还生成规则集的唯一触发条件的列表704。包括规则1和规则2的规则集的唯一触发条件的列表300是可生成的唯一触发条件的列表的例子。例如,当行中的触发条件序列存储为代码部分序列时,代码部分中的一个可以引导处理过程到在规则集的唯一触发条件的列表中编码的触发条件。

[0131] 在一些实现中,作为变换的一部分,还生成规则集的唯一输出的列表704。包括规则1和规则2的规则集的唯一输出的列表300是可生成的唯一输出的列表的例子。例如,当行中的输出存储为代码部分序列时,代码部分可以引导处理过程到在规则集的唯一输出的列表中的输出。

[0132] 在一些实现中,控制结构的行可以还包括这样的代码部分:当行的所有的触发条件满足并且行的(多个)输出已经生成时,其引导处理过程到接下来将处理的控制结构的不同行。

[0133] 例如,包括控制结构的变换可以通过在图6的执行环境604上运行的变换生成模块606生成(704)。

[0134] 所生成的变换,包括控制结构,可以被存储和/或传输706。在一些实现中,变换存储在一个存储器装置(例如,随机存取存储器)中,并传递给执行模块,该执行模块可以将变换施加到输入数据。例如,变换可以由变换生成模块606存储(706)在作为执行环境604一部分的易失性存储器装置中,其可以由图6中的执行模块612访问。在一些实现中,变换可以存储在包括非易失性存储器的数据存储装置(例如,数据库服务器或运行版本控制应用的服务器)中。例如,变换可以存储(706)在数据存储系统616中。在一些实现中,变换可以发送到远程设备(例如,通过电子通信网络)。例如,变换可以从运行在应用专家环境中的变换生成模块发送(706)到远程执行环境以施加到输入数据。

[0135] 一旦变换已经生成并可用于将要执行该变换的处理系统,则可以施加变换到输入数据。例如,变换可以由运行包括一个或多个实现变换的组件的数据流图的处理系统访问(例如,数据流图100的组件130和140)。可以从一个或多个数据源接收输入数据(708)(例如,数据源602)。在一些实现中,输入数据被预处理(例如,通过实现连接过程的组件120),以便创建创建可以传递到实现变换的一个或多个组件的数据流的工作单元。由于每个工作单元是基于所接收的输入数据准备的,所以可以被传递到变换。在一些实现中,工作单元组



被批量传递到变换。例如,可以通过图6中的执行环境604的网络接口接收输入数据(708)。

[0136] 执行变换(710)以处理所接收的输入数据。在一些实现中,也可以在处理新的输入数据时,在运行时解释和/或编译该变换。例如,可以执行变换(710)并使用相对于图8所描述的过程800将其施加到输入数据。将变换施加到输入数据可以包括针对输入数据,按照使用控制结构确定的顺序检查触发条件。例如,可以通过在图6的执行环境604中运行的执行模块606执行变换(710)。

[0137] 变换的执行可以持续(712)直到没有更多的输入数据可用。可以存储反映将变换施加到输入数据的结果的数据(714)(例如,作为写入到数据汇点的输出数据记录)。存储的结果数据可以是基于由控制结构指定的(多个)输出生成的。例如,该结果可以由执行模块612存储在图6中的数据存储系统614中。在一些实现中(未示出),基于控制结构指定的(多个)输出的结果数据通过电子通信网络(例如,通过执行环境的网络接口604)发送(例如,到应用专家环境622)。

[0138] 图8是用于执行基于规则集的变换的示例过程800的流程图。例如,过程800可以通过运行在图6中的执行环境604的执行模块612来执行。

[0139] 过程800包括检索(802)将施加到输入数据的变换的控制结构。在一些实现中,当实现变换的数据流图中的组件传递数据流中的工作单元时,检索控制结构(802)。在一些实现中,从存储器设备(例如,随机存取存储器)检索控制结构(802)。在一些实现中,从包括非易失性存储器(例如,数据库服务器或服务器上运行的版本控制程序)的数据存储设备检索控制结构(802)。在一些实现中,控制结构在运行时传递到解释器和/或编译器以准备用于执行的控制结构。

[0140] 针对工作单元的输入数据检查第一触发条件810。例如,可以解释和执行编码触发条件的DML表达式,以访问与工作单元相关联的(多个)记录的任何引用输入数据字段,并通过施加触发条件的逻辑来测试被访问的数据。该评估的结果可以是通过或不通过(真或假)。

[0141] 可以记录触发条件的评估结果(例如,用于测试、调试或优化的目的)。在一些实现中,可以记录触发条件的执行时间(例如,以微秒或处理器周期测量)。关于施加到输入数据的触发条件的不通过率或执行时间的数据可用于动态地更新控制结构,以尽力降低未来记录的平均处理时间。

[0142] 如果工作单元的输入数据没有满足触发条件815(例如,结果是不通过或假),则可以部分地基于与触发条件(例如,引用或包括该触发条件的代码部分)相关联的控制流代码,引导变换的执行到控制结构的不同行。按照该控制结构,执行可以跳转(820)到控制结构的不同行。例如,某些不通过的条件可能使得执行跳转(820)到控制结构的行序列中距离当前行多行之外的不同的行。以这种方式,可以跳过对某些行的触发条件和/或输出的执行,以降低工作单元的处理时间。可以随后检查(810)新行的下一个触发条件。在一些实现中(未示出),变换的执行可以跳转到没有触发条件的行(例如,对应于缺省执行案例),或直接到控制结构的结束。

[0143] 如果工作单元的输入数据满足触发条件(815)(例如,其结果是通过或真),则可以引导变换的执行到控制结构的当前行中的下一个元素。如果行的触发条件序列中有更多的触发条件(825),则检查行中的下一个触发条件(810)。否则,可以生成行的一个或多个输出



(830)。

[0144] 例如,可以解释和执行编码输出的DML表达式,以访问与工作单元相关联的记录中的任何引用输入数据字段和/或施加数据逻辑,以生成810一个或多个输出记录。(多个)结果输出记录可以是全新的,或可以更新或扩展现有记录,以包括另外的字段或其它数据。

[0145] 在生成(830)当前行的(多个)输出后,可以引导变换的执行到对应于另外的执行(多个)案例的不同的行。在一些实现中,行包括引导变换的执行到控制结构中的不同行的指针。如果还有执行案例待处理(835),则控制结构可以使得变换的执行跳转(840)到控制结构的不同行。例如,对于多发规则,可能需要处理对应于额外执行案例的额外的行。如果变换对应于具有多个规则的规则集,则控制结构可以使得变换的执行跳转(840)到控制结构中对应于不同规则的不同行。随后可以检查新行的下一个触发条件(810)。

[0146] 当没有更多的执行案例——因此没有更多的行——需要处理835时,可以基于处理后的输入数据的日志信息动态地更新控制结构。例如,可以部分地基于关于触发条件的执行时间的平均故障率的新的日志信息,对行的触发条件序列中的触发条件进行排序850。

[0147] 在一些实现中,可以基于用输入数据执行触发条件使用的时间,对唯一触发条件的列表中的触发条件的执行时间的测量进行更新。可以部分地基于更新后的触发条件的执行时间的测量,对行的触发条件序列中的触发条件进行排序(850)。在一些实现中,可以基于输入数据中的一个或多个记录是否满足该触发条件,对唯一触发条件的列表中的触发条件的不通过率测量进行更新。可以部分地基于更新的触发条件的不通过率测量,对行的触发条件序列中的触发条件进行排序(850)。

[0148] 可以存储(852)控制结构的更新版本以应用于未来的输入数据。在一些实现中,可以在存储器设备(例如,随机存取存储器)中存储852更新后的控制结构。在一些实现中,可以在包括非易失性存储器(例如,数据库服务器或运行版本控制应用的服务器)的数据存储设备中存储(852)更新后的控制结构。

[0149] 上述变换生成方法可以使用用于在计算机上执行的软件实现。例如,该软件形成一个或多个计算机程序中的过程,所述程序在一个或多个已编程或可编程的计算机系统(其可以是诸如分布式、客户端/服务器、或网格的各种架构)上运行,计算机系统的每一个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/或存储元素)、至少一个输入设备或端口、以及至少一个输出设备或端口。该软件可以形成较大程序的一个或多个模块,例如,提供与数据流图的设计和配置相关的其他服务。该图的节点和元素可以实现为存储在计算机可读介质中的数据结构、或符合在数据仓库中存储的数据模型的其他组织数据。

[0150] 该软件可以被提供在通用或专用目的的可编程计算机可读的诸如CD-ROM的存储介质上,或者通过网络的通信介质向运行它的计算机传递(在传播信号中编码)。所有的功能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。该软件可以按照分布式方式实现,其中该软件所规定的计算的不同部分由不同计算机执行。每一个这样的计算机程序优选存储在或下载到通用或专用可编程计算机可读的有形的、非暂时性的存储介质或设备(例如,固态存储器或介质、或者磁或光学介质),用于当存储介质或设备由计算机系统读取以执行其中描述的过程时,配置和操作计算机。本发明的系统也可以被认为实现为配置有计算机程序的计算机可读存储介质,其中如此配置的存储介质使得计算机系

统按照特定和预定义的方式操作以执行这里描述的功能。

[0151] 已经描述了本发明的多个实施例。然而,将理解的是,可以进行各种修改而不脱离本发明的精神和范围。例如,上述一些步骤可以是顺序无关的,并且由此可以按照与描述的顺序不同的顺序来执行。

[0152] 应当理解的是,前述描述旨在说明而不是限制由所附权利要求的范围限定的本发明的范围。例如,上面描述的多个功能步骤可以按照不同顺序执行,而基本上不影响总体处理。强调的是,在图2A和图2B的示例中描述的、以及本说明书中通篇引用的关于信用卡账户的特定的业务规则的细节,仅用于说明GUI 200和GUI 250的功能以及它们为其提供用户界面的变换生成系统。所呈现的特定的业务规则的细节不是必需的特征,并且不应被解释为限制权利要求的范围。其它实施例在以下权利要求的范围中。

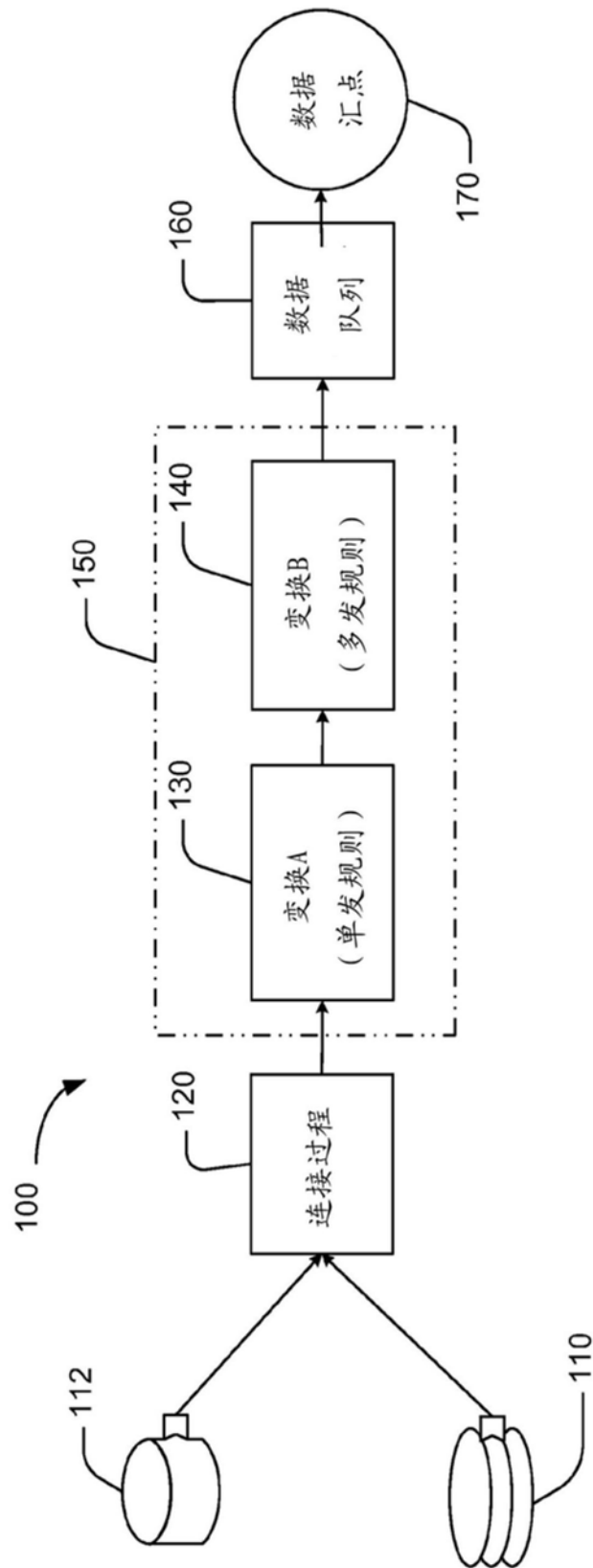


图1

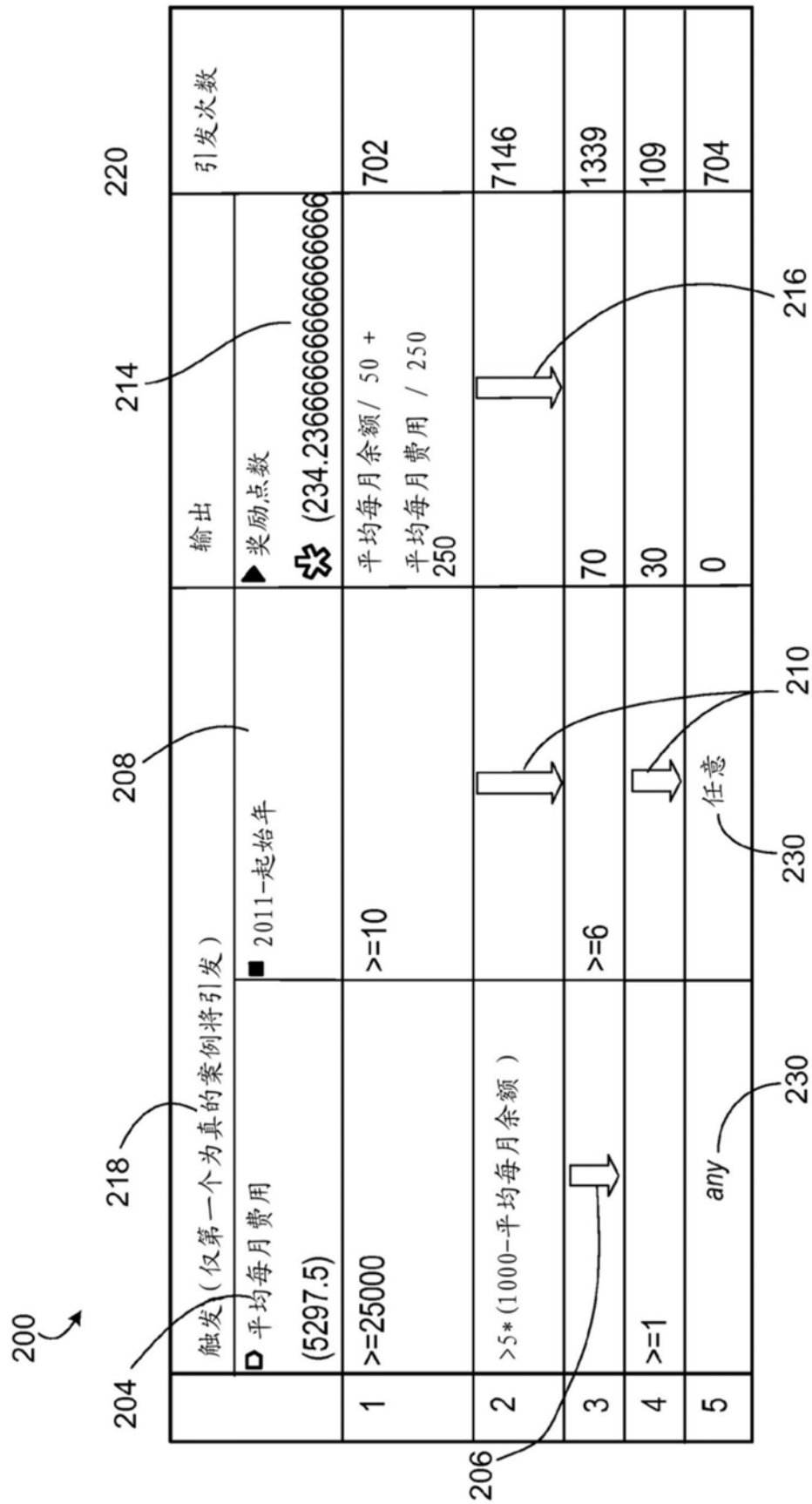


图2A

250

254 触发 (所有为真的案例都将引发)			258	268	264	270
为真:			县人口	输出 (目录)		
▶ 奖励点数			县人口	提供的产品		
🔱 (234.236666666666)			(801515)	🔱 (亚马逊Kindle)		
1	>=10	*卡类型 = 商务	非农村	“苹果 iPhone”		2530
2				“黑莓 Curve”		2530
3		*卡类型 = 亲和度 - 亚马逊	大型或都市	“亚马逊 Kindle”		428
4	>5	*卡类型 = 亲和度 - 航空		“便携式DVD播放器”		440
5			大型或都市	“Garmin 汽车GPS”		440

256

图2B

300

	DML表达式	使用 (rule_id, row_id, col_id)
1	in0.avg_purchases >=25000	(1,1,1)
2	ColumnHeader24* >= 10	(1,1,2) (1,2,2)
3	in0.avg_purchases > 5 * ( 1000 - in0.avg_balance)	(1,2,1) (1,2,1)
4	ColumnHeader24 >= 6	(1,3,2) (1,4,2)
5	in0.avg_purchases >= 1	(1,4,1)
6	Award_Points >= 10	(2,1,1) (2,2,1) (2,3,1)
7	in0.card_type == "biz"	(2,1,2) (2,2,2)
8	bzt_3(in0).CENSUS2000POP) <=10000	(2,1,3) (2,2,3)
9	in0.card_type == "amzn"	(2,3,2)
10	sql_or(sql_and((bzt_3(in0).CENSUS2000POP>=60001), (bzt_3(in0).CENSUS2000POP<=100000)), (bzt_3(in0).CENSUS2000POP>=100001)))	(2,3,3) (2,4,3) (2,5,3)
11	Award_Points >5	(2,4,1) (2,5,1)
12	in0.card_type == "air"	(2,4,2) (2,5,2)

310

320

图3A

350

	DML表达式	使用 (rule_id, row_id, col_id)
13	Award_Points=in0.avg_balance / 50 + in0.avg_purchases / 250	(1,1,3),(1,2,3)
14	Award_Points=70	(1,3,3)
15	Award_Points=30	(1,4,3)
16	Award_Points=0	(1,5,3)
17	Products_Offered=vector_append(Products_Offered,"Apple iPhone");	(2,1,4)
18	Products_Offered=vector_append(Products_Offered,"Blackberry Curve");	(2,2,4)
19	Products_Offered=vector_append(Products_Offered,"Amazon Kindle");	(2,3,4)
20	Products_Offered=vector_append(Products_Offered,"Portable DVD Player");	(2,4,4)
21	Products_Offered=vector_append(Products_Offered,"Garmin Auto GPS");	(2,5,4)

360

370

图3B

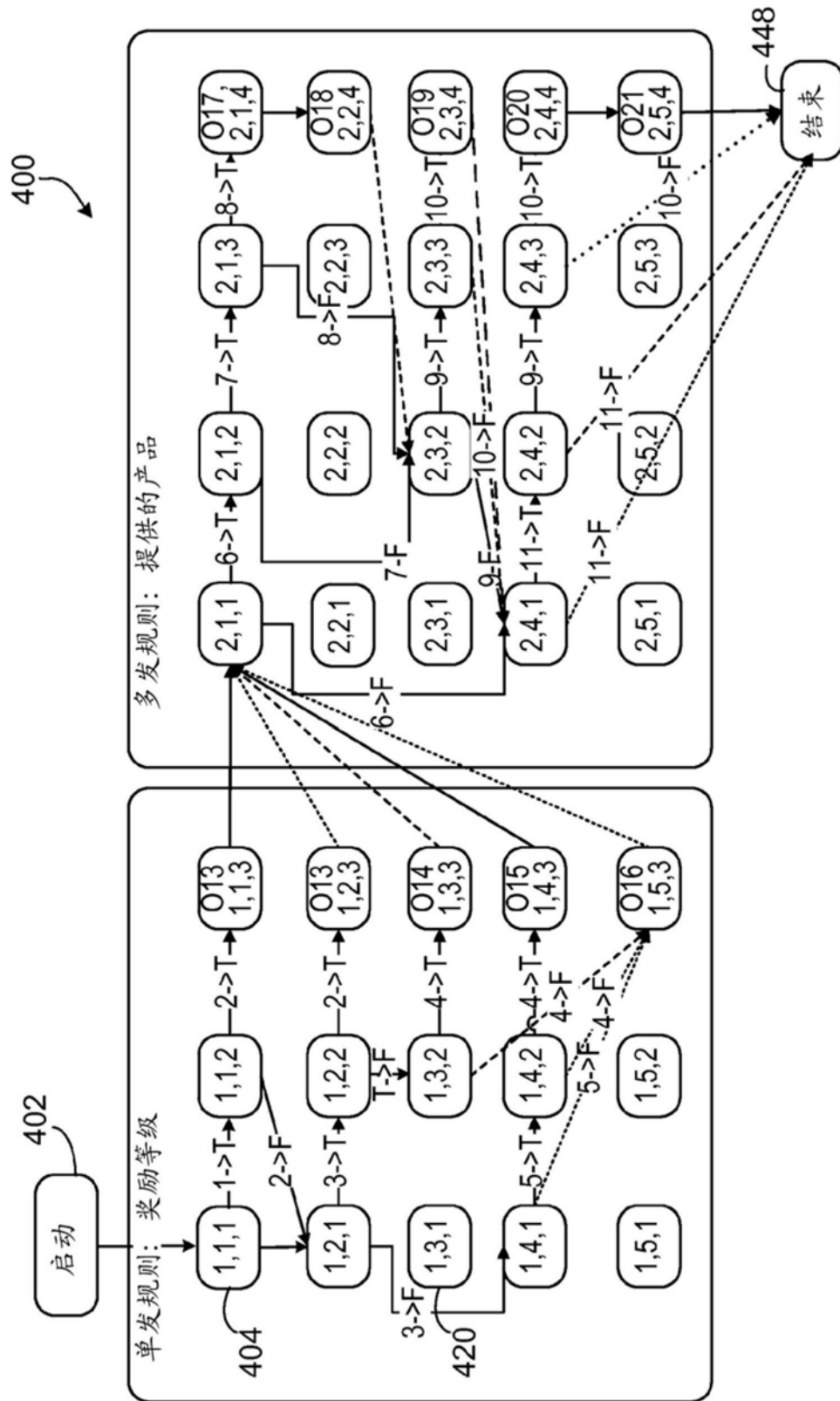


图4A



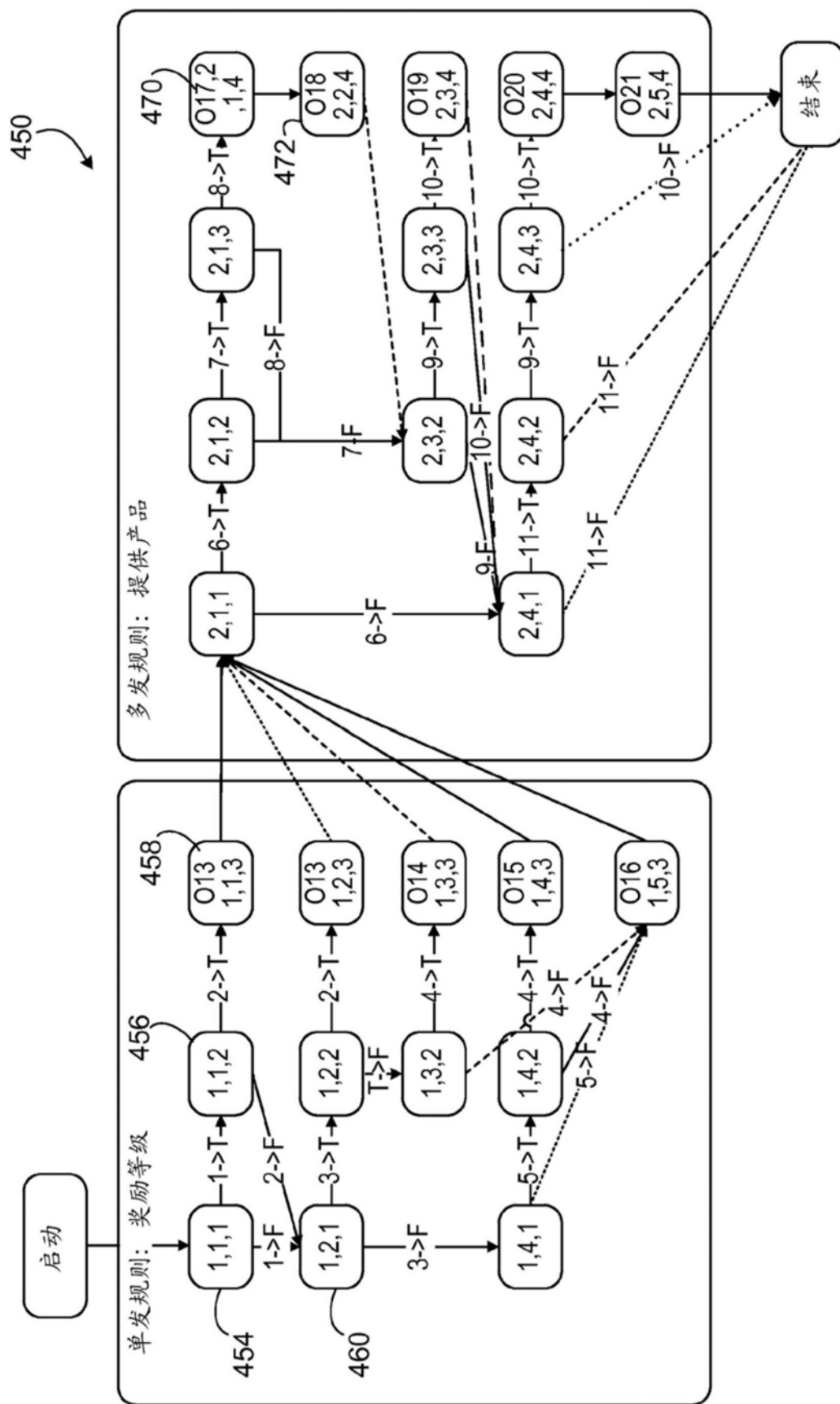


图4B

500

510 执行案例	520 触发	540 成功后的下一个 执行案例	530 成功后执行的输出
ec1	Check 1 on fail goto ec2 (检查1, 失败则跳转到ec2) Check 2 on fail goto ec2 (检查2, 失败则跳转到ec2)	ec6	13
ec2	Check 3 on fail goto ec4 (检查3, 失败则跳转到ec4) Check 2 on fail goto ec3 (检查2, 失败则跳转到ec3)	ec6	13
ec3	Check 4 on fail goto ec5 (检查4, 失败则跳转到ec5)	ec6	14
ec4	Check 5 on fail goto ec5 (检查5, 失败则跳转到ec5) Check 4 on fail goto ec5 (检查4, 失败则跳转到ec5)	ec6	15
ec5		ec6	16
ec6	Check 6 on fail goto ec9 (检查6, 失败则跳转到ec9) Check 7 on fail goto ec8 (检查7, 失败则跳转到ec8) Check 8 on fail goto ec8 (检查8, 失败则跳转到ec8)	ec7	17
ec7		ec8	18
ec8	Check 9 on fail goto ec9 (检查9, 失败则跳转到ec9) Check 10 on fail goto ec9 (检查10, 失败则跳转到ec9)	ec9	19
ec9	Check 11 on fail goto END (检查11, 失败则跳转到END) Check 9 on fail goto END (检查9, 失败则跳转到END) Check 10 on fail goto END (检查10, 失败则跳转到END)	ec10	20
ec10		结束	21

522

图5

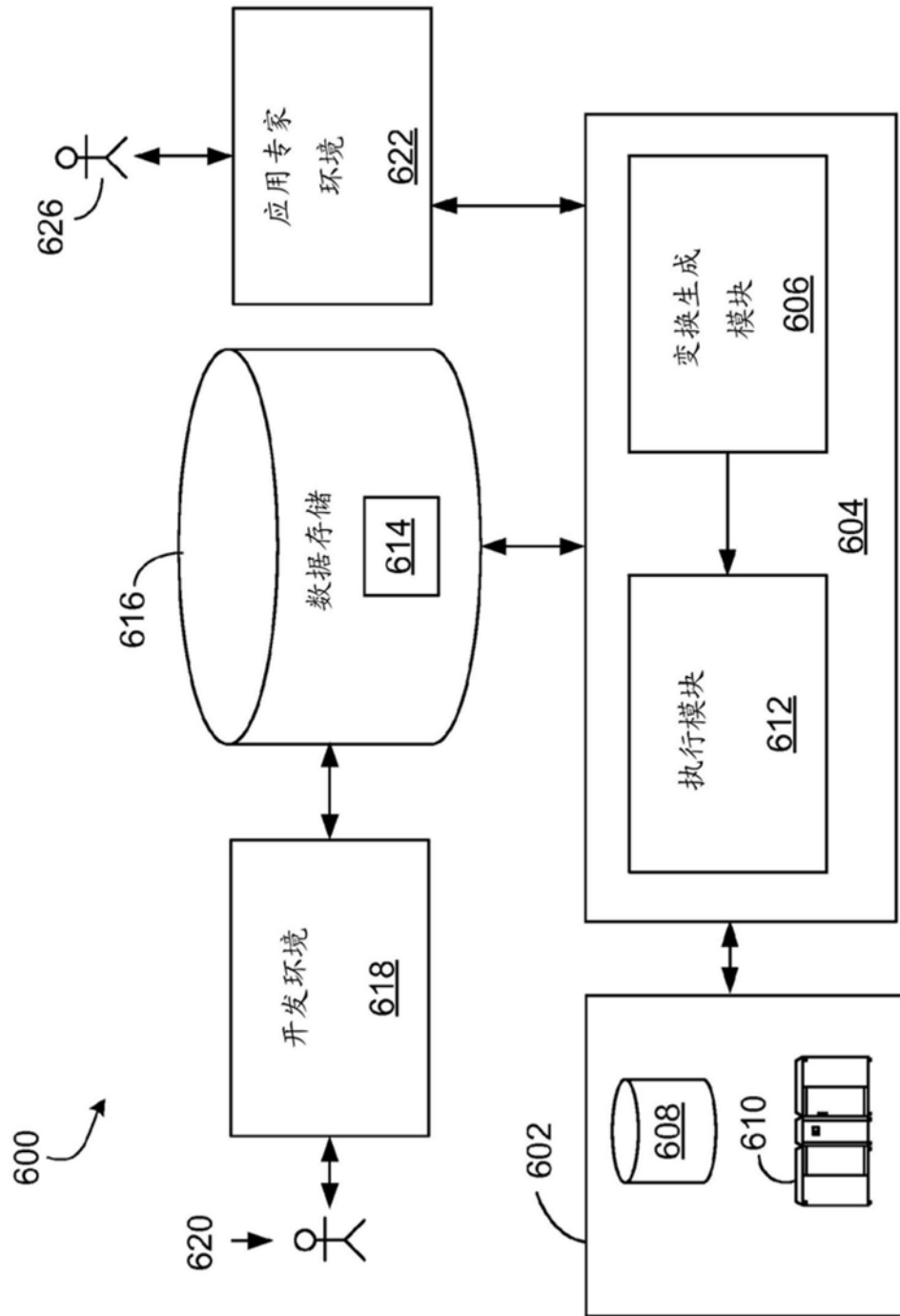


图6

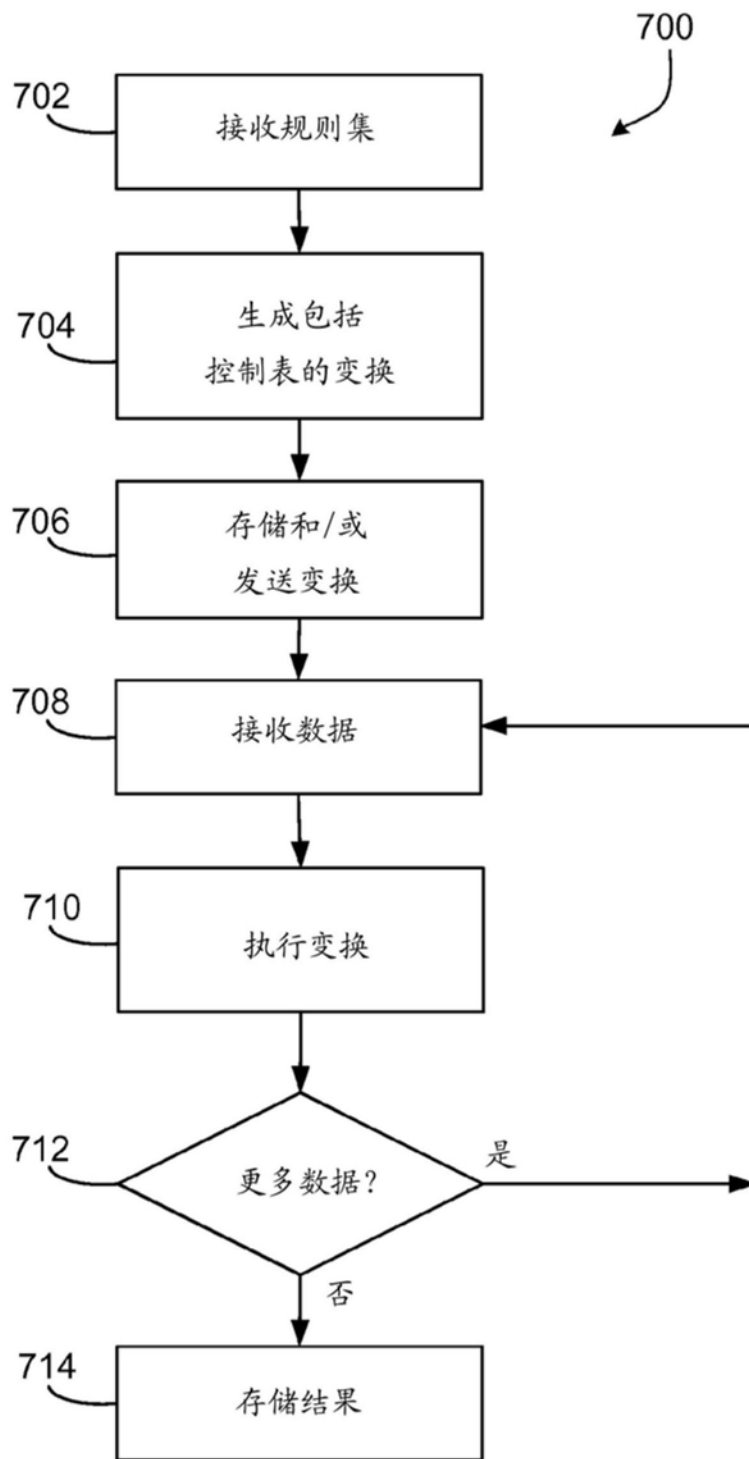


图7

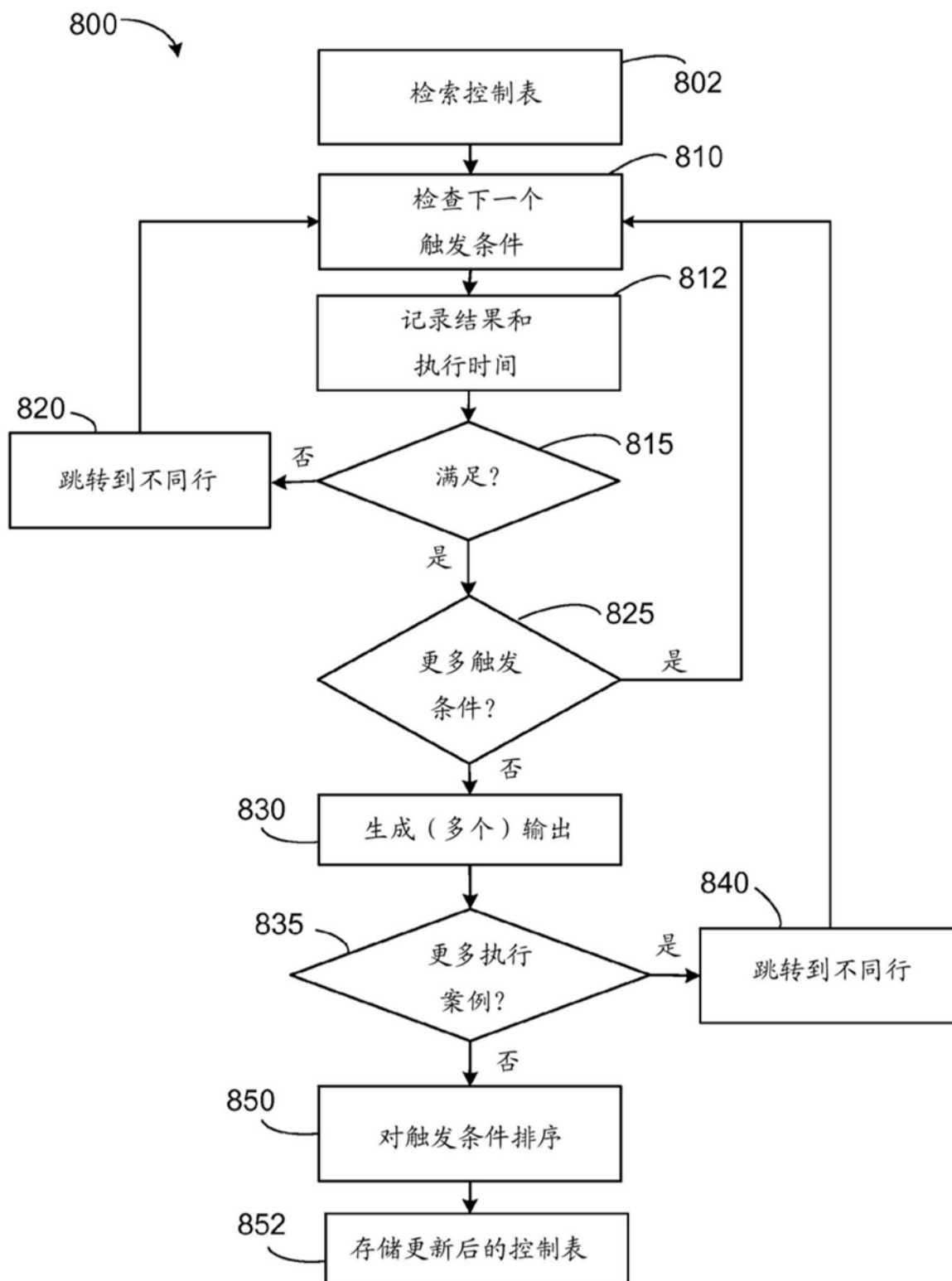


图8