(19)中华人民共和国国家知识产权局

(12)发明专利

(10)授权公告号 CN 107038519 B
(45)授权公告日 2022.04.29

(21)申请号 201610979053.1
(22)申请日 2016.11.08
(65)同一申请的已公布的文献号
申请公布号 CN 107038519 A
(43)申请公布日 2017.08.11
(30)优先权数据
14/935,922 2015.11.09 US
(73)专利权人 达索系统美国公司
地址 美国马萨诸塞州
(72)发明人 I·多宾森 P·海恩斯
(74)专利代理机构 永新专利商标代理有限公司 72002
代理人 刘瑜 王英
(51)Int.Cl.
G06Q 10/06(2012.01)
G06Q 10/08(2012.01)
G06F 16/25(2019.01)
(56)对比文件
US 2008071844 A1,2008.03.20
CN 101727320 A,2010.06.09
US 7801945 B1,2010.09.21
审查员 庄湧

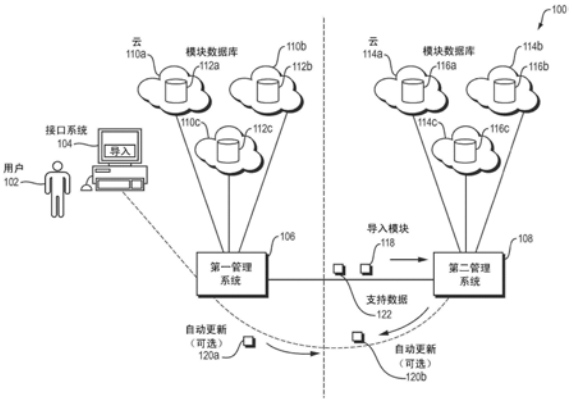
权利要求书3页 说明书15页 附图9页

(54)发明名称

用于系统之间的双向数据同步的方法和系统

(57)摘要

在实施例中,一种计算机实现的方法,用于在源系统中的对应模块变化时自动更新目的系统中的导入信息,所述方法包括:响应于所述目的系统检测到所述源系统中的源模块的变化而改变,其中,源模块对应于目的系统的导入模块,将对应于所述源模块的面包屑装载到所述源系统中。该计算机实现的方法还包括:基于所装载的面包屑来确定与所改变的源模块链接的目的模块。该计算机实现的方法还包括:从所述源系统导出对所述目的系统中的目的模块的改变。因此,在对任意系统做出改变时,数据保持在两个系统之间同步。



CN 107038519 B

1.一种计算机实现的方法,用于在源系统中的对应模块变化时自动更新目的系统中的导入信息,所述方法包括:

响应于用户对源系统中的源模块执行结构改变,调用触发程序,所述触发程序指示要发送到所述目的系统的所述结构改变;

响应于所述目的系统通过接收所述结构改变而检测到所述源系统中的源模块的变化,改变对应于所述目的系统中的导入模块的源模块;

将对应于所述源模块的面包屑装载到所述源系统中;

基于所装载的面包屑来确定与所改变的源模块链接的目的模块;以及

从所述源系统导出对所述目的系统中的目的模块的改变;

其中,(i)所述目的系统是产品生命周期管理PLM系统并且所述源系统是源代码管理SCM系统,或者(ii)所述目的系统是源代码管理SCM系统并且所述源系统是产品生命周期管理PLM系统。

2.如权利要求1所述的计算机实现的方法,还包括:

响应于导出对所述目的系统的改变而在所述目的系统中设置旗语标记;

在所述旗语标记被设置的同时在所述目的系统中阻止自动更新;以及

响应于导出对所述目的系统的改变的完成而清除所述目的系统中的旗语标记。

3.如权利要求2所述的计算机实现的方法,其中,阻止还包括:

阻止新更新线程在所述目的系统中开始;

等待其它运行线程完成;

阻止所述触发程序开始自动更新过程;

执行所述更新过程所需的更新;

解除对所述触发程序的阻止;以及

允许开始新线程。

4.如权利要求1所述的计算机实现的方法,其中,对所述源模块的改变是如下中的至少一个:(a)添加分级引用,(b)去除分级引用,以及(c)修改分级引用。

5.如权利要求4所述的计算机实现的方法,其中,导出对所述目的系统中的目的模块的改变还使得在所述目的系统中添加所述分级引用或者去除所述分级引用。

6.一种计算机实现的系统,用于在源系统中的对应模块变化时自动更新目的系统中的导入信息,所述系统包括:

处理器;以及

存储器,其中存储有计算机代码指令,所述存储器与所述处理器操作地耦合而使得计算机代码指令配置所述处理器来:

响应于用户对源系统中的源模块执行结构改变,调用触发程序,所述触发程序指示要发送到所述目的系统的所述结构改变;

响应于所述目的系统通过接收所述结构改变而检测到所述源系统中的源模块的变化,改变对应于所述目的系统中的导入模块的源模块;将对应于所述源模块的面包屑装载到所述源系统中;

基于所装载的面包屑来确定与所改变的源模块链接的目的模块;以及

从所述源系统导出对所述目的系统中的目的模块的改变;

其中，(i)所述目的系统是产品生命周期管理PLM系统并且所述源系统是源代码管理SCM系统，或者(ii)所述目的系统是源代码管理SCM系统并且所述源系统是产品生命周期管理PLM系统。

7.如权利要求6所述的计算机实现的系统，其中，所述处理器还被配置为：

响应于导出对所述目的系统的改变而在所述目的系统中设置旗语标记；如果所述旗语标记被设置，基于对所述源系统模块的初始改变而在所述源系统中阻止从所述目的系统的自动更新；以及响应于导出对所述目的系统的改变的完成而清除所述目的系统中的旗语标记。

8.如权利要求7所述的计算机实现的系统，其中，所述处理器被进一步配置为：

阻止新更新线程在所述目的系统中开始；

等待其它运行线程完成；

阻止触发程序开始自动更新过程；

执行所述更新过程所需的更新；

解除对所述触发程序的阻止；以及

允许开始新线程。

9.如权利要求6所述的计算机实现的系统，其中，对所述源模块的改变是如下中的至少一个：(a)添加分级引用，(b)去除分级引用，以及(c)修改分级引用。

10.如权利要求9所述的计算机实现的系统，其中，导出对所述目的系统中的目的模块的改变还使得在所述目的系统中添加所述分级引用或者去除所述分级引用。

11.一种非暂态计算机可读介质，其被配置为存储用于在源系统中的对应模块变化时自动更新目的系统中的导入信息的指令，所述指令在被处理器装载和执行时，使所述处理器：

响应于用户对源系统中的源模块执行结构改变，调用触发程序，所述触发程序指示要发送到所述目的系统的所述结构改变；

响应于所述目的系统通过接收所述结构改变而检测到所述源系统中的源模块的变化，改变对应于所述目的系统中的导入模块的源模块；

将对应于所述源模块的面包屑装载到所述源系统中；

基于所装载的面包屑来确定与所改变的源模块链接的目的模块；以及

从所述源系统导出对所述目的系统中的目的模块的改变；

其中，(i)所述目的系统是产品生命周期管理PLM系统并且所述源系统是源代码管理SCM系统，或者(ii)所述目的系统是源代码管理SCM系统并且所述源系统是产品生命周期管理PLM系统。

12.如权利要求11所述的非暂态计算机可读介质，其中，所述指令还使所述处理器：

响应于导出对所述目的系统的改变而在所述目的系统中设置旗语标记；以及

在所述旗语标记被设置的同时在所述源系统中阻止从所述目的系统的自动更新；以及

响应于导出对所述目的系统的改变的完成而清除所述目的系统中的旗语标记。

13.如权利要求12所述的非暂态计算机可读介质，其中，阻止还包括：

阻止新更新线程在所述目的系统中开始；

等待其它运行线程完成；

阻止所述触发程序开始自动更新过程；

执行所述更新过程所需的更新；

解除对所述触发程序的阻止；以及

允许开始新线程。

14.如权利要求11所述的非暂态计算机可读介质，其中，对所述源模块的改变是如下中的至少一个：(a)添加分级引用，(b)去除分级引用，以及(c)修改分级引用。

用于系统之间的双向数据同步的方法和系统

[0001] 相关申请

[0002] 本申请涉及共同提交的美国申请:转让给共同的受让人且与本申请在同一日提交的Ian Dobinson和Peter Haynes做出的“Exporting Hierarchical Data From A Product Lifecycle Management(PLM)System To A Source Code Management(SCM)System(从产品生命周期管理(PLM)系统导出分级数据到源代码管理(SCM)系统”(代理人档案号4659.1020-000)、以及转让给共同的受让人且与本申请在同一日提交的Ian Dobinson和Peter Haynes做出的“Exporting Hierarchical Data From A Source Code Management(SCM)System To A Product Lifecycle Management(PLM)System(从源代码管理(SCM)系统导出分级数据到产品生命周期管理(PLM)系统”(代理人档案号4659.1019-000)。

[0003] 上述申请的整个教导通过引用方式并入本文。

技术领域

[0004] 本申请涉及数据同步。具体而言,本申请涉及产品生命周期管理(PLM)系统与源代码管理(SCM)系统之间的双向数据同步。

背景技术

[0005] 硬件开发者,尤其是集成电路设计团体中的硬件开发者,通常将分级开发方法应用于设计,其中高级产品参考子产品或者构建于子产品之外。类似地,软件开发者通常使用分级,使得更高级的软件组件利用可重用软件块。

[0006] 源代码管理(SCM)系统在其最基础级上提供了对存储在存储器或数据库中的数据文件集合进行版本控制的手段,这允许跟踪那些文件随时间的变化。SCM系统的示例是:

[0007] a)DesignSync®系统,其是达索系统的Defect Management and Collaboration(缺陷管理和协作)产品的部分;

[0008] b)公共可用的Concurrent Versioning System(CVS)(并发版本系统);

[0009] c)公共可用的GIT修订控制系统;以及

[0010] d)在商业支持下可用的Subversion and Perforce系统。

[0011] 分级SCM系统这样的系统:其中数据文件能够被管理为包或模块,包或模块可以包含大量的单独的文件。模块或包还可以引用其他模块或包而形成数据分级。

[0012] 产品生命周期管理(PLM)系统提供了用于从开端、工程设计、制造、到所制造的产品的服务和处置来管理产品的整个生命周期的过程。PLM系统集成了人、数据、过程和商业系统,并且提供了用于公司及其扩展企业的产品信息骨干。PLM系统将数据表示为一系列由关系连接的一系列对象。例如,第一对象可以代表开发中的产品,其与表示正在开发产品的组织的第二对象有关系。这些对象还可以支持表示对象的不同发布的或者表示对象的开发状态的修订(或版本)(例如,不同的修订可能代表芯片的版本1.0或者芯片的后来的2.0版本,或者还可能表示处于朝向未来发布进展中的工作)。这些对象还可以经由相同或相似类型的对象之间的关系来支持分级。例如,表示芯片的对象可以与表示作为该芯片的一部分

的CPU模块的对象有关系。3DExperience®Platform旗下的由达索系统所提供的应用提供了PLM系统的示例。

发明内容

[0013] 在本发明的实施例中，系统和方法提供了用于将分级数据块集合的描述从SCM系统传递到PLM系统的方案。在本发明的实施例中，系统和方法将数据从PLM系统传输到支持分级设计的SCM系统。在本发明的实施例中，系统和方法发明手动地或者自动地将分级数据块集合的描述从PLM系统传递到SCM系统或者从SCM系统传递到PLM系统。

[0014] 传统地，PLM系统和SCM系统尚未链接。至少一些已知的解决方案提供了将两个系统中的数据链接的方式，并且提供了将数据从SCM系统拉入PLM系统的有限的支持。然而，先前的方式不高效，功能受限，并且不允许从SCM系统控制流程，不允许数据结构在PLM系统中定义然后下推至SCM系统，或者将一个系统中的数据自动更新随着对应的数据在另一系统中被修改，如本发明的实施例中的那样。

[0015] SCM至PLM

[0016] 本发明的一个实施例描述了一种将数据从支持分级设计的SCM系统传输到PLM或类似系统的手段。传递数据包括从SCM系统提取数据以及将数据转换成便携格式，包括分级连接以及那些连接是否是开发中的子块或版本的具体发布的细节。

[0017] 该系统进一步定制该数据，使得系统的单独的实现方式能够提供目标PLM系统的对象所需的数据。系统随后将该数据传输到PLM系统。

[0018] 该系统在PLM系统中创建表示来自SCM系统的数据块及其版本以及它们之间的分级连接的对象。

[0019] 该系统进一步经由SCM系统上的定制功能，基于所传递的附加数据来定制在PLM系统中创建的数据的类型、用于连接数据的关系以及PLM系统中的对象。

[0020] 系统随后将等同于SCM数据对象的、已经创建的对象的信息传递回到SCM系统，以使得该信息能够存储以用于跟踪目的。

[0021] 在实施例中，本发明根据SCM系统中的数据在PLM系统中创建新数据。

[0022] 本发明的实施例提供了对从SCM系统进行数据传输的初始化和控制。这允许有先进的定制和控制能力。该系统进一步提供用于存储关于SCM系统的跟踪信息的方案。

[0023] SCM系统管理定义系统级对象及其分级关系的数据。因此，有益的是数据在其创建于SCM系统中时从SCM系统插入PLM系统。现有的解决方案没有提供一种如本发明的实施例中那样利用从SCM系统控制的过程来直接在PLM系统内创建数据的手段。

[0024] 不同的PLM系统保存针对单独的对象的不同类型的信息(例如，性质/属性)，以及不同的SCM系统或者特定SCM系统的不同实现方式。PLM系统对能够用于在PLM系统中填充信息的不同的数据集具有访问权。因此，在本发明的实施例中，当从SCM系统传输数据到PLM系统时，系统能够聚集、生成且供给该附加信息。现有的解决方案还不支持创建数据和相关联的属性设置的具体定制。

[0025] 一旦数据已经传输到PLM系统，SCM保留到所创建的对象的链接。从SCM系统到PLM系统的链接能够通过允许SCM系统直接链接到PLM系统中的等同对象来实现先进的功能。例如，SCM系统能够实现如下功能：如果SCM系统得知该对象是什么，则直接显示等同的PLM系

统对象。现有的解决方案没有提供该跟踪(例如,面包屑(breadcrumb))信息在SCM系统内的存储。

[0026] PLM至SCM

[0027] 本发明的实施例是一种用于将分级数据块集合的描述从PLM系统传递到SCM系统的方案。传递描述包括:从PLM系统提取数据成便携格式,包括分级连接以及那些连接是否到子块的特定发布或者到开发中的版本的细节。系统进一步定制该数据,使得系统的单独的实现方式能够提供能够由目标SCM系统所使用的数据。该系统进一步将该数据传输到SCM系统。该系统进一步在SCM系统中创建表示来自SCM系统的对象以及它们之间的分级连接的容器对象。该系统进一步经由PLM系统上的定制功能,基于所传递的附加数据来在SCM系统中定制对象。该系统进一步将等同于PLM数据对象的、已经创建的对象的信息传递回到PLM系统,使得该信息能够被存储以用于跟踪目的。

[0028] PLM系统中的数据用于许多操作,例如,跟踪包含产品的件的开发状况以及跟踪针对包括产品的件中的任何一个所发现的问题(例如,缺陷、漏洞)。

[0029] 在许多情况下,采用汇总数据来得到产品的总体状况。例如,诸如计算机芯片的产品可能由多个组件构成,诸如随机存取存储器(RAM)和中央处理单元(CPU)。如果针对CPU发现问题并且提出对应的缺陷,则芯片的管理者期望汇总报告,该汇总报告表明该缺陷存在于一些子组件并且因此影响了芯片。因此,在系统中精确地跟踪PLM系统中的数据,尤其是较高级产品对子块的分级连接/使用。

[0030] PLM系统能够视为用于定义设计的块以及它们之间的分级关系的主机。也即,设计的结构创建于PLM系统中并且SCM系统用于管理用于包括结构的块的单独的数据文件。SCM系统包含其自身的对象以表示设计的块以及它们之间的分级关系,使得SCM系统的用户能够取得包括设计的正确的数据集并且对该数据执行它们的操作。

[0031] 本发明的实施例提供了利用已经在PLM系统内放到一起的定义来在SCM系统内创建块以及它们之间的连接的能力。现有的解决方案未提供利用从PLM系统控制的过程来直接在SCM系统内创建数据的手段。

[0032] 不同的PLM系统存储针对单独的对象的不同类型的信息(例如性质、属性),并且不同的SCM系统可以存储根据PLM系统中的信息定义的不同数据集。在本发明的实施例中,从PLM系统传输数据到SCM系统足够灵活来处理这种附加信息。现有的解决方案不支持数据的创建和相关联的属性设置的过程的详细定制。

[0033] 一旦数据传输到SCM系统,PLM系统保留到所创建的对象的链接并且SCM系统维护在PLM系统中的等同的源对象的知识。这使得通过允许SCM系统直接链接到PLM系统中的等同对象来实现先进的功能。例如,SCM系统能够实现如下一种功能:如果SCM系统知道该对象是什么则直接显示等同的PLM系统对象。现有的解决方案不能提供该跟踪(或面包屑)信息在SCM系统内的存储。本发明的实施例根据PLM系统中的数据在SCM系统中创建新数据。

[0034] 本发明的实施例通过如下改进了先前可用的解决方案:

[0035] a)允许对从PLM系统进行数据传输的初始化和控制;

[0036] b)允许基于PLM系统信息在SCM系统中创建新容器对象(例如,DesignSync系统中的模块)或现有对象的新分支;

[0037] c)提供先进的定制和控制能力;以及

[0038] d)提供存储关于SCM和PLM系统的跟踪信息的方案。

[0039] 从PLM到SCM的自动更新或者从SCM到PLM的自动更新

[0040] 本发明的实施例手动地或者自动地将分级数据块集合的描述从PLM系统传递到SCM系统，或者从SCM系统传递到PLM系统。该系统通过采用上述系统提取一个系统上的数据以及将该数据传输到另一系统来传递描述。该系统还提供对先前所描述的系统的扩展以允许在目标系统上数据的增量更新。对上述系统的扩展进一步允许用户手动开始以一侧的数据为起始的更新。该系统进一步自动标识变化且在两个系统之间同步变化。

[0041] 在实施例中，本发明允许对在PLM或SCM系统上的分级数据的不间断的开发以及对该数据的持续同步。在实施例中，本发明进一步允许自动同步数据，从而防止由于在任一系统内存在过期信息导致的任何问题。

[0042] 在实施例中，系统将数据从PLM系统传输到支持分级设计的SCM系统，反之亦然。在这些实施例中，传输是由用户启动的，并且旨在用于系统之间的初始的数据推送。系统手动地或者自动地更新两个系统上的数据并且同步数据。

[0043] 分级SCM系统允许将数据文件管理为包或模块，其可以包含大量的单独的文件。这些模块可以引用其它模块而形成数据的分级。

[0044] SCM系统存储其自身的表示设计的块和它们之间的分级关系的对象，使得SCM系统的用户能够取得包括设计的正确的数据集并且对该数据执行操作。

[0045] 随时间推移，结构的定义可以通过多种方式来变化。例如，产品的体系结构利用PLM系统可以决定采用通常由设计中的某其它组提供的附加块或者可替代的IP块(例如，模块、设计段或知识产权(IP)件)。作为另一示例，产品的开发者，利用SCM系统可以决定将块分割成两个部分用于实现的目的。作为又一示例，开发该设计的子块中的一个的团队可以发布该块的新版本并且可以决定在父产品中采用新版本。

[0046] 当使用本发明的实施例时，可以在开发期间的任何时候执行改变并且改变可以从PLM系统或SCM系统开始。如果对系统中的仅一个系统上的结构做了改变，则会出现挑战。例如，在PLM系统中所做的变化而没有在SCM系统中做出变化会导致构建错误的产品，对于产品可靠性或功能产生灾难性的影响。作为另一示例，在SCM系统中所做的改变，而没有在PLM系统中做出改变，会导致对问题的不正确跟踪，或者对IP使用的不正确跟踪。这会导致IP特许问题或者缺失对问题的精确跟踪，可能导致产品可靠性问题(在没有现有漏洞的知识的情况下运送产品)。

[0047] 现有的解决方案没有提供一种手动地或者自动地根据另一系统中的源定义来增量地更新一个系统中的数据的手段。

[0048] 在实施例中，一种计算机实现的方法，用于在源系统中的对应模块变化时自动更新目的系统中的导入信息，所述方法包括：响应于所述目的系统检测到所述源系统中的源模块的变化，而做出改变，其中，源模块对应于目的系统的导入模块，将对应于所述源模块的面包屑装载到所述源系统中。该计算机实现的方法还包括：基于所装载的面包屑来确定与改变的源模块链接的目的模块。该计算机实现的方法还包括：从所述源系统导出对所述目的系统中的目的模块的改变。

[0049] 在实施例中，所述目的系统是产品生命周期管理(PLM)系统，并且所述源系统是源代码管理(SCM)系统。

[0050] 在实施例中，所述目的系统是源代码管理(SCM)系统，并且所述源系统是产品生命周期管理(PLM)系统。

[0051] 在实施例中，计算机实现的方法还包括响应于导出对所述目的系统的变化而在所述源系统中设置旗语标记。该计算机实现的方法包括：如果旗语标记被设置，基于对所述源系统模块的初始改变而在所述源系统中阻止从所述目的系统自动更新。该阻止还可以包括：阻止新更新线程在所述源系统中开始；等待其他运行中的线程完成；阻止触发程序开始自动更新过程；执行所述更新过程所需的更新；解除对所述触发程序的阻止；以及允许开始新线程。

[0052] 在实施例中，对所述源模块的改变是如下中的至少一个：(a)添加分级引用，以及(b)去除分级引用。导出对所述目的系统中的目的模块的变化还使得在所述目的系统中添加分级引用或者去除分级引用。

[0053] 在实施例中，一种计算机实现的系统，用于在源系统中的对应模块变化时自动更新目的系统中的导入信息，所述系统包括处理器以及其中存储有计算机代码指令的存储器。该存储器与所述处理器可操作地耦合而使得计算机代码指令配置所述处理器来实现装载模块，其被配置为，响应于所述目的系统检测到所述源系统中的源模块的变化，而做出改变，所述源模块对应于所述目的系统的导入模块，将对应于所述源模块的面包屑装载到所述源系统中。该处理器被配置为实现链接模块，其被配置为，基于所装载的面包屑来确定与改变的源模块链接的目的模块。该处理器被进一步配置为导出模块，其被配置为从所述源系统导出所述目的系统中的目的模块的变化。

[0054] 在实施例中，所述目的系统是产品生命周期管理(PLM)系统，并且所述源系统是源代码管理(SCM)系统。

[0055] 在实施例中，所述目的系统是源代码管理(SCM)系统，并且所述源系统是产品生命周期管理(PLM)系统。

[0056] 在实施例中，计算机实现的系统包括旗语模块，其被配置为，响应于导出对所述目的系统的变化而在所述源系统中设置旗语标记；如果所述旗语标记被设置，基于对所述源系统模块的初始改变而在所述源系统中阻止从所述目的系统自动更新。该旗语模块可进一步配置为：阻止新更新线程在所述源系统中开始；等待其它运行中的线程完成；阻止触发程序开始自动更新过程；执行所述更新过程所需的更新；解除对所述触发程序的阻止；以及允许开始新线程。

[0057] 在实施例中，对所述源模块的改变是如下中的至少一个：(a)添加分级引用，以及(b)去除分级引用。所述导出模块可以被进一步配置为使得在所述目的系统中添加分级引用或者去除分级引用。

[0058] 在实施例中，一种非暂态计算机可读介质，其被配置为存储用于在源系统中的对应模块变化时自动更新目的系统中的导入信息的指令。所述指令在被处理器装载和执行时，使所述处理器：响应于所述目的系统检测到所述源系统中的源模块的变化而改变，所述目的系统的导入模块对应于所述源模块，将对应于所述源模块的面包屑装载到所述源系统中。指令进一步使所述处理器基于所装载的面包屑来确定与改变的源模块链接的目的模块。指令进一步使所述处理器从所述源系统导出对所述目的系统中的目的模块的改变。

[0059] 在实施例中，所述目的系统是产品生命周期管理(PLM)系统，并且所述源系统是源

代码管理(SCM)系统。

[0060] 在实施例中,所述目的系统是源代码管理(SCM)系统,并且所述源系统是产品生命周期管理(PLM)系统。

[0061] 在实施例中,所述指令还使所述处理器:响应于导出对所述目的系统的变化而在所述源系统中设置旗语标记。所述指令还使所述处理器:如果所述旗语标记被设置,基于对源系统的初始的改变,在所述源系统中阻止从所述目的系统自动更新。阻止还包括:阻止新更新线程在所述源系统中开始;等待其它运行中的线程完成;阻止触发程序开始自动更新过程;执行所述更新过程所需的更新;解除对所述触发程序的阻止;以及允许开始新线程。

[0062] 在实施例中,对所述源模块的改变是如下中的至少一个:(a)添加分级引用,以及(b)去除分级引用。

附图说明

[0063] 前面所述将根据本发明的示例性实施例的以下更具体的描述而变得显而易见,如附图所示的,在附图中相似的附图标记在不同的视图中指代相同的部件。附图不一定按比例绘制,相反重点应放在图示说明本发明的实施例。

[0064] 图1是示出了本发明的示范性的实施例的框图。

[0065] 图2是示出了PLM系统的分级的示例性实施例的框图。

[0066] 图3是示出了根据本发明的实施例的从SCM系统向PLM系统发送模块的框图。

[0067] 图4是示出了由本发明采用的过程的示例性实施例的流程图。

[0068] 图5是示出了将PLM系统对象数据导入SCM系统的示例性实施例的框图。

[0069] 图6是示出了由本发明采用用于从PLM系统导入数据到SCM系统的过程的示例性实施例的流程图。

[0070] 图7图示出本发明所采用的在另一系统中的对应模块变化时自动更新SCM或PLM系统中的导入的信息的过程的示例性实施例的流程图。

[0071] 图8示出了可以实现本发明的实施例的计算机网络或类似的数字处理环境。

[0072] 图9是在图8的计算机系统中计算机(例如,客户端处理器/设备或服务器计算机)的示例的内部结构的示意图。

具体实施方式

[0073] 本发明的示例性实施例的描述如下。

[0074] 图1是示出了本发明的示例性实施例的框图100。用户102采用与第一管理系统106和第二管理系统108耦合的接口系统104。第一管理系统106和第二管理系统108的示例分别可以是SCM或PLM。SCM和PLM二者均采用模块的分级集合来在SCM的情况下形成源代码解决方案或者在PLM的情况下形成产品。在许多情形下,PLM中的产品,诸如芯片,可能在其设计中需要源代码,并且源代码设计可能需要硬件资源。因此,用于将模块从第一管理系统106导入第二管理系统108的系统是期望的。

[0075] 第一管理系统106和第二管理系统108与能够任选地存储在相应的云网络110a-c和114a-c上的相应的模块数据库112a-c和116a-c耦合。在实施例中,模块数据库112a-c和116a-c还可以本地地存储(未示出),或者存储在单一远程服务器(未示出)上,或者在多个

远程服务器(未示出)上。用户102,经由接口系统104,引导待导入模块118从第一管理系统106发送到第二管理系统108。第一管理系统106还将支持数据122发送到第二管理系统108。支持数据122可以包括关于模块118的分级信息、支持模块以及关于模块的面包屑信息。

[0076] 此外,在一个实施例中,第一管理系统106和第二管理系统108可以将相应的自动更新120a-b发送给彼此。自动更新120a-b在一个系统上修改所导入的模块而使得在另一系统上需要更新时被发送。自动更新120a-b将在下文详细描述。

[0077] 图2是示出了PLM系统中的分级的示例性实施例的框图200。示例的分级包括芯片202,其与随机存取存储器(RAM)204和中央处理单元(CPU)206二者均相关。芯片202与RAM 204和CPU 206二者均具有“使用”关系,表明RAM 204和CPU 206是芯片22的子组件。

[0078] PLM系统中的数据用于操作,例如,跟踪构成产品的件(例如,芯片202、RAM 204和CPU 206)的开发状态。PLM还可以跟踪在构成产品的件诸如芯片202、RAM 204和CPU 206中所发现的问题208(例如,缺陷/漏洞)。问题208指示关于CPU 206所发现的问题。问题208还可以指示由于问题208所出现的缺陷210。缺陷210链接到CPU 206,因为问题208直接与CPU 206相关。缺陷210还与芯片202具有‘影响’关系,因为CPU 206是芯片202的子组件,因此CPU 206所存在的缺陷210影响芯片202。

[0079] 在这些情况中的多种情况下,汇总数据用于提供产品的总体状况。例如,诸如计算机芯片的产品可以由多个组件构成,诸如RAM存储器和中央处理单元(CPU)。如果发现CPU有问题并且出现了缺陷,则芯片的管理者可以请求表明该缺陷存在于子组件内并且因此影响具有CPU和RAM的芯片的汇总报告。PLM精确地跟踪包括较高级产品对子块的分级连接/使用的数据。

[0080] 图3是示出了根据本发明的实施例的从SCM系统301向PLM系统350发送模块的框图300。

[0081] SCM系统301包括接口模块302,接口模块302具有基于应用编程接口(API)的图形用户接口(GUI)和/或命令行接口(CLI)。GUI是为终端用户提供的以用于访问SCM系统301上的功能从而请求代码模块的具体分级与PLM系统350同步。GUI包括使得顶级代码模块的细节的用户输入能够同步以及使得版本的标识从SCM系统301推送到PLM的形式。系统允许同时或同期地推送对象的多个不同的版本。另外,该形式允许用户标识是推送代码模块的分级还是仅推送特定的级(例如,仅第一级)。该系统还支持干运行模式,其支持将假设操作的细节报告给用户,而不实施操作。该系统还支持报告模式,其控制返回的输出的细节。

[0082] 接口模块还提供CLI,例如利用由DesignSync®产品所支持的工具命令语言(TCL)扩展语言。CLI包括能够被提供有经由GUI形式规定的相同输入的命令行。

[0083] SCM系统301还包括分级服务模块304,其提取关于数据模块分级的信息以便与PLM系统350同步。在示例性实施例中,分级服务模块304例如采用SCM系统301的showhrefs函数来创建XML格式的结构以便传递到PLM系统350进行处理。分级服务模块304将所创建的结构传递给PLM系统350并且等待结果。返回结果包括所执行的操作的记录,其被报告给用户,加上由下文所述的面包屑管理系统310所处理的附加信息。

[0084] 该系统还包括与分级服务304可操作地耦合的定制层306。定制层306,在分级信息提取之后,允许修改或扩展数据。这包括任选的TCL函数,其能够由终端用户或系统管理者提供。TCL函数,在被调用时,将提取的数据作为变元来传递。TCL函数能够修改数据以改变

值，或者利用用于每个数据元素的附加信息来扩展数据。附加值随后传递到PLM系统350。

[0085] SCM系统301还包括过程配置模块308。PLM系统350被配置为响应于SCM系统301而实施的导入过程可以进一步从SCM系统301配置。例如，在PLM系统350上要创建的对象的类型可取决于源SCM系统301。过程配置模块从数据库读取过程配置设置并且在将数据发送到PLM系统350之前将过程配置设置应用于数据。示例的实现方式允许手动地或者经由图形工具在注册文件中设置配置参数。

[0086] SCM系统301还包括面包屑管理系统310。一旦PLM系统350处理了数据，PLM系统350将数据结构返回被增加有关于在PLM系统350中创建的对象的信息的SCM系统301。在示例性的实现方式中，增加包括用于PLM系统350中所创建的数据的唯一对象标识符。该信息随后由面包屑管理系统310来处理并且与SCM系统301的数据一起存储为性质，使得其可供其它操作使用以便在SCM系统301上使用。

[0087] SCM系统301还包括PLM服务器映射模块314以及标识映射模块316。SCM系统301知道PLM系统350的地址从而与其通信。服务器映射模块314允许PLM系统350以灵活的以及数据相关的方式提供寻址信息。组织通常地使用单个组织域PLM系统350，并且因此该系统的单一地址可满足。然而，服务器映射模块314支持具有多个PLM或者具有多个地址的分布式PLM的更复杂的设施。用户或管理者能够利用任选地为图形的配置工具来规定SCM系统301对象与PLM平台地址之间的映射。

[0088] 另外，用户的标识可在系统之间不同(例如，不同的用户名或用户ID)。例如，SCM系统301可以典型地采用用户的系统登录名来进行标识，但是PLM系统350可使用公司域标识或电子邮件地址来登录。任选的标识映射模块316提供了用于将用户的SCM系统301标识映射到PLM系统350上的标识的可定制系统。在特定的实施例中，ID在PLM系统350与SCM系统301之间相同并且不需要映射。

[0089] SCM系统301还包括通信层。在样本的实现中，RESTful Web服务318用于SCM系统301与PLM系统350之间的通信。可以使用提供用于将描述分级的数据以及另外用户标识信息从SCM系统301传递到PLM系统350的方案以及提供更新后信息的返回的其他通信手段。

[0090] PLM系统350包括标识映射模块336以便任选地执行同样到PLM系统350上的标识映射。提供到PLM系统350和SCM系统301上的标识映射系统允许更大的灵活性，因为a)如果来自全部SCM服务器的用户ID一致，则在PLM系统350中实现单一标识映射是足够的，以及b)如果单独的SCM服务器具有不同的用户ID，但是存在单个PLM系统350，则在每个SCM服务器处实现标识映射是恰当的。示例性实现的SCM系统301侧的标识映射模块将来自SCM系统301的标识映射到PLM系统350上的标识符。

[0091] 在PLM系统350上的分级服务324接受来自SCM系统301的数据并且控制整个过程。分级服务324调用例程来创建PLM系统350中的不同的对象并且将所创建的对象链接在一起而形成PLM系统350中的分级。分级服务324还为数据增加面包屑信息并且还将操作的记录返回SCM系统301。

[0092] 在新数据创建之后，PLM系统350的定制层326为数据增加已经从SCM系统301传递的附加信息或者允许以受原配置设置禁止的方式另外操纵数据。例如，如果功能的单独使用想要不同于缺省地命名对象，则定制层326可以重命名对象。向任选的重命名方法传递有关所导入的以及在PLM系统350中创建的等同新对象的全部信息。

[0093] PLM系统350可进一步通过提供由过程配置模块328读取和应用的设置来配置。例如,PLM系统350能够配置以此方式创建的对象的类型。例如,配置可以经由具有用于各种规定的配置参数的可定制值的性质文件。

[0094] 对象创建模块330在PLM系统350中创建对象。对象创建模块330利用来自SCM系统301的信息和配置参数来确定待创建的对象的细节。另外,创建父对象模块332可以创建相关联的父对象。在样本实现的情况下,这包括被配置为创建对象的附加版本以充当根版本或者在过程版本中工作以支持版本的整个派生树的自动创建模块334。自动创建最新模块340能够创建充当模块基于所采集的上述全部信息所创建的对象的容器的辅助父对象。创建历史模块334能够同样将模块的过去的版本信息装载到由自动创建最新模块340所创建的模块中。添加标签模块342还可以被配置为将标签或别名添加到所创建的条目上。

[0095] 例如,对象创建涉及到缺省的产品对象的创建以及表示来自SCM系统的数据的版本的修订,还有实际导入的版本。新对象连接在一起而形成产品分级。

[0096] 另外,产品能够连接到父模型和产品线组件,它们是在PLM系统中使用用于产品开发过程的一般管理的其它对象类型。

[0097] 图4是图示说明本发明所采用的过程的示例性实施例的流程图400。响应于PLM系统从SCM系统所开始的过程接收到来自SCM系统的导入数据,PLM系统处理导入数据中的条目以在PLM系统中创建对象(402)。PLM系统判定子条目是否存在(404)。如果是,则系统处理父条目中的每个子条目(406)并且将每个子条目连接到父条目(408)。然后,PLM系统判定每个所创建的子条目是否存在子条目(404)。如果不存在子条目,则过程结束当前子条目分支(410)。最终,全部的分支结束(410)并且该过程结束。

[0098] 图5是图示说明将PLM系统550对象数据导入SCM系统501的示例性实施例的框图500。

[0099] PLM系统550包括便于用户开始将对象分级导出到SCM系统501的图形用户接口(GUI)534。GUI 534包括允许用户规定在SCM系统501中所创建的对象是否连接作为锥形结构或对等结构的形式。锥形结构将分级中的子对象的文件组织成在父目录的目录下的物理目录。对等结构将分级中的子对象的文件组织成并行的物理目录。GUI形式还允许用户规定导出是初始导出还是更新导出。该设置控制关于条目是否在SCM系统501内重用的因素。

[0100] PLM系统550还存储SCM连接信息,其标识目标SCM系统服务器以及在该服务器上的哪里定位有表示来自PLM系统的顶级对象的对象。例如,这是经由:

[0101] a)库,其标识SCM服务器;

[0102] b)路径,其标识对象在该服务器上的位置;以及

[0103] c)选择器,其标识链接到PLM对象的SCM对象的版本。

[0104] PLM系统550的实现方式能够直接利用API函数来调用导出控制模块524,允许系统的全面定制控制。

[0105] PLM系统550还包括缺省分级提取模块530。例如,缺省提取模块530提供缺省提取过程,然而缺省过程可以被超驰(overridden)。缺省分级提取模块530提取关于PLM系统550中的数据对象分级的信息以便导出到SCM系统501。在样本实现方式中,提取采用数据库查找和查询来提取产品对象的分级。提取过程进一步创建传递到导出控制模块524的数据结构(例如,Java数据结构)。

[0106] PLM系统550还包括定制性质模块532。定制性质模块532能够提供对导入的定制，或者甚至提供对上述的缺省分级提取过程的定制。定制性质模块532修改一组性质或配置值，其能够控制例如相对于父或顶级对象的存储位置在哪里创建SCM系统中的新对象。

[0107] PLM系统550还包括作为缺省提取模块530的替代的模块定制提取模块528。定制提取模块528可以例如处理PLM系统550中的不同类型的对象，或者规定用于确定在SCM系统501中表示分级中的哪些对象以及哪些对象仅为PLM目的而呈现并且因此不需要拷贝到SCM系统501的定制方案。

[0108] 导出控制模块524随后控制其余的导出过程。导出控制模块524被配置为：

[0109] a)调用过程来执行在标识映射模块522处的任何标识映射；

[0110] b)调用web服务层520以将全部数据传递到SCM服务器；

[0111] c)接收自SCM系统501返回的结果；

[0112] d)将任何到数据的本地连接(例如，面包屑)存储到SCM服务器的面包屑管理模块508上；以及

[0113] e)显示返回给用户的整个过程的报告(或者将这作为记录脚本返回给调用程序)。

[0114] PLM系统550还包括对象连接模块526。从SCM系统501传回的数据包括匹配PLM对象的所创建的SCM对象的地址。每个创建的SCM对象的地址包括实际对象标识符(例如，为URL的形式)以及标识SCM对象的版本以便链接到PLM对象的选择器。该信息与PLM对象一起存储以形成与对应的SCM对象的连接。

[0115] PLM系统550包括标识映射模块522(任选地)。用户的标识可以在系统之间不同。例如，SCM系统501典型地能够使用用户的系统登录名来用于标识，但是PLM系统550可能使用组织域标识或电子邮件地址进行登录。标识映射模块522提供了可定制系统，用于将PLM系统550已知的用户标识映射到SCM系统501上的标识。标识映射是任选的：在特定的系统中，用户的标识在两个系统之间相同，并且映射是不必要的。

[0116] PLM系统550还包括通信层520(例如，基于SOAP的web服务)，用于在PLM系统550与SCM系统501之间通信。可替代地，SCM系统501、PLM系统550和SCM系统501、或者另一实体可以包括通信层520。可采用其它通信手段，提供将描述分级的数据和用户标识信息从PLM系统550传递到SCM系统510、返回更新的信息、记录输出以及返回值的方案。

[0117] SCM系统501可以包括标识映射模块502。标识映射任选地执行在SCM系统501以及PLM系统550上。在两个系统上提供标识映射允许有更大的灵活性：a)如果在全部SCM服务器上使用的用户名一致，则在PLM系统中实现单一标识映射是足够的；b)如果单独的SCM服务器具有不同的用户标识，但是存在单个PLM系统，则在每个SCM服务器上实现标识映射是恰当的。在样本实现方式的SCM侧的标识映射模块提供了待供给的定制函数，其将来自PLM系统的标识映射到SCM系统上的标识符。

[0118] SCM系统501包括具有定制层506的导入控制模块504。导入控制模块504接受来自PLM系统550的数据并且控制导入。导入控制模块504在SCM系统501中创建不同的对象并且将它们链接在一起而在SCM系统501中形成分级。导入控制模块504进一步为数据增加面包屑信息并且还将操作的记录返回PLM系统550。

[0119] 定制层506可以在导入数据之前或之后被激活。在导入之前，定制层506能够在其从PLM系统550接收到时修订数据。例如，定制层506标识并去除不适合在SCM系统510上表示

的条目。在导入后，向定制层506传递来自SCM系统501的数据以及关于新创建的SCM对象的信息。例如，定制层506接收来自PLM系统550的信息并且设置SCM对象上的性质(例如，作者值能够从PLM系统传递且在SCM对象上设置)。

[0120] SCM系统501还包括面包屑管理模块508。在SCM系统中创建数据时，面包屑被存储以将新SCM条目与PLM系统中的对象链接。在示例性的实现方式中，从PLM系统550发送的数据包括用于SCM对象的对象标识符(例如，整数是对象的唯一标识符)。来自PLM系统550的对象标识符被设置为在SCM系统501中所创建的对象(例如，模块、分支和/或版本)的性质。

[0121] SCM系统501还包括数据预校验模块。SCM系统另外包括模块(或对象)创建模块512。模块创建模块512被配置为，如果模块尚未存在，则在SCM系统501中创建模块表示来自PLM系统550的模块。分支创建与加标签模块514进一步能够基于其版本进展来创建模块的不同分支并且能够进一步将版本数据作为标签加到模块上。分级引用创建模块516能够创建所创建的模块与现有模块之间的引用表示SCM系统501中的模块当存在于PLM系统550上时模块之间的分级关系。

[0122] 图6是图示说明本发明所采用的用来从PLM系统导入数据到SCM系统的过程的示例性实施例的流程图600。SCM系统开始于预校验数据(602)。在实施例中，预校验可以包括校验数据的“过失”误差，这可通过“静态”校验来发现，从而防止过程后来失败。预校验中的标准的示例可以包括但不限于，校验所传入的“选择器”值和“名称”在目标系统上是有效的。如果数据预校验失败，则过程结束(604)。

[0123] 然而，如果预校验通过，则SCM系统处理数据的顶级条目(606)。SCM系统判定导入是初始导入模式还是现有导入模式(608)。如果这是初始导入，则系统判定SCM对象是否存在于SCM系统中(610)。如果是，则导入的模块可能是重新使用的预存在模块，并且因此在该点以下的全部数据已经是正确的，并且过程可以在该点结束(612)。如果对象不是预先存在的模块，而不是创建新的SCM对象，现有的SCM对象能够更新为到子对象的新链接集合。仍可以需要创建新的子对象。

[0124] 如果模块不存在(610)或者系统不处于初始模式(608)，则系统SCM系统如果有必要则创建表示PLM对象的对应模块(例如，如果不存在表示已经存在于SCM中的PLM对象的模块)(614)，创建具有包括初始版本的版本集合的所创建的SCM对象的开发分支(616)，并且经由标签模块来添加标签(618)。如果来自PLM系统的数据表明该对象是SCM数据的固定发布版本，则该系统加上版本数据这一标签，创建到SCM数据的版本的固定引用。然而，如果PLM数据表明对象不是进展中的工作，则实现到SCM数据的加标签分支的连接。换言之，其连接到该分支上的数据的最新版本。PLM系统能够因此表示固定/最终模块和进展中的开发二者。

[0125] SCM系统则将子条目连接到其父条目(620)。对于分级中的除了顶级条目之外的所有条目，系统利用SCM命令将子条目连接到其父对象以实现该连接。在示例的DesignSync®系统中，这包括创建从父模块到子模块的href(分级引用)、并且该href标识a)子模块，b)要使用的子模块版本(例如，作为版本选择器)，以及c)当子模块从SCM系统取回时指示相对于父模块在哪放置子模块数据的href路径。该最后一个是锥形或对等结构的href路径值得以实现的地方。对于锥形结构，href路径值是文件系统上的子目录名称，其将数据放到父模块数据的下方。对于对等结构，href路径值为“../directory-name”的形式，

意味着数据放置为与文件系统上的父模块数据并行。

[0126] SCM系统随后判定子条目是否存在(622)。在其它实施例中,过程的次序可以不同,因为取决于SCM系统及其能力,例如,可能需要在创建父对象之前递归地创建全部子条目。然而,在该示例中,递归步骤发生于创建父对象之前。如果子条目存在,则系统处理子条目(606)。如果不存在,则SCM系统结束过程(624)。

[0127] 图7图示说明了图示出本发明所采用的在另一系统中的对应模块变化时自动更新SCM或PLM系统中的导入信息的过程的示例性实施例的流程图700。自动更新过程在高级别上相似,无论是分别将数据从PLM移到SCM,还是从SCM移到PLM,然而实现细节可以不同。下面描述了SCM系统701的用户702开始的更新,导致结构变化704,然而,相似的描述可应用于PLM系统750,其用户722和对应的结构变化724。

[0128] 当SCM系统701的用户702执行操作而导致数据的结构变化时,过程开始(704)。该变化可以是如下的任意组合:

[0129] (1)从一个对象到子对象的分级引用的添加或去除。分级引用是从结构中的较高级对象到较低级对象的引用(例如,在芯片模块与CPU之间的链接);或者

[0130] (2)分级引用的修改(例如,在两个系统之间反映出的分级引用的一些方面的修改)。该修改可以是例如待使用的引用对象的版本的变化或者是待用于决定子对象数据相对于父对象放置在文件系统中的哪里的相对路径(例如,rel路径)的变化。

[0131] 结构变化会影响整体分级结构:添加或修改到子块的单一引用会从该点向下改变到整个分级。

[0132] 在手动更新中(706),在SCM系统701和PLM系统750上为用户所提供的GUI访问功能以请求对象的特定分级导出(708)到另一系统。

[0133] 为了提供自动更新,本发明的实施例在对数据做出改变时自动被触发(716)。对于修改数据的结构的全部改变都调用触发程序,确保总是执行数据的同步。触发程序被交互地调用,使得如果在单次用户操作中执行多个结构改变则仅调用触发程序一次。这通过将改变一次传递给目标系统而避免了不必要的工作/数据传输。在触发程序激活时,SCM系统701访问存储的面包屑以将PLM系统750中的对象链接到SCM系统701,或者反之亦然。这确保了将用于两个系统上所表示的数据的更新传递到目标系统。在第一系统中可能有不打算反映在第二系统上的数据。自动更新系统通过如下避免反映不必要的数据:手动地或者由于引用父对象的同步引起的块的第一同步,以及基于面包屑的存在的后续自动同步。

[0134] SCM系统701随后执行自动更新步骤(710),其调用导出过程,这在上文针对每个系统进行了说明(708)。

[0135] 在PLM系统上,更新过程(734)利用相同的API关于每个系统执行与终端用户所执行的相同类型的上述结构改变。重要的是,修改更新过程(734)以在调用系统上设置旗语标记(732)。旗语标记防止在对目标系统做出改变时自动触发程序击发,这使得数据被重新送回调用系统,可能导致无限循环。实现旗语(732)阻止了触发程序进行更新过程所做的更新,但是允许触发程序做出来自用户的其他改变。在多线程系统中,比如典型的SCM系统和PLM系统,存在触发程序在更新过程发生的同时被禁用的危险。禁用则阻止了对不同的线程的某些动作击发触发程序。旗语的实现取决于所使用的SCM系统和PLM系统的能力。然而,一个示例的实现包括(1)阻止全部新线程开始;(2)等待其他所有运行中的线程结束;(3)阻止

触发程序;(4)执行更新过程所需的更新;(5)解除对触发程序的阻止;以及(6)允许新线程开始。

[0136] 旗语过程(732)确保了在触发程序被阻止的同时不发生其他操作。更精巧的系统采用用于更新过程所需的全部数据库改变的交易并且仅在当该交易正在提交给数据库时的期间才阻止触发程序,利用了交易提交给数据库是单线程的事实。

[0137] 当PLM系统750的用户722执行导致改变数据的结构的操作时,沿其他方向的过程开始(724)。该改变可以是如下任意组合:

[0138] (1)从一个对象到子对象的分级引用的添加或去除。分级引用是从结构中的较高级对象到较低级对象的引用(例如,芯片模块与CPU之间的链接);或者

[0139] (2)分级引用的修改(例如,在两个系统之间反映的分级引用的某方面的修改)。该修改可以是例如待使用的引用对象的版本的改变或者是待用于决定子对象数据相对于父对象放入文件系统中的何处的相对路径(例如,rel路径)的改变。

[0140] 结构改变会影响整体分级结构:添加或修改对子块的单一引用能够从该点向下改变到整个分级。

[0141] 在手动更新(726)中,在PLM系统750和SCM系统701上为用户提供的GUI访问请求对象的特定分级导出(728)到另一系统的功能。

[0142] 为了提供自动更新,本发明的实施例在对数据做出改变时自动地被触发(736)。对于修改数据的结构的全部改变调用触发程序,确保总是执行数据的同步。触发程序被交互地调用,使得如果在单次用户操作中执行了多个结构改变,则仅调用触发程序一次。这通过将改变一次传递到目标系统来避免不必要的工作/数据传输。在触发程序激活时,PLM系统750访问存储的面包屑以将SCM系统701中的对象链接到PLM系统750,或反之亦然。这确保了对于在两个系统上所表示的数据,将更新传递到目标系统。在第一系统中存在不打算反映在第二系统上的数据。自动更新系统通过如下来避免反映不必要的数据:手动或者由于引用父对象的同步引起的块的第一同步,以及随后基于面包屑的存在的自动同步。

[0143] PLM系统750随后执行自动更新步骤(730),其调用上文对于每个系统所描述的导出过程(728)。

[0144] 在SCM系统701上,更新过程(714)利用相同的API关于每个系统执行与终端用户所执行的相同类型的上述结构改变。重要的是,修改更新过程(714)以在调用系统上设置旗语标记(712)。旗语标记防止在对目标系统做出改变时自动触发程序击发,这使得数据被重新送回调用系统,可能导致无限循环。实现旗语(712)阻止了触发程序进行更新过程所做的更新,但是允许触发程序做出来自用户的其他改变。在多线程系统中,比如典型的SCM系统和PLM系统,存在触发程序在更新过程发生的同时被禁用的危险。禁用则阻止了对不同的线程的某些动作击发触发程序。旗语的实现方式取决于所使用的SCM系统和PLM系统的能力。然而,一个示例的实现方式包括(1)阻止全部新线程开始;(2)等待其他所有运行的线程结束;(3)阻止触发程序;(4)执行更新过程所需的更新;(5)解除对触发程序的阻止;以及(6)允许新线程开始。

[0145] 旗语过程(712)确保了在触发程序被阻止的同时不发生其他操作。更精巧的系统采用用于更新过程所需的全部数据库改变的交易并且仅在当该交易正在提交给数据库时的期间才阻止触发程序,利用了交易提交给数据库是单线程的事实。

[0146] 对于自动更新修改上述的导出过程和导入过程。具体地，过程的导出部分从源系统采集数据。在自动更新中，任何现有的面包屑或跟踪数据连同结构一起被提取。过程的导入侧执行工作的大部分来创建或修改目标系统上的数据。在自动更新中，当检查对象是否已经存在于系统上时，使用面包屑数据。如果存在面包屑数据，则执行存在检查以确保对象存在。如果对象已经存在，则从该对象到子对象的引用仍需要更新。新的子对象可能已经添加，并且已经创建了对应的新引用。到子对象的引用可能已经在源系统上去除，并且在目标系统上的对应的引用可能也需要去除。引用的属性(例如，当取得数据时所使用的href路径或目标路径的版本)可能已经改变。

[0147] 同样，如果目标对象是静态版本，则也不可能修改目标对象。在PLM系统750中的对象的静态版本代表了发布的条目。这些静态版本不能修改，因为它们代表了在过去的固定点的状态。改变可能对于这些条目而言是不允许的，因为SCM系统701本身防止在这样的情况下的改变。如果在静态版本上的改变被尝试，则导入过程精确地将问题报告回给SCM系统701，以便用户在该级处进行校正。

[0148] 对于自动更新修改上述的导出和导入过程。具体地，过程的导出部分从源系统采集数据。在自动更新中，任何现有的面包屑或跟踪数据连同结构一起被提取。过程的导入侧执行工作的大部分以创建或修改目标系统上的数据。在自动更新中，当检查对象是否已经存在于系统上时，使用面包屑数据。如果存在面包屑数据，则执行存在检查以确保对象存在。如果对象已经存在，则从该对象到子对象的引用仍需要更新。新的子对象可能已经添加，并且可能已经创建了对应的新引用。对子对象的引用可能已经在源系统上去除，并且在目标系统上的对应的引用也需要去除。引用的属性(例如，当取得数据时所使用的目标路径或href路径的版本)可能已经改变。

[0149] 同样，如果目标对象是静态版本，则也不可能修改目标对象。在SCM系统中的对象的静态版本代表了发布的条目。这些静态版本不能修改，因为它们代表了在过去的固定点的状态。改变可能对于这些条目而言是不允许的，因为PLM系统本身防止在这样的情况下的改变。如果在静态版本上的改变被尝试，则导入过程精确地将问题报告回给PLM系统，以便用户在该级处进行校正。

[0150] 理想地，结构的改变在单次操作内限于设计的单层。例如，操作可能添加从顶级芯片到新ALU块的引用，但是不会同时修改芯片已经引用的CPU块下的结构。如果这得以保障，则可以优化整个更新过程以使结构的单层在系统之间传递且一次更新。然而，在大多数系统中，这不能得到保障，并且在给定的示例中，ALU可以是在另一系统中所要表示的整个新分级结构。

[0151] 另外，为了安全以及处理任何错误状况，过程强制整个分级的重新同步。因此，系统对于任意操作总是发送整个结构(例如，从顶级修改的对象向下)，并且在导入时针对目标系统对此进行校验。该方法从性能和系统要求方面看更繁重，但是更精确。

[0152] 图8图示说明了可以在其中实现本发明的实施例的计算机网络或类似的数字处理环境。

[0153] 客户端计算机/设备50以及服务器计算机60提供了执行应用程序等的处理、存储和输入/输出设备。客户端计算机/设备50还能够通过通信网70链接到其他计算设备，包括其他的客户端设备/过程50以及服务器计算机60。通信网70可以是远程访问网络的部分、全

局网络(例如,因特网),世界范围内的计算机的集合、局域网或广域网、以及当前使用相应的协议(TCP/IP,Bluetooth®等)彼此通信的网关。其他电子设备/计算机网络体系结构是适合的。

[0154] 图9是在图8的计算机系统中的计算机(例如,客户端处理器/设备50或者服务器计算机60)的示例的内部结构的示意图。每个计算机50、60包含了系统总线79,其中总线是用于计算机或处理系统的组件之间的数据传输的硬件线路的集合。系统总线79实质上是连接实现元件之间的信息传输的计算机系统的不同元件(例如,处理器、磁盘存储设备、存储器、输入/输出端口、网络端口等)的共享管道。附接到系统总线79的是I/O设备接口82,其用于将各个输入和输出设备(例如,键盘、鼠标、显示器、打印机、扬声器等)连接到计算机50、60。网络接口86允许计算机连接到与网络(例如,图8的网络70)附接的其他各种设备。存储器90提供了用于实现本发明的实施例(例如,上文详述的提取模块、创建模块和映射模块)的计算机软件指令92和数据94的易失性存储。磁盘存储设备95提供了用于实现本发明的实施例的计算机软件指令92和数据94的非易失性存储。中央处理单元84也附接到系统总线79并且提供计算机指令的执行。

[0155] 在一个实施例中,处理器例程92和数据94是计算机程序产品(通常表示为92),包括提供用于本发明系统的软件指令的至少一部分的非暂态计算机可读介质(例如,可移除存储介质,诸如一个或多个DVD-ROM、CD-ROM、磁盘、磁带等)。计算机程序产品92能够由任何适合的软件安装程序来安装,这是本领域公知的。在另一实施例中,软件指令的至少部分还可以经由电缆通信和/或无线连接下载。在其他实施例中,本发明的程序是具体体现为传播介质(例如,无线电波、红外波、激光波、声波或经由诸如因特网的全局网络或其他网络传播的电波)上的传播信号的计算机程序传播信号产品。该载波介质或信号可以用来提供用于本发明例程/程序92的软件指令的至少一部分。

[0156] 虽然参考本发明的示例性实施例特别地显示和描述了本发明,但是本领域技术人员将理解的是,可以在其中做出形式和细节上的各种改变,而不背离随附的权利要求所涵盖的本发明的范围。

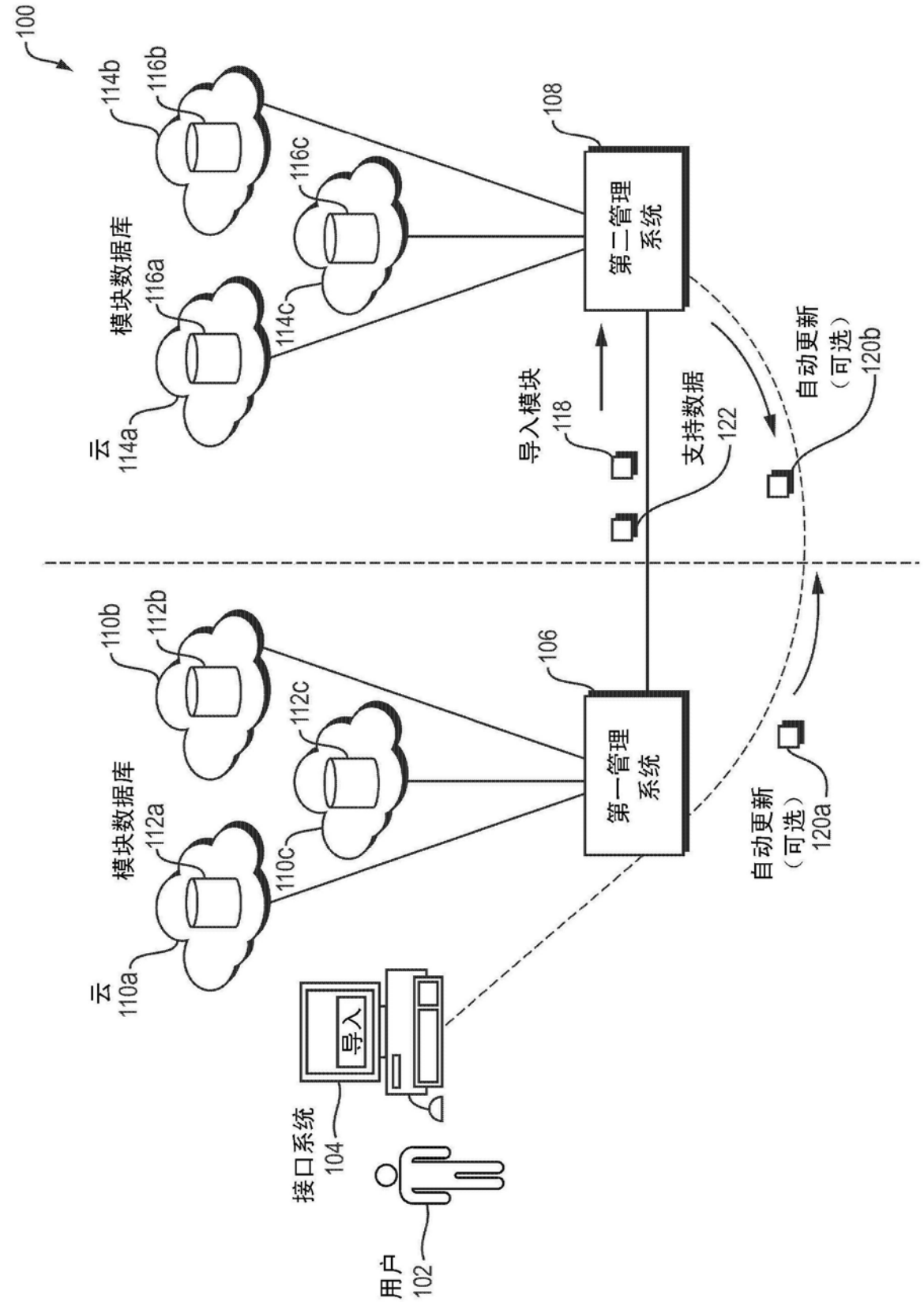

100
114b
116b
116c
模块数据库
116a
114c
云
114a
108
第二管理
系统
导入模块
118
支持数据
122
自动更新
（可选）
120b
110b
112b
112c
106
模块数据库
112a
110c
第一管理
系统
自动更新
（可选）
120a
云
110a
导入
接口系统
104
用户
102

图1

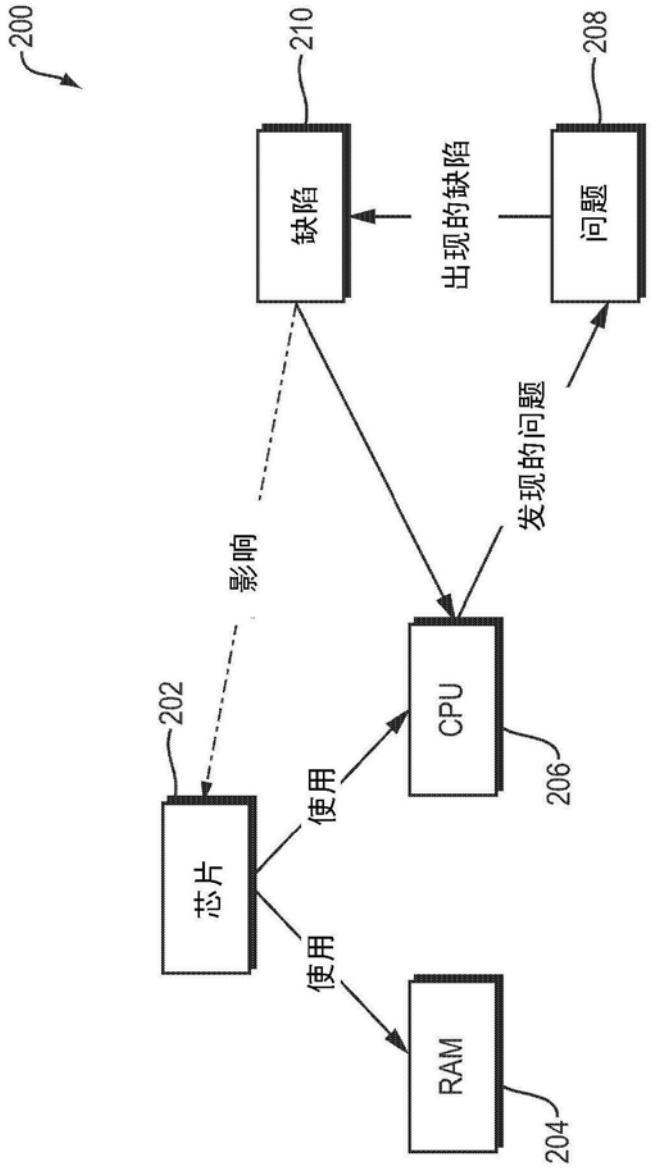


图2

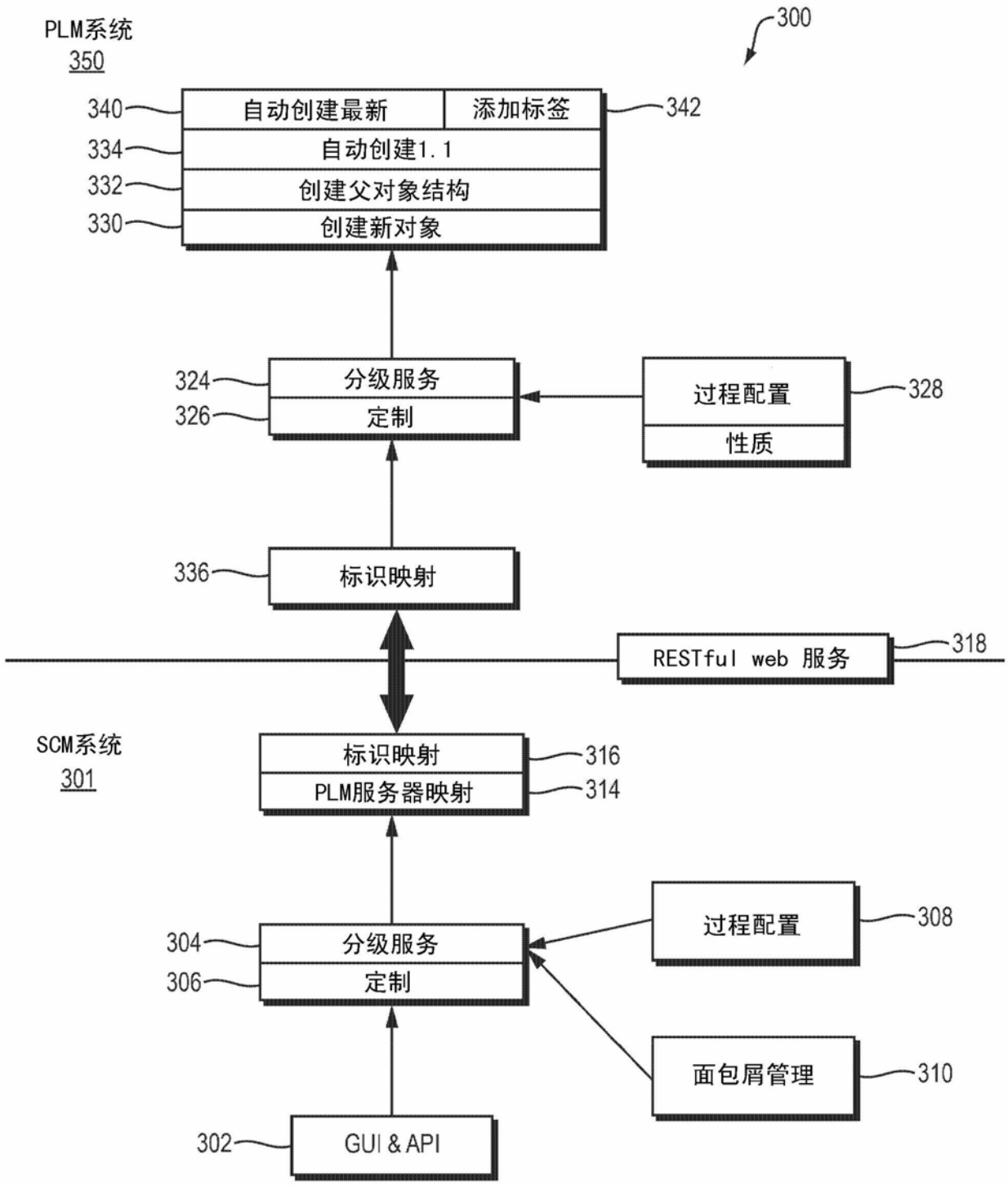

PLM系统
350
300
340
自动创建最新
添加标签
342
334
自动创建1.1
332
创建父对象结构
330
创建新对象
324
分级服务
326
定制
过程配置
328
性质
336
标识映射
RESTful web 服务
318
SCM系统
301
标识映射
316
PLM服务器映射
314
304
分级服务
306
定制
过程配置
308
面包屑管理
310
302
GUI & API

图3

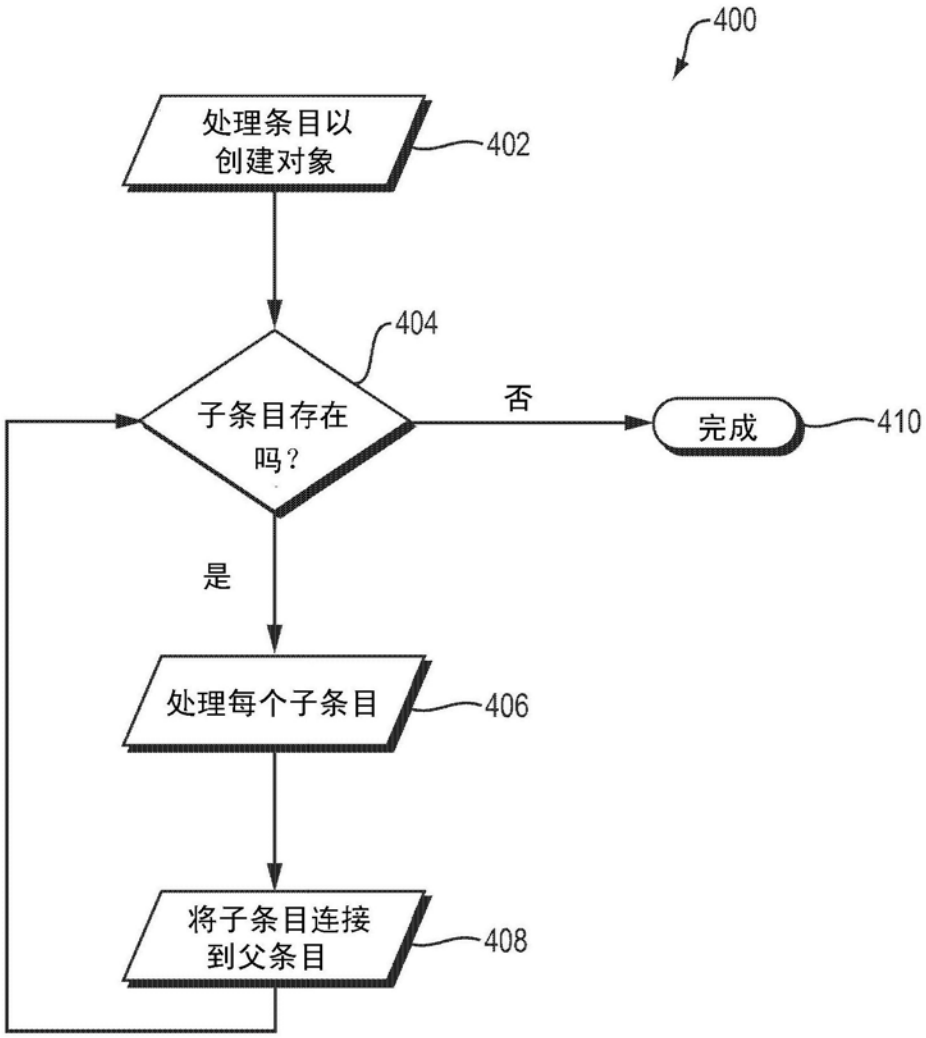

400
处理条目以
创建对象
402
404
子条目存在
吗？
否
完成
410
是
处理每个子条目
406
将子条目连接
到父条目
408

图4

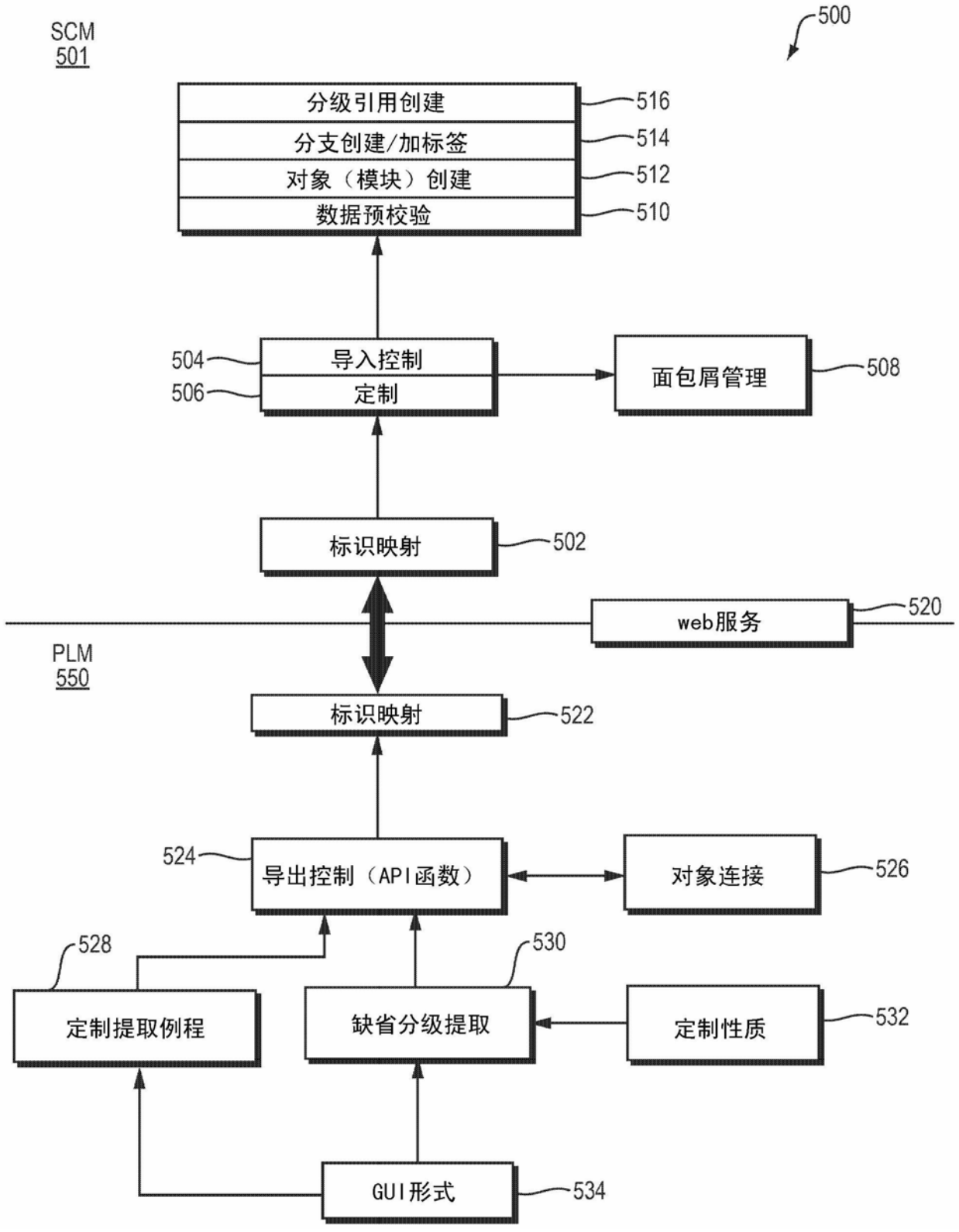

500
SCM
501
分级引用创建
分支创建/加标签
对象（模块）创建
数据预校验
516
514
512
510
504
506
导入控制
定制
面包屑管理
508
标识映射
502
web服务
520
PLM
550
标识映射
522
524
导出控制（API函数）
对象连接
526
528
530
定制提取例程
缺省分级提取
定制性质
532
GUI形式
534

图5

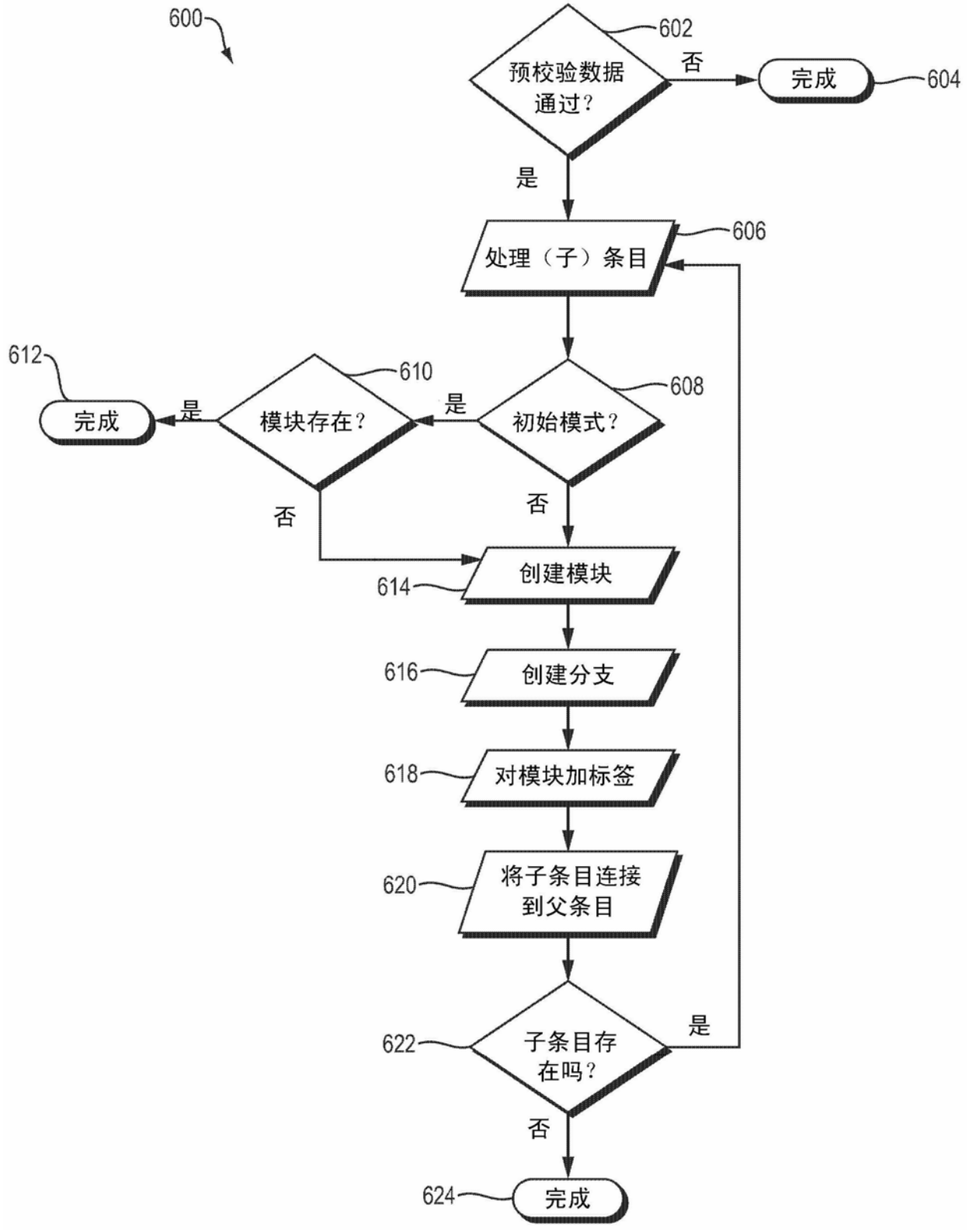

600
602
预校验数据通过？
否
完成
604
是
606
处理（子）条目
612
610
608
完成
是
模块存在？
是
初始模式？
否
否
创建模块
614
616
创建分支
618
对模块加标签
620
将子条目连接到父条目
622
子条目存在吗？
是
否
624
完成

图6

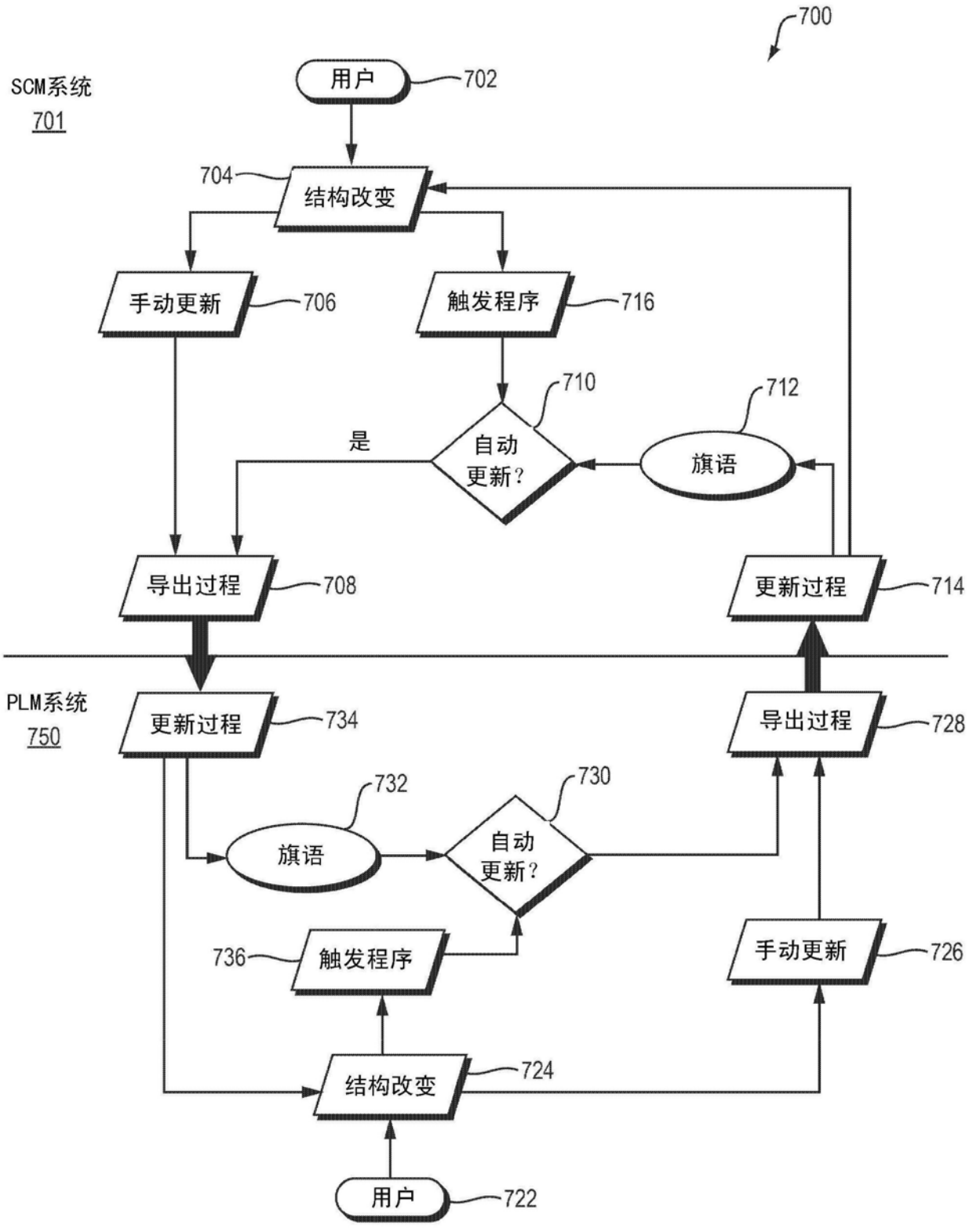

700
SCM系统
701
用户
702
704
结构改变
手动更新
706
触发程序
716
710
712
是
自动
更新?
旗语
导出过程
708
更新过程
714
PLM系统
750
更新过程
734
导出过程
728
732
730
旗语
自动
更新?
736
触发程序
手动更新
726
结构改变
724
用户
722

图7

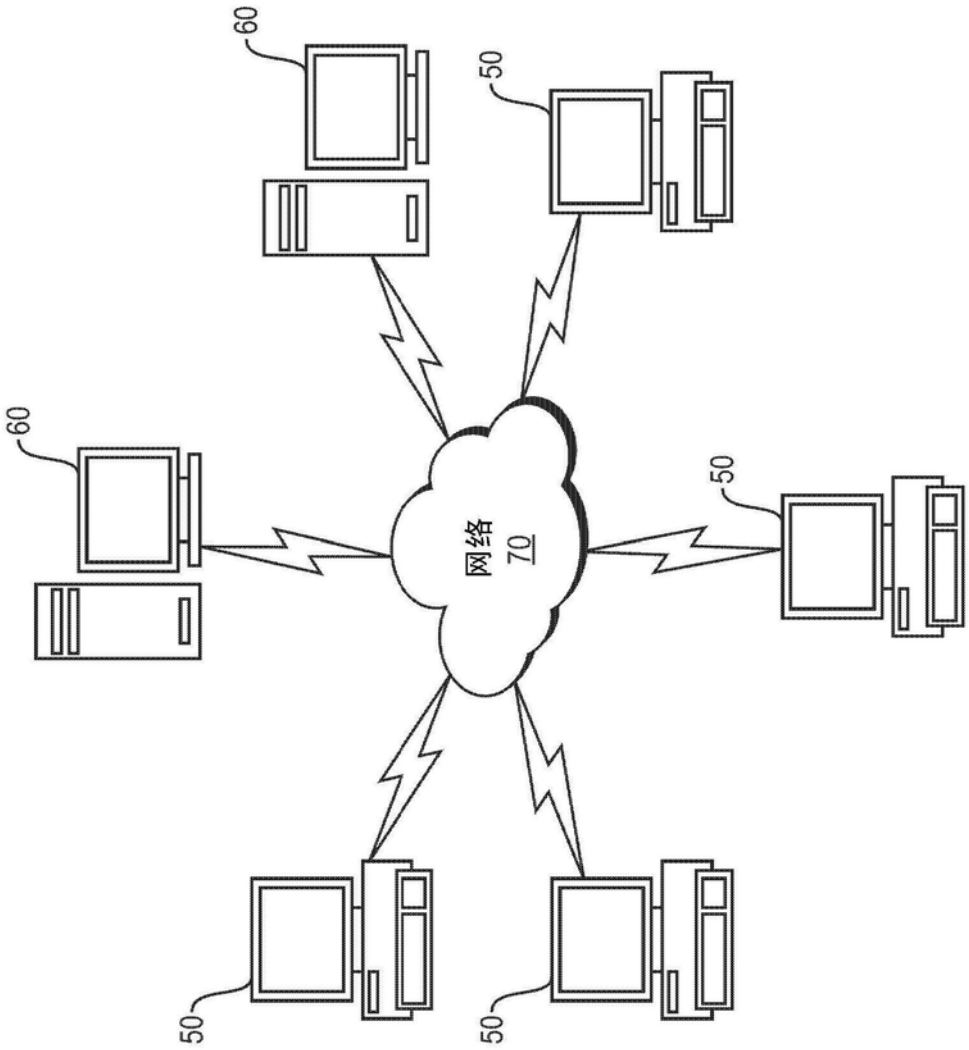


图8

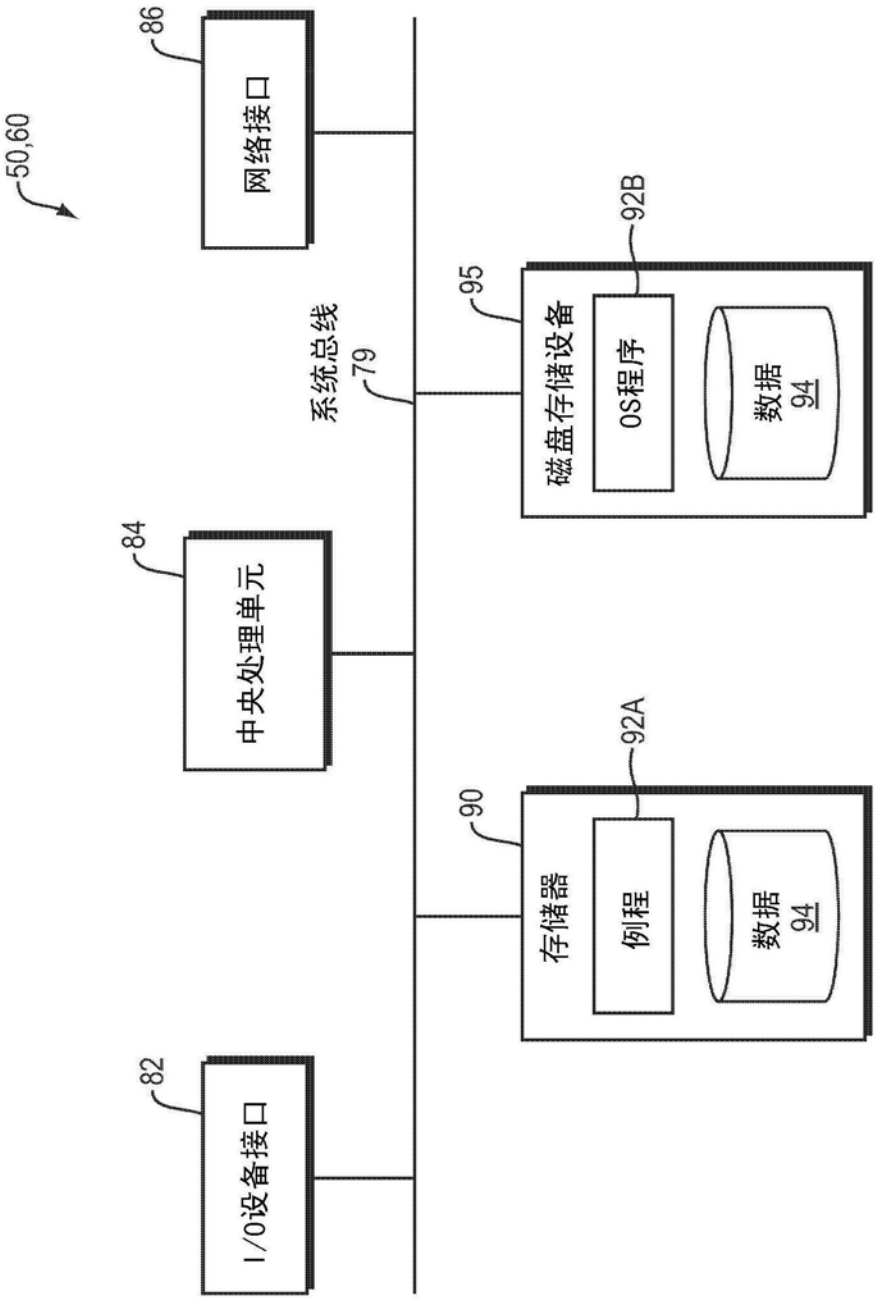

50,60
网络接口
86
系统总线
79
中央处理单元
84
I/O设备接口
82
磁盘存储设备
95
OS程序
92B
数据
94
存储器
90
例程
92A
数据
94

图9