(19)国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 120226304 A
(43)申请公布日 2025.06.27

(21)申请号 202380079992.9

(22)申请日 2023.11.13

(30)优先权数据
18/058290 2022.11.23 US

(85)PCT国际申请进入国家阶段日
2025.05.19

(86)PCT国际申请的申请数据
PCT/CN2023/131187 2023.11.13

(87)PCT国际申请的公布数据
WO2024/109575 EN 2024.05.30

(71)申请人 国际商业机器公司
地址 美国

(72)发明人 L·A·巴腾 E·K·巴特勒
M·H·科克 C·D·库克
A·K·萨顿 S·C·汤普森

(74)专利代理机构 北京市中咨律师事务所 11247
专利代理师 于静 刘薇

(51)Int.Cl.
H04L 9/32(2006.01)

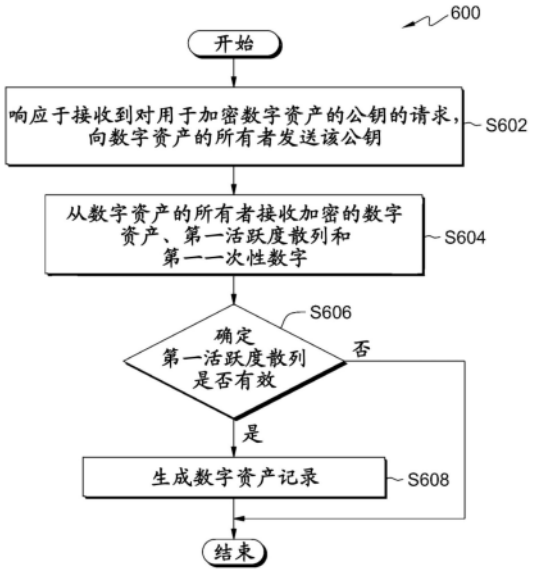
权利要求书4页 说明书26页 附图16页

(54)发明名称

验证数字资产的所有权和认证数字资产的方法

(57)摘要

公开了一种用于验证数字资产的所有权的计算机实现的方法。该计算机实现的方法包括:响应于接收到对用于加密数字资产的公钥的请求,向数字资产的所有者发送公钥。该计算机实现的方法还包括:从数字资产的所有者接收加密的数字资产和第一活跃度散列,其中,数字资产是使用公钥加密的,第一活跃度散列是至少部分地基于未加密形式的数字资产和第一一次性数字来生成的。该计算机实现的方法还包括:确定第一活跃度散列是否有效。该计算机实现的方法还包括:响应于确定第一活跃度散列有效,生成数字资产记录,其中,数字资产记录包括加密的数字资产和第一活跃度散列。



CN 120226304 A

1.一种用于验证数字资产的所有权的计算机实现的方法,所述计算机实现的方法包括:

由一个或多个处理器响应于接收到对用于加密数字资产的公钥的请求,向所述数字资产的所有者发送所述公钥;

由所述一个或多个处理器从所述数字资产的所述所有者接收加密的数字资产和第一活跃度散列,其中,所述数字资产是使用所述公钥加密的,所述第一活跃度散列是至少部分地基于未加密形式的所述数字资产和第一一次性数字来生成的;

响应于从来自所述数字资产的所述所有者的所述加密的数字资产中接收到所述第一一次性数字和所述第一活跃度散列,确定所述第一活跃度散列是否有效;以及

由所述一个或多个处理器响应于确定所述第一活跃度散列有效,生成数字资产记录,其中,所述数字资产记录包括所述加密的数字资产和所述第一活跃度散列。

2.根据权利要求1所述的计算机实现的方法,其中,发送用于加密所述数字资产的所述公钥进一步包括:生成公钥/私钥对,其中,所述公钥/私钥对中的所述公钥被用于加密所述数字资产,并且所述公钥/私钥对中的私钥被用于解密所述加密的数字资产。

3.根据权利要求1所述的计算机实现的方法,还包括:

将所述数字资产记录存储在分布式账本或区块链环境外部的存储服务中的至少一个中。

4.根据权利要求1所述的计算机实现的方法,其中,确定所述第一活跃度散列是否有效包括:

由所述一个或多个处理器使用与所述数字资产相关联的私钥将所述加密的数字资产变换回未加密的数字资产;

由所述一个或多个处理器基于所述未加密的数字资产和所述第一一次性数字,生成第二活跃度散列;以及

由所述一个或多个处理器将所述第一活跃度散列与所述第二活跃度散列进行匹配。

5.根据权利要求1所述的计算机实现的方法,还包括:

由所述一个或多个处理器从请求实体接收用于所述数字资产的所述所有者提供所述数字资产的所有权证明的请求,其中,所述请求包括第二一次性数字;以及

响应于接收到用于所述数字资产的所述所有者提供所述数字资产的所有权证明的所述请求:

由所述一个或多个处理器将所述第二一次性数字发送到所述数字资产的所述所有者。

6.根据权利要求5所述的计算机实现的方法,还包括:

由所述一个或多个处理器从所述数字资产的所述所有者接收与数字映像相关联的映像id和第三活跃度散列,其中,所述第二活跃度散列是至少部分地基于未加密形式的所述数字资产和所述第二一次性数字而生成的。

7.根据权利要求6所述的计算机实现的方法,还包括:

由所述一个或多个处理器至少部分地基于以下操作来验证所述数字资产的所有权证明:

由所述一个或多个处理器使用所述映像id从所述数字资产记录中取得所述加密的数字资产;

由所述一个或多个处理器使用与所述数字资产相关联的所述私钥将所述加密的数字资产变换回未加密的数字资产;

由所述一个或多个处理器基于所述未加密的数字资产和所述第二一次性数字来生成第四活跃度散列;以及

由所述一个或多个处理器将所述第一活跃度散列与所述第二活跃度散列进行匹配。

8.根据权利要求4所述的计算机实现的方法,还包括:

由所述一个或多个处理器响应于验证所述数字资产的所述所有权证明,向所述请求实体发送所述数字资产的所述所有者有效的认证。

9.根据权利要求1所述的计算机实现的方法,还包括:

由所述一个或多个处理器使用在区块链账本中的可信执行环境中执行的智能合约来确定所述第一活跃度散列的共识。

10.根据权利要求1所述的计算机实现的方法,还包括:

由所述一个或多个处理器访问与所述数字资产相关联的智能合约,以验证所述第一活跃度散列与所述数字资产和所述第一一次性数字的散列相匹配。

11.根据权利要求1所述的计算机实现的方法,其中,所述数字资产是映像或多媒体。

12.根据权利要求5所述的计算机实现的方法,还包括:

将真实所有权的信息从智能合约TEE发送到智能合约。

13.根据权利要求9所述的计算机实现的方法,还包括:

将真实所有权的信息从所述智能合约发送到注册表。

14.一种用于验证数字资产的所有权的计算机程序产品,所述计算机程序产品包括一个或多个计算机可读存储介质和存储在一个或多个计算机可读存储介质上的程序指令,所述程序指令包括用于以下操作的指令:

响应于接收到对用于加密数字资产的公钥的请求,向所述数字资产的所有者发送所述公钥;

从所述数字资产的所述所有者接收加密的数字资产和第一活跃度散列,其中,所述数字资产是使用所述公钥加密的,所述第一活跃度散列是至少部分地基于未加密形式的所述数字资产和第一一次性数字来生成的;

响应于从来自所述数字资产的所述所有者的所述加密的数字资产中接收到所述第一一次性数字和所述第一活跃度散列,确定所述第一活跃度散列是否有效;以及

响应于确定所述第一活跃度散列有效,生成数字资产记录,其中,所述数字资产记录包括所述加密的数字资产和所述第一活跃度散列。

15.根据权利要求14所述的计算机程序产品,其中,发送用于加密所述数字资产的所述公钥的指令进一步包括生成公钥/私钥对的指令,其中,所述公钥/私钥对中的所述公钥被用于加密所述数字资产,并且所述公钥/私钥对中的私钥被用于解密所述加密的数字资产。

16.根据权利要求14所述的计算机程序产品,还包括用于以下操作的指令:

将所述数字资产记录存储在分布式账本或区块链环境外部的存储服务中的至少一个中。

17.根据权利要求14所述的计算机程序产品,其中,确定所述第一活跃度散列是否有效的指令包括用于以下操作的指令:

使用与所述数字资产相关联的私钥将所述加密的数字资产变换回未加密的数字资产;

基于所述未加密的数字资产和所述第一一次性数字,生成第二活跃度散列;以及

将所述第一活跃度散列与所述第二活跃度散列进行匹配。

18.根据权利要求14所述的计算机程序产品,还包括用于以下操作的指令:

从请求实体接收用于所述数字资产的所述所有者提供所述数字资产的所有权证明的请求,其中,所述请求包括第二一次性数字;以及

响应于接收到用于所述数字资产的所述所有者提供所述数字资产的所有权证明的所述请求:

将所述第二一次性数字发送到所述数字资产的所述所有者。

19.根据权利要求14所述的计算机程序产品,还包括用于以下操作的指令:

从所述数字资产的所述所有者接收与数字映像相关联的映像id和第三活跃度散列,其中,所述第二活跃度散列是至少部分地基于未加密形式的所述数字资产和所述第二一次性数字而生成的。

20.一种用于验证数字资产的所有权的计算机系统,包括:

一个或多个计算机处理器;

一个或多个计算机可读存储介质;

计算机程序指令;

所述计算机程序指令被存储在所述一个或多个计算机可读介质上以用于由所述一个或多个计算机处理器执行;以及

所述计算机程序指令包括用于以下操作的指令:

响应于接收到对用于加密数字资产的公钥的请求,向所述数字资产的所有者发送所述公钥;

从所述数字资产的所述所有者接收加密的数字资产和第一活跃度散列,其中,所述数字资产是使用所述公钥加密的,所述第一活跃度散列是至少部分地基于未加密形式的所述数字资产和第一一次性数字来生成的;

响应于从来自所述数字资产的所述所有者的所述加密的数字资产中接收到所述第一一次性数字和所述第一活跃度散列,确定所述第一活跃度散列是否有效;以及

响应于确定所述第一活跃度散列有效,生成数字资产记录,其中,所述数字资产记录包括所述加密的数字资产和所述第一活跃度散列。

21.一种用于验证数字资产的所有权的计算机实现的方法,所述计算机实现的方法包括:

由一个或多个处理器从请求实体接收用于数字资产的所有者提供所述数字资产的所有权证明的请求,其中,所述请求包括第一一次性数字;以及

响应于接收到用于所述数字资产的所述所有者提供所述数字资产的所有权证明的所述请求:

由所述一个或多个处理器将所述第一一次性数字发送到所述数字资产的所述所有者;

由所述一个或多个处理器从所述数字资产的所述所有者接收与数字映像相关联的映像id和第一活跃度散列;以及

由所述一个或多个处理器验证所述数字资产的所有权证明。

22.根据权利要求21所述的计算机实现的方法,其中,所述第一活跃度散列是至少部分地基于未加密形式的所述数字资产和所述第一一次性数字而生成的。

23.根据权利要求21所述的计算机实现的方法,其中,由所述一个或多个处理器验证所述数字资产的所有权证明是至少部分地基于:

由所述一个或多个处理器使用所述映像id从所述数字资产记录中取得所述加密的数字资产;

由所述一个或多个处理器使用与所述数字资产相关联的所述私钥将所述加密的数字资产变换回未加密的数字资产;

由所述一个或多个处理器基于所述未加密的数字资产和第二一次性数字来生成第二活跃度散列;以及

由所述一个或多个处理器将所述第一活跃度散列与所述第二活跃度散列进行匹配。

24.根据权利要求21所述的计算机实现的方法,还包括:

由所述一个或多个处理器响应于验证所述数字资产的所述所有权证明,向所述请求实体发送所述数字资产的所述所有者有效的认证。

25.根据权利要求21所述的计算机实现的方法,其中所述数字资产是映像或多媒体。

验证数字资产的所有权和认证数字资产的方法

背景技术

[0001] 本发明一般涉及数字资产领域,尤其涉及认证数字资产和验证数字资产的所有权。

[0002] 区块链是共享的、不可变的账本,其促进了在商业网络中记录交易和跟踪资产的过程。资产可以是有形的(房屋、汽车、现金、土地)或无形的(知识产权、专利、版权、品牌)。不可替换代币(NFT)是与特定数字或物理资产相关联的区块链上的记录。NFT铸造是取得数字文件并将其变成可以被存储在区块链上的数字资产的过程。一旦NFT变成数字资产,就可以使NFT进入流通,并且可以经由智能合约来销售NFT。智能合约是计算机程序或交易协议,其旨在根据合约或协议的条款自动地执行、控制或记录法律相关事件和动作。例如,当NFT从一个加密货币钱包被转移到另一个时,附加到NFT的智能合约被执行。加密货币钱包是存储用于加密货币交易的公钥和/或私钥的设备、物理介质、程序或服务。NFT的所有权被记录在区块链中,并且可以由所有者转移,从而允许NFT被销售和交易。

[0003] 在铸造过程期间,当区块链将包含加密散列(其是标识数据集的字符集)的记录连接到先前记录时,NFT被创建,从而创建可标识的数据区块链。该加密交易过程通过提供跟踪NFT的所有权的数字签名来确保每个数字资产的认证。数字资产通常可从公知的储存库下载。数字资产也可以在许多不同的平台或市场上被持有、出售、交易、许诺销售。这意味着,某人可以从一个平台取得数字资产并在另一个平台或市场上创建原始数字资产的映像的伪造品。

[0004] 加密货币钱包是存储用于加密货币交易的公钥和/或私钥的设备、物理介质、程序或服务。加密货币钱包附有公钥和私钥。公钥的作用类似于电子邮件地址,这意味着它可以安全地与其他人共享,从而允许你发送或接收资产。然而,私钥是使得数字资产的持有者能够进行交易并证明其数字资产的所有权的安全码。私钥通常是一串字母和数字。公钥允许你接收加密货币交易。虽然任何人都可以将交易发送到公钥,但是你需要私钥来“解锁”它们,并且证明你是在交易中接收到的加密货币的所有者。

[0005] 随着区块链技术的出现,内容创建者已经能够将他们的创建数字化并将它们作为NFT在市场中出售。NFT的缺点之一是任何人都可以使用NFT中的链接来访问URL并从储存库下载资产。然后,他们可以在另一个NFT市场上铸造新的NFT,从而宣称为资产的所有者。然后,挑战变成证明资产的真实所有权。

[0006] US20200242105A1描述了“一种用于创建结合了影响和外延(“KNFT”)的可操作的数字资产和代币的分布式计算平台和方法。KNFT应用服务器可以被配置为通过分布式计算网络从远程计算节点接收对新的不可替代代币的请求,其中KNFT包括唯一KNFT标识符、至少一个元数据元素、以及至少一个社交向量…社交动作可以包括用户评论、连接、直接消息、点赞、或好评,并且KNFT的所有权的改变可以由KNFT API写入社交向量。社交向量可以包括来自至少一个先前所有者的社交向量数据,并且KNFT还可以包括合并了KNFT的所有权历史的流通踪迹向量。”该参考文献未能解决在账本上提供安全机制以认证数字资产交易的问题。本发明的实施例是有利的,并且认识到通过提供用于认证数字资产交易的安全机

制来保护数字资产的方法的需求和重要性。本发明的实施例通过使用用于验证的活跃度散列(liveness hash)来解决账本上的这个问题。

[0007] 防止数字资产欺诈的另一种方式是检查其它已注册或已知的市场。当前,NFT被生成,并且下载原始资产所需的元数据通常被嵌入作为NFT的一部分。因此,有权访问元数据的任何人都能够下载原始资产。这为伪造、假冒和其他漏洞提供了机会。此外,资产本身可以被存储在某个第三方网站中,该网站在某个时刻可能停止运营,这可能导致原始资产的丢失。然而,本发明的实施例认识到,用加密映像来铸造数字资产增加了确定数字资产的真实性的可能性。

[0008] 现有的NFT创建和验证过程对公众是开放的。现有的NFT铸造过程包括对象存储库中的映像散列和映像的URL。这意味着,任何人都可以通过NFT中的资产散列或地址来验证散列,并计算该散列以检验匹配。此外,通过复制资产、创建散列、以及在同一个或不同的平台用相同的映像铸造另一个NFT,很容易实施NFT盗窃。本发明的实施例通过资产位置和活跃度散列来解决这个问题。此外,所引用的资产被加密。在本发明的实施例中,所有权验证不依赖于NFT中的数据。在本发明的实施例中,请求者提供一次性数字(nonce),并且所有者储存库执行活跃度散列以便证明所有权。

发明内容

[0009] 根据本发明的一个实施例,公开了一种用于验证数字资产的所有权的计算机实现的方法。所述计算机实现的方法包括:由一个或多个处理器响应于接收到对用于加密数字资产的公钥的请求,向所述数字资产的所有者发送所述公钥。所述计算机实现的方法还包括:由所述一个或多个处理器从所述数字资产的所述所有者接收加密的数字资产和第一活跃度散列,其中,所述数字资产是使用所述公钥加密的,所述第一活跃度散列是至少部分地基于未加密形式的所述数字资产和第一一次性数字来生成的。所述计算机实现的方法还包括:响应于从来自所述数字资产的所述所有者的所述加密的数字资产中接收到所述第一一次性数字和所述第一活跃度散列,确定所述第一活跃度散列是否有效。所述计算机实现的方法还包括:由所述一个或多个处理器响应于确定所述第一活跃度散列有效,生成数字资产记录,其中,所述数字资产记录包括所述加密的数字资产和所述第一活跃度散列。本发明的实施例有利于生成活跃度散列来代替散列。

[0010] 根据本发明的另一个实施例,公开了一种用于验证数字资产的所有权的计算机程序产品。所述计算机程序产品包括一个或多个计算机可读存储介质和存储在一个或多个计算机可读存储介质上的程序指令。所述程序指令包括响应于接收到对用于加密数字资产的公钥的请求,向所述数字资产的所有者发送所述公钥的指令。所述程序指令还包括从所述数字资产的所述所有者接收加密的数字资产和第一活跃度散列的指令,其中,所述数字资产是使用所述公钥加密的,所述第一活跃度散列是至少部分地基于未加密形式的所述数字资产和第一一次性数字来生成的。所述程序指令还包括响应于从来自所述数字资产的所述所有者的所述加密的数字资产中接收到所述第一一次性数字和所述第一活跃度散列,确定所述第一活跃度散列是否有效的指令。所述程序指令还包括响应于确定所述第一活跃度散列有效,生成数字资产记录的指令,其中,所述数字资产记录包括所述加密的数字资产和所述第一活跃度散列。本发明的实施例有利于生成活跃度散列来代替散列。

[0011] 根据本发明的另一个实施例,公开了一种用于验证数字资产的所有权的计算机系统。所述计算机系统包括一个或多个计算机处理器、一个或多个计算机可读存储介质以及计算机程序指令,所述计算机程序指令被存储在所述一个或多个计算机可读介质上以用于由所述一个或多个计算机处理器执行。所述程序指令包括响应于接收到对用于加密数字资产的公钥的请求,向所述数字资产的所有者发送所述公钥的指令。所述程序指令还包括从所述数字资产的所述所有者接收加密的数字资产和第一活跃度散列的指令,其中,所述数字资产是使用所述公钥加密的,所述第一活跃度散列是至少部分地基于未加密形式的所述数字资产和第一一次性数字来生成的。所述程序指令还包括响应于从来自所述数字资产的所述所有者的所述加密的数字资产中接收到所述第一一次性数字和所述第一活跃度散列,确定所述第一活跃度散列是否有效的指令。所述程序指令还包括响应于确定所述第一活跃度散列有效,生成数字资产记录的指令,其中,所述数字资产记录包括所述加密的数字资产和所述第一活跃度散列。本发明的实施例有利于生成活跃度散列来代替散列。

[0012] 根据本发明的另一个实施例,公开了一种用于验证数字资产的所有权的计算机实现的方法。所述计算机实现的方法包括:由一个或多个处理器从请求实体接收用于数字资产的所有者提供所述数字资产的所有权证明的请求,其中,所述请求包括第一一次性数字。所述计算机实现的方法还包括:响应于接收到用于所述数字资产的所述所有者提供所述数字资产的所有权证明的所述请求。所述计算机实现的方法还包括:由所述一个或多个处理器将所述第一一次性数字发送到所述数字资产的所述所有者。所述计算机实现的方法还包括:由所述一个或多个处理器从所述数字资产的所述所有者接收与数字映像相关联的映像id和第一活跃度散列。所述计算机实现的方法还包括:由所述一个或多个处理器验证所述数字资产的所有权证明。本发明的实施例有利于生成活跃度散列来代替散列。

附图说明

[0013] 本公开中包括的附图并入说明书中并形成说明书的一部分。它们示出了本公开的实施例,并且与说明书一起用于解释本公开的原理。附图仅说明特定实施例而不限制本公开。

[0014] 图1是根据本发明的至少一个实施例的适于执行在执行本发明的方法中涉及的至少一些计算机代码(例如,数字资产所有权验证代码150)的计算环境(总体上被指定为100)的功能框图。

[0015] 图2A示出了根据本发明的至少一个实施例的总体上被指定为200的示例区块链架构配置。

[0016] 图2B示出了根据本发明的至少一个实施例的总体上被指定为250的区块链交易流。

[0017] 图3是根据本发明的至少一个实施例的适于数字资产交易程序301的操作的数字资产所有权验证系统(总体上被指定为300)的功能框图。

[0018] 图4是描绘根据本发明的至少一个实施例的数字资产所有权验证程序301的操作步骤的流程图(总体上被指定为400)。

[0019] 图5是描绘根据本发明的至少一个实施例的数字资产所有权验证程序301的操作步骤的流程图(总体上被指定为500)。

[0020] 图6是描绘根据本发明的至少一个实施例的用于注册映像过程的操作步骤的流程图(总体上被指定为600)。

[0021] 图7是描绘根据本发明的至少一个实施例的用于注册映像过程的操作步骤的流程图(总体上被指定为700)。

[0022] 图8A示出了根据本公开的至少一个实施例的被配置为执行本文描述的一个或多个操作的示例系统(总体上被指定为800)。

[0023] 图8B示出了根据本公开的至少一个实施例的被配置为执行本文描述的一个或多个操作的另一个示例系统(总体上被指定为840)。

[0024] 图8C示出了根据本公开的至少一个实施例的被配置为利用智能合约的另一个示例系统(总体上被指定为850)。

[0025] 图8D示出了根据本公开的至少一个实施例的被配置为利用区块链的又一个示例系统(总体上被指定为860)。

[0026] 图9A示出了根据本公开的至少一个实施例的用于新的区块被添加到分布式账本的过程(总体上被指定为900)。

[0027] 图9B示出了根据本发明的至少一个实施例的新的数据区块(总体上被指定为930)的内容。

[0028] 图9C示出了根据本公开的至少一个实施例的数字内容的区块链(总体上被指定为970)。

[0029] 图9D示出了根据本公开的至少一个实施例的可以表示区块链中的区块的结构的区块(总体上被指定为990)。

[0030] 虽然本文所述的实施例可具有各种修改和替代形式,但其细节已在附图中以示例的方式示出并将被详细描述。然而,应当理解,所描述的特定实施例不应被理解为限制性的。相反,本发明将涵盖落入本公开的精神和范围内的所有修改物、等效物和替代物。

具体实施方式

[0031] 本发明一般涉及数字资产领域,尤其涉及认证数字资产和验证数字资产的所有权。

[0032] NFT是表示唯一的物理或数字(不可替换)资产的区块链账本条目/代币。当前的NFT铸造过程通过在区块链账本上发布资产来直接揭示数字资产,或者通过在账本条目上发布资产的位置和访问信息连同资产的散列来间接揭示数字资产。数字资产验证是通过将存储在NFT中的散列与原始资产的散列进行比较来执行的。

[0033] 所有权证明是从另一客户端向当前所有者的质询,以用于验证所有者具有对原始数字资产的访问权。NFT的功能类似于所有权证书,具有经由智能合约、资产指针和其他信息进行验证的手段。正如其他证书一样,这不能防止欺诈或盗窃,因为该解决方案是单方面的。资产仅可针对其被存储的区块链实例来验证,这不能防止资产被窃取以及新的NFT在另一区块链实例上的后续铸造。对于太大而不能被编码为NFT的一部分的数字资产来说尤其如此。在没有从资产引用NFT信息的能力的情况下,不可能证明资产所有权,这又将不会减少欺诈。当前解决方案包括用于验证NFT或检查NFT是否存在于区块链实例中、向NFT添加社交网络向量、或扫描公共源以寻找类似技巧的服务。当NFT被铸造时,区块链上的代币id被

返回。NFT包含对原作的引用,并且可以包含访问原作所需的相关信息或当存储在账本中时对原作的代币。本发明的实施例包括一种用于证明原始资产的所有权而不暴露所有权凭证或原始资产的解决方案。

[0034] 当前NFT铸造过程允许由客户端和账本注册表进行本地资产验证以用于跟踪所有权而无需证明。资产验证是通过执行资产的散列并将结果与存储在NFT中的散列进行比较来执行的。这要求资产是世界可见的,这使得任何不良行为人能够下载资产、铸造主张所有权的新NFT并将其发布到账本。本发明的实施例通过使用活跃散列而不是常规散列来扩充NFT铸造过程。

[0035] 本发明的实施例在铸造NFT时使用加密的数字资产,并使得所有权验证能够确保NFT的任何潜在买家都可以相信NFT的当前所有者是合法的。由于数字资产被加密,所以映像验证是由智能合约执行的。智能合约在可信执行环境(TEE)内的节点上被执行以确保未加密资产永远不会被泄露,这最终导致所有方都信任的验证过程。然后,相同的验证智能合约被用于证明数字资产的所有权。

[0036] 本发明的实施例生成一次性数字。一次性数字是在密码通信中仅可使用一次的任意数。一次性数字通常是在认证协议中使用的随机或伪随机数,以确保旧的通信不能被欺诈性地使用。本发明的实施例要求请求者提供一次性数字,并且所有者储存库执行活跃度散列以便证明所有权。本发明的实施例通过使用派生散列而不是原始散列来改进当前方法,从而使得欺骗性复制更加困难。

[0037] 本发明的实施例通过使用活跃度散列而不是常规散列来改进现有NFT铸造过程。本发明的实施例通过从不将原始数字资产泄露给潜在的不良行为人、通过引用原始资产的加密实例来保护所有操作、通过使用活跃度散列来防止验证欺诈、将智能合约用于铸造和验证过程、以及利用安全环境执行智能合约,来防止NFT欺诈和盗窃。本发明的实施例在账本中执行注册表操作,从而防止单点故障并且增强由区块链环境提供的安全性。

[0038] 本发明的实施例通过将数字资产的所有者提供的活跃度散列与智能合约创建的活跃度散列进行比较来执行验证。活跃度散列由经解密的数字映像和一次性数字的组合散列的散列组成。请求验证的实体提供将在活跃度散列中使用的一次性数字。本发明的实施例使用一次性数字作为散列算法中的种子。提供一次性数字保证了将必须计算活跃度散列。此外,防止在验证期间由数字资产的所有者使用已兑现值(cashed value)。

[0039] 本发明的各个方面通过包括在计算机程序产品(CPP)实施例中的说明文本、流程图、计算机系统的方框图和/或机器逻辑的方框图来描述。对于任何流程图,根据所涉及的技术,可以按照与给定流程图中显示的顺序不同的顺序执行操作。例如,同样根据所涉及的技术,在连续流程图块中显示的两个操作可以以相反顺序执行、作为单个集成步骤、同时执行,或者以在时间上至少部分重叠的方式执行。

[0040] 计算机程序产品实施例("CPP实施例"或"CPP")是本发明中使用的术语,用于描述一个或多个存储介质(也称为"介质")的任何集合,所述存储介质(也称为"介质")共同包括在一个或多个存储设备的集合中,所述存储设备共同包括与用于执行给定CPP权利要求中指定的计算机操作的指令和/或数据相对应的机器可读代码。"存储设备"是可以保留和存储指令以供计算机处理器使用的任何有形设备。计算机可读存储介质可以是电子存储介质,磁存储介质,光存储介质,电磁存储介质,半导体存储介质,机械存储介质或上述任意合

适的组合。包括这些介质的一些已知类型的存储设备包括:软盘,硬盘,随机存取存储器(RAM),只读存储器(ROM),可擦除可编程只读存储器(EPROM或Flash存储器),静态随机存取存储器(SRAM),光盘只读存储器(CD-ROM),数字多功能盘(DVD),记忆棒,软盘,机械编码设备(例如穿孔卡或在盘的主表面中形成的凹坑/岛)或上述任意合适的组合。作为本发明中使用的术语,计算机可读存储介质不被解释为其本身的过渡信号形式的存储,例如无线电波或其它自由传播的电磁波,通过波导传播的电磁波,通过光缆的光脉冲,通过导线传送的电信号和/或其它传输介质。如本领域技术人员所理解,数据通常在存储设备的正常操作期间,例如在访问,碎片整理或垃圾收集期间,在某些偶尔的时间点被移动,但这不会使存储设备成为暂时的,因为数据在存储时不是暂时的。

[0041] 已经出于说明的目的给出了本发明的各种实施例的描述,但是其不旨在是穷尽的或限于所公开的实施例。在不背离所描述的实施例的范围和精神的情况下,许多修改和变化对于本领域的普通技术人员将是显而易见的。选择本文所使用的术语以最好地解释实施例的原理、实际应用或对市场上存在的技术的技术改进,或使本领域的其他普通技术人员能够理解本文所公开的实施例。

[0042] 附图中的流程图和框图示出了根据本发明的各种实施例的系统、方法和计算机程序产品的可能实现的架构、功能和操作。在这点上,流程图或框图中的每个框可以表示指令的模块、段或部分,其包括用于实现指定的逻辑功能的一个或多个可执行指令。在一些替代实施方案中,框中所提及的功能可不按图中所提及的次序发生。例如,连续示出的两个框实际上可以基本上同时执行,或者这些框有时可以以相反的顺序执行,这取决于所涉及的功能。还将注意,框图和/或流程图图示的每个框以及框图和/或流程图图示中的框的组合可以由执行指定功能或动作或执行专用硬件和计算机指令的组合的专用的基于硬件的系统来实现。

[0043] 现在将参考附图详细描述本发明。图1是根据本发明的至少一个实施例的适于执行在执行本发明的方法中涉及的至少一些计算机代码(例如,存储在块150中的数字资产所有权验证代码)的计算环境(总体上被指定为100)的功能框图。除了块150之外,计算环境100还包括例如计算机101,广域网(WAN)102,最终用户设备(EUD)103,远程服务器104,公共云105和私有云106。在本实施例中,计算机101包括处理器组110(包括处理电路120和高速缓存121),通信结构111,易失性存储器112,持久性存储装置113(包括操作系统122和块150,如上所述),外围设备组114(包括用户接口(UI)设备组123,存储装置124和物联网(IoT)传感器组125),以及网络模块115。远程服务器104包括远程数据库130。公共云105包括网关140,云编排模块141,主机物理机组142,虚拟机组143和容器组144。

[0044] 计算机101可以采取台式计算机,膝上型计算机,平板计算机,智能电话,智能手表或其他可穿戴计算机,大型机计算机,量子计算机或任何其他形式的计算机或移动设备的形式,这些计算机或移动设备现在已知或将来开发,能够运行程序,访问网络或查询数据库,例如远程数据库130。如计算机技术领域所理解的那样,根据该技术,计算机实现的方法的执行可以分布在多个计算机之间和/或多个位置之间。另一方面,在这个计算环境100的呈现中,详细讨论集中在单个计算机,特别是计算机101上,以保持呈现尽可能简单。计算机101可能位于云中,尽管图1中它没有显示在云中。另一方面,计算机101不需要在云中,除非在任何程度上都可以肯定地指出。

[0045] 处理器组110包括一个或多个现在已知或将来要开发的任何类型的计算机处理器。处理电路120可以分布在多个封装上,例如多个协调的集成电路芯片。处理电路120可以实现多个处理器线程和/或多个处理器核心。高速缓存121是位于处理器芯片封装中的内存,通常用于处理器组110上运行的线程或核心应该可以快速访问的数据或代码。根据与处理电路的相对接近度,通常将高速缓冲存储器组织成多个级别。替代地,处理器组的某些或全部高速缓存可能位于"片外"。在某些计算环境中,处理器组110可以被设计用于使用量子位并执行量子计算。

[0046] 计算机可读程序指令通常被加载到计算机101上,以使得计算机101的处理器组110执行一系列操作步骤,从而实现计算机实现的方法,使得这样执行的指令将实例化本文档中包含的计算机实现的方法的流程图和/或说明描述中指定的方法(统称为"本发明的方法")。这些计算机可读程序指令存储在各种类型的计算机可读存储介质中,例如高速缓存121和下面讨论的其他存储介质。程序指令和相关数据由处理器组110访问以控制和直接执行本发明的方法。在计算环境100中,用于执行本发明方法的至少一些指令可以存储在持久性存储装置113中的块150中。

[0047] 通信结构111是允许计算机101的各种组件相互通信的信号传导路径。通常,该结构由开关和导电路径组成,例如构成总线,桥接器,物理输入/输出端口等的开关和导电路径。可以使用其它类型的信号通信路径,例如光纤通信路径和/或无线通信路径。

[0048] 易失性存储器112是目前已知或未来开发的任何类型的易失性存储器。示例包括动态类型随机存取存储器(RAM)或静态类型RAM。通常,易失性存储器112以随机存取为特征,但除非明确指示,否则这不是必需的。在计算机101中,易失性存储器112位于单个封装中,并且在计算机101内部,但是,可选地或者附加地,易失性存储器可以分布在多个封装上和/或相对于计算机101位于外部。

[0049] 持久性存储装置113是现在已知或者将来将要开发的计算机的任何形式的非易失性存储装置。该存储装置的非易失性意味着不管是否向计算机101供电和/或是否直接向持久性存储装置113供电,都将保持所存储的数据。持久性存储装置113可以是只读存储器(ROM),但通常至少一部分持久存储装置允许数据的写入,数据的删除和数据的重写。一些常见的持久性存储装置形式包括磁盘和固态存储设备。操作系统122可以采取多种形式,例如各种已知的专有操作系统或采用内核的开源可移植操作系统接口类型操作系统。包括在块150中的代码通常包括执行本发明的方法所涉及的至少一些计算机代码。

[0050] 外围设备组114包括计算机101的外围设备组。外围设备与计算机101的其他部件之间的数据通信连接可以以各种方式实现,例如蓝牙连接,近场通信(NFC)连接,电缆连接(例如通用串行总线(USB)类型电缆),插入式连接(例如安全数字(SD)卡),通过局域通信网进行连接,甚至通过互联网等广域网进行连接。在各种实施例中,UI设备组123可以包括诸如显示屏,扬声器,麦克风,可佩戴设备(例如护目镜和智能手表),键盘,鼠标,打印机,触摸板,游戏控制器和触觉设备等组件。存储装置124是外部存储装置(例如外部硬盘驱动器)或可插入存储器(例如SD卡)。存储装置124可以是持久性和/或易失性的。在一些实施例中,存储装置124可以采取量子计算存储设备的形式,用于以量子比特的形式存储数据。在需要计算机101具有大量存储(例如,计算机101在本地存储和管理大型数据库)的实施例中,该存储装置可以由被设计用于存储非常大量数据的外围存储设备提供,例如多个地理上分散的

计算机所共享的存储区域网络(SAN)。IoT传感器组125包括可在物联网应用中使用的传感器。例如,一个传感器可以是温度计,另一个传感器可以是运动检测器。

[0051] 网络模块115是计算机软件,硬件和固件的集合,允许计算机101通过WAN 102与其他计算机通信。网络模块115可以包括硬件,例如调制解调器或Wi-Fi信号收发机,用于对用于通信网络传输的数据进行分组和/或解分组的软件,和/或用于通过因特网传送数据的网络浏览器软件。在一些实施例中,网络模块115的网络控制功能和网络转发功能在同一物理硬件设备上执行。在其它实施例(例如,利用软件定义网络(SDN)的实施例)中,在物理分离设备上执行网络模块115的控制功能和转发功能,使得控制功能管理多个不同的网络硬件设备。用于执行本发明的方法的计算机可读程序指令通常可以通过网络模块115中包含的网络适配卡或网络接口从外部计算机或外部存储设备下载到计算机101。

[0052] WAN 102是任何广域网(例如互联网),能够通过现在已知或将来开发的任何用于传送计算机数据的技术在非本地距离上传送计算机数据。在一些实施例中,WAN 102可由局域网(LAN)替换和/或补充,局域网(LAN)被设计为在位于诸如Wi-Fi网络的局部区域中的设备之间传送数据。WAN和/或LAN通常包括计算机硬件,例如铜传输电缆,光传输纤维,无线传输,路由器,防火墙,交换机,网关计算机和边缘服务器。

[0053] 最终用户设备(EUD)103是由最终用户(例如,操作计算机101的企业的客户)使用和控制的任何计算机系统,并且可以采取以上结合计算机101讨论的任何形式。EUD 103通常从计算机101的操作接收有助和有用的数据。例如,在计算机101被设计为向最终用户提供建议的假设情况下,该建议通常将从计算机101的网络模块115通过WAN 102传送到EUD 103。这样,EUD 103可以向最终用户显示或以其他方式呈现推荐。在一些实施例中,EUD 103可以是客户机设备,例如瘦客户机,重客户机,大型机,台式机等。

[0054] 远程服务器104是向计算机101提供至少一些数据和/或功能的任何计算机系统。远程服务器104可以由操作计算机101的同一实体控制和使用。远程服务器104表示用于收集和存储有用数据以供其他计算机(例如计算机101)使用的机器。例如,在计算机101被设计和编程以提供基于历史数据的推荐的假设情况下,该历史数据可以从远程服务器104的远程数据库130被提供给计算机101。

[0055] 公共云105是可供多个实体使用的任何计算机系统,它提供计算机系统资源和/或其他计算机能力(特别是数据存储(云存储)和计算能力)的按需可用性,而不需要用户直接主动管理。云计算通常利用资源共享来实现一致性和规模经济。公共云105计算资源的直接主动管理由云编排模块141的计算机硬件和/或软件执行。由公共云105提供的计算资源通常由在构成主机物理机组142的计算机的各种计算机上运行的虚拟计算环境实现,主机物理机组142是公共云105中的和/或可用于公共云105的各种物理计算机。虚拟计算环境(VCE)通常采取来自虚拟机组143的虚拟机和/或来自容器组144的容器的形式。可以理解,这些VCE可以被存储为映像,并且可以作为映像或者在VCE实例化之后在各个物理机主机之间传输。云编排模块141管理映像的传输和存储,部署VCE的新实例以及管理VCE部署的活动实例。网关140是计算机软件、硬件和固件的集合,其允许公共云105通过WAN102进行通信。

[0056] 现在将提供虚拟化计算环境(VCE)的一些进一步解释。VCE可以被存储为“映像”。可以从映像实例化VCE的新活动实例。两种熟悉的VCE类型是虚拟机和容器。容器是使用操作系统级别虚拟化的VCE。这是指操作系统特征,其中内核允许存在多个隔离的用户空间实

例(称为容器)。从运行在其中的程序的角度来看,这些隔离的用户空间实例通常表现为真实计算机。在普通操作系统上运行的计算机程序可以利用该计算机的所有资源,例如连接的设备、文件和文件夹、网络共享、CPU能力和可量化的硬件能力。但是,在容器内运行的程序只能使用该容器的内容和分配给该容器的设备,这一特征称为容器化。

[0057] 私有云106类似于公共云105,只是计算资源仅供单个企业使用。虽然私有云106被描绘为与WAN 102通信,但是在其他实施例中,私有云可以完全与因特网断开连接,并且只能通过本地/私有网络访问。混合云是不同类型(例如私有,社区或公共云类型)的多个云的组合,通常由不同的供应商分别实现。多个云中的每一个都仍然是一个独立的离散实体,但更大的混合云架构通过标准化或专有技术绑定在一起,支持多个组成云之间的编排、管理和/或数据/应用可移植性。在本实施例中,公共云105和私有云106都是较大混合云的一部分。

[0058] 图2A示出了根据本发明的至少一个实施例的示例区块链架构配置。区块链架构200可以包括特定区块链元件,例如,一组区块链节点202。区块链节点202可包括一个或多个节点204和210(这四个节点仅通过示例来描绘)。这些节点参与多个活动,诸如区块链交易添加和验证过程(共识)。区块链节点可启动区块链认证并设法写入存储在区块链层216中的区块链不可变账本,该不可变账本的副本也可存储在底层物理基础设施214上。区块链配置可包括一个或多个应用224,一个或多个应用224被链接到应用编程接口(API)222以访问和执行所存储的程序/应用代码220(例如,链代码、智能合约等),这些程序/应用代码220可根据参与者所寻求的定制配置被创建,并且可维护它们自己的状态、控制它们自己的资产、以及接收外部信息。这可以被部署为交易,并且经由附加到分布式账本而被安装在所有区块链节点204-210上。

[0059] 区块链基础或平台212可包括区块链数据、服务(例如,密码信任服务、虚拟执行环境等)和底层物理计算机基础设施的各种层,这些层可用于接收和存储新的交易并向试图访问数据条目的审核员提供访问。区块链层216可公开提供对处理程序代码和参与物理基础设施214所必需的虚拟执行环境的访问的接口。加密信任服务218可用于验证诸如资产交换交易之类的交易并保持信息私密。

[0060] 图2A的区块链架构配置可经由区块链平台212所公开的一个或多个接口以及区块链平台212所提供的服务来处理和执行程序/应用代码220。代码220可以控制区块链资产。例如,代码220可以存储和传输数据,并且可以由节点204-210以智能合约和相关链码的形式执行,相关链码具有受限于其执行的条件或其他代码元素。作为非限制性示例,智能合约可以被创建为执行存储空间的生成、存储空间的预留、对当前交易协议的更新等。智能合约本身可以用于识别与账本的授权和访问要求以及使用相关联的规则。例如,文档属性信息226可以由包括在区块链层216中的一个或多个处理实体(例如,虚拟机)处理。结果228可以包括多个链接的共享文档(例如,其中每个链接的共享文档记录智能合约的发行等)。物理基础设施214可以用于取得本文所述的任何数据或信息。

[0061] 智能合约可以经由高级应用和编程语言来创建,并且然后被写入区块链中的区块。智能合约可以包括向区块链(例如,区块链对等方的分布式网络)注册、存储和/或复制的可执行代码。交易是智能合约代码的执行,交易可以响应于与智能合约相关联的条件被满足而被执行。智能合约的执行可以触发对数字区块链账本的状态的(一个或多个)可信修

改。由智能合约执行引起的对区块链账本的(一个或多个)修改可以通过一个或多个共识协议在整个区块链对等方的分布式网络中被自动复制。

[0062] 智能合约可以以键-值对的格式将数据写入区块链。此外,智能合约代码可以读取存储在区块链中的值,并在应用操作中使用这些值。智能合约代码可以将各种逻辑运算的输出写入区块链中。该代码可用于在虚拟机或其它计算平台中创建临时数据结构。被写入到区块链的数据可以是公共的和/或可以被加密并被保持私有。由智能合约使用/生成的临时数据由所提供的执行环境保持在存储器中,然后一旦区块链所需的数据被标识,该临时数据就被删除。

[0063] 链码可以包括智能合约的代码解释并且具有附加特征。如本文所述,链码可以是被部署在计算网络上的程序代码,其中它在共识过程期间由链验证器一起执行和验证。该链码接收散列并从区块链中取得与通过使用先前存储的特征提取器创建的数据模板相关联的散列。如果散列标识符的散列与从所存储的标识符模板数据创建的散列相匹配,则链码向所请求的服务发送授权密钥。链码可将与密码细节相关联的数据写入区块链(例如,从而在用户和许可方之间建立新的智能合约)。

[0064] 图2B示出了根据本发明的至少一个实施例的区块链的节点之间的区块链交易流250的示例。参考图2B,交易流可以包括由应用客户端节点260发送到背书对等节点281的交易提议291(例如,在一些实施例中,可以发送交易提议291以用于确定数字资产的所有权或认证)。背书对等方281可以检验客户端签名并执行链码功能以发起交易。输出可以包括链码结果、在链码中读取的一组键/值版本(读取集)以及在链码中写入的一组键/值(写入集)。如果被批准,则提议响应292连同背书签名一起被发送回客户端260。客户端260将背书组装成交易有效载荷293,并将其广播到排序服务节点284。然后,排序服务节点284将经排序的交易作为区块在通道上递送给所有对等方281-283。在提交到区块链之前,每个对等方281-283可以验证交易。例如,对等方可以检查交易条款、所有权或其他数字资产数据,以确保用于交易有效载荷293的资产的正确分配和分发。

[0065] 再次参考图2B,客户端节点260通过构造请求并将该请求发送到作为背书者的对等节点281来发起交易291。客户端260可以包括利用支持的软件开发工具包(SDK)的应用,该应用利用可用的API来生成交易提议。该提议是调用链码函数的请求,使得数据可以被读取和/或写入到账本(例如,写入用于资产的新的键值对)。SDK可以用作适配层(shim)以将交易提议封装成适当的架构化格式(例如,远程过程调用(RPC)上的协议缓冲区)并采取客户端的密码证书来产生交易提议的唯一签名。

[0066] 作为响应,背书对等节点281可以检验(a)交易提议是否格式正确,(b)过去尚未提交该交易(重放攻击保护),(c)签名有效,以及(d)提交者(在该示例中为客户端260)被正确授权以在该通道上执行所提议的操作。背书对等节点281可以将交易提议输入作为所调用的链码函数的自变量。然后,对照当前状态数据库执行该链码,以产生包括响应值、读取集和写入集的交易结果。然而,此时不对账本进行更新。在292,该组值连同背书对等节点281的签名一起作为提议响应292被传递回客户端260的SDK,该SDK解析有效载荷以供应用使用。

[0067] 作为响应,客户端260的应用检查/检验背书对等方签名,并比较该提议响应以确定该提议响应是否相同。如果该链码仅查询账本,则该应用将检查查询响应,并且通常不向

排序节点服务284提交交易。如果客户端应用意图将交易提交给排序节点服务284以更新账本,则应用在提交之前确定所指定的背书策略是否已经被满足(例如,交易所需的所有对等节点是否对该交易进行背书)。在此,客户端可以包括交易的多方中的仅一方。在这种情况下,每个客户端可以具有它们自己的背书节点,并且每个背书节点将需要背书该交易。该架构使得即使应用选择不检查响应或以其它方式转发未经背书的交易,背书策略仍将由对等方实施并在提交确认阶段得到支持。

[0068] 在成功检查之后,在步骤293中,客户端260将背书组装成交易,并在交易消息内将交易提议和响应广播到排序节点284。交易可以包含读/写集、背书对等方签名以及通道ID。排序节点284不需要检查交易的全部内容以便执行其操作,而是排序节点284可以简单地从网络中的所有通道接收交易,按通道的时间顺序对它们排序,并且按通道创建交易区块。

[0069] 交易区块在通道上从排序节点284被递送到所有对等节点281-283。区块内的交易294被验证,以确保任何背书策略被满足,并且确保自读取集由交易执行生成以来,没有对读取集变量的账本状态进行改变。区块中的交易被标记为有效或无效。此外,在步骤295,每个对等节点281-283将区块附加到该区块的链,并且对于每个有效的交易,写入集被提交给当前状态数据库。事件被发出,以向客户端应用通知该交易(调用)已经被不可变地附加到链,以及通知该交易有效还是无效。

[0070] 图3是根据本发明的至少一个实施例的适于数字资产交易程序301的操作的数字资产所有权验证系统(总体上被指定为300)的功能框图。数字资产所有权验证系统300可以在诸如参考图1所描述的计算环境100之类的计算环境中实现,图3提供了仅一种实施方式的图示,并且不暗示关于其中可以实现不同实施例的环境的任何限制。本领域技术人员可以对所描绘的环境进行许多修改,而不偏离如权利要求所述的本发明的范围。

[0071] 数字资产所有权验证系统300包括通过诸如WAN 102的网络互连的用户设备310、服务器320、智能合约330、不可信账本350、账本360A和360B、区块链370A和370B、区块链节点380A-380N和TEE 390A-390N。通常,用户设备310可以表示能够执行机器可读程序指令并经由诸如WAN 102的网络与服务器320和其他设备(未示出)通信的任何可编程电子设备或可编程电子设备的组合。在一个实施例中,用户设备310是终端用户设备,诸如图1中描绘的EUD 103,并且可以是移动设备、膝上型计算机、平板计算机、上网本计算机、个人计算机、台式计算机、个人数字助理(PDA)、智能电话、可穿戴设备(例如,智能眼镜、智能手表、电子纺织品、AR头戴式耳机等)或者本领域已知的任何可编程计算机系统。

[0072] 用户设备310还包括用户接口312、应用314和钱包316。用户接口312是提供诸如用户设备310的终端用户设备的用户与驻留在设备上的多个应用(例如,应用314)之间的接口的程序。诸如用户接口312的用户接口涉及程序呈现给用户的信息(诸如图形、文本和声音),以及用户用于控制程序的控制序列。存在各种类型的用户接口。在一个实施例中,用户接口312是图形用户接口。图形用户接口(GUI)是一种用户接口,其允许用户通过图形图标和视觉指示符(例如辅助符号)与电子设备(例如计算机键盘和鼠标)交互,这与基于文本的接口、键入的命令标签或文本导航相反。在计算中,GUI是作为对需要在键盘上键入命令的命令行接口的所感知的陡峭学习曲线的反应而引入的。GUI中的动作通常通过直接操纵图形元素来执行。在另一实施例中,用户接口312是脚本或应用编程接口(API)。

[0073] 应用314可以表示在用户设备310上操作的一个或多个应用(例如,应用套件)。在

一个实施例中,应用314表示位于用户设备310上的一个或多个应用(例如,资产持有应用、资产市场应用和资产认证应用)。例如,用户经由应用314访问资产持有软件以购买数字资产。在另一个示例中,用户经由应用314在线上传数字资产。在各种示例实施例中,应用314可以是用户设备310的用户用于访问资产市场网站并且发布用于销售、交易、要约或购买数字资产的应用。在一个实施例中,应用314可以是与在服务器320上运行的服务器侧应用相关联的客户端侧应用(例如,与数字资产所有权验证程序301相关联的客户端侧应用)。在一个实施例中,应用314可以操作以执行数字资产所有权验证程序301的处理步骤(即,应用314可以表示在用户设备310上操作的数字资产所有权验证程序301)。

[0074] 钱包316是数字或加密货币钱包。在一个实施例中,钱包316包括与对应于数字资产的一个或多个公钥和私钥相关联的信息。在一个实施例中,钱包316包括关于一个或多个数字资产的信息。在一个实施例中,数字资产包括NFT、加密货币、资金或其他数字资产。在一个实施例中,钱包316是硬件加密货币钱包。

[0075] 服务器320被配置为向诸如用户设备310的各种计算设备提供资源。通常,服务器320表示能够执行机器可读程序指令并彼此通信以及与用户设备310、智能合约330和诸如WAN 102的网络内的其它计算设备(未示出)通信的任何可编程电子设备或可编程电子设备的组合。在一个实施例中,服务器320是能够运行程序并访问网络或查询数据库的独立设备,诸如图1中描绘的计算机101。在一个实施例中,服务器320可以是能够接收、发送和处理数据的管理服务器、web服务器、应用服务器、移动设备或任何其他电子设备或计算系统。在一个实施例中,服务器320表示利用多个计算机作为服务器系统的服务器计算系统。在一个实施例中,服务器320表示利用充当单个无缝资源池的集群计算机和组件(例如,数据库服务器计算机、应用服务器计算机、web服务器计算机、web邮件服务器计算机、媒体服务器计算机等)的计算系统。服务器320还包括对象存储库322。在示例性实施例中,服务器320还包括数字资产所有者验证程序301。

[0076] 在一个实施例中,对象存储库322存储关于数字资产的信息。在一个实施例中,对象存储库322是在一个或多个服务器上运行的对象存储服务。

[0077] 智能合约330包括关于附加到数字资产或与数字资产相关联的一个或多个智能合约的信息。在一个实施例中,智能合约330包括利用区块链注册、存储和/或复制的可执行代码。交易是智能合约代码的执行,其可以响应于与智能合约相关联的条件被满足而被执行,诸如将NFT从一个加密货币钱包转移到另一个。在一个实施例中,数字资产所有者验证程序301访问与数字资产相关联的智能合约,例如智能合约330,以验证第一活跃度散列与数字资产和第一一次性数字的散列相匹配。

[0078] 在一个实施例中,智能合约330包括数字资产所有权验证程序301。在一个实施例中,数字资产所有权验证程序301是注册表,并且是智能合约330的子集。数字资产所有权验证程序301可以至少部分地从参考图1所描绘和描述的数字资产所有权验证码150形成。在一个实施例中,数字资产所有权验证程序301与智能合约330分离。例如,在一些实施例中,数字资产所有权验证程序301被包括在服务器320中。在一个实施例中,智能合约330和智能注册表可以互换使用。

[0079] 在一个实施例中,数字资产所有者验证程序301可以被配置为访问各种数据源,诸如用户数字钱包,其可以包括个人数据、内容、上下文数据或用户不想被处理的信息。个人

数据包括个人识别信息或敏感的个人信息以及用户信息,例如位置跟踪或地理位置信息。处理指任何操作(自动的或非自动的)或操作集合,例如收集、记录、组织、构造、存储、适配、改变、检索、咨询、使用个人数据,通过传输公开、传播、或以其他方式使个人数据可用,组合、限制、擦除或破坏个人数据。在一个实施例中,数字资产所有者验证程序301使得能够对个人数据进行授权和安全的处理。在一个实施例中,数字资产所有者验证程序301提供知情同意,并且通知个人数据的收集,从而允许用户选择处理个人数据或选择不处理个人数据。同意可以采取几种形式。选择加入同意可以强制用户在个人数据被处理之前采取肯定动作。或者,选择退出同意可强制用户采取肯定动作以在个人数据被处理之前防止处理个人数据。在一个实施例中,数字资产所有者验证程序301提供关于个人数据和处理的性质(例如,类型、范围、目的、持续时间等)的信息。在一个实施例中,数字资产所有者验证程序301向用户提供所存储的个人数据的副本。在一个实施例中,数字资产所有者验证程序301允许校正或完成不正确或不完整的个人数据。在一个实施例中,数字资产所有者验证程序301允许立即删除个人数据。

[0080] 在一个实施例中,智能合约330以键-值对的形式被写入区块链。此外,智能合约代码可以被构造为读取存储在区块链中的值,并在应用操作中使用这些值。智能合约代码可以被构造为将各种逻辑运算的输出写入区块链中。该代码可用于在虚拟机或其它计算平台中创建临时数据结构。被写入到区块链的数据可以是公共的和/或可以被加密并被保持为私有。由智能合约使用/生成的临时数据由所供应的执行环境保存在存储器中,然后一旦区块链所需的数据被标识,该临时数据就被删除。

[0081] 在一个实施例中,智能合约330在具有TEE 390A-390N的节点上执行。在一个实施例中,智能合约330接收加密的数字资产、一次性数字和活跃度散列。在一个实施例中,活跃度散列是从原始散列得到的散列。在一个实施例中,智能合约330确定与数字资产或钱包相关联的私钥。在一个实施例中,智能合约330执行TEE动作,并将结果返回给TEE 390A-390N。

[0082] 在一个实施例中,智能合约330在区块链节点380A-380N上的TEE 390A-390N中执行。在一个实施例中,智能合约330接收加密的数字资产、一次性数字、私钥和活跃度散列。在一个实施例中,一次性数字是仅被使用一次的随机生成的数。在一个实施例中,智能合约330解密数字资产(A),计算活跃度散列H'(诸如sha256(sha256(A)||一次性数字)),并且返回验证H==H'。

[0083] 在一个实施例中,数字资产所有权验证程序301是智能合约,例如智能合约330。在这些实施例中,数字资产所有权验证程序301在区块链对等方内运行。在一个实施例中,服务器320是智能合约,例如智能合约330。在这些实施例中,键-值存储或对象存储被配置为在每个智能合约330内运行。在一个实施例中,键-值存储或对象存储是存储在区块链对等方或可由区块链对等方作为预言机(oracle)访问的分散式对象存储系统内的对象。区块链预言机是将区块链连接到外部系统从而使得智能合约能够基于输入和输出而执行的实体。

[0084] 分布式账本350包括一个或多个独立的计算机或节点(诸如账本360A和360B以及区块链节点380A-380N),其用于共享和同步它们相应的电子账本中的交易。在一个实施例中,分布式账本350被存储在本地区块链(诸如区块链370A或370B)中。

[0085] 账本360A和360B包括能够执行区块链(诸如区块链370A和370B)的一个或多个账本。

[0086] 区块链370A和370B可以被配置为使用管理多个参与节点的交易的一个或多个智能合约,例如智能合约330。在一些实施例中,神经网络和/或任何形式的机器学习可以由云服务提供商用于分析智能合约和/或交易请求以确定交易条款或认证信息。在一个实施例中,区块链370A和370B可以存储要在诸如区块链节点380之类的节点之间被共享的数据。在一个实施例中,区块链370A和370B可以由如参考图2A所描述的区块链架构配置200来表示。

[0087] 区块链节点380A-380N包括一个或多个节点。在一个实施例中,区块链节点380A-380N可以由如先前参考图2A描述的区块链节点202来表示。

[0088] TEE 390A-390N是区块链节点380A-380N的可信执行环境和安全区域。在一个实施例中,TEE 390A-390N保护代码和数据机密性。在一个实施例中,智能合约330由区块链节点380A-380N中的TEE 390A-390N执行。

[0089] 在一个实施例中,数字资产所有权验证程序301接收数字资产,并且生成用于数字资产的一次性数字。在一个实施例中,数字资产是映像或其他多媒体,并且数字资产所有权验证程序301在验证数字资产的请求中接收一次性数字。在一个实施例中,一次性数字是仅被使用一次的随机生成的数。在一个实施例中,数字资产所有权验证程序301将一次性数字添加到加密的映像或数字资产的散列。在一个实施例中,映像加密是用加密算法对映像进行编码以使得未授权用户不能访问该映像的过程。在一个实施例中,数字资产所有权验证程序301基于映像和一次性数字而生成活跃度散列,诸如活跃度散列=sha256(sha256(映像)||一次性数字)。在一个实施例中,活跃度散列包括作为由用户发送的质询的一次性数字值,并且所返回的映像散列和一次性数字的散列是对质询的响应代币(response token)。

[0090] 在一个实施例中,数字资产所有权验证程序301接收加密公钥(PubKey)。在一个实施例中,公钥是与私钥配对的公钥。在一个实施例中,数字资产所有权验证程序301基于一次性数字和活跃度散列对映像和公钥进行加密,例如Encrypt(image,PubKey),nonce,Liveness Hash(加密(映像,PubKey),一次性数字,活跃度散列)。

[0091] 在一个实施例中,数字资产所有权验证程序301对加密映像和私钥进行解密,例如image=decrypt(enc image,PrivKey)(映像=解密(加密映像,PrivKey))。在一个实施例中,数字资产所有权验证程序301访问智能合约以验证活跃度散列与映像和一次性数字的散列相匹配,诸如验证:活跃度散列==sha256(sha256(映像)||一次性数字))(liveness hash==sha256(sha256(image)||nonce))。在一个实施例中,数字资产所有权验证程序301将包括活跃度散列初始和加密映像的加密映像记录存储在区块链上,诸如存储映像记录:<活跃度散列初始,映像>。

[0092] 在一个实施例中,数字资产所有权验证程序301接收认证数字资产的所有权的请求。在一个实施例中,数字资产所有权验证程序301用私钥解密接收到的映像id,例如image=decrypt(get(image id),PrivKey)。在一个实施例中,数字资产所有权验证程序301验证第二活跃度散列与映像和一次性数字的散列相匹配,诸如验证:活跃度散列2==sha256(sha256(映像)||一次性数字2)(liveness hash 2==sha256(sha256(image)||nonce2))。在一个实施例中,如果第二活跃度散列与映像和一次性数字的活跃度散列相匹配,则数字资产所有权验证程序301验证所有权。

[0093] 在一个实施例中,数字资产所有权验证程序301从一个用户接收验证另一用户当

前拥有的数字资产的请求。在一个实施例中,数字资产验证程序301从数字资产所有者接收散列原始资产和一次性数字的活跃度散列。在一个实施例中,数字资产验证程序301从数字资产所有者接收活跃度散列、一次性数字以及对加密映像的引用(映像ID)。在一个实施例中,数字资产所有权验证程序301生成原始资产的加密实例和活跃度散列。在一个实施例中,数字资产所有权验证程序301在区块链账本上发布原始资产的加密实例,并且引用原始资产的活跃度散列。在一个实施例中,数字资产所有权验证程序301使用区块链账本中的智能合约(例如在诸如TEE 390A-390N的可信执行环境中执行的智能合约330)来确定活跃度散列的共识。

[0094] 在一个实施例中,从一个用户向所有者直接请求所有权证明,或者该请求被转发到所有者和数字资产验证程序301。在另一个实施例中,数字资产所有权验证程序301接收来自用户的针对数字资产的验证请求,并将该请求发送至注册表(NFT,活跃度散列(H),一次性数字(N))。在一个实施例中,数字资产所有权验证程序301确定或接收用于数字资产的智能合约。在一个实施例中,数字资产所有权验证程序301接收或确定加密数字资产(Ae)。在一个实施例中,数字资产所有权验证程序301验证散列(Ae,H,N)。在一个实施例中,数字资产所有权验证程序301确定或接收注册表私钥(RPk)。在一个实施例中,数字资产所有权验证程序301验证散列(Ae,H,N,RPk)。在一个实施例中,数字资产所有权验证程序301对数字资产进行解密。在一个实施例中,数字资产所有权验证程序301计算数字资产的活跃度散列(H')。在一个实施例中,数字资产所有权验证程序301至少部分地基于第一活跃度散列H与第二活跃度散列H'相匹配(H==H')而执行验证。在一个实施例中,数字资产所有权验证程序301向用户发送所有权的验证。在一个实施例中,数字资产所有权验证程序301从智能合约TEE向智能合约发送真实所有权信息。在一个实施例中,数字资产所有权验证程序301将真实所有权的信息从智能合约发送到注册表。

[0095] 在一个示例中,数字资产所有权验证程序301使用公钥和私钥认证来利用账本认证数字资产。数字资产所有权验证程序301请求注册公钥(Rpk)。数字资产所有权验证程序301生成公钥和私钥PAR,并注册该密钥对。数字资产所有权验证程序301生成一次性数字(Nr)。数字资产所有权验证程序301生成数字资产的散列(Ah),例如sha256(资产)。数字资产所有权验证程序301通过确定一次性数字(Nr)和活跃度散列(Ah),例如SHA256(Ah|Nr),来生成活跃度散列(LHr)。数字资产所有权验证程序301使用注册表公钥(Aenc)=加密(Rpk)(资产)来加密资产。数字资产所有权验证程序301将加密资产与包括活跃度散列和一次性数字的唯一密钥(LHr:Nr)一起发送到注册表存储库,诸如图3中所描绘的对象存储库322。例如,注册表可以使用其中地址为LHr:Nr并且值是加密资产的对象存储库或其他键/值存储库。数字资产所有权验证程序301接收将NFT数字资产铸入注册表的请求,其中输入包括公共所有者、所有者公钥、存储密钥(Sk)=LH:Nr、存储库URL、活跃度散列(LHr)和一次性数字(Nr)。

[0096] 在另一个示例中,数字资产所有权验证程序301确定用于数字资产的验证一次性数字(Nv)。数字资产所有权验证程序301将验证请求发送到具有验证一次性数字(Nv)和数字资产地址的数字资产的所有者。所有者处理该请求,并且数字资产所有权验证程序301接收响应。数字资产所有权验证程序301至少部分地基于资产散列(Ah)=sha256(映像)和活跃度散列(H)=sha256(Ah|Nv)来检索数字访问并且使用验证一次性数字(Nv)计算活跃度

散列。数字资产所有权验证程序301向用户发送具有新的活跃度散列(H)的回复。用户将接收到的具有验证一次性数字(Nv)和数字资产引用的活跃度散列(H)发送到注册表以进行验证。注册表接收所有权验证请求,其中输入请求包括来自用户的数字资产引用H和一次性数字标志(Nv)。注册表从地址(URL)检索加密资产,并从数字资产检索智能合约引用。注册表用具有所有者活跃度散列(H)、加密资产和一次性数字(Nv)的输入来调用智能合约。数字资产所有权验证程序301将来自智能合约的共识结果发送到用户。

[0097] 在一个示例中,所有者向智能合约330请求用于加密映像的公钥。在此,公钥由智能合约返回(存储在区块链环境中的公钥/私钥对)。所有者创建一次性数字,计算原始映像和一次性数字的活跃度散列,用公钥加密原始映像,并发送加密映像、一次性数字和对注册智能合约的活跃度散列。在此,数字资产所有权验证程序301从所有者接收加密映像、一次性数字和对注册智能合约的活跃度散列。数字资产所有权验证程序301执行活跃度散列的验证。数字资产所有权验证程序301使用初始活跃度散列作为密钥将加密映像存储到存储服务或储存库。数字资产所有权验证程序301创建NFT,该NFT包括存储服务地址和加密映像的密钥(初始活跃度散列)。数字资产所有权验证程序301将加密映像存储在区块链环境外部的存储服务(诸如云对象存储(COS))中。

[0098] 在另一个示例中,数字资产所有权验证程序301从用户接收映像或NFT、所生成的一次性数字、所计算的活跃度散列。在一个实施例中,活跃度散列用于创建NFT以代替NFT铸造过程输入中的标准散列(stand hash)。输入可以包括存储地址、所有者代币或活跃度散列。

[0099] 在一个实施例中,数字资产所有权验证程序301执行注册过程。在一个实施例中,响应于从第一用户接收到对公钥的请求,数字资产所有权验证程序301将公钥发送到第一用户。在一个实施例中,数字资产所有权验证程序301生成公钥/私钥对,并将公钥发送到第一用户。在一个实施例中,数字资产所有权验证程序301从用户A接收加密映像、一次性数字、以及活跃度散列。在一个实施例中,数字资产所有权验证程序301解密映像,并利用从用户A接收的一次性数字来计算映像的活跃度散列。在一个实施例中,如果数字资产所有权验证程序301确定用户A接收到的有效性散列与计算出的有效性散列相匹配,则数字资产所有权验证程序301存储该映像。在一个实施例中,原始活跃度散列被用作用于存储系统中的加密映像的密钥。

[0100] 在一个实施例中,数字资产所有权验证程序301执行所有权验证过程。在一个示例中,第二用户将具有一次性数字的所有权请求证明发送到第一用户。在一个示例中,第一用户利用从第二用户接收的一次性数字来确定活跃度散列。在一个实施例中,数字资产所有权验证程序301将所确定的具有从第二用户接收到的一次性数字的活跃度散列发送到第二用户。在一个实施例中,数字资产所有权验证程序301从第二用户接收包括一次性数字、映像id和所返回的活跃度散列的验证请求。在一个实施例中,数字资产所有权验证程序301解密映像,并使用接收到的一次性数字来确定活跃度散列。在一个实施例中,如果两个活跃度散列匹配,则数字资产所有权验证程序301确定所有权有效。在一个实施例中,如果活跃度散列不匹配,则数字资产所有权验证程序301确定所有权有效。

[0101] 图4是描绘根据本发明的至少一个实施例的数字资产所有权验证程序301的操作步骤的流程图(总体上被指定为400)。图4仅提供了一种实施方式的说明,并且不暗示对其

中可实现不同实施例的环境的任何限制。本领域技术人员可以对所描绘的环境进行许多修改,而不偏离权利要求所陈述的本发明的范围。

[0102] 在步骤S402,用户A获得映像。在一个实施例中,映像是数字资产。在一个实施例中,用户A接收映像。

[0103] 在步骤S404,用户A生成一次性数字。在一个实施例中,一次性数字是随机生成的数。

[0104] 在步骤S406,用户A生成第一活跃度散列。在一个实施例中,用户A基于映像和一次性数字来生成活跃度散列,诸如活跃度散列=sha256(sha256(映像)||一次性数字)。

[0105] 在步骤S408,数字资产所有权验证程序301接收加密公钥(PubKey)。在一个实施例中,数字资产所有权验证程序301响应于从用户A接收到对用于加密数字资产的公钥的请求而生成加密公钥。在一个实施例中,公钥被与用于用户的加密货币钱包的私钥配对。在一个实施例中,数字资产所有权验证程序301从用户A接收加密公钥,并将所确定的加密公钥发送到注册表智能合约。

[0106] 在步骤S410,数字资产所有权验证程序301发送用于映像的加密公钥。在一个实施例中,数字资产所有权验证程序301将公钥从注册表智能合约发送到用户A。

[0107] 在步骤S412,数字资产所有权验证程序301利用一次性数字和第一活跃度散列来加密数字资产映像和公钥。在一个实施例中,数字资产所有权验证程序301利用公钥对数字资产进行加密。在一个实施例中,数字资产所有权验证程序301利用映像散列和一次性数字确定活跃度散列。在一个实施例中,数字资产所有权验证程序301接收具有加密映像、一次性数字和活跃度散列的注册请求。

[0108] 在步骤S414,数字资产所有权验证程序301对经加密的数字资产映像和私钥进行解密。

[0109] 在步骤S416,数字资产所有权验证程序301验证第一活跃度散列=sha256(sha256(映像)||一次性数字)。

[0110] 在步骤S418,数字资产所有权验证程序301存储映像记录:<活跃度散列初始,映像>。

[0111] 在步骤S420,数字资产所有权验证程序301接收对所有权证明的请求,一次性数字2。在一个实施例中,数字资产所有权验证程序301从用户B接收对所有权证明的请求。

[0112] 在步骤S422,数字资产所有权验证程序301确定活跃度散列2=sha256(sha256(映像)||一次性数字2)、映像id。在一个实施例中,数字资产所有权验证程序301将所确定的活跃度散列2=sha256(sha256(映像)||一次性数字2)、映像id从用户A发送到用户B。

[0113] 在步骤S424,数字资产所有权验证程序301验证活跃度散列2,映像id。在一个实施例中,数字资产所有权验证程序301将经验证的活跃度散列2、映像id从用户B发送到注册表智能合约。

[0114] 在步骤S426,数字资产所有权验证程序301确定映像=解密(获得(映像id),PrivKey)。在一个实施例中,数字资产所有权验证程序301对数字资产和私钥进行解密。

[0115] 在步骤S428,数字资产所有权验证程序301验证活跃度散列2==sha256(sha256(映像)||一次性数字2)。在一个实施例中,数字资产所有权验证程序301确定第一活跃度散列与第二活跃度散列相匹配。

[0116] 在判定步骤S430,数字资产所有权验证程序301确定数字资产是否具有真实的所有权。在一个实施例中,如果第一活跃度散列与第二活跃度散列相匹配,则数字资产所有权验证程序301确定数字资产具有真实的所有权。在一个实施例中,如果第一活跃度散列与第二活跃度散列不匹配,则数字资产所有权验证程序301确定数字资产不具有真实的所有权。如果数字资产所有权验证程序301确定数字资产具有真实的所有权(判定步骤S430的“是”分支),则数字资产所有权验证程序301前进到步骤S432。如果数字资产所有权验证程序301确定数字资产不具有真实的所有权(判定步骤S430的“否”分支),则数字资产所有权验证程序301前进到步骤S434。

[0117] 在步骤S432,数字资产所有权验证程序301将与真实的所有权相关联的信息发送到用户A或用户B。

[0118] 在步骤S434,数字资产所有权验证程序301将与非真实的所有权相关联的信息发送到用户A或用户B。

[0119] 图5是描绘根据本发明的至少一个实施例的数字资产所有权验证程序301的操作步骤的流程图(总体上被指定为500)。图5仅提供了一种实施方式的说明,并且不暗示对其中可实现不同实施例的环境的任何限制。本领域技术人员可以对所描绘的环境进行许多修改,而不偏离权利要求所陈述的本发明的范围。

[0120] 在步骤S502,数字资产所有权验证程序301接收针对数字资产的验证请求(NFT,活跃度散列(H),一次性数字(N))。在一个实施例中,数字资产所有权验证程序301从用户A接收针对数字资产的验证请求。

[0121] 在步骤S504,数字资产所有权验证程序301确定智能合约。

[0122] 在步骤S506,数字资产所有权验证程序301确定加密资产(Ae)。在一个实施例中,数字资产所有权验证程序301接收加密资产(Ae)。

[0123] 在步骤S508,数字资产所有权验证程序301验证散列(Ae,H,N)。

[0124] 在步骤S510,数字资产所有权验证程序301确定注册表私钥(RPK)。

[0125] 在步骤S512,数字资产所有权验证程序301验证散列(Ae,H,N,Rpk)。

[0126] 在步骤S514,数字资产所有权验证程序301对资产进行解密。

[0127] 在步骤S516,数字资产所有权验证程序301计算活跃度散列(H')。

[0128] 在步骤S518,数字资产所有权验证程序301执行验证(H==H')。在一个实施例中,数字资产所有权验证程序301确定散列(H)和活跃度散列(H')相同。

[0129] 在步骤S520,数字资产所有权验证程序301验证结果。在一个实施例中,数字资产所有权验证程序301将验证结果从智能合约TEE发送到智能合约。

[0130] 在步骤S522,数字资产所有权验证程序301验证结果。在一个实施例中,数字资产所有权验证程序301将验证结果从智能合约发送到注册表。在一个实施例中,数字资产所有权验证程序301将验证结果从注册表发送到验证请求者。

[0131] 图6是描绘根据本发明的至少一个实施例的用于注册映像过程的操作步骤的流程图(总体上被指定为600)。图6仅提供了一种实施方式的说明,并且不暗示对其中可实现不同实施例的环境的任何限制。本领域技术人员可以对所描绘的环境进行许多修改,而不偏离权利要求所陈述的本发明的范围。

[0132] 在步骤S602,数字资产所有权验证程序301响应于接收到对用于加密数字资产的

公钥的请求,向数字资产的所有者发送该公钥。在一个实施例中,发送用于加密数字资产的公钥进一步包括生成公钥/私钥对。在一个实施例中,公钥/私钥对中的公钥被用于加密数字资产。在一个实施例中,公钥/私钥对中的私钥被用于解密加密的数字资产。

[0133] 在步骤S604,数字资产所有权验证程序301从数字资产的所有者接收加密的数字资产、第一活跃度散列和第一一次性数字。在一个实施例中,数字资产是使用公钥加密的。在一个实施例中,第一活跃度散列是至少部分地基于未加密形式的数字资产和第一一次性数字来生成的。

[0134] 在判定步骤S606,数字资产所有权验证程序301确定第一活跃度散列是否有效。在一个实施例中,数字资产所有权验证程序301使用与数字资产相关联的私钥将加密的数字资产变换回未加密的数字资产。在一个实施例中,数字资产所有权验证程序301基于未加密的数字资产和第一一次性数字来生成第二活跃度散列。在一个实施例中,数字资产所有权验证程序301将第一活跃度散列与第二活跃度散列进行匹配。如果数字资产所有权验证程序301确定第一活跃度散列有效(判定步骤S606的“是”分支),则数字资产所有权验证程序301前进到步骤S608。如果数字资产所有权验证程序301确定第一活跃度散列无效(判定步骤S606的“否”分支),则数字资产所有权验证程序301结束。

[0135] 在步骤S608,数字资产所有权验证程序301生成数字资产记录。在一个实施例中,数字资产记录包括加密的数字资产和第一活跃度散列。在一个实施例中,数字资产所有权验证程序301将数字资产记录存储在分布式账本或区块链环境外部的存储服务中的至少一个中。

[0136] 图7是描绘根据本发明的至少一个实施例的用于注册映像过程的操作步骤的流程图(总体上被指定为700)。图7仅提供了一种实施方式的说明,并且不暗示对其中可实现不同实施例的环境的任何限制。本领域技术人员可以对所描绘的环境进行许多修改,而不偏离权利要求所陈述的本发明的范围。

[0137] 在步骤S702,数字资产所有权验证程序301从请求实体接收用于数字资产的所有者提供数字资产的所有权证明的请求。在一个实施例中,该请求包括第二一次性数字。

[0138] 在步骤S704,响应于接收到用于数字资产的所有者提供数字资产的所有权证明的请求,数字资产所有权验证程序301将第二一次性数字发送到数字资产的所有者。

[0139] 在步骤S706,数字资产所有权验证程序301从数字资产的所有者接收与数字映像相关联的映像id和第三活跃度散列。在一个实施例中,第二活跃度散列是至少部分地基于未加密形式的数字资产和第二一次性数字而生成的。

[0140] 在步骤S708,数字资产所有权验证程序301验证数字资产的所有权证明。在一个实施例中,验证数字资产的所有权证明至少部分地基于使用映像id从数字资产记录中取得加密的数字资产。在一个实施例中,验证数字资产的所有权证明至少部分地基于使用与数字资产相关联的私钥将加密的数字资产变换回未加密的数字资产。在一个实施例中,验证数字资产的所有权证明至少部分地基于根据未加密的数字资产和第二一次性数字来生成第四活跃度散列。在一个实施例中,验证数字资产的所有权证明至少部分地基于将第一活跃度散列与第二活跃度散列进行匹配。

[0141] 在步骤S710,响应于验证数字资产的所有权证明,数字资产所有权验证程序301向请求实体发送数字资产的所有者有效的认证。

[0142] 图8A示出了根据本公开的实施例的包括被配置为执行各种操作的物理基础设施810的示例系统800。参考图8A,物理基础设施810包括模块812和模块814。模块814包括区块链820和智能合约830(其可以驻留在区块链820上),它们可以执行包括在任何示例实施例中的任何操作步骤808(在模块812中)。步骤/操作808可以包括所描述或描绘的一个或多个实施例,并且可以表示从一个或多个智能合约830和/或区块链820写入或读取的输出或写入信息。物理基础设施810、模块812和模块814可以包括一个或多个计算机、服务器、处理器、存储器和/或无线通信设备。此外,模块812和模块814可以是同一个模块。

[0143] 图8B示出了根据本公开的实施例的被配置为执行各种操作的另一示例系统840。参考图8B,系统840包括模块812和模块814。模块814包括区块链820和智能合约830(其可以驻留在区块链820上),它们可以执行包括在任何示例实施例中的任何操作步骤808(在模块812中)。步骤/操作808可以包括所描述或描绘的一个或多个实施例,并且可以表示从一个或多个智能合约830和/或区块链820写入或读取的输出或写入信息。物理模块812和模块814可以包括一个或多个计算机、服务器、处理器、存储器和/或无线通信设备。此外,模块812和模块814可以是同一个模块。

[0144] 图8C示出根据本公开的实施例的被配置为在订约方之间利用智能合约配置的示例系统和被配置为在区块链上强制执行智能合约条款的调解服务器。参考图8C,配置850可以表示由显式识别一个或多个用户设备852和/或856的智能合约830驱动的通信会话、资产转移会话或处理或过程。智能合约执行的执行、操作和结果可以由服务器854管理。智能合约830的内容可能需要作为智能合约交易的各方的实体852和856中的一个或多个的数字签名。智能合约执行的结果可以作为区块链交易被写到区块链820。智能合约830驻留在可以驻留在一个或多个计算机、服务器、处理器、存储器和/或无线通信设备上的区块链820上。

[0145] 图8D示出了根据本公开的实施例的包括区块链的系统860。参考图8D的示例,应用编程接口(API)网关862提供用于访问区块链逻辑(例如,智能合约830或其他链码)和数据(例如,分布式账本等)的公共接口。在该示例中,API网关862是用于通过将一个或多个实体852和856连接到区块链对等方(例如,服务器854)来在区块链上执行交易(例如,调用、查询等)的公共接口。在此,服务器554是区块链网络对等组件,该区块链网络对等组件保存世界状态和分布式账本的副本,从而允许客户端552和556查询关于世界状态的数据以及将交易提交到区块链网络中,在区块链网络中,取决于智能合约830和许可条款,被授权的交易将运行智能合约830。

[0146] 以上实施例能够以硬件、由处理器执行的计算机程序、固件或以上项的组合来实现。计算机程序可体现在计算机可读介质(例如存储介质)上。例如,计算机程序可以驻留在随机存取存储器(“RAM”)、闪存、只读存储器(“ROM”)、可擦除可编程只读存储器(“EPROM”)、电可擦除可编程只读存储器(“EEPROM”)、寄存器、硬盘、可移动盘、紧凑盘只读存储器(“CD-ROM”)、或本领域已知的任何其他形式的存储介质中。

[0147] 示意性存储介质可耦接到处理器,使得处理器可从存储介质读取信息和将信息写入到存储介质。在替代方案中,存储介质可与处理器成一体。处理器和存储介质可驻留在专用集成电路(“ASIC”)中。在替代方案中,处理器和存储介质可作为分离组件而驻留。

[0148] 图9A示出了根据本公开的实施例的(例如,当新的智能合约被生成时)新的区块被添加到分布式账本920的过程900,并且图9B示出了根据本公开的实施例的用于区块链的新

的数据区块结构930的内容。新的数据区块930可以包含文档链接数据。

[0149] 参考图9A,客户端(未示出)可向区块链节点911、912和/或913提交交易。客户端可以是从任何源接收以制定在区块链922上的活动的指令。作为示例,客户端可以是代表请求者(例如设备、人或实体)行动以提议对区块链的交易的应用。多个区块链对等方(例如,区块链节点911、912和/或913)可维护区块链网络的状态和分布式账本920的副本。不同类型的区块链节点/对等方可以存在于区块链网络中,包括:仿真和授权由客户端提议的交易的节点;利用自然语言处理技术并推荐要与用户自动签约的实体的推荐节点;以及验证交易并将交易提交给分布式账本920的提交对等方。在该示例中,区块链节点911、912和/或913可以执行背书者节点、提交者节点、推荐者节点或所有这三个节点的角色。

[0150] 分布式账本920包括在区块中存储不可变的排序记录的区块链922和维护区块链922的当前状态的状态数据库924(当前世界状态)。每个通道可以存在一个分布式账本920,并且每个对等方针对它们作为成员的每个通道而维护它自己的分布式账本920副本。区块链922是交易日志,该交易日志被构造为散列链接的区块,其中每个区块包含具有N个交易的序列。区块可包括诸如图9B中所示的各种组件。区块的链接(由图9A中的箭头示出)可以通过在当前区块的区块头部内添加先前区块的头部的散列来生成。以此方式,区块链922上的所有交易被排序并被加密地链接在一起,从而防止篡改区块链数据而不破坏散列链接。此外,由于链接,区块链922中的最新区块代表在它之前出现的每个交易。区块链922可以被存储在支持仅附加的区块链工作负载的对等文件系统(本地或附加存储)上。

[0151] 区块链922的当前状态和分布式账本920可以被存储在状态数据库924中。此处,当前状态数据表示曾经被包括在区块链922的链交易日志中的所有键的最新值。链码调用对照状态数据库924中的当前状态来执行交易。为了使这些链码交互极其高效,所有键的最新值被存储在状态数据库924中。状态数据库924可以包括到区块链922的交易日志中的索引视图,因此可以在任何时间从链中重新生成该索引视图。在交易被接受之前,在对等方启动后,状态数据库924可自动被恢复(或者被生成,如果需要的话)。

[0152] 节点从客户端接收交易,并基于仿真结果来授权交易。节点持有仿真交易提议的智能合约。当节点验证数字资产所有权时,节点创建交易背书,该交易背书是从该节点到客户端应用的指示数字资产所有权的验证的签名响应。验证数字资产所有权的方法取决于可以在链码内指定的活跃度散列、一次性数字、私钥和公钥对中的一个或多个。不同的通道可具有不同的许可条款。经授权的交易由客户端应用转发到排序服务910。

[0153] 排序服务910接受经授权的交易,将它们排序到区块中,以及向提交对等方递送区块。例如,当已经达到交易阈值、定时器超时或另一条件时,排序服务910可以发起新区块。在图9A的示例中,区块链节点912是提交对等方,其已接收到新数据区块930以供存储在区块链920上。区块链中的第一个区块可以被称为起源区块,该起源区块包括关于区块链、其成员、存储在其中的数据等的信息。

[0154] 排序服务910可以包括排序器集群。排序服务910不处理交易、智能合约或维护共享账本。相反,排序服务910可以接受经授权的交易并且指定这些业务被提交到分布式账本920的顺序。可以设计区块链网络的架构,使得‘排序’的具体实现变成可插拔组件。

[0155] 交易以一致的顺序被写到分布式账本920。交易的顺序被确立以确保对状态数据库924的更新在这些更新被提交至网络时是有效的。与其中通过求解密码谜题或挖掘而发

生排序的加密货币区块链系统不同,在本示例中,分布式账本920的各方可以选择最适合该网络的排序机制。

[0156] 当排序服务910初始化新数据区块930时,新数据区块930可被广播到提交对等方(例如,区块链节点911、912和913)。当交易被授权时,交易被写到分布式账本920上的区块链922,并且状态数据库924被用来自读-写集的写数据来更新。如果交易失败,即,如果提交对等方发现读-写集与状态数据库924中的当前世界状态不匹配,则被排序到区块中的交易将仍然被包括在该区块中,但该交易将被标记为无效,并且状态数据库924将不被更新。

[0157] 参考图9B,被存储在分布式账本920的区块链922上的新数据区块930(也被称为数据区块)可以包括多个数据段,诸如区块头部940、区块数据950和区块元数据960。应当理解,图9B中所示的各种所描绘的块及其内容(诸如新数据区块930及其内容)仅仅是示例,并且不意味着限制示例实施例的范围。新数据区块930可以在区块数据950内存储N个交易(例如,1、10、100、500、1000、2000、3000等)的交易信息。新数据区块930还可以包括到区块头部940内的前一个区块(例如,在图9A中的区块链922上)的链接。具体地说,区块头部940可包括前一个区块的头部的散列。区块头部940还可包括唯一区块号、新数据区块930的区块数据950的散列等。新数据区块930的区块号可以是唯一的并且以不同顺序被分配,诸如从零开始的递增/顺序排序。

[0158] 数据区块950可以存储被记录在新数据区块930内的每个交易的交易信息。例如,交易数据可以包括交易类型、版本、时间戳、分布式账本920的通道ID、交易ID、时期、有效载荷可见性、链码路径(部署tx)、链码名称、链码版本、输入(链码和函数)、诸如公钥和证书的客户端(创建者)标识、客户端的签名、许可方的签名、许可方的标识、提议散列、链码事件、响应状态、命名空间、读取集(由交易读取的键和版本的列表等)、写入集(键和值的列表等)、开始键、结束键、键列表、Merkel树查询概要等。可针对N个交易中的每个交易存储交易数据。

[0159] 在一些实施例中,区块数据950还可以存储向区块链922中的散列链接的区块链添加附加信息的新数据962。附加信息包括本文描述或描绘的步骤、特征、过程和/或动作中的一个或多个。因此,新数据962可以被存储在分布式账本920上的区块的不可变日志中。存储此新数据962的益处中的一些反映于本文所公开和描绘的各种实施例中。尽管在图9B中,新数据962被描绘在数据区块950中,但是也可以位于区块头部940或块元数据690中。新数据962可包括被用于链接组织内的文档的文档合成键(document composite key)。

[0160] 区块元数据960可存储元数据的多个字段(例如,作为字节阵列等)。元数据字段可以包括关于区块创建的签名、对最后配置区块的引用、标识区块内的有效和无效交易的交易过滤器、对区块进行排序的排序服务的存留的最后偏移等。签名、最后配置区块、以及经排序的元数据可以由排序服务910来添加。同时,区块的提交者(诸如区块链节点912)可以基于背书策略、读/写集的验证等来添加有效性/无效性信息。交易过滤器可包括大小等于数据区块950中的交易的数量的字节阵列和标识交易是否有效/无效的验证码。

[0161] 图9C示出了根据本文描述的实施例的用于数字内容的区块链970的实施例。数字内容可包括一个或多个文件和相关联的信息。文件可包含媒体、图像、视频、音频、文本、链接、图形、动画、网页、文档或其他形式的数字内容。区块链的不可变的、仅附加的方面充当保护措施来保护数字内容的完整性、有效性和真实性,从而使数字内容适合用于法律程序

(其中可接受性规则适用)或其中考虑证据或其中数字信息的呈现和使用以其他方式感兴趣的其他设置中。在这种情况下,数字内容可以被称为数字证据。

[0162] 可以以各种方式形成区块链。在一个实施例中,数字内容可以被包括在区块链本身中以及可以从区块链本身来访问。例如,区块链的每个区块可以连同相关联的数字内容一起存储引用信息(例如,头部、值等)的散列值。然后,散列值和相关联的数字内容可以一起被加密。因此,每个区块的数字内容可以通过解密区块链中的每个区块来访问,并且每个区块的散列值可以用作引用前一个区块的基础。这可以说明如下:

[0163]

区块 1	区块 2	.....	区块 N
散列值 1	散列值 2		散列值 N

[0164] 数字内容 1 数字内容 2 数字内容 N

[0165] 在一个实施例中,数字内容可以不被包括在区块链中。例如,区块链可在没有任何数字内容的情况下存储每个区块的内容的经加密的散列。数字内容可以被存储在与原始文件的散列值相关联的另一个存储区域或存储器地址中。另一个存储区域可以是用于存储区块链的同一个存储设备,或者可以是不同的存储区域或者甚至是单独的关系数据库。通过获得或查询感兴趣区块的散列值,然后在存储区域中查找与该实际数字内容相对应地存储的值,可以引用或访问每个区块的数字内容。此操作可以例如由数据库网守(gatekeeper)来执行。这可以说明如下:

[0166]

区块链	存储区
区块 1 散列值	区块 1 散列值...内容
.	.
.	.
.	.
区块 N 散列值	区块 N 散列值...内容

[0167] 在图9C的示例实施例中,区块链970包括被加密地链接在有序序列中的多个区块978_1、978_2、…、978_N,其中N≥1。用于链接区块978_1、978_2、…、978_N的加密可以是多个键控或非键控散列函数中的任何函数。在一个实施例中,区块978_1、978_2、…、978_N经受散列函数,该散列函数从基于区块中的信息的输入产生n位字母数字输出(其中,n是256或另一个数)。这样的散列函数的示例包括但不限于SHA型(SHA代表安全散列算法)算法、Merkle-Damgard算法、HAIFA算法、Merkle-tree算法、基于一次性数字的算法、以及防冲突的PRF算法。在另一个实施例中,区块978_1、978_2、…、978_N可通过与散列函数不同的函数被加密地链接。为了说明的目的,参考散列函数(例如,SHA-2)进行以下描述。

[0168] 区块链中的区块978_1、978_2、…、978_N中的每一个包括头部、文件的版本、以及值。作为区块链中的散列的结果,头部和值对于每个区块是不同的。在一个实施例中,该值可被包括在头部中。如下文更详细描述的,文件的版本可以是原始文件或原始文件的不同版本。

[0169] 区块链中的第一个区块978_1被称为起源区块,并且包括头部972_1、原始文件974_1和初始值976_1。用于起源区块以及实际上在所有后续区块中的散列方案可以变化。例如,第一区块978_1中的所有信息可被一起散列并且一次散列,或者第一区块978_1中的信息的每一个或一部分可被分开散列,然后可执行被分开散列的部分的散列。

[0170] 头部972_1可以包括一个或多个初始参数,其例如可以包括版本号、时间戳、一次性

数字、根信息、难度级别、共识协议、持续时间、媒体格式、源、描述性关键字、和/或与原始文件974_1和/或区块链相关联的其他信息。头部972_1可以自动地(例如,通过区块链网络管理软件)或由区块链参与者手动地生成。与区块链中的其他区块978_2至978_N中的头部不同,起源区块中的头部972_1不引用前一个区块,这只是因为不存在前一个区块。

[0171] 起源区块中的原始文件974_1可以是例如在被包括在区块链中之前在具有或没有处理的情况下由设备捕获的数据。通过系统的接口从设备、媒体源或节点接收原始文件974_1。原始文件974_1与元数据相关联,元数据例如可以由用户、设备和/或系统处理器手动地或自动地生成。元数据可以被包括在与原始文件974_1相关联的第一区块978_1中。

[0172] 起源区块中的值976_1是基于原始文件974_1的一个或多个唯一属性生成的初始值。在一个实施例中,一个或多个唯一属性可以包括原始文件974_1的散列值、原始文件974_1的元数据、以及与文件相关联的其他信息。在一个实施例中,初始值976_1可基于以下唯一属性:1)SHA-2计算的原始文件的散列值;2)发端设备ID;3)原始文件的开始时间戳;4)原始文件的初始存储位置;以及5)用于软件当前控制原始文件和相关联元数据的区块链网络成员ID。

[0173] 区块链中的其他区块978_2至978_N也具有头部、文件和值。然而,与第一区块972_1不同,其他区块中的头部972_2至972_N中的每一个包括紧接在前的区块的散列值。紧接在前的区块的散列值可以仅是前一个区块的头部的散列或者可以是整个前一个区块的散列值。通过在剩余区块的每一个中包括前一个区块的散列值,可以在逐块的基础上执行从第N个区块返回到起源区块(和相关联的原始文件)的跟踪,如箭头980所指示的,以建立可审计的和不可变的监管链。

[0174] 其他区块中的头部972_2至972_N中的每一个还可以包括其他信息,例如,版本号、时间戳、一次性数字、根信息、难度级别、共识协议、和/或总体上与对应的文件和/或区块链相关联的其他参数或信息。

[0175] 其它区块中的文件974_2至974_N可以等于原始文件或者可以是起源区块中的原始文件的修改版本,这取决于例如所执行的处理的类型。所执行的处理的类型可从区块到区块变化。该处理可以涉及例如对在前区块中的文件的任何修改,如编辑信息或以其他方式改变文件的内容、将信息从这些文件移除、或将信息添加或附加到这些文件上。

[0176] 附加地或可替代地,处理可以涉及只是从在前区块复制文件、改变文件的存储位置、分析来自一个或多个在前区块的文件、将文件从一个存储或存储器位置移动到另一个、或执行相对于区块链的文件和/或其相关联的元数据的动作。涉及分析文件的处理可以包括例如附加、包括各种分析、统计或与该文件相关联的其他信息,或以其他方式将各种分析、统计或与该文件相关联的其他信息相关联。

[0177] 其它区块中的其他区块976_2至976_N中的每一个中的值是唯一的值,并且作为所执行的处理的结果都是不同的。例如,任何一个区块中的值对应于前一个区块中的值的更新版本。该更新反映在被赋值的区块的散列中。区块的值因此提供在区块中执行了什么处理的指示,以及还允许通过区块链返回到原始文件的跟踪。该跟踪确认文件在整个区块链中的监管链。

[0178] 例如,考虑先前区块中的文件的多个部分被编辑、阻挡或像素化以便保护在文件中显示的人的身份的情况。在这种情况下,包括被编辑文件的区块将包括与被编辑文件相

关联的元数据,例如,如何执行编辑、谁执行编辑、编辑发生的时间戳等。元数据可以被散列以形成值。因为区块的元数据与被散列以形成前一个区块中的值的信息不同,所以这些值彼此不同且可在解密时被恢复。

[0179] 在一个实施例中,当发生以下任何一者或多者时,可以更新前一个区块的值(例如,计算新的散列值)以形成当前区块的值。在此示例实施例中,可以通过对以下指出的信息的全部或一部分进行散列来计算新的散列值。

[0180] a)如果文件已经以任何方式被处理(例如,如果文件被编辑、复制、更改、访问或采取一些其他动作),则新SHA-2计算的散列值

[0181] b)文件的新存储位置

[0182] c)所识别的与该文件相关联的新元数据

[0183] d)从一个区块链参与者向另一个区块链参与者转移对文件的访问或控制

[0184] 图9D示出根据一个实施例的可表示区块链990中的区块结构的区块的实施例。该区块(即,区块$_i$)包括头部972_i、文件974_i和值976_i。

[0185] 头部972_i包括前一个区块即区块$_{i-1}$的散列值和附加参考信息,该附加参考信息例如可以是本文所讨论的任意类型的信息(例如,包括引用、特性、参数等的头部信息)。所有区块引用前一个区块的散列,当然起源区块除外。前一个区块的散列值可以仅是前一个区块中的头部的散列或前一个区块中的信息的全部或部分(包括文件和元数据)的散列。

[0186] 文件974_i包括多个数据,如按顺序的数据1、数据2、…、数据N。使用描述与数据相关联的内容和/或特性的元数据1、元数据2、…、元数据N来标记数据。例如,每个数据的元数据可以包括用于指示数据的时间戳、处理数据、指示数据中描绘的人员或其他内容的关键词、和/或可以有助于作为整体建立文件的有效性和内容(特别是其作为数字证据的使用)的其他特征的信息,例如,如结合以下讨论的实施例所描述的。除了元数据之外,每个数据可以被标记有对前一个数据的引用$_1$、引用$_2$、…、引用$_N$以防止篡改、文件中的间隙、以及通过文件的顺序引用。

[0187] 一旦元数据被分配给数据(例如,通过智能合约),在没有散列改变(这可以容易地被识别以用于无效)的情况下不能更改元数据。因此,元数据创建了可被访问以供区块链中的参与者使用的信息的数据日志。

[0188] 值976_i是散列值或基于先前讨论的任何类型的信息计算的其他值。例如,对于任何给定区块$_i$,该区块的值可被更新以反映针对该区块执行的处理,例如,新的散列值、新的存储位置、相关联的文件的新的元数据、控制或访问的转移、标识符、或将被添加的其他动作或信息。尽管每个区块中的值被示出为与文件的数据的元数据和头部分离,但是在另一个实施例中,该值可以部分地或整体地基于此元数据。

[0189] 一旦形成了区块链990,在任何时间点,可以通过查询区块链以获得跨区块的值的交易历史来获得文件的不可变监管链。该查询或跟踪过程可开始于解密最近被包括的区块(例如,最后(第N个)区块)的值,然后继续解密其他区块的值,直到到达起源区块并恢复原始文件为止。解密还可包括解密每个区块处的头部和文件以及相关联的元数据。

[0190] 解密是基于在每个区块中发生的加密的类型来执行的。这可涉及使用私钥、公钥、或公钥-私钥对。例如,当使用非对称加密时,网络中的区块链参与者或处理器可使用预定算法生成公钥和私钥对。公钥和私钥通过某种数学关系相互关联。公钥可公开被分发以用

作从其他用户接收消息的地址,例如,IP地址或家庭地址。私钥保密并且用于对发送到其他区块链参与者的消息进行数字签名。签名被包括在消息中,使得接收方可以使用发送方的公钥来验证。这样,接收方可以确信只有发送方可能已经发送该消息。

[0191] 生成密钥对可类似于在区块链上创建账户,但不必实际上在任何位置进行注册。此外,在区块链上执行的每个交易由发送方使用它们的私钥进行数字签名。此签名确保仅账户的所有者能够跟踪和处理(如果在由智能合约所确定的许可的范围内)区块链的文件。

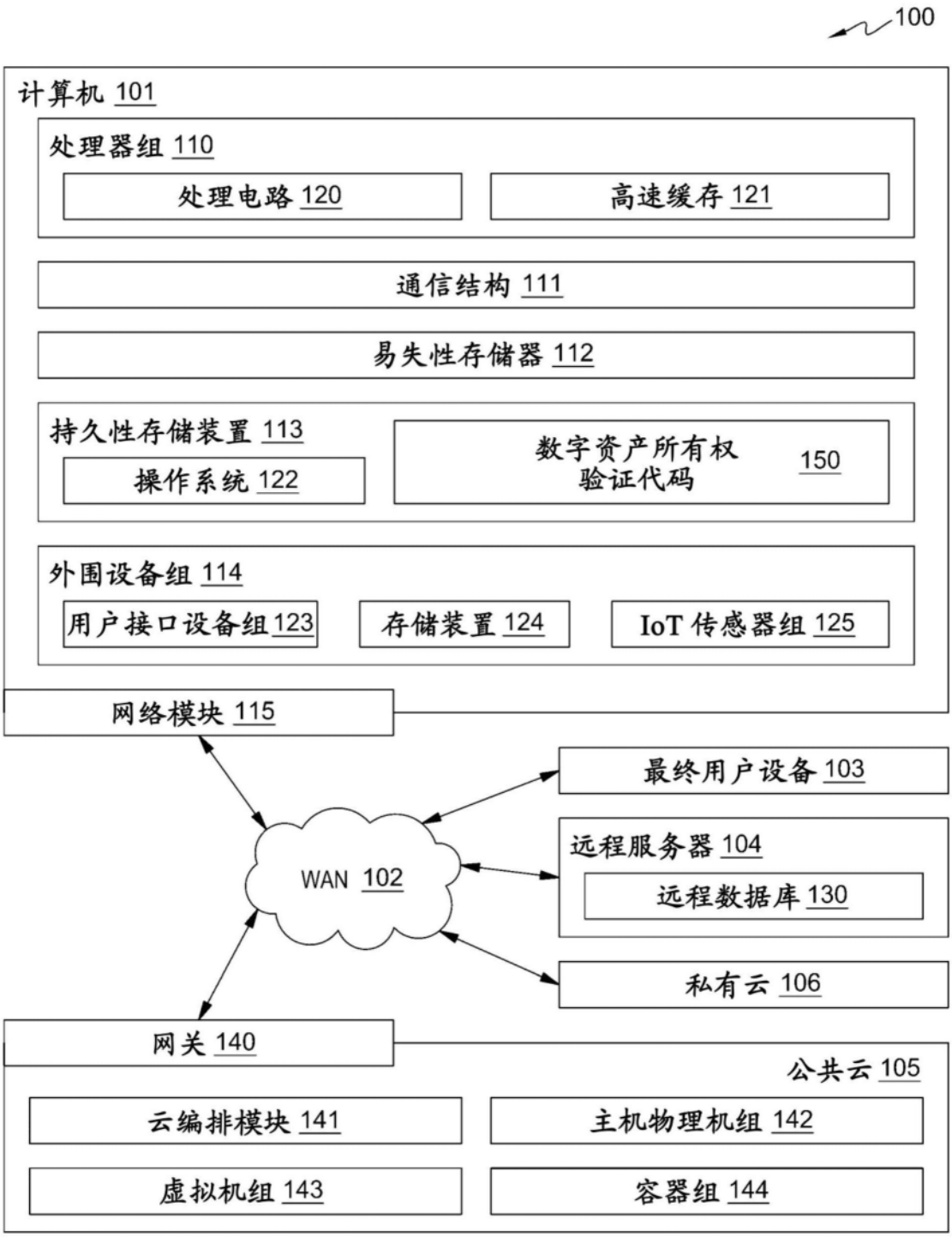

100
计算机 101
处理器组 110
处理电路 120
高速缓存 121
通信结构 111
易失性存储器 112
持久性存储装置 113
操作系统 122
数字资产所有权
验证代码
150
外围设备组 114
用户接口设备组 123
存储装置 124
IoT 传感器组 125
网络模块 115
最终用户设备 103
WAN 102
远程服务器 104
远程数据库 130
私有云 106
网关 140
公共云 105
云编排模块 141
主机物理机组 142
虚拟机组 143
容器组 144

图1

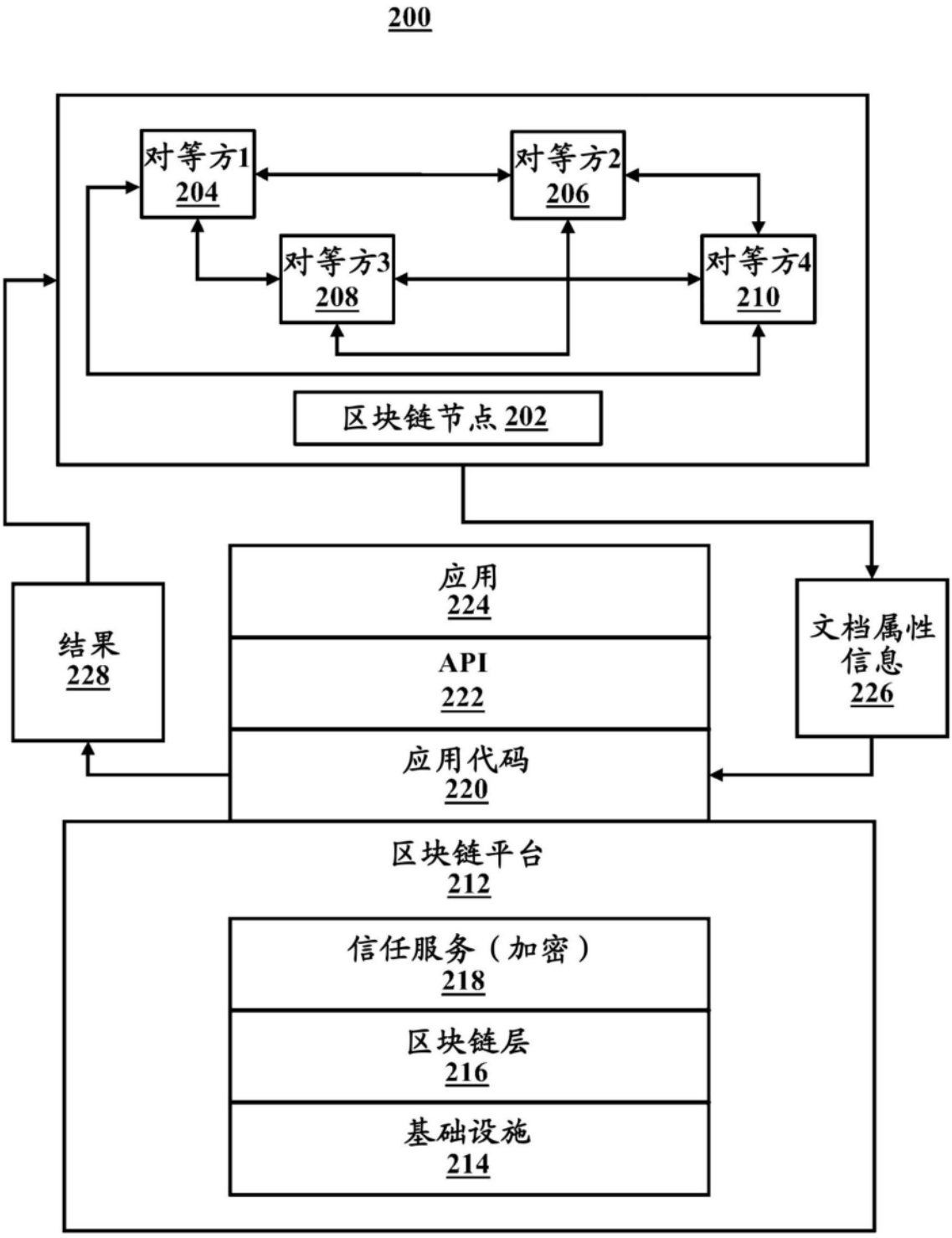

200
对等方1
204
对等方2
206
对等方3
208
对等方4
210
区块链节点 202
结果
228
应用
224
API
222
应用代码
220
文档属性
信息
226
区块链平台
212
信任服务（加密）
218
区块链层
216
基础设施
214

图2A

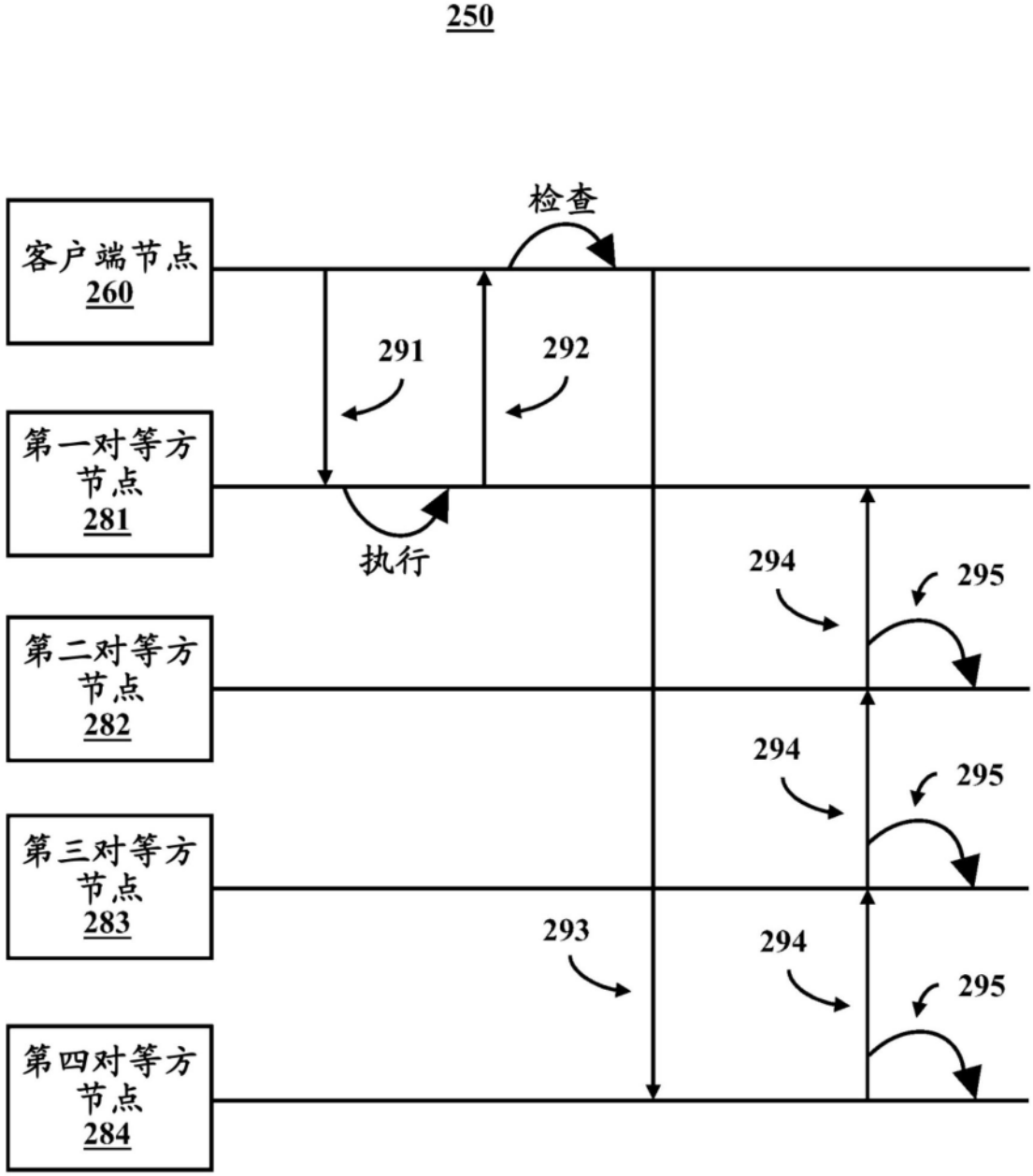

250
检查
客户端节点
260
291
292
第一对等方
节点
281
执行
294
295
第二对等方
节点
282
294
295
第三对等方
节点
283
293
294
295
第四对等方
节点
284

图2B

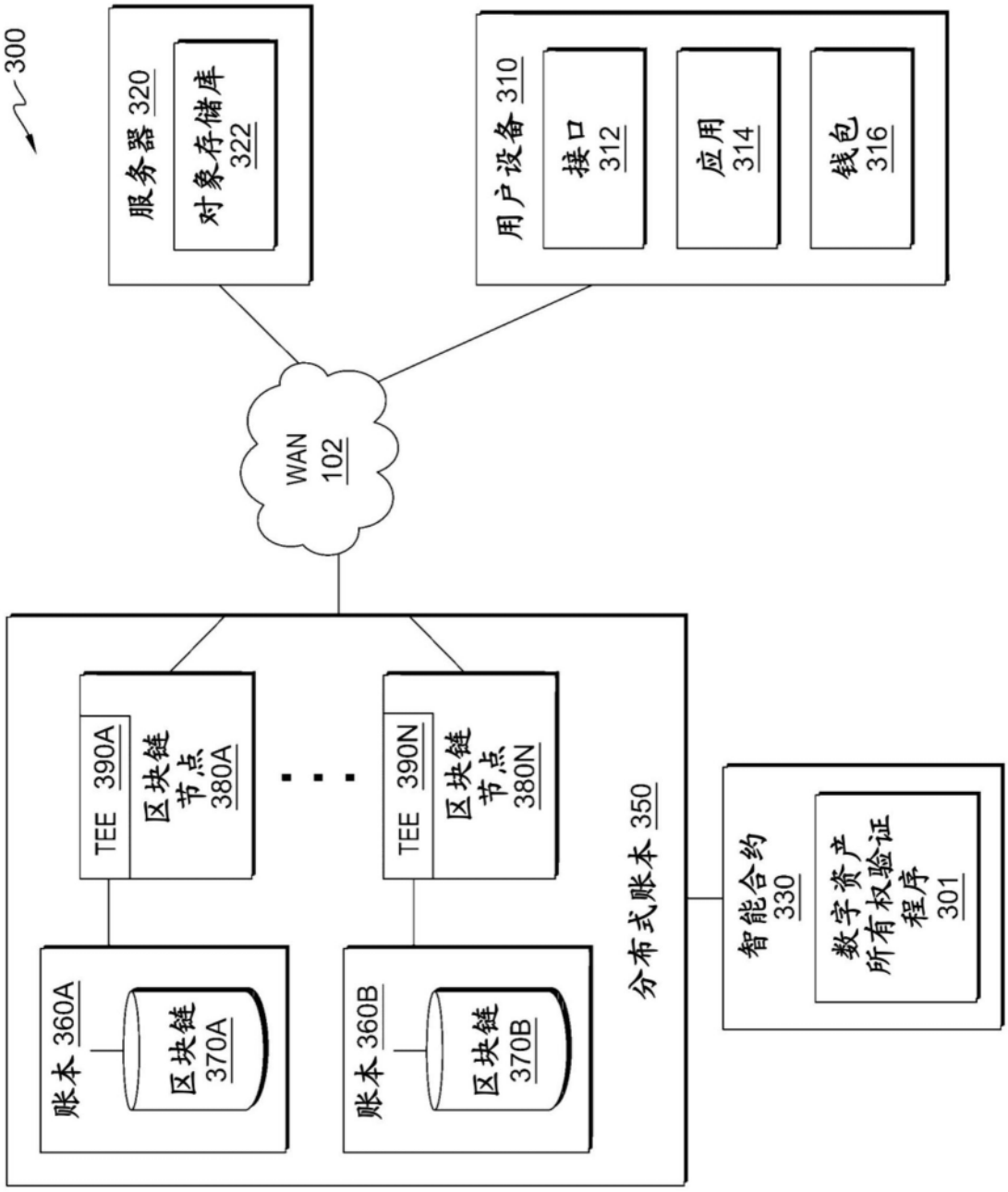

300
服务器 320
对象存储库 322
用户设备 310
接口 312
应用 314
钱包 316
WAN 102
TEE 390A
区块链节点 380A
TEE 390N
区块链节点 380N
分布式账本 350
账本 360A
区块链 370A
账本 360B
区块链 370B
智能合约 330
数字资产所有权验证程序 301

图3

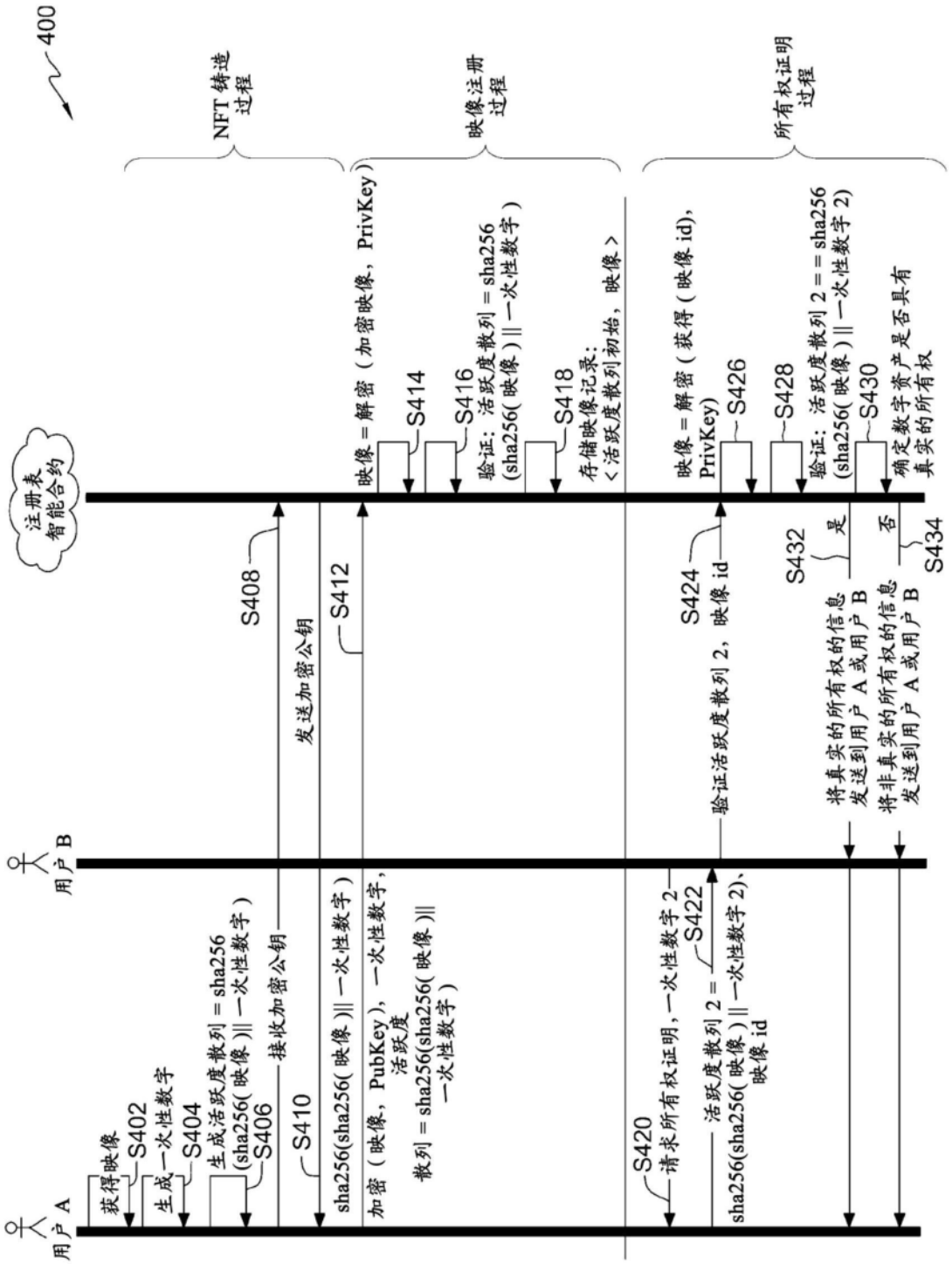

400
NFT 铸造过程
映像注册过程
所有权证明过程
注册表智能合约
用户 A
用户 B
获得映像
S402
生成一次性数字
S404
生成活跃度散列 = sha256(sha256(映像)|| 一次性数字)
S406
S408
接收加密公钥
发送加密公钥
S410
sha256(sha256(映像)|| 一次性数字)
S412
加密(映像，PubKey)，一次性数字，活跃度散列 = sha256(sha256(映像)|| 一次性数字)
映像 = 解密(加密映像，PrivKey)
S414
S416
验证：活跃度散列 = sha256(sha256(映像)|| 一次性数字)
S418
存储映像记录：<活跃度散列初始，映像>
S420
请求所有权证明，一次性数字 2
S422
活跃度散列 2 = sha256(sha256(映像)|| 一次性数字 2)，映像 id
S424
验证活跃度散列 2，映像 id
映像 = 解密(获得(映像 id)，PrivKey)
S426
S428
验证：活跃度散列 2 == sha256(sha256(映像)|| 一次性数字 2)
S430
确定数字资产是否具有真实的所有权
S432
是
将真实的所有权的信息发送到用户 A 或用户 B
否
S434
将非真实的所有权的信息发送到用户 A 或用户 B

图4

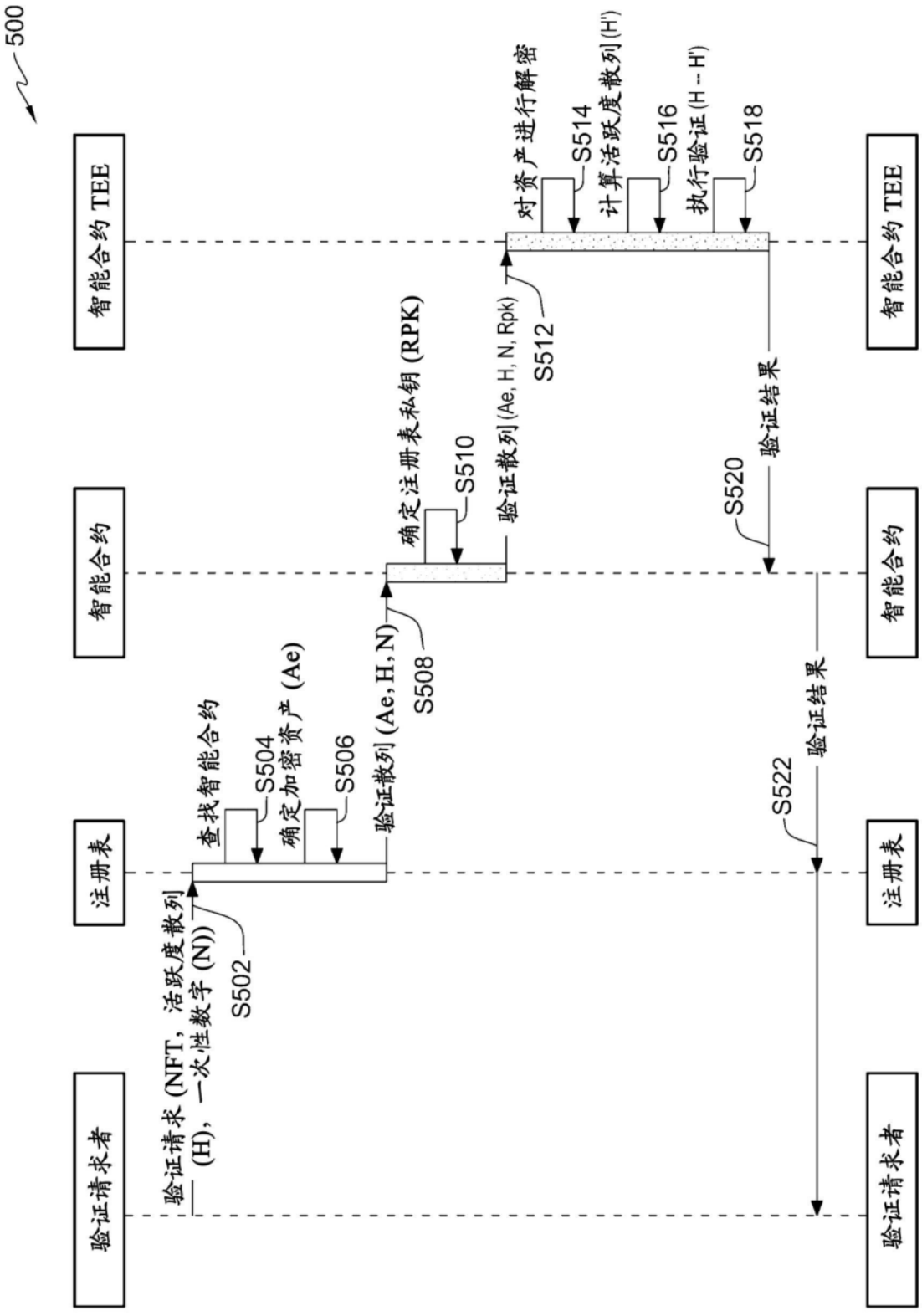

500
验证请求者
注册表
智能合约
智能合约 TEE
验证请求 (NFT，活跃度散列 (H)，一次性数字 (N))
S502
查找智能合约
S504
确定加密资产 (Ae)
S506
验证散列 (Ae, H, N)
S508
确定注册表私钥 (RPK)
S510
验证散列 (Ae, H, N, Rpk)
S512
对资产进行解密
S514
计算活跃度散列 (H')
S516
执行验证 (H -- H')
S518
验证结果
S520
验证结果
S522
验证请求者
注册表
智能合约
智能合约 TEE

图5

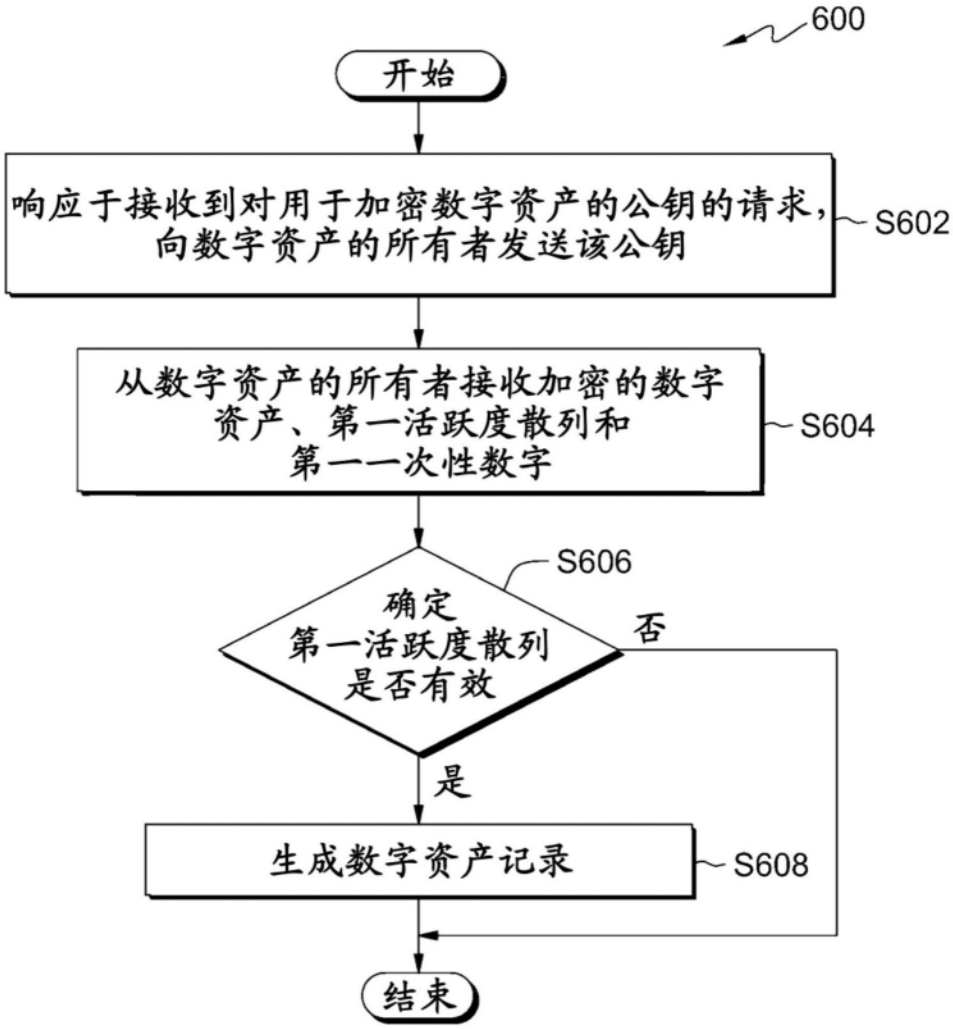

600
开始
响应于接收到对用于加密数字资产的公钥的请求，向数字资产的所有者发送该公钥
S602
从数字资产的所有者接收加密的数字资产、第一活跃度散列和第一一次性数字
S604
S606
确定第一活跃度散列是否有效
否
是
生成数字资产记录
S608
结束

图6

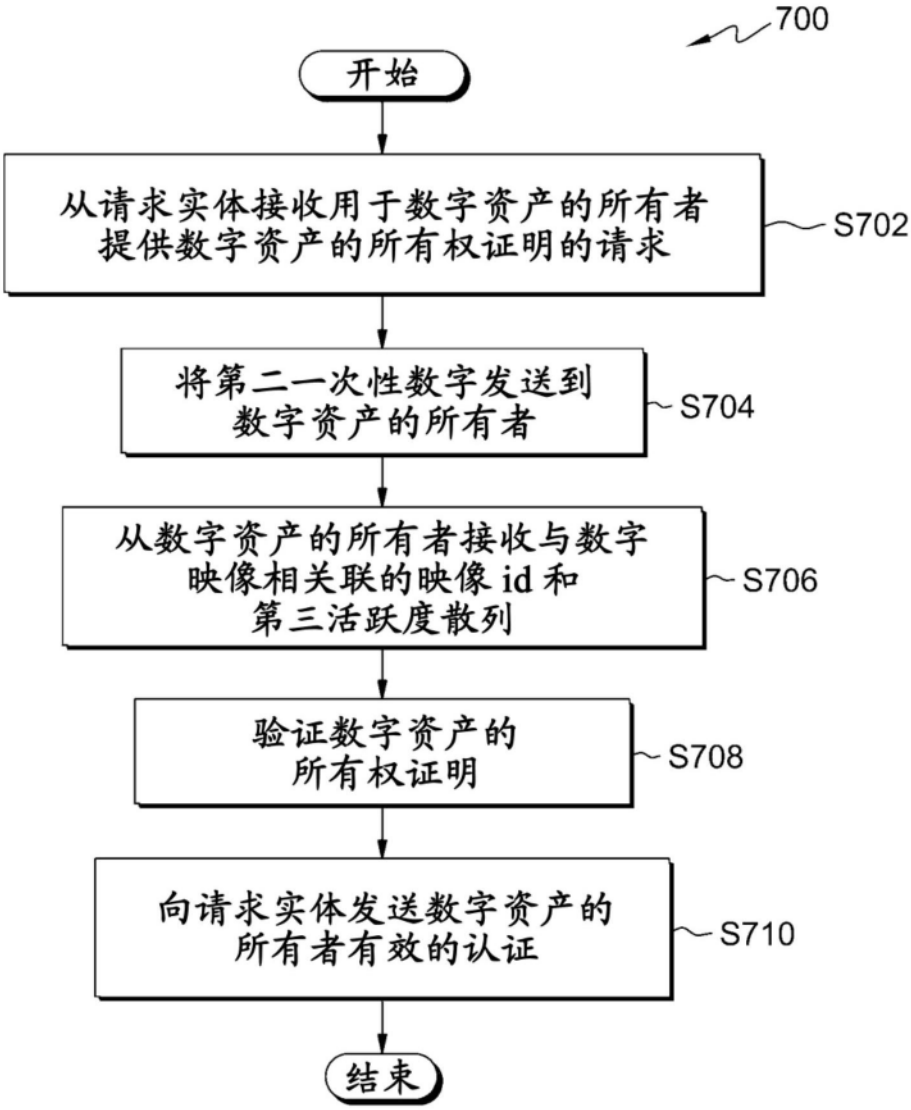

700
开始
从请求实体接收用于数字资产的所有者提供数字资产的所有权证明的请求
S702
将第二一次性数字发送到数字资产的所有者
S704
从数字资产的所有者接收与数字映像相关联的映像 id 和第三活跃度散列
S706
验证数字资产的所有权证明
S708
向请求实体发送数字资产的所有者有效的认证
S710
结束

图7

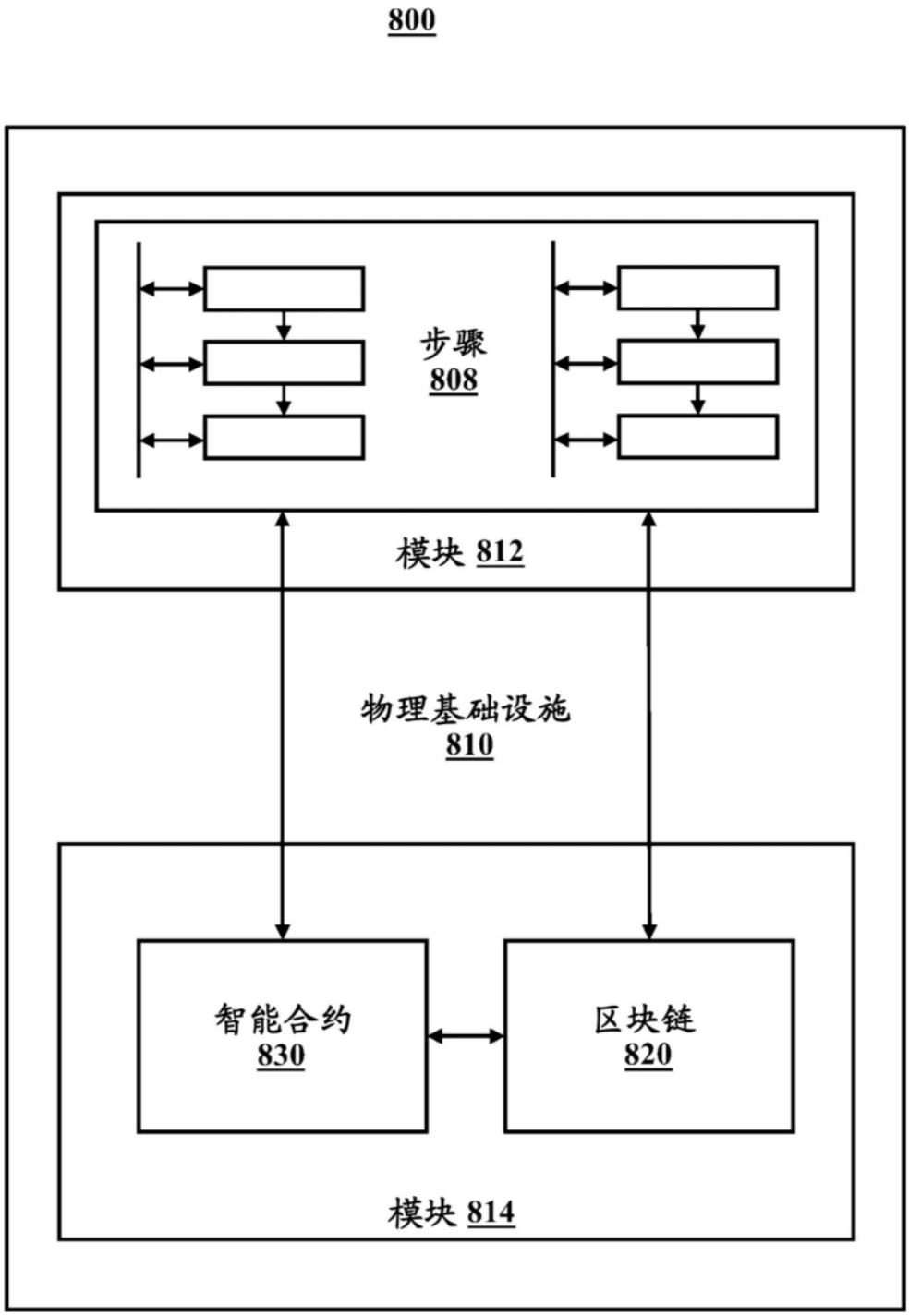


图8A

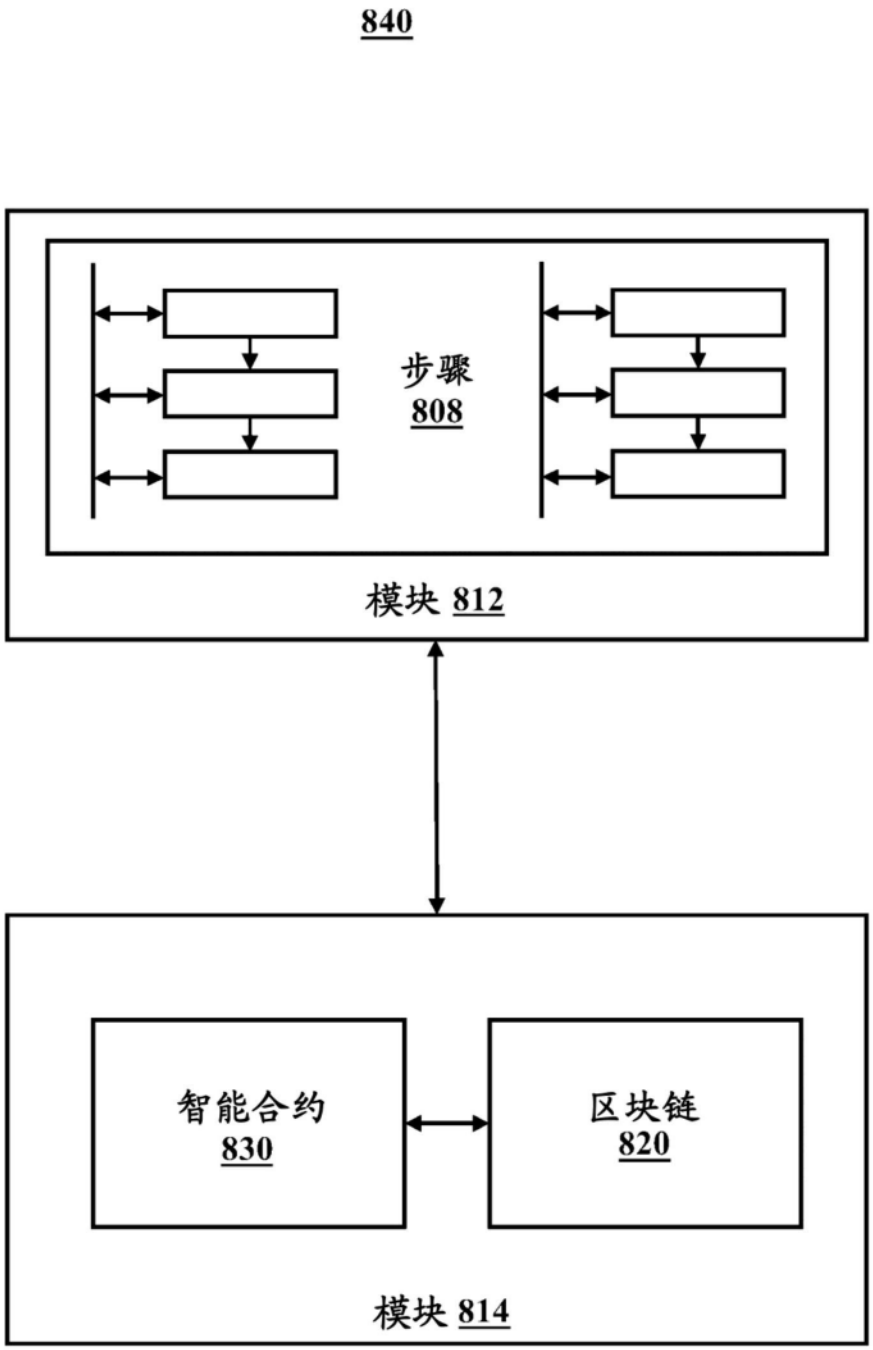


图8B

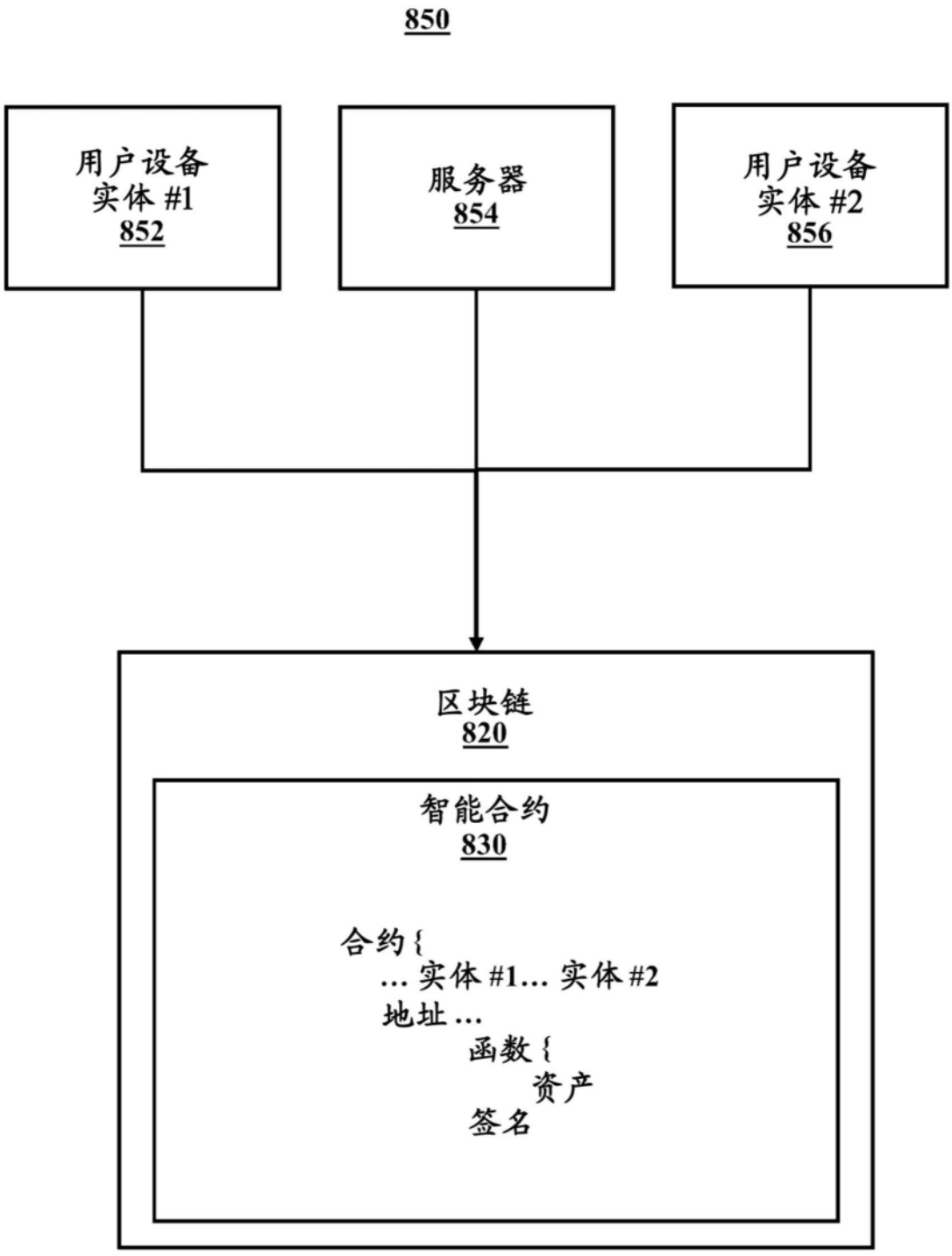

850
用户设备
实体 #1
852
服务器
854
用户设备
实体 #2
856
区块链
820
智能合约
830
合约{
... 实体 #1... 实体 #2
地址 ...
函数{
资产
签名

图8C

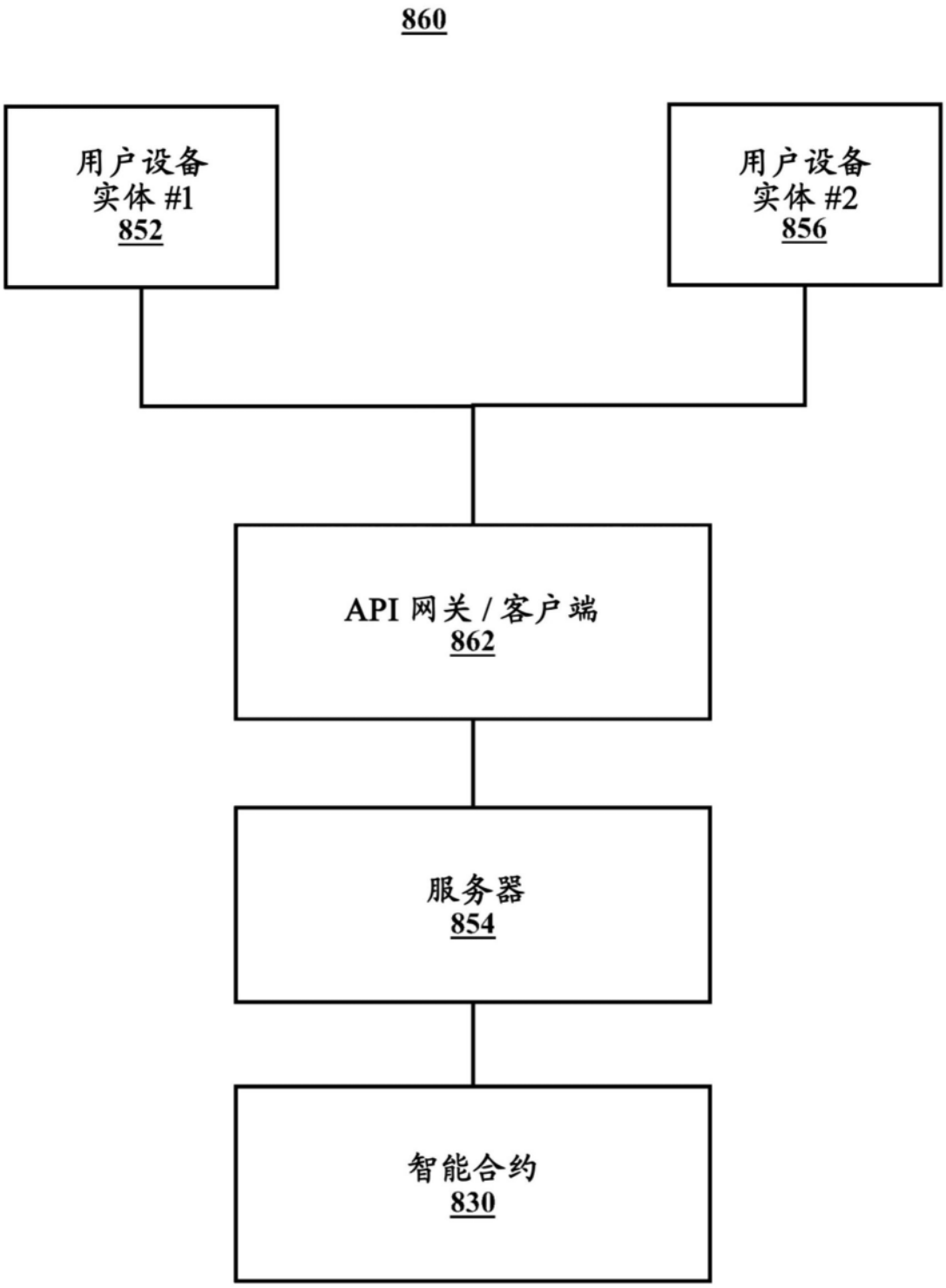


图8D

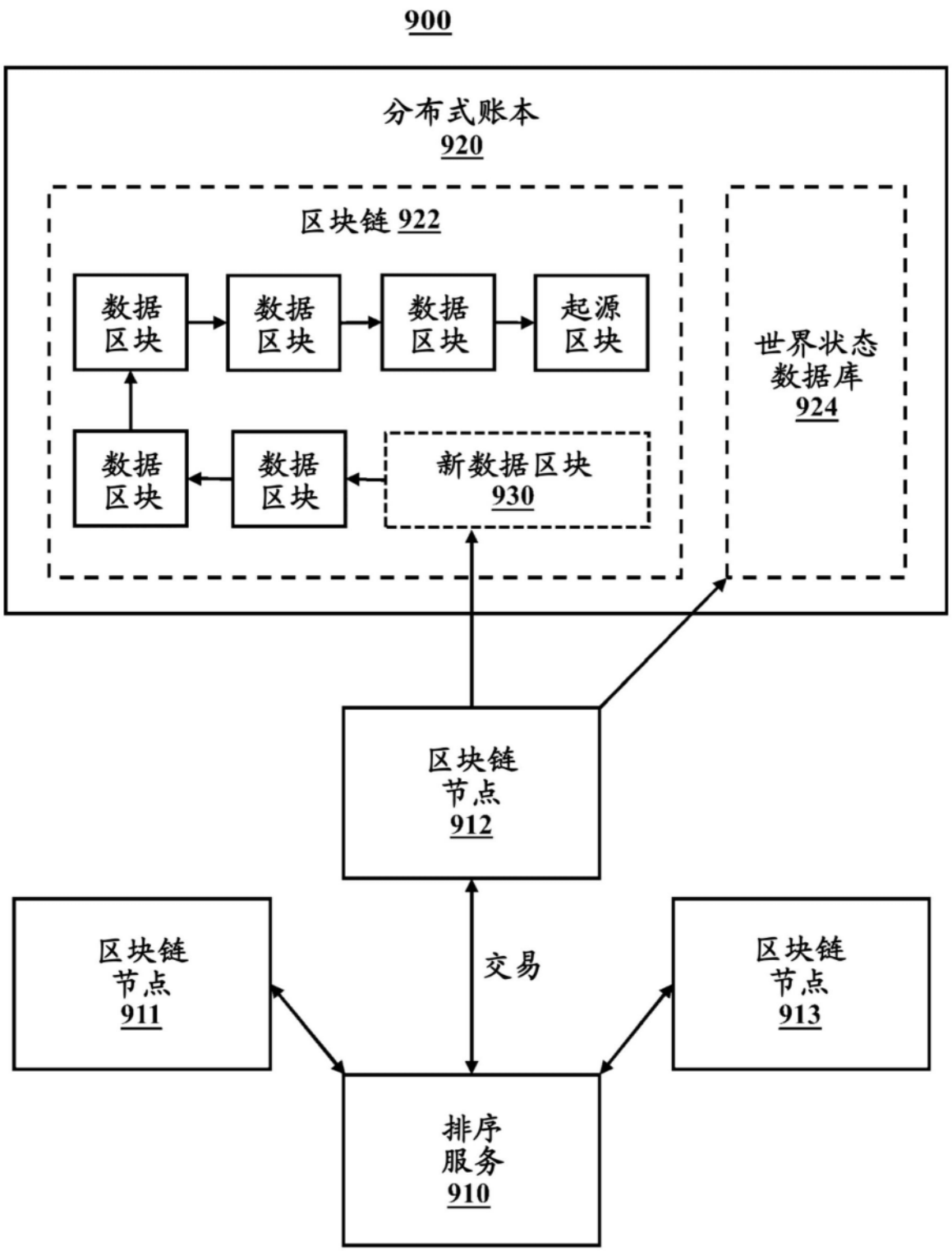

900
分布式账本
920
区块链 922
数据区块
数据区块
数据区块
起源区块
数据区块
数据区块
新数据区块
930
世界状态数据库
924
区块链节点
912
区块链节点
911
交易
区块链节点
913
排序服务
910

图9A

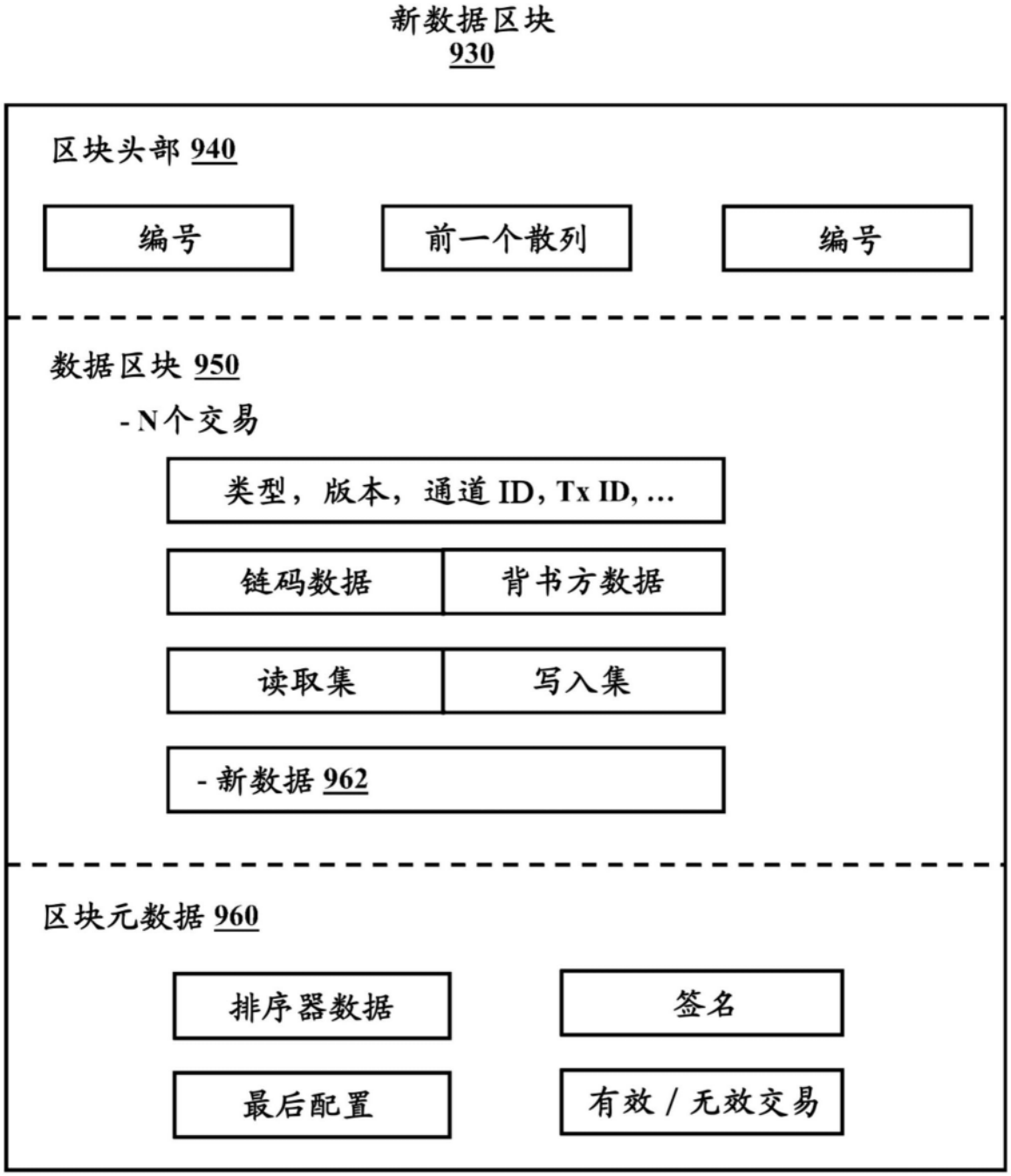

新数据区块
930
区块头部 940
编号
前一个散列
编号
数据区块 950
- N个交易
类型，版本，通道 ID, Tx ID, ...
链码数据
背书方数据
读取集
写入集
- 新数据 962
区块元数据 960
排序器数据
签名
最后配置
有效 / 无效交易

图9B

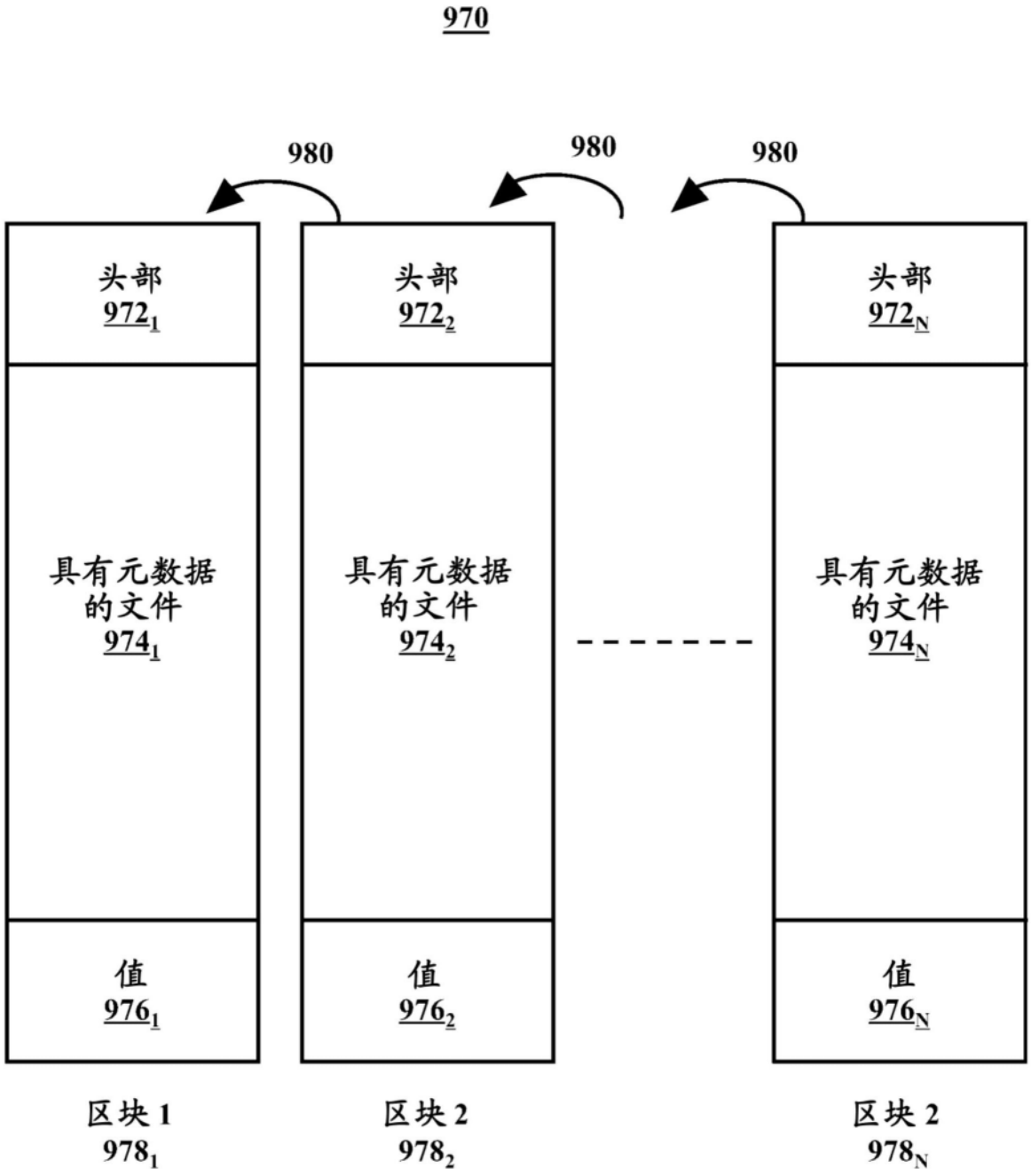

970
980
980
980
头部
972_1
头部
972_2
头部
972_N
具有元数据
的文件
974_1
具有元数据
的文件
974_2
具有元数据
的文件
974_N
值
976_1
值
976_2
值
976_N
区块 1
978_1
区块 2
978_2
区块 2
978_N

图9C

990

区块$_i$

头部 972$_i$ - 前一个区块的散列值 - 引用信息
文件（以及元数据）974$_i$ 数据 1 引用 1 元数据 1 数据 2 引用 2 元数据 2 . . . . 数据 N 引用 N 元数据 N
值 976$_i$（基于以下一个或多个） - 文件的新散列值 - 文件的新存储位置 - 被分配给文件的新元数据 - 访问 / 控制向新区块链参与者的转移 - 文件的新的 / 现有 / 更改所有权

图9D