

一种基于区块链的用户余额查询方法及计算机系统

说明书

技术领域

本说明书涉及计算机技术领域，尤其涉及一种基于区块链的用户余额查询方法及计算机系统。

背景技术

目前，区块链技术正在日趋成熟，其应用范围也逐渐扩展和普及，在供应链管理、线上资产、金融衍生品等领域中都得到了广泛应用。区块链技术已经成为现阶段数字经济时代中的关键基础技术之一。

然而，伴随着区块链用户的日益增多，区块链网络压力的不断攀升，交易处理效率上的局限性问题逐渐显露。在网络使用高峰期阶段，用户在查询账号个人数据中的余额信息时，往往需要向区块链网络直接提出查询申请。而在查询申请的执行过程中可能会因为网络拥堵而被迫延长查询时间，从而导致用户的交易决策受到影响。这种查询方式不仅影响了用户体验，使得用户无法实时获取准确的账户状态信息，同时也极大程度上降低了交易效率，严重影响交易任务的正常执行。甚至在高频高额的交易场景中，较长时间的余额查询延迟和不稳定性可能会给交易双方带来重大经济损失。

因此，如何能够实时且高效率的查询区块链用户的余额信息，是至关重要的。

发明内容

本说明书实施例提供一种基于区块链的用户余额查询方法及计算机系统，以部分解决上述现有技术存在的问题。

本说明书提供了一种基于区块链的用户余额查询方法，包括：获取余额查询请求，所述余额查询请求用于查询用户在区块链网络中的虚拟货币的余额；根据所述余额查询请求，从预设的存储节点中查询出所述用户在区块链网络中虚拟货币的余额信息和余额存储时间信息，所述余额存储时间信息用于表示所述余额信息存储在所述存储节点的时间；根据所述余额存储时间信息，确定所述余额存储时间信息对应的时间与当前时间的时间间隔；若确定所述时间间隔不大于预设的存储时间阈值，则将所述存储节点中存储的所述的余额信

息返回给所述用户。

本说明书提供了一种计算机系统，所述计算机系统包括存储器、处理器及存储在所述存储器上并可在所述处理器上运行的计算机程序，所述处理器执行所述程序时实现如下步骤：获取余额查询请求，所述余额查询请求用于查询用户在区块链网络中的虚拟货币的余额；根据所述余额查询请求，从预设的存储节点中查询出所述用户在区块链网络中虚拟货币的余额信息和余额存储时间信息，所述余额存储时间信息用于表示所述余额信息存储在所述存储节点的时间；根据所述余额存储时间信息，确定所述余额存储时间信息对应的时间与当前时间的时间间隔；若确定所述时间间隔不大于预设的存储时间阈值，则将所述存储节点中存储的所述的余额信息返回给所述用户。

本说明书提供了一种可读存储介质，所述存储介质存储有计算机程序，所述计算机程序被处理器执行时实现上述基于区块链的用户余额查询方法。

本说明书采用的上述至少一个技术方案能够达到以下有益效果：

在本说明书实施例提供的基于区块链的用户余额查询方法中，能够将用户在区块链网络中的虚拟货币的余额信息存储在存储节点中，使得用户在想要对自己账户的余额信息进行查询时，可以直接从存储节点中获取，无需再向区块链网络发送查询请求。通过此方法可以有效提高用户查询余额信息时的查询速度，提升用户进行信息查询时的体验，并且一定程度上减轻了区块链网络的网络负载，提高了区块链网络中交易场景中的交易执行效率。

附图说明

此处所说明的附图用来提供对本说明书的进一步理解，构成本说明书的一部分，本说明书的示意性实施例及其说明用于解释本说明书，并不构成对本说明书的不当限定。在附图中：

图 1 为本说明书中提供的一种用户从区块链网络查询余额信息的实例示意图；

图 2 为本说明书中提供的一种基于区块链的用户余额查询方法的流程示意图；

图 3 为本说明书中提供的一种用户在数字钱包上发起余额查询请求的示意图；

图 4 为本说明书中提供的一种根据存储更新任务对存储节点对进行更新的流程示意图；

图 5 为本说明书中提供的一种区块链网络交易场景下存储节点的更新过程的流程示意图；

图 6 为本说明书中提供的一种用户通过基于区块链的用户余额查询方法进行余额查询时的实例示意图；

图 7a, 图 7b, 图 7c 为本说明书中提供的一种将存储节点中的余额信息返回给用户的流程示意图;

图 8a 为本说明书中提供的一种利用区块链网络进行跨境交易的实例示意图;

图 8b 为本说明书中提供的一种跨境交易中支付方式选取界面的示意图;

图 9 为本说明书实施例提供的计算机系统的结构示意图。

具体实施方式

为使本说明书的目的、技术方案和优点更加清楚,下面将结合本说明书具体实施例及相应的附图对本说明书技术方案进行清楚、完整地描述。显然,所描述的实施例仅是本说明书一部分实施例,而不是全部的实施例。基于本说明书中的实施例,本领域普通技术人员在没有做出创造性劳动前提下所获得的所有其他实施例,都属于本说明书保护的范围。

目前,区块链技术已经日渐成熟,在诸如数字资产交易、金融衍生品市场等数字经济领域展都展现出了极其强大的影响力和实用性。伴随着区块链技术的不断普及和广泛应用,使用用户的增长速率也随之加快,用户基数的不断提升给整体区块链网络带来的网络压力和拥堵问题也在日渐增多。

在现阶段区块链技术中,用户一般都会对自己在区块链网络上的账户余额保持严格把控,从而会随时或定期向区块链网络进行余额信息的查询,查询实例具体可以参考图 1 中所示。

图 1 为本说明书中提供的一种用户从区块链网络查询余额信息的实例示意图。

从图 1 的实例所示可以看出,用户可以通过自身所使用的终端设备与区块链网络进行信息交互,通过发送余额查询请求,以从区块链网络中查询出用户在区块链网络中虚拟货币的余额信息。此种查询手段虽然可以获得较为准确的余额信息,但查询效率却会因为区块链整体网络问题而受到影响。之前提到过的,随着现阶段区块链网络中使用用户的不断增多,导致网络压力日渐攀升。而在众多用户同期使用区块链网络执行各类操作的情况下,单个用户对自己在区块链网络中虚拟货币的余额信息进行查询时,可能会面临网络拥堵和信息返回延迟的问题,即用户在向区块链发出查询余额信息的请求后,可能会延迟很久时间才收到区块链网络返回的余额信息。此种情况的出现严重影响到了用户获取余额信息时的操作体验,并且信息返回延迟的问题可能会导致用户接收到的余额信息的实时性受到影响。在涉及高频率高交易额的区块链交易场景中,过长时间的余额查询延迟和余额信息的延迟误差都可能对整体交易环节造成十分严重的负面影响。

另外，在用户通过上述手段直接向区块链网络进行查询的时候，可能会遇到进行信息交互的区块链节点中并不存有完整的区块链总体交易记录的情况，而在这种情况下，区块链网络会先对节点所记录的交易信息进行更新，再对用户的余额信息进行查询和返回，整体所花费的时间更长，过程也更加繁琐，严重降到了用户进行余额查询时的查询效率。综上所述可以看出，如何能够实时且高效的查询区块链网络中用户的余额信息，是个至关重要的问题。

为此，本说明书提供了一种基于区块链的用户余额查询方法，用于解决上述在对区块链网络进行用户余额查询时存在的查询效率问题。其中，本说明书提供的方法所采用的执行主体可以是诸如台式电脑、笔记本电脑等终端设备，也可以是服务器，而除此之外，本说明书的执行主体也可以为软件形式的主体，如安装在终端设备中的客户端等。而为了便于说明，下面本说明书仅以终端设备作为执行主体，对所提供的基于区块链的用户余额查询方法进行说明。

以下结合附图，详细说明本说明书各实施例提供的技术方案。

图 2 为本说明书中提供的一种基于区块链的用户余额查询方法的流程示意图，包括以下步骤：

S201：获取余额查询请求。

S202：根据所述余额查询请求，从预设的存储节点中查询出所述用户在区块链网络中虚拟货币的余额信息和余额存储时间信息。在本说明书中，终端设备可以根据用户发起的余额查询请求，从预设的存储节点中查询出用户在区块链网络中虚拟货币的余额信息和对应的余额存储时间信息。其中，余额存储时间信息用于表示余额信息存储在存储节点中的具体时间。

其中，终端设备获取余额查询请求的具体方式可以是在用户对区块链客户端中可交互的查询余额控件进行操作时，终端设备生成并获取的。当前，还可以是监测到用户通过区块链网络在终端设备中的链下网页进行主动余额查询的操作行为等，在本说明书中不对用户发起的余额查询请求的具体形式和发起途径进行严格限定，根据实际应用场景的不同可进行灵活调整。为了便于理解用户发起余额查询请求的具体过程，下面以一个数字钱包的示意图，来进行介绍，如图 3 所示。

图 3 为本说明书中提供的一种用户在数字钱包上发起余额查询请求的示意图。

如图 3 中所示，用户在此实施例中，可以通过点击图 3 界面中的查询余额控件，向数字钱包应用所处的终端设备发起余额查询请求。终端设备确定出用户的余额信息后，可以

将余额信息显示在图 3 中的“¥XXX”处，以使得用户可以直观的看到自己在区块链网络中的虚拟货币的具体余额数值。

另外，上述提及的存储节点可以是用户所使用的终端设备，当前，也可以是设立在具有后公信力的机构的服务器，终端设备可以向该服务器发送余额查询请求，以进行余额查询。因此在本说明书中，存储节点的具体数据形式和类型在本说明书中不进行严格限定，可根据实际应用需求和应用场景进行灵活设定。而为了便于理解存储节点中的所包含的具体信息和格式，下面以一个存储节点中存储信息的实例进行介绍和说明，具体如表 1 所示。

bridge_token_hold_info
gmt_create:timestamp gmt_modified:timestamp balance_update_time:timestamp
id:bigint unsigned(20) holder_address:varchar(128) chain_id:varchar(64) token_id:varchar(64) bank_account_id_list:varchar(65535) token_hold_status:varchar(16) creator:varchar(32) updater:varchar(32) balance:bigint(20)

表 1

表 1 为存储节点存储信息一种形式。其中，“balance_update_time”用于表示存储节点中存储的余额信息对应的余额存储时间信息，“holder_address”用于表示用户在区块链网络中的地址信息，“chain_id”用于表示用户所使用的区块链网络的 id 信息，“token_id”用于表示用户在区块链网络中的虚拟货币的 id 信息，“balance”用于表示用户在区块链网络中的虚拟货币的具体数值，即用户的余额信息。

基于表 1 所示的存储节点的实例来说，假设此实例中的存储节点设立在终端设备中，终端设备可以根据获取到的余额查询请求，从存储节点中查询出用户在区块链网络中的虚拟货币的余额信息，以及余额信息所对应的余额存储时间信息，即表 1 中所示的“balance”和“balance_update_time”。

另外，需要说明的是，如果存储节点为用户所使用的终端设备，那么，该存储节点只会存储该用户在区块链网络中的信息数据，并不会将区块链网络中其他用户的信息进行存

储。这样可以有效地保证用户的信息安全。

除此之外，在本说明书中，终端设备在根据余额查询请求从存储节点查询出用户的余额信息和余额存储时间信息之前，可以先确定存储节点中是否存储有用户的余额信息和余额存储时间信息。而若是确定出存储节点中并未存储有用户的余额信息和余额存储时间信息，终端设备可以直接根据余额查询请求，向区块链网络发送针对用户余额信息的获取请求，进而从区块链网络中查询出用户的当前余额信息。

在从区块链网络中查询出用户的当前余额信息后，终端设备可以直接跳过后续步骤，将当前余额信息返回给余额查询请求对应的用户，并同时当前余额信息存储在存储节点中，以及记录当前余额信息存储在存储节点时的余额存储时间信息。

S203：根据所述余额存储时间信息，确定所述余额存储时间信息对应的时间与当前时间的时间间隔。

S204：若确定所述时间间隔不大于预设的存储时间阈值，则将所述存储节点中存储的所述余额信息返回给所述用户。

在本说明书中，终端设备可以根据存储节点中的余额存储时间信息，确定出余额存储时间信息所示对应的时间与当前时间的时间间隔的大小。在确定时间间隔不大于预设的存储时间阈值后，终端设备可以将存储节点中对应的余额信息返回给用户。

而当终端设备确定时间间隔大于预设的存储时间阈值时，终端设备可以认为此时存储在存储节点中的余额信息已经不是用户的实时余额信息。终端设备可以向区块链网络发送针对用户余额信息的获取请求，进而从区块链网络中查询出用户在当前时间的余额信息，并可以直接返回给用户。

具体的，终端设备在确定时间间隔大于预设的存储时间阈值时，可以生成针对存储节点的存储更新任务。然后，终端设备可以分配出用于执行存储更新任务的任务进程，以使得终端设备可以通过任务进程从区块链网络中查询出用户在当前时间区块链网络中虚拟货币的余额信息，并将当前余额信息返回给用户。

举例来说，假设预设的存储时间阈值为 24 小时，存储节点中的余额信息对应的余额存储时间信息为当天上午时间 9:00，而在当天下午时间 19:00 的时候终端设备获取到了用户发起的余额查询请求，上午时间 9:00 到下午时间 19:00 之间为 10 小时，并不超过为 24 小时的存储时间阈值，此时终端设备可以认为此时存储节点中的余额信息是用户在区块链网络中虚拟货币的真实余额信息，并将存储节点中的余额信息返回给用户。

而假如用户在第二天的上午时间 10:00 向终端设备发起了余额查询请求，当天上午时间

9:00 到第二天的上午时间 10:00 之间为 25 小时，已经大于为 24 小时的存储时间阈值，此时终端设备可以认为此时存储节点中的余额信息是与区块链网络中用户的真实余额信息是不相符的，进而生成针对存储节点的存储更新任务，以对存储节点中用户的余额信息进行更新。需要说明的是，在本说明书中不对存储时间阈值的具体时长进行限定，可根据实际场景和应用需求进行灵活设定。

另外，终端设备在获取到用户的当前余额信息后，可以将先前用于执行存储更新任务的任务进程进行释放，以使得任务进程可以用于执行其他任务。并且，终端设备还可以根据用户的当前余额信息，对原来存储在存储节点的余额信息进行更新，并同时当前余额信息存储在存储节点的时间信息作为存储节点中余额信息更新后的余额存储时间信息。

除此之外，还需要进行说明的是，在本说明书中，终端设备可能会面临同时处理多个存储更新任务的情况。而为了保证终端设备中任务进程所执行的存储更新任务都是有效且实用的，终端设备可以在执行存储更新任务前，根据存储更新任务的任务状态，确定存储更新任务是否需要被执行。

具体的，终端设备可以确定出存储更新任务的生成时间，并判断存储更新任务的任务生成时间是否是晚于存储节点中余额存储时间信息所对应的的时间。若确定存储更新任务是晚于是余额存储时间信息所对应的的时间后生成的，在可以确定出存储更新任务的任务状态为需要执行的状态，终端设备便可以直接通过上述步骤执行存储更新任务。而若是确定存储更新任务是早于是余额存储时间信息所对应的的时间后生成的，终端设备则可以认为存储节点中的余额信息已经是更新后的，并不需要再执行存储更新任务。

需要说明的是，除了上述根据存储更新任务的生成时间确定存储更新任务的任务状态外，终端设备还可以对存储更新任务施加用于表示不同任务状态的标签信息，以用于区分出需要执行的存储更新任务和不需要执行的存储更新任务。例如，终端设备可以在存储更新任务被创立时赋予标签 0，标签 0 用于表示该存储更新任务是处于未执行的任务状态，需要被执行。而在存储更新任务被执行完毕后，终端设备可以将标签信息改为标签 1，标签 1 用于表示该存储更新任务是处于执行完毕的任务状态，不需要被执行。终端设备可以在面临同时处理多个存储更新任务的情况下，根据存储更新任务的标签信息确定任务状态，进而选择需要被执行的存储更新任务分配任务进程。

为了便于理解上述终端设备根据存储更新任务对存储节点中的余额信息进行更新的总体流程，下面以一个根据存储更新任务对存储节点对进行更新的流程示意图，来进行说明，如图 4 所示。

图 4 为本说明书中提供的一种根据存储更新任务对存储节点进行更新的流程示意图。

如图 4 中所示，终端设备可以根据存储更新任务的任务状态确定存储更新任务是否需要被执行，确定需要被执行后，可以分配相应任务进程用于执行存储更新任务，具体可通过向各任务进程发送操作指令的方式实现任务进程的分配。然后，通过任务进程从区块链网络中查询出用户的当前余额信息，根据当前余额信息对存储节点进行更新，并记录余额存储时间信息。在更新完成后，将任务进程进行释放，从而完成更新过程。

还需进行单独说明的是，上述步骤中提及的用于确定存储节点是否需要更新的存储时间阈值还可以用于将终端设备设定为自动定时更新。终端设备可以根据存储时间阈值的时长大小，对存储节点中的用户余额信息进行一定时间间隔的自动更新。此处的自动更新过程无需通过用户发起的余额查询请求进行触发，完全由终端设备根据存储时间阈值对存储节点进行定期更新，提高用户查询效率的同时，也保证了用户从存储节点获取到的余额信息的实时性和准确性。

另外，在本说明书中，在用户从区块链网络中进行货币交易时，终端设备可以根据区块链网络中用户交易后的余额信息对存储节点进行及时更新，以保证后续用户在进行余额查询时可以获取到最实时且准确的余额信息。为了便于理解终端设备在区块链网络交易操作时对存储节点的更新过程，下面以一个区块链网络交易场景下存储节点的更新过程的示意图，来进行说明，如图 5 所示。

图 5 为本说明书中提供的一种区块链网络交易场景下存储节点的更新过程的流程示意图。

如图 5 中所示，终端设备在获取到用户发起的交易操作后，可以将交易请求发送到对应的区块链网络中，以使区块链网络可以根据用户的交易请求完成交易。在区块链网络完成交易操作后，终端设备可以直接针对存储节点生成对应的存储更新任务，并从区块链网络中查询出用户的交易后余额信息，根据交易后余额信息对存储节点进行更新。此处提及的交易操作可以是如转账交易、合约交易、治理投票等区块链网络中常见的交易行为，在本说明书中不对交易操作的具体类型进行严格限定。

综上所述，为了便于理解本说明书中基于区块链的用户余额查询方法的整体流程，下面以一个用户通过基于区块链的用户余额查询方法进行余额查询时的实例，来进行说明和介绍，如图 6 所示。

图 6 为本说明书中提供的一种用户通过基于区块链的用户余额查询方法进行余额查询时的实例示意图。

如图 6 中所示, 某一区块链的使用用户向终端设备发起了一次余额查询请求, 终端设备在获取到该用户发起的余额查询请求后, 可以先确定存储节点中是否存储有该用户的余额信息和余额存储时间信息。若确定存储节点中存储有余额信息和余额存储时间信息, 终端设备便可以先根据余额存储时间信息, 判断存储节点中的信息是否需要进行更新, 再决定是否将余额信息返回给该用户。而若是确定此时刻的存储节点需要更新, 终端设备则会生成存储更新任务, 以针对存储节点进行更新, 并将更新后当前余额信息返回给该用户, 使得该用户获取到实时且准确的余额信息。

而若是确定此时的存储节点不需要进行更新, 终端设备便可以直接将存储节点中的余额信息返回给该用户, 以供该用户进行查看。另外, 若是之前确定存储节点中并未存储有余额信息和余额存储时间信息, 终端设备可以直接向区块链网络进行查询, 获取到该用户的当前余额信息后, 将当前余额信息返回给用户进行查看。

除上述内容外, 在本说明书中, 终端设备还可以在确定存储节点中存储有用户的余额信息时, 先直接将存储节点中的余额信息返回给用户, 然后在根据余额存储时间信息判断存储节点中的余额信息是否需要更新。若确定需要更新, 终端设备可以生成相应的存储更新任务, 进而从区块链网络中查询出用户的当前余额信息, 对存储节点进行更新。与上述方法和步骤不同的是, 终端设备在对存储节点进行更新后不会即刻将用户的当前余额信息返回给用户, 而是在获取到用户下一次的余额查询请求时, 终端设备才可以将存储节点中更新后用户的当前余额信息返回给用户。具体过程可以参考图 7a, 图 7b, 图 7c

图 7a, 图 7b, 图 7c 为本说明书中提供的一种将存储节点中的余额信息返回给用户的流程示意图。

如图 7a 中所示, 终端设备可以通过检测用户对图 7a 中的查询余额控件进行的触控操作, 生成对应的余额查询请求, 并根据余额查询请求对存储节点中的余额信息进行查询。假设某一用户在存储节点中的余额信息为 100, 余额存储时间信息对应为前一天的上午 10:00, 而预设的存储时间阈值为 24 小时, 若该用户在第二天的中午 12:00 通过触控图 7a 中所示的查询余额控件, 使得终端设备生成了相应的余额查询请求, 进而对存储节点进行了查询操作。由上述内容可知, 存储节点中的余额信息的存储时间已经超过了存储时间阈值, 但终端设备在获取到该用户的余额查询请求后, 可以直接将存储节点中原有的余额信息向用户进行展示, 然后再对存储节点中的余额信息进行更新, 更新完成后终端设备可以在展示界面中显示提示用户余额信息已发生变动的提示框, 从而提示用户再次进行余额查询操作, 如图 7b 中所示。

紧接着，用户可以通过图 7b 中的更新余额控件对显示界面的余额进行更新，假设区块链网络中用户的当前余额信息为 200，当用户通过图 7b 中的更新余额控件再次提出余额查询请求后，终端设备可以将先前更新后的存储节点中的当前余额信息展示给用户，如图 7c 中所示。需要另行说明的是，此实例中用户在区块链网络中虚拟货币的余额信息发生了变更是为了体现整体流程的变化，实际情况还可能是，用户在区块链网络中虚拟货币的余额信息在一定时间段中并未发生改变，但终端设备根据余额存储时间确定存储节点中的余额信息是需要进行更新的，同样可以执行上述流程，从而确保用户接收到余额信息的准确性和实时性。

另外，在图 7a，图 7b，以及图 7c 中所显示的虚拟货币 A 和虚拟货币 B，用于表示终端设备的使用用户可能会涉及到在多种不同的区块链网络上分别拥有虚拟货币。在这种情况下，用户在进行余额信息的查询时，可以先确定和选择自己想要查询的虚拟货币类型，再进行查询，例如，图 7a，图 7b，以及图 7c 的虚拟货币 A 可以是如比特网中的比特币（Bitcoin），而虚拟货币 B 则可以是如以太坊（Ethereum）中的以太币（Ether，ETH），当用户想要查询自己在以太坊网络中虚拟货币以太币的余额信息时，可以通过触控相应的查询余额控件进行查询操作。

综上所述，还需说明的是，应用本说明书中提供的基于区块链的用户余额查询方法的终端设备所对应的应用场景具体可根据实际需求而定，例如，在用户进行余额信息确认的场景中，区块链网络的使用用户可能会不定时的对自己在区块链网络上虚拟货币的余额进行确认，以保证个人资产的实时安全以及对自身余额信息的时刻掌握。终端设备可以在用户发起余额查询请求后，从存储节点中查询出用户的余额信息并返回给用户，使得用户可以高效确认自己的余额信息是否正确，保障用户的财产安全。

再例如，在区块链网络进行跨境交易的场景中，在跨境交易的过程中交易双方用户可能会对各自在区块链网络的余额信息进行确认，以确定余额足够，从而保障跨境交易可以顺利进行。为了便于理解，下面以一个在通过区块链网络进行跨境交易的场景中应用本说明书中提供的方法的应用实例，来进行说明，具体可以参考图 8a。

图 8a 为本说明书中提供的一种利用区块链网络进行跨境交易的实例示意图。

如图 8a 中所示，在区块链网络的跨境交易中，用户可以在 A 国通过所使用的终端设备向所属 B 国的购物平台提出购物申请，而购物平台则会根据购物申请返回给用户进行订单支付方式选取的界面，使得用户可以自行选取和确认支付方式，支付方式选取界面中可以包含有多种支付方式，具体可以参考图 8b 所示。

图 8b 为本说明书中提供的一种跨境交易中支付方式选取界面的示意图。

如图 8b 中所示，购物平台可以向用户返回如图 8b 中的支付方式选取界面，其中可以包含有虚拟货币的支付方式，即利用用户在区块链网络中的虚拟货币进行订单支付。其中，用户在确定支付前可以选择进行余额查询，确保自己账户上的余额满足订金金额，具体可通过触控图 8b 中的余额查询控件进行查询。

当用户选择虚拟货币支付方式进行支付时，可以点击对应的余额查询控件，终端设备即可以获取到用户的余额查询请求，并对存储节点中存储的余额信息进行查询，如图 8a 中所示。若存储节点中存储有用户在区块链网络中虚拟货币的余额信息，终端设备会判断此时的存储节点是否需要进行更新，若确定不需要更新，终端设备便可以将存储节点中的余额信息返回给用户。而若是确定需要更新，终端设备便会生成相应的存储更新任务，以从区块链网络中查询到用户的当前余额信息，并将当前余额信息返回给用户，以供其查看，使得用户可以及时确定自己在区块链网络中虚拟货币的具体余额，以便根据具体的余额信息对跨境交易行为（如图 8a 和图 8b 中的跨境购物支付等）进行决策，有效提高了交易效率，并且一定程度上避免了因为网络拥堵导致交易出现问题的情况的发生。

从上述内容可以看出，本说明书提供的基于区块链的用户余额查询方法可以将用户在区块链网络中的虚拟货币的余额信息存储在终端设备的存储节点中。当用户需要查询自身账户余额时，无需直接向区块链网络发起查询请求，而可以从存储节点快速获取余额信息。这种方法能够显著提升用户查询余额信息的查询速度，既优化了用户体验，同时也有助于减少对区块链网络整体的网络压力，使得在区块链交易场景中提高了交易处理效能。

The specification also provides a computer-readable non-transitory storage medium, wherein the storage medium stores a computer program, which, when executed by a processor, can be used to perform one or more steps of one or more methods described or illustrated herein or provides functionality described or illustrated herein. Herein, the computer-readable non-transitory storage medium or media may include one or more semiconductor-based or other integrated circuits (ICs) (such, as for example, field-programmable gate arrays (FPGAs) or application-specific ICs (ASICs)), hard disk drives (HDDs), hybrid hard drives (HHDs), optical discs, optical disc drives (ODDs), magneto-optical discs, magneto-optical drives, floppy diskettes, floppy disk drives (FDDs), magnetic tapes, solid-state drives (SSDs), RAM-drives, SECURE DIGITAL cards or drives, any other suitable computer-readable non-transitory storage media, or any suitable combination of two or more of these, where appropriate. A computer-readable non-transitory storage medium may be volatile, non-volatile, or a combination of volatile and non-volatile, where appropriate.

The specification also provides a computer system. FIG. 9 illustrates an example computer system **900**. In particular embodiments, one or more computer systems **900** perform one or more steps of one or more methods described or illustrated herein. In particular embodiments, one or more computer systems **900** provide functionality described or illustrated herein. In particular embodiments, software running on one or more computer systems **900** performs one or more steps of one or more methods described or illustrated herein or provides functionality described or illustrated herein. Particular embodiments include one or more portions of one or more computer systems **900**. Herein, reference to a computer system may encompass a computing device, and vice versa, where appropriate. Moreover, reference to a computer system may encompass one or more computer systems, where appropriate.

This disclosure contemplates any suitable number of computer systems **900**. This disclosure contemplates computer system **900** taking any suitable physical form. As example and not by way of limitation, computer system **900** may be an embedded computer system, a system-on-chip (SOC), a single-board computer system (SBC) (such as, for example, a computer-on-module (COM) or system-on-module (SOM)), a desktop computer system, a laptop or notebook computer system, an interactive kiosk, a mainframe, a mesh of computer systems, a mobile telephone, a personal digital assistant (PDA), a server, a tablet computer system, or a combination of two or more of these. Where appropriate, computer system **900** may include one or more computer systems **900**; be unitary or distributed; span multiple locations; span multiple machines; span multiple data centers; or reside in a cloud, which may include one or more cloud components in one or more networks. Where appropriate, one or more computer systems **900** may perform without substantial spatial or temporal limitation one or more steps of one or more methods described or illustrated herein. As an example and not by way of limitation, one or more computer systems **900** may perform in real time or in batch mode one or more steps of one or more methods described or illustrated herein. One or more computer systems **900** may perform at different times or at different locations one or more steps of one or more methods described or illustrated herein, where appropriate.

In particular embodiments, computer system **900** includes a processor **902**, memory **904**, storage **906**, an input/output (I/O) interface **908**, a communication interface **910**, and a bus **912**. Although this disclosure describes and illustrates a particular computer system having a particular number of particular components in a particular arrangement, this disclosure contemplates any suitable computer system having any suitable number of any suitable components in any suitable arrangement.

In particular embodiments, processor **902** includes hardware for executing instructions, such

as those making up a computer program. As an example and not by way of limitation, to execute instructions, processor **902** may retrieve (or fetch) the instructions from an internal register, an internal cache, memory **904**, or storage **906**; decode and execute them; and then write one or more results to an internal register, an internal cache, memory **904**, or storage **906**. In particular embodiments, processor **902** may include one or more internal caches for data, instructions, or addresses. This disclosure contemplates processor **902** including any suitable number of any suitable internal caches, where appropriate. As an example and not by way of limitation, processor **902** may include one or more instruction caches, one or more data caches, and one or more translation lookaside buffers (TLB s). Instructions in the instruction caches may be copies of instructions in memory **904** or storage **906**, and the instruction caches may speed up retrieval of those instructions by processor **902**. Data in the data caches may be copies of data of **80** in memory **904** or storage **906** for instructions executing at processor **902** to operate on; the results of previous instructions executed at processor **902** for access by subsequent instructions executing at processor **902** or for writing to memory **904** or storage **906**; or other suitable data. The data caches may speed up read or write operations by processor **902**. The TLBs may speed up virtual-address translation for processor **902**. In particular embodiments, processor **902** may include one or more internal registers for data, instructions, or addresses. This disclosure contemplates processor **902** including any suitable number of any suitable internal registers, where appropriate. Where appropriate, processor **902** may include one or more arithmetic logic units (ALUs); be a multi-core processor; or include one or more processors **902**. Although this disclosure describes and illustrates a particular processor, this disclosure contemplates any suitable processor.

In particular embodiments, memory **904** includes main memory for storing instructions for processor **902** to execute or data for processor **902** to operate on. As an example and not by way of limitation, computer system **900** may load instructions from storage **906** or another source (such as, for example, another computer system **900**) to memory **904**. Processor **902** may then load the instructions from memory **904** to an internal register or internal cache. To execute the instructions, processor **902** may retrieve the instructions from the internal register or internal cache and decode them. During or after execution of the instructions, processor **902** may write one or more results (which may be intermediate or final results) to the internal register or internal cache. Processor **902** may then write one or more of those results to memory **904**. In particular embodiments, processor **902** executes only instructions in one or more internal registers or internal caches or in memory **904** (as opposed to storage **906** or elsewhere) and operates only on data in one or more internal registers or internal caches or in memory **904** (as opposed to storage

906 or elsewhere). One or more memory buses (which may each include an address bus and a data bus) may couple processor **902** to memory **904**. Bus **912** may include one or more memory buses, as described below. In particular embodiments, one or more memory management units (MMUs) reside between processor **902** and memory **904** and facilitate accesses to memory **904** requested by processor **902**. In particular embodiments, memory **904** includes random access memory (RAM). This RAM may be volatile memory, where appropriate. Where appropriate, this RAM may be dynamic RAM (DRAM) or static RAM (SRAM). Moreover, where appropriate, this RAM may be single-ported or multi-ported RAM. This disclosure contemplates any suitable RAM. Memory **904** may include one or more memories **904**, where appropriate. Although this disclosure describes and illustrates particular memory, this disclosure contemplates any suitable memory.

In particular embodiments, storage **906** includes mass storage for data or instructions. As an example and not by way of limitation, storage **906** may include a hard disk drive (HDD), a floppy disk drive, flash memory, an optical disc, a magneto-optical disc, magnetic tape, or a Universal Serial Bus (USB) drive or a combination of two or more of these. Storage **906** may include removable or non-removable (or fixed) media, where appropriate. Storage **906** may be internal or external to computer system **900**, where appropriate. In particular embodiments, storage **906** is non-volatile, solid-state memory. In particular embodiments, storage **906** includes read-only memory (ROM). Where appropriate, this ROM may be mask-programmed ROM, programmable ROM (PROM), erasable PROM (EPROM), electrically erasable PROM (EEPROM), electrically alterable ROM (EAROM), or flash memory or a combination of two or more of these. This disclosure contemplates mass storage **906** taking any suitable physical form. Storage **906** may include one or more storage control units facilitating communication between processor **902** and storage **906**, where appropriate. Where appropriate, storage **906** may include one or more storages **906**. Although this disclosure describes and illustrates particular storage, this disclosure contemplates any suitable storage.

In particular embodiments, I/O interface **908** includes hardware, software, or both, providing one or more interfaces for communication between computer system **900** and one or more I/O devices. Computer system **900** may include one or more of these I/O devices, where appropriate. One or more of these I/O devices may enable communication between a person and computer system **900**. As an example and not by way of limitation, an I/O device may include a keyboard, keypad, microphone, monitor, mouse, printer, scanner, speaker, still camera, stylus, tablet, touch screen, trackball, video camera, another suitable I/O device or a combination of two or more of these. An I/O device may include one or more sensors. This disclosure contemplates any suitable I/O devices and any suitable I/O interfaces **908** for them. Where appropriate, I/O interface **908**

may include one or more device or software drivers enabling processor **902** to drive one or more of these I/O devices. I/O interface **908** may include one or more I/O interfaces **908**, where appropriate. Although this disclosure describes and illustrates a particular I/O interface, this disclosure contemplates any suitable I/O interface.

In particular embodiments, communication interface **910** includes hardware, software, or both providing one or more interfaces for communication (such as, for example, packet-based communication) between computer system **900** and one or more other computer systems **900** or one or more networks. As an example and not by way of limitation, communication interface **910** may include a network interface controller (NIC) or network adapter for communicating with an Ethernet or other wire-based network or a wireless NIC (WNIC) or wireless adapter for communicating with a wireless network, such as a WI-FI network. This disclosure contemplates any suitable network and any suitable communication interface **910** for it. As an example and not by way of limitation, computer system **900** may communicate with an ad hoc network, a personal area network (PAN), a local area network (LAN), a wide area network (WAN), a metropolitan area network (MAN), or one or more portions of the Internet or a combination of two or more of these. One or more portions of one or more of these networks may be wired or wireless. As an example, computer system **900** may communicate with a wireless PAN (WPAN) (such as, for example, a BLUETOOTH WPAN), a WI-FI network, a WI-MAX network, a cellular telephone network (such as, for example, a Global System for Mobile Communications (GSM) network), or other suitable wireless network or a combination of two or more of these. Computer system **900** may include any suitable communication interface **910** for any of these networks, where appropriate. Communication interface **910** may include one or more communication interfaces **910**, where appropriate. Although this disclosure describes and illustrates a particular communication interface, this disclosure contemplates any suitable communication interface.

In particular embodiments, bus **912** includes hardware, software, or both coupling components of computer system **900** to each other. As an example and not by way of limitation, bus **912** may include an Accelerated Graphics Port (AGP) or other graphics bus, an Enhanced Industry Standard Architecture (EISA) bus, a front-side bus (FSB), a HYPERTRANSPORT (HT) interconnect, an Industry Standard Architecture (ISA) bus, an INFINIBAND interconnect, a low-pin-count (LPC) bus, a memory bus, a Micro Channel Architecture (MCA) bus, a Peripheral Component Interconnect (PCI) bus, a PCI-Express (PCIe) bus, a serial advanced technology attachment (SATA) bus, a Video Electronics Standards Association local (VLB) bus, or another suitable bus or a combination of two or more of these. Bus **912** may include one or more buses **912**, where appropriate. Although this disclosure describes and illustrates a particular bus, this

disclosure contemplates any suitable bus or interconnect.

以上所述仅为本说明书的实施例而已，并不用于限制本说明书。对于本领域技术人员来说，本说明书可以有各种更改和变化。凡在本说明书的精神和原理之内所作的任何修改、等同替换、改进等，均应包含在本申请的权利要求范围之内。

权 利 要 求 书

1、一种基于区块链的用户余额查询方法，包括：

获取余额查询请求，所述余额查询请求用于查询用户在区块链网络中的虚拟货币的余额；

根据所述余额查询请求，从预设的存储节点中查询出所述用户在区块链网络中虚拟货币的余额信息和余额存储时间信息，所述余额存储时间信息用于表示所述余额信息存储在所述存储节点的时间；

根据所述余额存储时间信息，确定所述余额存储时间信息对应的时间与当前时间的时间间隔；

若确定所述时间间隔不大于预设的存储时间阈值，则将所述存储节点中存储的所述的余额信息返回给所述用户。

2、如权利要求 1 所述的方法，所述方法还包括：

若确定所述时间间隔大于所述存储时间阈值，则从所述区块链网络中查询出所述用户在所述区块链网络中虚拟货币的当前余额信息，并返回给所述用户。

3、如权利要求 2 所述的方法，从所述区块链网络中查询出所述用户在所述区块链网络中虚拟货币的当前余额信息，具体包括：

生成针对所述存储节点的存储更新任务；

根据所述存储更新任务的任务状态信息，确定所述存储更新任务是否需要被执行；

若确定所述存储更新任务需要被执行，则确定执行所述存储更新任务的任务进程；

通过所述任务进程，从所述区块链网络中查询出所述用户在所述区块链网络中虚拟货币的当前余额信息。

4、如权利要求 3 所述的方法，所述方法还包括：

根据所述当前余额信息，对所述存储节点存储所述用户在所述区块链网络中虚拟货币的余额信息进行更新，并记录所述当前余额信息存储在所述存储节点的余额存储时间信息。

5、如权利要求 4 所述的方法，所述方法还包括：

将所述任务进程进行释放。

6、如权利要求 3 所述的方法，根据所述存储更新任务的任务状态信息，确定所述存储更新任务是否需要被执行，具体包括：

确定所述存储更新任务的任务生成时间；

判断所述任务生成时间是否晚于所述余额存储时间信息对应的时间；

若是，则确定所述存储更新任务需要被执行；

若否，则确定所述存储更新任务不需要被执行。

7、如权利要求 1 所述的方法，所述方法还包括：

若确定所述存储节点未存储有所述用户在区块链网络中虚拟货币的余额信息和余额存储时间信息，则从所述区块链网络中查询出所述用户在所述区块链网络中虚拟货币的当前余额信息，并返回给所述用户，以及，将查询出的当前余额信息以及存储所述查询出的当前余额信息的余额存储时间信息存储在所述存储节点中。

8、一种计算机系统，所述计算机系统包括存储器、处理器及存储在所述存储器上并可在所述处理器上运行的计算机程序，所述处理器执行所述程序时实现如下步骤：

获取余额查询请求，所述余额查询请求用于查询用户在区块链网络中的虚拟货币的余额；

根据所述余额查询请求，从预设的存储节点中查询出所述用户在区块链网络中虚拟货币的余额信息和余额存储时间信息，所述余额存储时间信息用于表示所述余额信息存储在所述存储节点的时间；

根据所述余额存储时间信息，确定所述余额存储时间信息对应的时间与当前时间的时间间隔；

若确定所述时间间隔不大于预设的存储时间阈值，则将所述存储节点中存储的所述的余额信息返回给所述用户。

9、如权利要求 8 所述的计算机系统，所述计算机系统还用于：

若确定所述时间间隔大于所述存储时间阈值，则从所述区块链网络中查询出所述用户在所述区块链网络中虚拟货币的当前余额信息，并返回给所述用户。

10、如权利要求 9 所述的计算机系统，从所述区块链网络中查询出所述用户在所述区块链网络中虚拟货币的当前余额信息，具体包括：

生成针对所述存储节点的存储更新任务；

根据所述存储更新任务的任务状态信息，确定所述存储更新任务是否需要被执行；

若确定所述存储更新任务需要被执行，则确定执行所述存储更新任务的任务进程；

通过所述任务进程，从所述区块链网络中查询出所述用户在所述区块链网络中虚拟货币的当前余额信息。

11、如权利要求 10 所述的计算机系统，所述计算机系统还用于：

根据所述当前余额信息，对所述存储节点存储所述用户在所述区块链网络中虚拟货币

的余额信息进行更新，并记录所述当前余额信息存储在所述存储节点的余额存储时间信息。

12、如权利要求 11 所述的计算机系统，所述计算机系统还用于：

将所述任务进程进行释放。

13、如权利要求 10 所述的计算机系统，根据所述存储更新任务的任务状态信息，确定所述存储更新任务是否需要被执行，具体包括：

确定所述存储更新任务的任务生成时间；

判断所述任务生成时间是否晚于所述余额存储时间信息对应的时间；

若是，则确定所述存储更新任务需要被执行；

若否，则确定所述存储更新任务不需要被执行。

14、如权利要求 8 所述的计算机系统，所述计算机系统还用于：

若确定所述存储节点未存储有所述用户在区块链网络中虚拟货币的余额信息和余额存储时间信息，则从所述区块链网络中查询出所述用户在所述区块链网络中虚拟货币的当前余额信息，并返回给所述用户，以及，将查询出的当前余额信息以及存储所述查询出的当前余额信息的余额存储时间信息存储在所述存储节点中。

15、一种计算机可读存储介质，所述存储介质存储有计算机程序，所述计算机程序被处理器执行时实现上述权利要求 1~7 任一项所述的方法。