(19)中华人民共和国国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 112800132 A
(43)申请公布日 2021.05.14

(21)申请号 202110037518.2

(22)申请日 2021.01.12

(71)申请人 东北大学
地址 110819 辽宁省沈阳市和平区文化路3号巷11号

(72)发明人 刘园 陈俣欣 谭立元

(74)专利代理机构 沈阳东大知识产权代理有限公司 21109
代理人 梁焱

(51)Int.Cl.
*G06F 16/27*(2019.01)
*H04L 29/06*(2006.01)
*H04L 29/08*(2006.01)

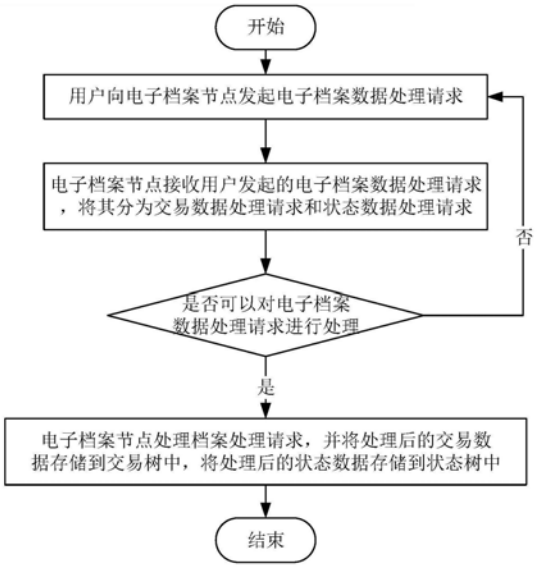
权利要求书2页 说明书7页 附图5页

(54)发明名称

一种电子档案的区块链存储方法

(57)摘要

本发明公开了一种电子档案的区块链存储方法，属于电子档案的存储技术领域。电子档案节点接收电子档案数据处理请求，将其分为交易数据处理请求和状态数据处理请求；判断是否可以对电子档案数据处理请求进行相应的处理，若是，则继续处理请求，若否，则需重新向电子档案节点发起电子档案数据处理请求；针对接收的电子档案数据处理请求进行相应处理，并将处理后的交易数据存储到交易树，将处理后的状态数据存储到状态树，状态树为存储状态数据的MPT树，交易树为存储交易数据的MPT树；将需存储的电子档案数据按照数据映射关系进行存储，查询电子档案数据时按照相应的映射关系进行查询。可以有效地保障电子档案数据的真实性、完整性和隐私性。




CN 112800132 A

1.一种电子档案的区块链存储方法,其特征在于,包括以下步骤:

步骤1:用户向电子档案节点发起电子档案数据处理请求;

所述电子档案数据处理请求,包括:增加、修改、删除或查询电子档案数据;

步骤2:电子档案节点接收用户发起的电子档案数据处理请求,将其分为交易数据处理请求和状态数据处理请求;

将需存储的电子档案数据分为交易数据和状态数据,将用户向电子档案节点发起的增加、修改或删除电子档案数据都视为一笔交易,对应的电子档案数据称为交易数据,交易数据需存储在电子档案节点的区块体中;状态数据是记录电子档案当前状态的数据,状态数据需存储在电子档案节点本地;

步骤3:电子档案节点判断是否可以对电子档案数据处理请求进行相应的处理,若是,则执行步骤4,若否,则用户需重新向电子档案节点发起电子档案数据处理请求,转至步骤1;

若电子档案节点接收的是增加、修改或删除电子档案数据的请求,即电子档案节点接收到一笔交易,则执行步骤3.1;若电子档案节点接收的是查询档案数据的请求,则执行步骤3.2;

步骤3.1:将交易广播给其他电子档案节点,进行交易共识,若达成交易共识,则转至步骤3.2,若未达成交易共识,则拒绝处理该交易,用户需重新向电子档案节点发起电子档案数据处理请求,转至步骤1;

步骤3.2:判断电子档案节点已存储的交易数据和状态数据是否被篡改,若是,则该电子档案节点需从其他电子档案节点重新获取未被篡改的交易数据和状态数据,若否,执行步骤4;

步骤4:针对步骤2中接收的电子档案数据处理请求进行相应处理,并将处理后的交易数据存储到交易树中,将处理后的状态数据存储到状态树中;

所述状态树是指存储状态数据的MPT树,交易树是指存储交易数据的MPT树。

2.根据权利要求1所述的电子档案的区块链存储方法,其特征在于,在步骤2中,将用户向电子档案节点发起的增加、修改或删除电子档案数据请求视为用户向电子档案节点发起的增加交易数据请求、以及分别视为用户向电子档案节点发起的增加、修改或删除状态数据请求;用户在电子档案节点发起的查询电子档案数据请求,视为用户向电子档案节点发起的查询交易数据的请求和查询状态数据的请求。

3.根据权利要求1所述的电子档案的区块链存储方法,其特征在于,步骤3.2所述的判断电子档案节点已存储的交易数据和状态数据是否被篡改的方法如下:

电子档案节点判断当前区块存储的前一区块交易树根节点哈希值与前一区块存储的当前区块交易树根节点哈希值是否相同,若否,则说明该电子档案节点已存储的交易数据被篡改,若是,则说明该电子档案节点已存储的交易数据未被篡改;

电子档案节点判断当前区块存储的前一区块状态树根节点哈希值与前一区块存储的当前区块状态树根节点哈希值是否相同,若否,则说明该电子档案节点已存储的状态数据被篡改,若是,则说明该电子档案节点已存储的状态数据未被篡改。

4.根据权利要求2所述的电子档案的区块链存储方法,其特征在于,在所述步骤4中:

针对步骤2中接收的增加交易数据请求,在电子档案节点中增加相应的交易数据,并将

增加的交易数据存储到交易树中；

针对步骤2中接收的增加、修改或删除状态数据请求，在电子档案节点中增加、修改或删除相应的状态数据，并将增加或修改的状态数据存储到状态树中；

针对步骤2中接收的查询交易数据请求和查询状态数据请求，电子档案节点从交易数据中查询到历史电子档案数据、从状态数据中查询到当前电子档案数据。

5.根据权利要求4所述的电子档案的区块链存储方法，其特征在于，在所述步骤4中，增加交易数据的过程包括：电子档案节点将为增加的交易生成交易ID，同时将交易数据按照一定的映射关系与交易ID绑定。

6.根据权利要求4所述的电子档案的区块链存储方法，其特征在于，在所述步骤4中，电子档案节点将需存储的电子档案数据按照数据映射关系进行存储，查询电子档案数据时按照相应的映射关系进行查询。

7.根据权利要求6所述的电子档案的区块链存储方法，其特征在于，在所述步骤4中，在电子档案节点中通过映射关系从交易数据中查询到历史电子档案数据和通过映射关系从状态数据中查询到当前电子档案数据。

8.根据权利要求6所述的电子档案的区块链存储方法，其特征在于，通过隐藏数据的映射关系对特定的数据进行保护。

# 一种电子档案的区块链存储方法

## 技术领域

[0001] 本发明属于电子档案的存储技术领域，具体涉及一种电子档案的区块链存储方法。

## 背景技术

[0002] 传统的档案管理采用纸质档案进行存储和管理，其管理制度已经相对完善，但如今网络的快速发展提供了一种基于网络的新型管理模式，那就是电子档案。相对于传统的纸质档案，电子档案可以在一定程度上提升档案的利用率、流通性以及便捷性，在存储和查阅上也拥有更大的优势。

[0003] 但是，电子档案的真实性、完整性以及隐私性有待考量。纸质档案可以通过判断材质的时间以及填写人的笔记来验证档案的产生时间以及真伪，通过档案室保存档案保证档案的隐私性。

## 发明内容

[0004] 本发明为了解决上述问题，提供了一种电子档案的区块链存储方法，旨在利用区块链每次交易都会验证数据的真实性、完整性以及隐私性的特点，对电子档案进行存储，以保障电子档案数据的真实性、完整性以及隐私性。

[0005] 为解决上述技术问题，本发明提供一种技术方案：

[0006] 一种电子档案的区块链存储方法，包括以下步骤：

[0007] 步骤1：用户向电子档案节点发起电子档案数据处理请求；

[0008] 所述电子档案数据处理请求，包括：增加、修改、删除或查询电子档案数据；

[0009] 步骤2：电子档案节点接收用户发起的电子档案数据处理请求，将其分为交易数据处理请求和状态数据处理请求；

[0010] 将需存储的电子档案数据分为交易数据和状态数据，将用户向电子档案节点发起的增加、修改或删除电子档案数据都视为一笔交易，对应的电子档案数据称为交易数据，交易数据需存储在电子档案节点的区块体中；状态数据是记录电子档案当前状态的数据，状态数据需存储在电子档案节点本地；

[0011] 步骤3：电子档案节点判断是否可以对电子档案数据处理请求进行相应的处理，若是，则执行步骤4，若否，则用户需重新向电子档案节点发起电子档案数据处理请求，转至步骤1；

[0012] 若电子档案节点接收的是增加、修改或删除电子档案数据的请求，即电子档案节点接收到一笔交易，则执行步骤3.1；若电子档案节点接收的是查询档案数据的请求，则执行步骤3.2；

[0013] 步骤3.1：对于电子档案节点接收到增加、修改或删除电子档案数据后，即电子档案节点接收到一笔交易，交易会被广播给其他电子档案节点，进行交易共识，若达成交易共识，则转至步骤3.2，若未达成交易共识，则拒绝处理该交易，则用户需重新向电子档案节点

发起电子档案数据处理请求,转至步骤1;

[0014] 步骤3.2:判断电子档案节点已存储的交易数据和状态数据是否被篡改,若是,则该电子档案节点需从其他电子档案节点重新获取未被篡改的交易数据和状态数据,若否,执行步骤4;

[0015] 步骤4:针对步骤2中接收的电子档案数据处理请求进行相应处理,并将处理后的交易数据存储到交易树中,将处理后的状态数据存储到状态树中;

[0016] 所述状态树是指存储状态数据的MPT树,交易树是指存储交易数据的MPT树。

[0017] 进一步地,根据所述的电子档案的区块链存储方法,在步骤2中,将用户向电子档案节点发起的增加、修改或删除电子档案数据请求视为用户向电子档案节点发起的增加交易数据请求、以及分别视为用户向电子档案节点发起的增加、修改或删除状态数据请求;用户在电子档案节点发起的查询电子档案数据请求,视为用户向电子档案节点发起的查询交易数据的请求和查询状态数据的请求。

[0018] 进一步地,根据所述的电子档案的区块链存储方法,步骤3.2所述的判断电子档案节点已存储的交易数据和状态数据是否被篡改的方法如下:

[0019] 电子档案节点判断当前区块存储的前一区块交易树根节点哈希值与前一区块存储的当前区块交易树根节点哈希值是否相同,若否,则说明该电子档案节点已存储的交易数据被篡改,若是,则说明该电子档案节点已存储的交易数据未被篡改;

[0020] 电子档案节点判断当前区块存储的前一区块状态树根节点哈希值与前一区块存储的当前区块状态树根节点哈希值是否相同,若否,则说明该电子档案节点已存储的状态数据被篡改,若是,则说明该电子档案节点已存储的状态数据未被篡改。

[0021] 进一步地,根据所述的电子档案的区块链存储方法,在所述步骤4中:

[0022] 针对步骤2中接收的增加交易数据请求,在电子档案节点中增加相应的交易数据,并将增加的交易数据存储到交易树中;

[0023] 针对步骤2中接收的增加、修改或删除状态数据请求,在电子档案节点中增加、修改或删除相应的状态数据,并将增加或修改的状态数据存储到状态树中;

[0024] 针对步骤2中接收的查询交易数据请求和查询状态数据请求,电子档案节点从交易数据中查询到历史电子档案数据、从状态数据中查询到当前电子档案数据。

[0025] 进一步地,根据所述的电子档案的区块链存储方法,在所述步骤4中,增加交易数据的过程包括:电子档案节点将为增加的交易生成交易ID,同时将交易数据按照一定的映射关系与交易ID绑定。

[0026] 进一步地,根据所述的电子档案的区块链存储方法,在所述步骤4中,电子档案节点将需存储的电子档案数据按照数据映射关系进行存储,查询电子档案数据时按照相应的映射关系进行查询。

[0027] 进一步地,根据所述的电子档案的区块链存储方法,在所述步骤4中,在电子档案节点中通过映射关系从交易数据中查询到历史电子档案数据和通过映射关系从状态数据中查询到当前电子档案数据。

[0028] 进一步地,根据所述的电子档案的区块链存储方法,通过隐藏数据的映射关系对特定的数据进行保护。

[0029] 本发明的有益效果:1、在电子档案数据存储形式上分为交易数据存储和状态数据

存储，可以将每一次的档案操作增添进行记录，记录到区块链中保证数据的完整性与真实性。交易记录了每一笔档案更新，即所有的交易构成了档案。并可以将每一次的档案更新反映给节点，获取到最新的档案信息。这样可以保证档案查看的便捷性，更加符合电子档案的需求。2、采用MPT数据存储结构存储数据，可以保障数据存储的完整性和真实性，通过观察根节点的Hash值，就可以判断数据是否被篡改，保障电子档案的安全性和完整性。3、采取了数据映射存储的方式，来保障数据的隐私性，使只有有权利的用户节点才能查阅到档案的真实用户信息，其他节点只能查看到分散的数据，不能知道真实的数据组合，保障数据的隐私性。

## 附图说明

[0030] 图1是本发明电子档案的区块链存储方法流程图；

[0031] 图2是本发明用户向电子档案节点发起处理请求情况示意图；

[0032] 图3是本发明一档案数据示例图；

[0033] 图4是本发明区块链系统中数据存储结构示意图；

[0034] 图5是本发明电子档案节点对数据一致性判断的示意图

[0035] 图6是本发明另一档案数据组成结构示意图；

[0036] 图7是本发明电子档案节点修改状态数据示例图。

## 具体实施方式

[0037] 为了便于理解本申请，下面将参照相关附图对本申请进行更全面的描述。附图中给出了本申请的较佳实施方式。但是，本申请可以以许多不同的形式来实现，并不限于本文所描述的实施方式。相反地，提供这些实施方式的目的是使对本申请的公开内容理解的更加透彻全面。

[0038] 区块链中存储的数据是不可被篡改的，一旦数据被存储到区块链中，就不能再进行修改，在区块链每次交易都会验证数据的真实性、完整性以及隐私性。区块链交易中会对每项交易都进行记录，区块链中存储的数据具有“不可伪造”“可以追溯”的特征，非常适用于存储电子档案。

[0039] 本实施方式的电子档案的区块链存储方法，如图1所示，包括以下步骤：

[0040] 步骤1、用户向电子档案节点发起电子档案数据处理请求；

[0041] 如图2所示，本实施方案中，用户可以向电子档案节点发起以下四种档案处理请求，本发明也是将以用户向电子档案节点发起的以下四种档案处理请求为例对本发明电子档案的区块链存储方法进行详细说明，当然，本领域的技术人员容易理解到除了这四种档案处理请求以外的其他档案处理请求也适用于本发明电子档案的区块链存储方法。

[0042] A.用户向电子档案节点发起增加电子档案数据的请求；

[0043] 例如：用户A请求增加电子档案数据“用户A->档案ID:0001->姓名:张三;身份证号:123456789;身高:160cm”，其中“->”代表“档案ID:0001”是“用户A”附属，如图3所示，“;”代表相互并列关系。电子档案节点将为用户A生成增加该电子档案数据的请求。

[0044] B.用户向电子档案节点发起修改电子档案数据的请求；

[0045] 例如：对于电子档案数据“用户A->档案ID:0001->姓名:张三;身份证号：

123456789;身高:160cm”,用户A请求将其中的“身高:160cm”修改为“身高:170cm”,电子档案节点将为用户A生成修改电子档案数据的请求,如:“用户A->档案ID:0001->姓名:张三;身份证号:123456789;身高:170cm”。

[0046] C.用户向电子档案节点发起删除电子档案数据的请求;

[0047] 例如:对于电子档案数据“用户A->档案ID:0001->姓名:张三;身份证号:123456789;身高:160cm”,用户A请求删除电子档案数据“档案ID:0001”,电子档案节点将为用户A生成删除“档案ID:0001”档案数据的请求。

[0048] D.用户向电子档案节点发起查询档案数据的请求;

[0049] 例如:用户A请求查询电子档案数据“档案ID:0001”,电子档案节点将为用户A生成查询“档案ID:0001”电子档案数据的请求。

[0050] 步骤2、电子档案节点接收用户发起的电子档案数据处理请求,将其分为交易数据处理请求和状态数据处理请求;

[0051] 为了便于用户在电子档案节点查询历史档案数据和当前档案数据,本实施方式将需存储的电子档案数据分为交易数据和状态数据。将用户向电子档案节点发起的增加、修改或删除电子档案数据都视为一笔交易,对应的电子档案数据称为交易数据,交易数据需存储在电子档案节点的区块体中;状态数据是记录电子档案当前状态的数据,状态数据需存储在电子档案节点本地。

[0052] 如图4所示,区块链系统中的区块(Block)分为区块头(Header)和区块体(Body)。区块头中封装了当前区块状态树根节点哈希值、前一区块状态树根节点哈希值、当前区块交易树根节点哈希值和前一区块交易树根节点哈希值。其中,状态树是指存储状态数据的MPT树,交易树是指存储交易数据的MPT树。MPT(Merkle Patricia Tree,MPT树)是Merkle Tree和Patricia Tree的结合。MPT树根节点的哈希值随着存储数据内容的变化而变化。如果数据发生篡改或者缺失,MPT树根节点的哈希值将会发生变化,所以可以保障数据的真实性和完整性,更加适用于电子档案数据的存储。如图4所示,区块体中封装了交易数据的集合,可视为一个交易池(transaction pool)。

[0053] 对于步骤1中,用户向电子档案节点发起的增加、修改或删除档案数据请求,对应到区块链系统中,则是电子档案节点接收用户发起的交易数据处理请求和状态数据处理请求。

[0054] 电子档案节点接收用户发起的交易数据处理请求:交易数据存储在区块体中,因为区块链中存储的数据只能增加不能修改,所以用户向电子档案节点发起的增加、修改或删除请求都视为增加交易数据请求。例如,1)与增加档案数据请求对应地是在电子档案节点请求增加交易数据,例如,请求增加交易数据“用户A->档案ID:0001->姓名:张三;身份证号:123456789;身高:160cm”。2)与修改档案数据请求对应地是在电子档案节点中请求增加交易数据,例如,请求增加交易数据“用户A->档案ID:0001->姓名:张三;身份证号:123456789;身高:170cm”档案数据。3)与删除档案数据处理请求对应地是在电子档案节点请求增加交易数据,例如,请求增加交易数据“用户A->null->null;null;null”档案数据。

[0055] 电子档案节点接收用户发起的状态数据处理请求:状态数据存储在电子档案节点本地,状态数据是记录电子档案当前状态的数据,所以用户向电子档案节点发起的增加、修改或删除数据请求分别视为增加、修改或删除状态数据请求。例如:1)与增加电子档案数据

请求对应地是在电子档案节点请求增加状态数据，例如，请求增加状态数据“用户A->档案ID:0001->姓名:张三;身份证号:123456789;身高:160cm”档案数据。2)与修改电子档案数据请求对应地是在电子档案节点请求修改状态数据，例如，请求将档案数据“用户A->档案ID:0001->姓名:张三;身份证号:123456789;身高:160cm”修改为档案数据“用户A->档案ID:0001->姓名:张三;身份证号:123456789;身高:170cm”。3)与删除电子档案数据请求对应地是在电子档案节点请求删除状态数据，例如，与用户A删除“档案ID:0001”档案数据请求对应地是在电子档案节点请求删除状态数据“档案ID:0001”。

[0056] 同理，对于步骤1中，用户向电子档案节点发起的查询档案数据请求，对应到区块链系统中，则是电子档案节点接收用户发起的查询交易数据的请求和查询状态数据的请求。例如，电子档案节点同时接收用户发起的查询“档案ID:0001”历史数据和查询“档案ID:0001”当前数据的请求。

[0057] 步骤3:电子档案节点判断是否可以对电子档案数据处理请求进行相应的处理，若是，则执行步骤4，若否，则用户需重新向电子档案节点发起电子档案数据处理请求，转至步骤1;

[0058] 若电子档案节点接收的是增加、修改或删除电子档案数据的请求，即电子档案节点接收到一笔交易，则执行步骤3.1:

[0059] 步骤3.1:对于电子档案节点接收到增加、修改或删除电子档案数据后，即电子档案节点接收到一笔交易，交易会被广播给其他电子档案节点，进行交易共识，若其他电子档案节点认可该交易，则达成交易共识，可以继续处理该交易，转至步骤3.2。所述交易共识是指对于交易的验证，其他电子档案节点是否达成一致的验证过程。若其他节点对该交易的验证达成一致，则可以继续处理该交易。若其他节点对该交易的验证未达成一致，则未达成交易共识，拒绝处理该交易，则用户需重新向电子档案节点发起电子档案数据处理请求，转至步骤1。

[0060] 步骤3.2:判断电子档案节点已存储的交易数据和状态数据是否被篡改，若是，则该电子档案节点需从其他电子档案节点重新获取未被篡改的交易数据和状态数据，若否，执行步骤4;

[0061] 如图5所示，电子档案节点首先判断当前区块存储的前一区块交易树根节点哈希值与前一区块存储的当前交易树根节点哈希值是否相同，若否，则说明该电子档案节点已存储的交易数据被篡改，则该电子档案节点需从其他电子档案节点重新获取未被篡改的交易数据，若是，则可以继续处理交易数据请求。如图4所示，区块头中存储了当前区块交易树根节点哈希值和前一区块交易树根节点哈希值。交易数据存储在区块体中，区块链中存储的数据只能增加不能修改，若恶意电子档案节点试图篡改区块链中存储的数据，那么前一区块存储的当前区块交易树根节点哈希值将会发生变化，与当前区块存储的前一区块交易树根节点哈希值将会不同。若不同，则说明交易数据被篡改，则需要从其他电子档案节点重新获取交易数据，再进行处理交易数据请求。若相同，则说明该电子档案节点已存储的交易数据未被篡改，可以继续处理交易数据请求。

[0062] 如图5所示，电子档案节点接下来应判断最新区块存储的前一区块状态树根节点哈希值与上一区块存储的状态树根节点哈希值是否相同，若否，则则说明该电子档案节点已存储的状态数据被篡改，需要从其他电子档案节点重新获取状态数据，若是，则说明该电

子档案节点已存储的状态数据被篡改，则可以继续处理状态数据请求。如图4所示，区块头中存储了当前区块状态树根节点哈希值和前一区块状态树根节点哈希值。状态数据存储在电子档案节点本地中，若恶意电子档案节点篡改状态数据，那么前一区块存储的当前区块状态树根节点哈希值将会发生变化，与当前区块存储的前一区块状态树根节点哈希值将会不同。若不同，则说明状态数据被篡改，则需要从其他电子档案节点重新获取状态数据，再进行处理状态数据请求。若相同，则说明状态数据没有被篡改，可以继续处理状态数据请求。

[0063] 对于用户向电子档案节点发起的查询档案数据请求，电子档案节点接收后，首先判断当前区块存储的前一区块交易树根节点哈希值与前一区块存储的当前区块交易树根节点哈希值是否相同。若不同，则说明交易数据被篡改，则需要从其他电子档案节点重新获取交易数据，再进行处理交易数据请求。若相同，则说明交易数据没有被篡改，可以继续处理交易数据请求；接下来判断当前区块存储的前一区块状态树根节点哈希值与上一区块存储的当前区块状态树根节点哈希值是否相同。若不同，则说明状态数据被篡改，则需要从电子档案节点重新获取状态数据，在进行处理状态数据请求。若相同，则说明状态数据没有被篡改，可以继续处理状态数据请求。

[0064] 步骤4、针对步骤2接收的电子档案数据处理请求进行相应处理，并将处理后的交易数据存储到交易树中，将处理后的状态数据存储到状态树中；

[0065] 电子档案节点将需存储的档案数据按照数据映射关系进行存储，查询时也是按照数据映射关系进行查询。

[0066] 本实施方式中，将档案数据的组成部分分为元数据与映射关系。元数据是逻辑上的定义，代表档案数据存储时的不可拆分最小数据。例如，对于档案数据“用户A->档案ID:0001->姓名:张三;身份证号:123456789;身高:160cm”，将“档案ID:0001”对应的数据“姓名:张三”、“身份证号:123456789”和“身高:160cm”称为元数据，存在三个映射关系“档案ID:0001”->“姓名:张三”、“档案ID:0001”->“身份证号:123456789”和“档案ID:0001->“身高:160cm”。各个元数据之间存在长度至少为1的映射链，例如，如图6所示的数据“用户B->档案ID:0002->姓名:李四;身高:160cm;高考分数:600分->语文:120分”，其中“档案ID:0002”分别对应的数据“姓名:李四”、“身高:160cm”、“高考分数:600分”、“语文:120分”均称为元数据，“档案ID:0002”->“姓名:李四”、“档案ID:0002”->“身高:160cm”均为长度为1的映射链，“档案ID:0002”->“高考分数:600分”->“语文:120分”为长度为2的映射链，即该映射链上的元数据个数即为映射链长度。

[0067] 这种存储方法能更好的保障档案数据的隐私性，可以使涉及到隐私的敏感数据得到更好的保护。因为非法操作获取得到的信息是无效的，例如，数据“身高:160cm”其没有对应的映射关系，没有获得授权的电子档案节点将不知道这个数据对应的是谁的身高，这样的信息将是没有意义的。所述授权是指对电子档案节点授予操作电子档案的权限，如增加、修改、删除和查询电子档案操作的权限。

[0068] 电子档案节点处理用户发起的增加、修改或删除档案数据请求，包括处理用户向电子档案节点发起的交易数据请求和处理用户向电子档案节点发起的状态数据请求。

[0069] 针对步骤2中所述的增加档案数据、修改档案数据和删除档案数据的增加交易数据请求，在电子档案节点中都会对应地增加交易数据，并将增加的交易数据存储到交易树

中。

[0070] 增加交易数据的过程包括:电子档案节点将为增加的交易生成交易ID,同时将交易数据与交易ID绑定。例如,为增加的交易数据“用户A->档案ID:0001->姓名:张三;身份证号:123456789;身高:160cm”生成“交易ID:1001”。电子档案节点可以通过1001[0]获取到数据“姓名:张三”,其中1001代表交易ID,[0]代表数据存储的映射位置,可以通过1001[1]获取到数据“身份证号:123456789”,可以通过1001[2]获取到数据“身高:160cm”,其中0,1,2为数据存储的位置,数据存储的位置可以随机选取,通过映射关系,就可以找到对应的数据。

[0071] 进一步地,还可以通过隐藏数据的映射关系对较为敏感的信息进行保护。例如,对于“身份证号:123456789”这个较为敏感的信息,可以通过1001[1]存储数据“身份证号”,通过1001[47]存储数据“123456789”,使得交易ID与交易数据存储之间是无序关联的,这样可以在一定程度上保证数据的安全。

[0072] 针对步骤2中所述的增加档案数据请求,对应地,电子档案节点需增加状态数据,并将增加的状态数据存储到状态树中。状态数据是通过对应的交易数据存储位置获取的。例如,数据“用户A->档案ID:0001->姓名:张三;身份证号:123456789;身高:160cm”,根据上述通过1001[0]获取到数据“姓名:张三”,通过1001[1]获取到数据“身份证号:123456789”,通过1001[2]获取到数据“身高:160cm”,那么在状态数据中也是通过1001[0]获数据“姓名:张三”,通过1001[1]获取数据“身份证号:123456789”,通过1001[2]获取数据“身高:160cm”。

[0073] 针对步骤2中所述的修改档案数据请求,对应地,电子档案节点需修改相应的状态数据,并将修改的状态数据存储到状态树中。例如,如图7所示的数据“用户A->档案ID:0001->姓名:张三;身份证号:123456789;身高:160cm”修改为数据“用户A->档案ID:0001->姓名:张三;身份证号:123456789;身高:170cm”。根据在交易数据中,通过1001[0]获取到数据“姓名:张三”,通过1001[1]获取到数据“身份证号:123456789”,通过1002[0]获取到数据“身高:170cm”,那么,在状态数据中也通过1001[0]获取到数据“姓名:张三”,通过1001[1]获取数据“身份证号:123456789”,通过1002[0]获取数据“身高:160cm”。

[0074] 针对步骤2中所述的删除档案数据请求,对应地,电子档案节点需删除相关的状态数据,且电子档案节点直接删除相关的状态数据即可。

[0075] 针对步骤2中所述的用户向电子档案节点发起的查询档案数据请求,对应到电子档案节点,则是查询交易数据和查询状态数据两个方面的请求。根据上述内容,档案数据是被封装在交易数据中,交易数据中存储着历史电子档案数据,则可通过映射关系从交易数据中查询到历史电子档案数据。根据上述内容,档案数据同时还被封装在状态数据中,状态数据中存储着当前电子档案数据,则可通过映射关系从状态数据中查询到当前电子档案数据。

[0076] 至此,用户向电子档案节点发起的档案处理请求处理完成,电子档案节点将处理结果返回给电子档案节点。

[0077] 应当理解的是,本领域技术人员在本发明技术构思的启发下,在不脱离本发明内容的基础上,可以根据上述说明做出各种改进或变换,这仍落在本发明的保护范围之内。

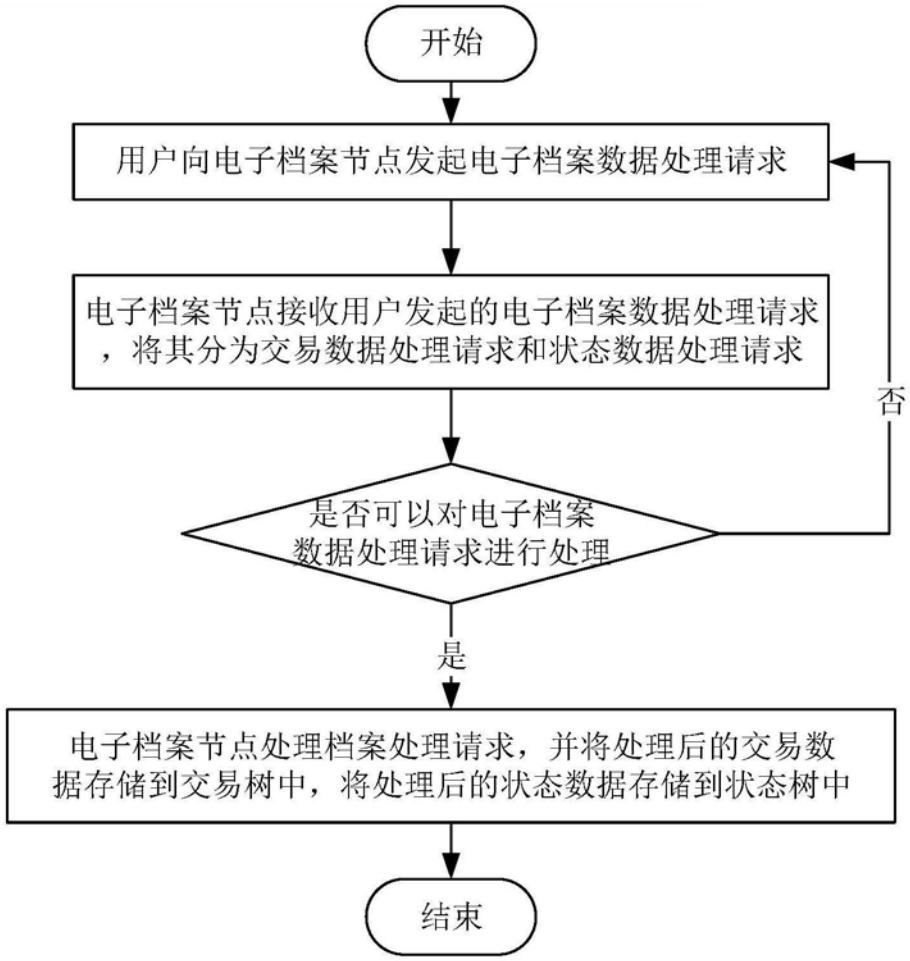

开始
用户向电子档案节点发起电子档案数据处理请求
电子档案节点接收用户发起的电子档案数据处理请求，将其分为交易数据处理请求和状态数据处理请求
是否可以对电子档案数据处理请求进行处理
否
是
电子档案节点处理档案处理请求，并将处理后的交易数据存储到交易树中，将处理后的状态数据存储到状态树中
结束

图1

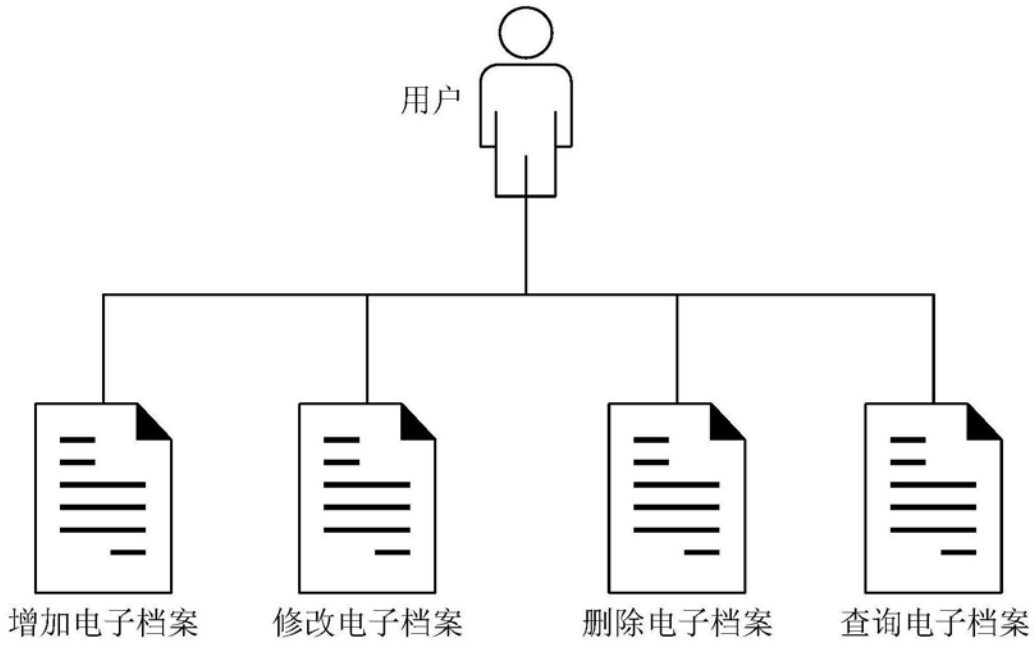

用户
增加电子档案
修改电子档案
删除电子档案
查询电子档案

图2

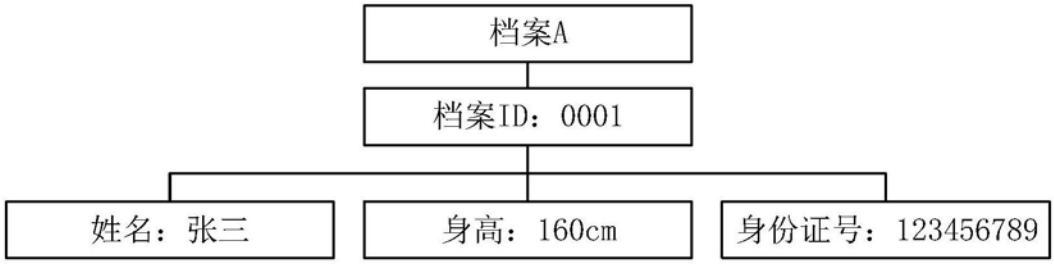

档案A
档案ID：0001
姓名：张三
身高：160cm
身份证号：123456789

图3

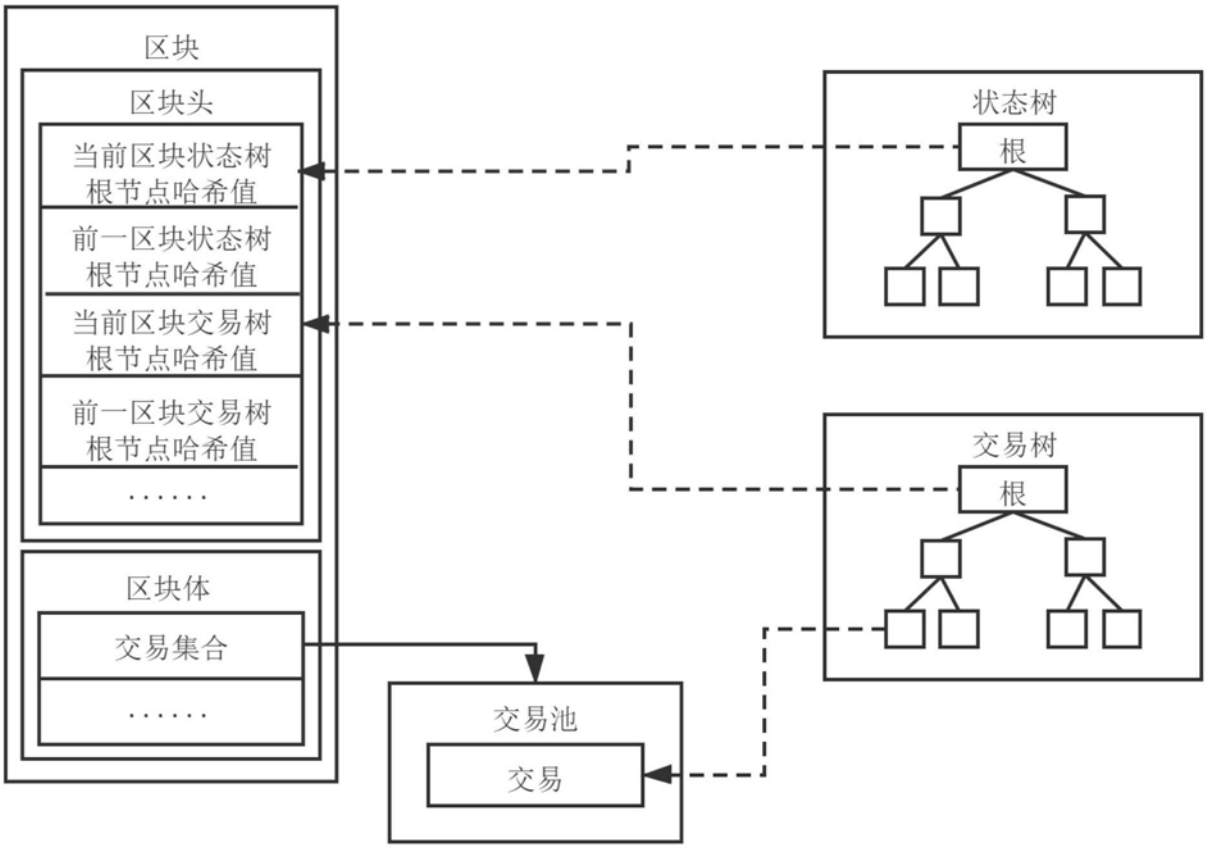

区块
区块头
当前区块状态树
根节点哈希值
前一区块状态树
根节点哈希值
当前区块交易树
根节点哈希值
前一区块交易树
根节点哈希值
.....
区块体
交易集合
.....
状态树
根
交易树
根
交易池
交易

图4

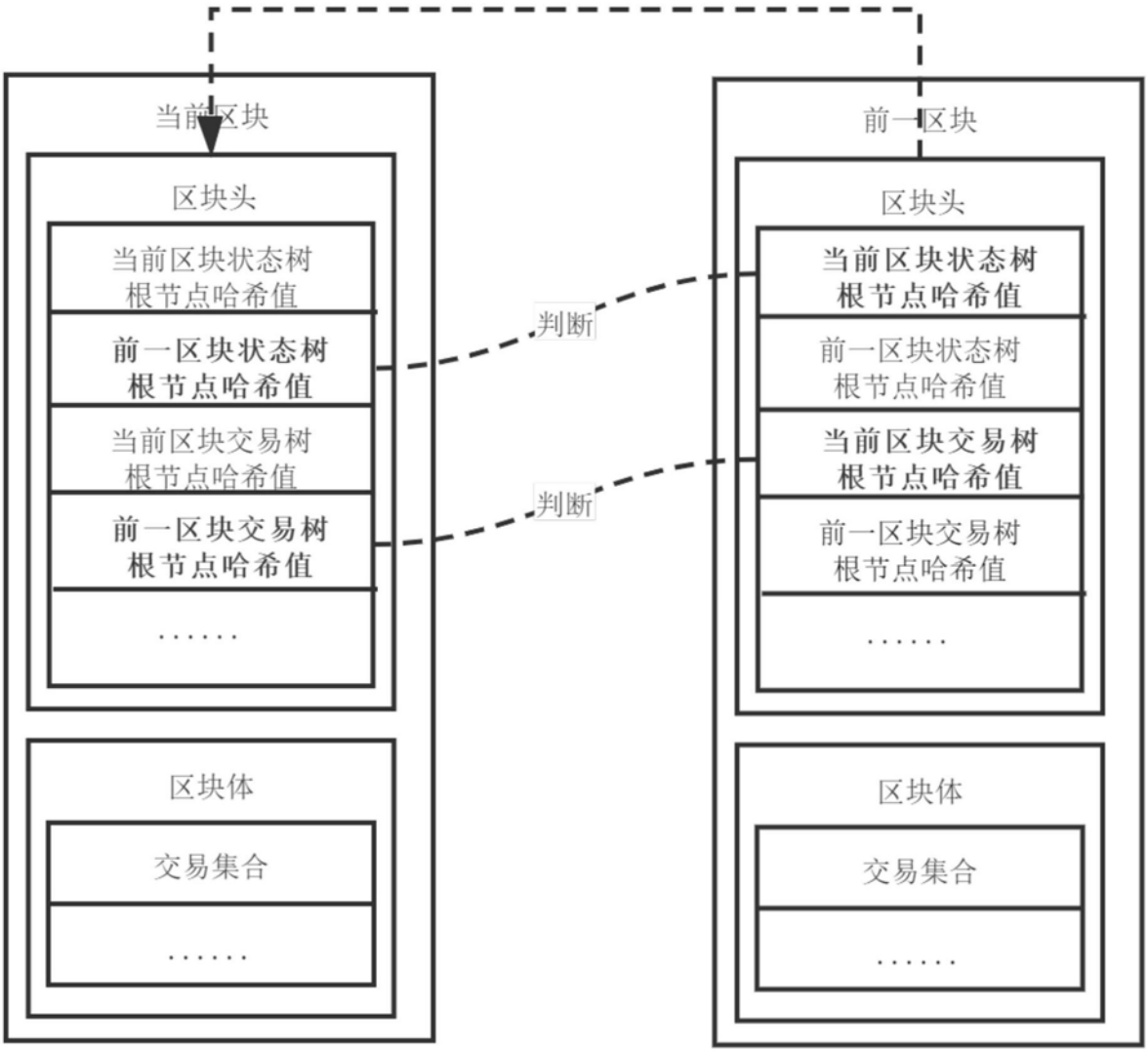

当前区块
区块头
当前区块状态树
根节点哈希值
前一区块状态树
根节点哈希值
当前区块交易树
根节点哈希值
前一区块交易树
根节点哈希值
.....
区块体
交易集合
.....
判断
判断
前一区块
区块头
当前区块状态树
根节点哈希值
前一区块状态树
根节点哈希值
当前区块交易树
根节点哈希值
前一区块交易树
根节点哈希值
.....
区块体
交易集合
.....

图5

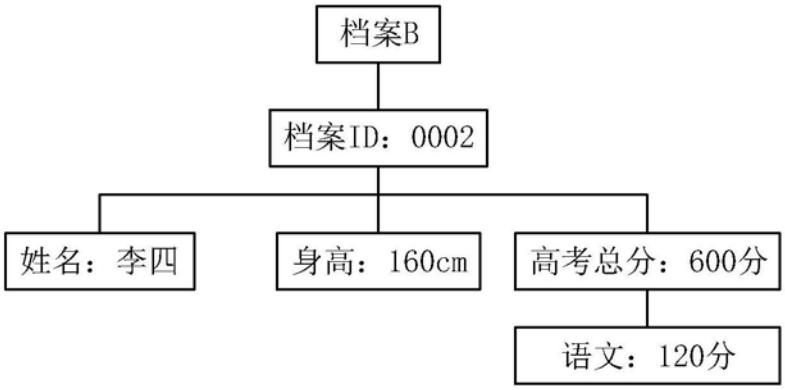

档案B
档案ID：0002
姓名：李四
身高：160cm
高考总分：600分
语文：120分

图6

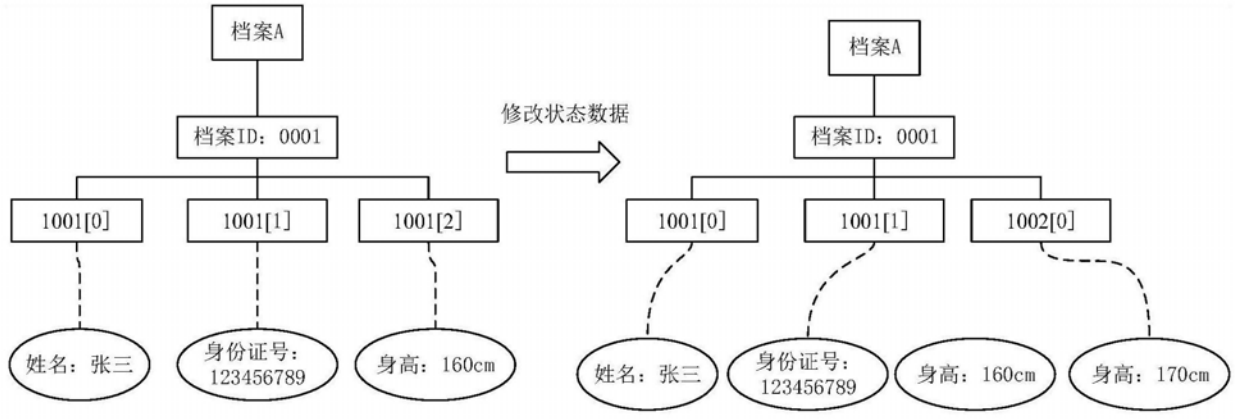

档案A
档案ID：0001
1001[0]
1001[1]
1001[2]
姓名：张三
身份证号：
123456789
身高：160cm
修改状态数据
档案A
档案ID：0001
1001[0]
1001[1]
1002[0]
姓名：张三
身份证号：
123456789
身高：160cm
身高：170cm

图7