

区块链的智能合约执行方法及装置和电子设备

技术领域

[01] 本说明书实施例涉及区块链技术领域，尤其涉及一种区块链的智能合约执行方法及装置和电子设备。

5 背景技术

[02] 区块链技术，也被称之为分布式账本技术，是一种由若干台计算设备共同参与“记账”，共同维护一份完整的分布式数据库的新兴技术。由于区块链技术具有去中心化、公开透明、每台计算设备可以参与数据库记录、并且各计算设备之间可以快速的进行数据同步的特性，使得区块链技术在众多的领域中广泛的进行应用。

10 发明内容

[03] 本说明书实施例提供一种区块链的智能合约执行方法及装置和电子设备：

[04] 根据本说明书实施例的第一方面，提供一种区块链的智能合约执行方法，所述区块链对接有可信第三方，所述可信第三方中部署有执行智能合约的合约引擎，所述方法包括：

15 [05] 区块链中的节点设备接收客户端发起的用于调用目标智能合约的目标交易；

[06] 响应于所述目标交易，从所述区块链的分布式数据库中获取所述目标智能合约的合约源码；

[07] 将所述合约源码发送给所述可信第三方，以使所述可信第三方的合约引擎执行所述合约源码，并向所述节点设备返回执行结果。

20 [08] 可选的，所述目标交易包括所述目标智能合约名称；以及，所述目标智能合约中被调用的目标函数名称；

[09] 所述从所述区块链的分布式数据库中获取所述目标智能合约的合约源码，具体包括：

[10] 基于所述目标交易包含的目标智能合约名称，从所述区块链的分布式数据库中获取该目标智能合约名称对应的合约源码；

25 [11] 基于所述目标智能合约中被调用的目标函数名称，从所获取的合约源码获取该目标

函数名称对应的合约源码。

[12]可选的，所述交易包含有输入参数；

[13]所述将所述合约源码发送给所述可信第三方，以使所述可信第三方的合约引擎执行所述合约源码，包括：

- 5 [14]将所述合约源码以及所述输入参数发送给所述可信第三方，以使所述可信第三方向所述节点设备获取与所述输入参数对应的输入数据，将所述输入数据加载到所述合约源码，并执行所述合约源码。

[15]可选的，所述区块链对接有作为校验方的另一可信第三方；

[16]所述方法还包括：

- 10 [17]将所述执行结果发送给所述另一可信第三方，以由所述另一可信第三方的合约引擎，针对所述执行结果进行校验；

[18]如果所述执行结果校验通过，将所述执行结果发布至所述区块链中的所有节点设备进行共识处理，并在共识通过后，将所述执行结果存储至所述区块链。

- 15 [19]可选的，所述执行结果为所述可信第三方基于所述合约源码的执行结果生成的结果集；其中，所述结果集包括执行合约源码过程中产生的读写集和执行合约源码后得到的执行结果；

[20]针对所述执行结果的校验过程，包括：

[21]所述另一可信第三方的合约引擎从所述读写集中读取输入数据，将读取到的输入数据加载到所述合约源码，并模拟执行所述合约源码；

- 20 [22]确定模拟执行所述合约源码产生的执行结果，与所述结果集中的执行结果是否一致；

[23]如果是，确定所述执行结果校验通过。

[24]可选的，所述执行结果为所述可信第三方基于所述合约源码的执行结果生成的结果集；其中，所述结果集包括执行合约源码过程中产生的读写集和执行合约源码后得到的执行结果；

- 25 [25]针对所述执行结果的校验过程，包括：

[26]所述另一可信第三方的合约引擎从所述读写集中读取输入数据，将读取到的输入数据加载到所述合约源码，并模拟执行所述合约源码；

- [27]确定模拟执行所述合约源码产生的读写集,与所述结果集中的读写集是否一致;
- [28]如果是,确定所述执行结果校验通过。
- [29]可选的,所述区块链为联盟链。
- [30]根据本说明书实施例的第二方面,提供一种区块链的智能合约执行方法,所述区块链对接有可信第三方,所述可信第三方中部署有执行智能合约的合约引擎,所述方法包括:
- 5 [31]所述可信第三方的合约引擎接收所述区块链中的节点设备发送的合约源码;其中,所述节点设备为所述区块链中响应客户端发起的用于调用目标智能合约的目标交易请求的节点,所述合约源码为所述目标智能合约的合约源码;
- 10 [32]执行所述合约源码并向所述节点设备,并向所述节点设备返回执行结果。
- [33]可选的,所述交易包含有输入参数;
- [34]所述接收所述区块链中的节点设备发送的合约源码,具体包括:
- [35]接收所述区块链中的节点设备发送的合约源码以及所述输入参数;
- [36]所述执行所述合约源码,具体包括:
- 15 [37]向所述节点设备获取与所述输入参数对应的输入数据,将所述输入数据加载到所述合约源码,并执行所述合约源码。
- [38]可选的,所述方法还包括:
- [39]在接收到所述节点设备发送的针对其它可信第三方的合约引擎的结果集进行的校验请求时,对所述结果集进行校验;其中,所述结果集包括执行合约源码过程中产生的读写集和执行合约源码后得到的执行结果;
- 20 [40]如果所述结果集校验通过,将所述结果集中的执行结果发布至所述区块链中的所有节点设备进行共识处理,并在共识通过后,将所述结果集中的执行结果存储至所述区块链。
- [41]可选的,针对所述第二执行结果的校验过程,包括:
- 25 [42]从所述读写集中读取输入数据,将读取到的输入数据加载到所述合约源码,并模拟执行所述合约源码;
- [43]确定模拟执行所述合约源码产生的执行结果,与所述结果集中的执行结果是否一致;

- [44]如果是，确定所述执行结果校验通过。
- [45]可选的，针对所述第二执行结果的校验过程，包括：
- [46]从所述读写集中读取输入数据，将读取到的输入数据加载到所述合约源码，并模拟执行所述合约源码；
- 5 [47]确定模拟执行所述合约源码产生的读写集，与所述结果集中的读写集是否一致；
- [48]如果是，确定所述执行结果校验通过。
- [49]可选的，所述区块链为联盟链。
- [50]根据本说明书实施例的第三方面，提供一种区块链的智能合约执行装置，所述区块链对接有可信第三方，所述可信第三方中部署有执行智能合约的合约引擎，所述装置包
- 10 括：
- [51]接收单元，区块链中的节点设备接收客户端发起的用于调用目标智能合约的目标交易；
- [52]响应单元，响应于所述目标交易，从所述区块链的分布式数据库中获取所述目标智能合约的合约源码；
- 15 [53]执行单元，将所述合约源码发送给所述可信第三方，以使所述可信第三方的合约引擎执行所述合约源码，并向所述节点设备返回执行结果。
- [54]根据本说明书实施例的第四方面，提供一种区块链的智能合约执行装置，所述区块链对接有可信第三方，所述可信第三方中部署有执行智能合约的合约引擎，所述方法包括：
- 20 [55]接收单元，所述可信第三方的合约引擎接收所述区块链中的节点设备发送的合约源码；其中，所述节点设备为所述区块链中响应客户端发起的用于调用目标智能合约的目标交易请求的节点，所述合约源码为所述目标智能合约的合约源码；
- [56]执行单元，执行所述合约源码并向所述节点设备，并向所述节点设备返回执行结果。
- [57]根据本说明书实施例的第五方面，提供一种电子设备，包括：
- 25 [58]处理器；
- [59]用于存储处理器可执行指令的存储器；
- [60]其中，所述处理器被配置为上述任一项区块链的智能合约执行方法。

[61]本说明书实施例，提供了一种区块链的智能合约执行方案，将智能合约与区块链节点解耦合，区块链中节点设备不再存储智能合约也不负责智能合约的运行，而是由区块链对接的可信第三方中部署的合约引擎负责运行智能合约；如此，当区块链中响应客户端发起交易的节点设备需要执行相关业务逻辑时，可以通过调用可信第三方的合约引擎，

5 由合约引擎运行相关的智能合约从而得到执行结果。一方面，通过将智能合约与节点设备解绑，从而将智能合约分离出来由合约引擎运行，简化了节点设备功能从而降低了对节点设备性能的要求（比如，性能不高的设备也可以作为节点设备接入区块链）。另一方面，区块链中节点设备间进行共识时，不再依赖智能合约进行共识，而是将结果集同世界状态比较就可以实现共识，无需每个节点设备参与智能合约运算，大大提升了共识

10 效率。再一方面，区块链对接的可信第三方作为专用于运行智能合约的节点，可以预先设置强大的存储和计算能力，以此部署大量的智能合约，并且智能合约的业务逻辑可以设计的较为复杂，以应对计算量较大、业务复杂度较高的业务需求。

附图说明

[62]图 1 是本说明书一实施例提供的区块链的智能合约执行方法的流程图；

15 [63]图 2 是本说明书一实施例提供的区块链系统的架构示意图；

[64]图 3 是本说明书一实施例提供的区块链的智能合约执行方法的流程图；

[65]图 4 是本说明书一实施例提供的区块链的智能合约执行装置的硬件结构图；

[66]图 5 是本说明书一实施例提供的区块链的智能合约执行装置的模块示意图；

[67]图 6 是本说明书一实施例提供的区块链的智能合约执行装置的模块示意图。

20 具体实施方式

[68]这里将详细地对示例性实施例进行说明，其示例表示在附图中。下面的描述涉及附图时，除非另有表示，不同附图中的相同数字表示相同或相似的要素。以下示例性实施例中所描述的实施方式并不代表与本说明书相一致的所有实施方式。相反，它们仅是与如所附权利要求书中所详述的、本说明书的一些方面相一致的装置和方法的例子。

25 [69]在本说明书使用的术语是仅仅出于描述特定实施例的目的，而非旨在限制本说明书。在本说明书和所附权利要求书中所使用的单数形式的“一种”、“所述”和“该”也旨在包括多数形式，除非上下文清楚地表示其他含义。还应当理解，本文中使用的术语“和

/或”是指并包含一个或多个相关联的列出项目的任何或所有可能组合。

[70]应当理解,尽管在本说明书可能采用术语第一、第二、第三等来描述各种信息,但这些信息不应限于这些术语。这些术语仅用来将同一类型的信息彼此区分开。例如,在不脱离本说明书范围的情况下,第一信息也可以被称为第二信息,类似地,第二信息也可以被称为第一信息。取决于语境,如在此所使用的词语“如果”可以被解释成为“在...时”或“当...时”或“响应于确定”。

[71]智能合约(Smart contract)是一种旨在应用在可以部署在区块链上的以信息化方式传播、验证或执行合同的计算机协议。通过在智能合约中声明业务逻辑可以实现执行相应操作。智能合约允许在没有第三方的情况下进行可信交易。这些交易可追踪且不可逆转。

10 智能合约能够提供优于传统合同方法的安全,并减少与合同相关的其他交易成本。

[72]首先,介绍传统区块链中智能合约的部署方式。传统智能合约是直接部署在区块链中节点设备上,并在节点设备执行交易请求时,直接运行本地部署的相应智能合约。对于需要行共识的情况,同样将执行结果抛出由区块链中所有节点设备进行共识,每个节点设备都需要运行本地部署的用于共识的智能合约。可见,一方面,传统智能合约与节点设备之间是强耦合的,每个节点设备都需要部署智能合约并需要运行智能合约,对于节点设备来说无疑增加了许多负担(节点设备需要具备一定的计算能力);另一方面,在节点设备间共识时,每个节点设备均需要重新执行一次智能合约判断执行结果是否正确,导致共识效率不高;再一方面,由于节点设备存储、计算能力有限,相应就限制了智能合约的数量以及复杂程度,传统智能合约无法执行计算量、复杂度较大的业务逻辑。

20 [73]本说明书提出了一种区块链的智能合约执行方案,通过对传统智能合约进行优化,可以提升包括区块链中节点设备的性能和效率,节点设备间进行共识的效率,以及智能合约执行的效率等,从而提升区块链整体性能。

[74]在本说明实现方式中,将智能合约与区块链节点解耦合,区块链中节点设备不再存储智能合约也不负责智能合约的运行,而是由区块链对接的可信第三方中部署的合约引擎负责运行智能合约;如此,当区块链中响应客户端发起交易的节点设备需要执行相关业务逻辑时,可以通过调用可信第三方的合约引擎,由合约引擎运行相关的智能合约从而得到执行结果。一方面,通过将智能合约与节点设备解绑,从而将智能合约分离出来由合约引擎运行,简化了节点设备功能从而降低了对节点设备性能的要求(比如,性能不高的设备也可以作为节点设备接入区块链)。另一方面,区块链中节点设备间进行共识时,不再依赖智能合约进行共识,而是将结果集同世界状态比较就可以实现共识,无

25

30

需每个节点设备参与智能合约运算，大大提升了共识效率。再一方面，区块链对接的可信第三方作为专用于运行智能合约的节点，可以预先设置强大的存储和计算能力，以此部署大量的智能合约，并且智能合约的业务逻辑可以设计的较为复杂，以应对计算量大、业务复杂度较高的业务需求。

- 5 [75]在本说明书所描述的区块链，具体可以包括私有链、共有链以及联盟链等，在本说明书中不进行特别限定。

- [76]例如，在一个场景中，上述区块链具体可以是由总部机构、分支机构、代理机构、节点设备等作为联盟成员组成的一个联盟链；该联盟链的运营方可以依托于该联盟链，来部署相应的业务；而以上所描述的作为联盟成员的总部机构、分支机构、代理机构都可以作为上述业务的一个业务节点。各个业务节点可以将自身所产生或者接收到的与执行结果以交易的形式在联盟链中进行发布，并在该交易经过联盟链中的共识节点的共识处理之后，在联盟链中的分布式数据库进行存储，完成上述交易信息的“上链”存证。
- 10

[77]其中，需要说明的是，在本说明书中所描述的交易（Transaction），是指通过区块链的客户端创建，并需要最终发布至区块链的分布式数据库中的一笔数据。

- 15 [78]区块链中的交易，通常存在狭义的交易以及广义的交易之分。狭义的交易是指用户向区块链发布的一笔价值转移；例如，在传统的比特币区块链网络中，交易可以是用户在区块链中发起的一笔转账。而广义的交易是指用户向区块链发布的一笔具有业务意图的业务数据；例如，运营方可以基于实际的业务需求搭建一个联盟链，依托于联盟链部署一些与价值转移无关的其它类型的在线业务（比如，宽泛的可以分为查询业务、调用业务等），而在这类联盟链中，交易可以是用户在联盟链中发布的一笔具有业务意图的业务消息或者业务请求。
- 20

[79]上述客户端，可以包括任意类型的以区块链中存储的底层业务数据作为数据支撑，来实现特定的业务功能的上层应用。

- [80]请参考图1，图1为本说明书一实施例提供的区块链的智能合约执行方法的流程图，所述区块链对接有可信第三方，所述可信第三方中部署有执行智能合约的合约引擎，所述方法可以包括以下步骤：
- 25

[81]步骤 110：区块链中的节点设备接收客户端发起的用于调用目标智能合约的目标交易；

[82]步骤 120：响应于所述目标交易，从所述区块链的分布式数据库中获取所述目标智

能合约的合约源码;

[83] 步骤 130: 将所述合约源码发送给所述可信第三方, 以使所述可信第三方的合约引擎执行所述合约源码, 并向所述节点设备返回执行结果。

[84] 以下结合图 2 所示的区块链系统的示意图为例加以说明:

- 5 [85] 客户端向区块链发起一笔目标交易, 经由 SDK 调用智能合约函数, SDK 将该目标交易发送到区块链的一个节点设备上, 该节点设备可以称为调用节点; 所述目标交易可以携带有相关的执行参数, 例如可以包括合约名称、调用函数名、输入参数、交易 ID、时间戳等。

- 10 [86] 如前所述, 本说明书中节点设备不负责智能合约运行; 因此, 调用节点在接收到客户端发起的目标交易后, 需要调用可信第三方的合约引擎来执行相应智能合约的业务逻辑。

[87] 其中, 该调用节点首先需要获取供合约引擎执行相应智能合约的合约源码:

[88] 在一实施例中, 所述目标交易包括所述目标智能合约名称; 以及, 所述目标智能合约中被调用的目标函数名称;

- 15 [89] 所述从所述区块链的分布式数据库中获取所述目标智能合约的合约源码, 具体包括:

[90] 基于所述目标交易包含的目标智能合约名称, 从所述区块链的分布式数据库中获取该目标智能合约名称对应的合约源码;

[91] 基于所述目标智能合约中被调用的目标函数名称, 从所获取的合约源码获取该目标函数名称对应的合约源码。

- 20 [92] 也就是说, 基于所述目标智能合约名称和目标调用函数名称, 节点设备就可以从所述区块链的分布式数据库中获取所述目标智能合约所需的合约源码。

[93] 其中, 所述合约源码是由一个个函数构成的, 每个函数具有一定的逻辑功能, 可以在合约引擎中执行相应逻辑; 通过目标调用函数名称可以确定相应的函数, 以保证确定合约源码包含的函数与目标调用函数一致。

- 25 [94] 该实施例中, 对于可信第三方的合约引擎来说, 为了避免合约引擎作恶, 智能合约的合约源码是存放在区块链的分布式数据库中的; 当需要使用智能合约时, 调用节点首先从分布式数据库中获取相应的合约源码, 然后再将合约源码发送给可信第三方的合约引擎; 如此, 每次合约引擎执行的智能合约都是临时由调用节点提供的。

[95]在一实施例中，所述目标交易还包含有输入参数；

[96]所述将所述合约源码发送给所述可信第三方，以使所述可信第三方的合约引擎执行所述合约源码，包括：

[97]将所述合约源码以及所述输入参数发送给所述可信第三方，以使所述可信第三方向
5 所述节点设备获取与所述输入参数对应的输入数据，将所述输入数据加载到所述合约源码，并执行所述合约源码。

[98]在实际应用中，有的业务类型可以不需要传输数据，则节点设备无需传输输入参数（或者，输入参数可以为空），合约引擎也无需向节点设备获取输入数据。

[99]对于有些复杂业务，需要用到输入数据，则需要用户在发起交易时提交输入参数。

10 [100] 如图 2 所示，节点设备在调用合约引擎时，需要提供了合约源码和输入参数。此时，合约引擎可以根据该输入参数从节点设备处获取输入数据。其中，所述输入参数可以是由客户端（即用户）提供的，所述输入数据可以是执行智能合约所需的其它数据。

[101] 以下举例说明输入参数和输入数据的区别：

[102] 假设客户端发送的交易请求是，令 $a+1$ 。则字段 a 和加 1 即为输入参数，而 a
15 当前的值即为输入数据。在实际运算 $a+1$ 时，首先需要获取 a 的原始值（输入数据），然后在原始值基础上加 1 才能得到执行结果。

[103] 由于区块链具有不可篡改的特性，以及节点设备存储有全量账本（即数据），因此合约引擎可以向节点设备获取该节点设备本地存储的输入参数相关的输入数据。

[104] 合约引擎可以注入合约源码，在获取到输入数据后，将所述输入数据加载到所
20 述合约源码，并执行所述合约源码得到执行结果。

[105] 需要说明的是，合约引擎在执行合约源码过程中还会产生读写集。所述读写集可以包括执行合约源码时需要的输入数据，和输入数据对应的输出数据。对于复杂的合约源码，输入数据和输出数据可以包含多组，即读写集是合约源码执行过程中所需的的所有输入数据和对应的输出数据，构成的数据集合，可以真实反映合约源码的整个执行计
25 算过程。请参考后续表 1 或表 2 所示的示例。

[106] 在一实施例中，所述执行结果为所述可信第三方基于所述合约源码的执行结果生成的结果集；其中，所述结果集包括执行合约源码过程中产生的读写集和执行合约源码后得到的执行结果。

[107] 其中，合约引擎可以将输入参数、合约源码（或者合约源码 hash）、读写集和执行结果打包为结果集，并为所述结果集添加本合约引擎的签名；将所述签名后的结果集作为执行结果返回给所述节点设备。

5 [108] 相应地，节点设备接收所述合约引擎返回的结果集；其中，所述结果集为所述合约引擎将所述输入参数、合约源码（或者合约源码 hash）、读写集和执行结果打包后添加本合约引擎的签名的集合。

[109] 该实施例中，合约引擎需要为结果集添加签名，以保证智能合约执行的有效性。对于没有签名的结果集，节点设备是不予确认的，视为无效。

10 [110] 值得一提的是，合约源码 hash 是指基于合约源码得到的 hash 值；所述合约源码 hash 可以用于校验合约引擎调用的合约源码是否存在篡改。举例说明，假设节点设备提供的合约源码为 A，正确的 hash 为 (A) hash；那么节点设备或者其他节点只需要将目标智能合约的合约源码的 hash 值与结果集中合约源码 hash 值比对，就可以判断合约引擎是否篡改了合约源码。假设合约引擎擅自篡改了合约源码，假设篡改后的合约源码为 A+1；那么返回的合约源码 hash 就变为 (A+1) hash，显然与原始合约源码的 (A) hash
15 不一致。

[111] 在一实施例中，所述区块链对接有作为校验方的另一可信第三方；

[112] 所述方法还包括：

[113] 将所述执行结果发送给所述另一可信第三方，以由所述另一可信第三方的合约引擎，针对所述执行结果进行校验；

20 [114] 如果所述执行结果校验通过，将所述执行结果发布至所述区块链中的所有节点设备进行共识处理，并在共识通过后，将所述执行结果存储至所述区块链。

[115] 为了便于区分，以下将另一可信第三方的合约引擎简称为第二合约引擎，并将前述运行智能合约的合约引擎简称为第一合约引擎。

25 [116] 如图 2 所示，节点设备可以通过第二合约引擎对第一合约引擎的执行结果进行检验。

[117] 在一实施例中，针对所述执行结果的校验过程，包括：

[118] 所述第二合约引擎从所述读写集中读取输入数据，将读取到的输入数据加载到所述合约源码，并模拟执行所述合约源码；

- [119] 确定模拟执行所述合约源码产生的执行结果，与所述结果集中的执行结果是否一致；
- [120] 如果是，确定所述执行结果校验通过。
- [121] 还可以为第一合约引擎的结果集添加该第二合约引擎的签名。
- 5 [122] 在一实施例中，针对所述执行结果的校验过程，包括：
- [123] 针对所述执行结果的校验过程，包括：
- [124] 所述另一可信第三方的合约引擎从所述读写集中读取输入数据，将读取到的输入数据加载到所述合约源码，并模拟执行所述合约源码；
- [125] 确定模拟执行所述合约源码产生的读写集，与所述结果集中的读写集是否一致；
- 10 [126] 如果是，确定所述执行结果校验通过。
- [127] 还可以为第一合约引擎的结果集添加该第二合约引擎的签名。
- [128] 举例说明，针对 $a+2$ ， $a+2$ ；相应地读写集如下表 1 所示：

读	写
$a=0$	$a=2$
$a=2$	$a=4$

- [129] 上述表 1 的含义，首先计算 $a+2$ ；产生第一次的读数据 $a=0$ ，运行 $a+1$ 后，产生第一次的写数据 $a=2$ ；然后计算 $a+2$ ；产生第二次的读数据 $a=2$ ，运行 $a+2$ ，产生第二次的写数据 $a=4$ 。
- 15 [130] 如果第一合约引擎作恶篡改了计算过程，将 $a+2$ ， $a+2$ ；篡改为 $a+1$ ， $a+3$ ；那么读写集如下表 2 所示：

读	写
$a=0$	$a=1$
$a=1$	$a=4$

- 20 [131] 假设表 2 为第一合约引擎返回的结果集中的读写集，表 1 为第二合约引擎模拟合约源码运行后得到的读写集；对比表 1 和表 2 就可以确定表 1 存在问题，说明第一合约引擎存在篡改。

- [132] 而如果仅通过比对执行结果,则表1和表2的执行结果均为4,容易造成误判。
- [133] 通过第二合约引擎判断第一合约引擎返回的读写集是否真实有效,如前所述读写集包括一系列的输入和输出,从读写集中读取输入数据模拟执行得到输出数据后,逐个确认得到的输出数据是否跟读写集一致(即整个过程各个环节的运算结果都符合预期);如果全部一致,证明读写集有效,没有被人篡改,此时才会进一步将最终的执行结果,与结果集中的结果进行匹配。如此校验过程可以避免第一合约引擎作弊。
- [134] 以下介绍本说明书中节点设备间共识的机制:
- [135] 如前所述如果所述执行结果校验通过,节点设备需要将所述执行结果发布至所述区块链中的所有节点设备进行共识处理,并在共识通过后,将所述执行结果存储至所述区块链。
- [136] 这里的共识过程,可以为:
- [137] 首先将执行结果在本地的状态数据库保存(更新自己的世界状态)。在对智能合约的执行结果进行共识时,可以将收到的其它节点设备发送的执行结果与本地的状态数据库中的执行结果进行比较,来对执行结果进行共识。
- [138] 如果所有节点设备返回的执行结果,或者大多数节点设备返回的执行结果,与自己的世界状态中的执行结果一致,则共识通过。
- [139] 在一可选的实施例中,除了保存执行结果还可以保存读写集:
- [140] 首先将返回的读写集在本地的状态数据库保存(更新自己的世界状态)。在对智能合约的执行结果进行共识时,可以将收到的其它节点设备发送的读写集与本地的状态数据库中的读写集进行比较,来对执行结果进行共识。
- [141] 如果所有节点设备返回的读写集,或者大多数节点设备返回的读写集,与自己的世界状态中的读写集一致,则共识通过。
- [142] 该实施例,在区块链中节点设备间进行共识时,不再需要重新执行智能合约,对智能合约的执行结果进行共识校验,而是直接将结果集同世界状态进行比较就可以完成对智能合约的执行结果的共识;因此可以大大提升共识效率。
- [143] 请参考图3,图3为本说明书一实施例提供的区块链的智能合约执行方法的流程图,所述区块链对接有可信第三方,所述可信第三方中部署有执行智能合约的合约引擎,所述方法可以应用于合约引擎,所述方法可以包括以下步骤:

[144] 步骤 210: 所述可信第三方的合约引擎接收所述区块链中的节点设备发送的合约源码; 其中, 所述节点设备为所述区块链中响应客户端发起的用于调用目标智能合约的目标交易请求的节点, 所述合约源码为所述目标智能合约的合约源码;

5 [145] 步骤 220: 执行所述合约源码并向所述节点设备, 并向所述节点设备返回执行结果。

[146] 该实施例是以合约引擎为执行主体进行描述的, 相关的节点设备可以参考前述图 1 所示实施例, 在该实施例不再进行赘述。

[147] 在一实施例中, 所述交易包含有输入参数;

[148] 所述接收所述区块链中的节点设备发送的合约源码, 具体包括:

10 [149] 接收所述区块链中的节点设备发送的合约源码以及所述输入参数;

[150] 所述执行所述合约源码, 具体包括:

[151] 向所述节点设备获取与所述输入参数对应的输入数据, 将所述输入数据加载到所述合约源码, 并执行所述合约源码。

[152] 在一实施例中, 所述方法还包括:

15 [153] 在接收到所述节点设备发送的针对其它可信第三方的合约引擎的结果集进行的校验请求时, 对所述结果集进行校验; 其中, 所述结果集包括执行合约源码过程中产生的读写集和执行合约源码后得到的执行结果;

[154] 如果所述结果集校验通过, 将所述结果集中的执行结果发布至所述区块链中的所有节点设备进行共识处理, 并在共识通过后, 将所述结果集中的执行结果存储至所述
20 区块链。

[155] 在一实施例中, 针对所述第二执行结果的校验过程, 包括:

[156] 从所述读写集中读取输入数据, 将读取到的输入数据加载到所述合约源码, 并模拟执行所述合约源码;

[157] 确定模拟执行所述合约源码产生的执行结果, 与所述结果集中的执行结果是否
25 一致;

[158] 如果是, 确定所述执行结果校验通过。

[159] 在一实施例中, 针对所述第二执行结果的校验过程, 包括:

- [160] 从所述读写集中读取输入数据，将读取到的输入数据加载到所述合约源码，并模拟执行所述合约源码；
- [161] 确定模拟执行所述合约源码产生的读写集，与所述结果集中的读写集是否一致；
- [162] 如果是，确定所述执行结果校验通过。
- 5 [163] 综上所述实施例，提供了一种区块链的智能合约执行方案，将智能合约与区块链节点解耦合，区块链中节点设备不再存储智能合约也不负责智能合约的运行，而是由区块链对接的可信第三方中部署的合约引擎负责运行智能合约；如此，当区块链中响应客户端发起交易的节点设备需要执行相关业务逻辑时，可以通过调用可信第三方的合约引擎，由合约引擎运行相关的智能合约从而得到执行结果。一方面，通过将智能合约与
- 10 节点设备解绑，从而将智能合约分离出来由合约引擎运行，简化了节点设备功能从而降低了对节点设备性能的要求（比如，性能不高的设备也可以作为节点设备接入区块链）。另一方面，区块链中节点设备间进行共识时，不再依赖智能合约进行共识，而是将结果集同世界状态比较就可以实现共识，无需每个节点设备参与智能合约运算，大大提升了共识效率。再一方面，区块链对接的可信第三方作为专用于运行智能合约的节点，可以
- 15 预先设置强大的存储和计算能力，以此部署大量的智能合约，并且智能合约的业务逻辑可以设计的较为复杂，以应对计算量较大、业务复杂度较高的业务需求。
- [164] 与前述区块链的智能合约执行方法实施例相对应，本说明书还提供了区块链的智能合约执行装置的实施例。所述装置实施例可以通过软件实现，也可以通过硬件或者软硬件结合的方式实现。以软件实现为例，作为一个逻辑意义上的装置，是通过其所在
- 20 设备的处理器将非易失性存储器中对应的计算机业务程序指令读取到内存中运行形成的。从硬件层面而言，如图 4 所示，为本说明书区块链的智能合约执行装置所在设备的一种硬件结构图，除了图 4 所示的处理器、网络接口、内存以及非易失性存储器之外，实施例中装置所在的设备通常根据区块链的智能合约执行实际功能，还可以包括其他硬件，对此不再赘述。
- 25 [165] 请参见图 5，为本说明书一实施例提供的区块链的智能合约执行装置的模块图，所述装置对应了图 1 所示实施例，所述区块链对接有可信第三方，所述可信第三方中部署有执行智能合约的合约引擎，所述装置包括：
- [166] 接收单元 310，区块链中的节点设备接收客户端发起的用于调用目标智能合约的目标交易；

- [167] 响应单元 320, 响应于所述目标交易, 从所述区块链的分布式数据库中获取所述目标智能合约的合约源码;
- [168] 执行单元 330, 将所述合约源码发送给所述可信第三方, 以使所述可信第三方的合约引擎执行所述合约源码, 并向所述节点设备返回执行结果。
- 5 [169] 可选的, 所述目标交易包括所述目标智能合约名称; 以及, 所述目标智能合约中被调用的目标函数名称;
- [170] 所述响应单元 320, 具体包括:
- [171] 响应于所述目标交易, 基于所述目标交易包含的目标智能合约名称, 从所述区块链的分布式数据库中获取该目标智能合约名称对应的合约源码; 并, 基于所述目标智能合约中被调用的目标函数名称, 从所获取的合约源码获取该目标函数名称对应的合约源码。
- 10 [172] 可选的, 所述交易包含有输入参数;
- [173] 所述执行单元 330, 包括:
- [174] 将所述合约源码以及所述输入参数发送给所述可信第三方, 以使所述可信第三方向所述节点设备获取与所述输入参数对应的输入数据, 将所述输入数据加载到所述合约源码, 并执行所述合约源码; 以及向所述节点设备返回执行结果。
- 15 [175] 可选的, 所述区块链对接有作为校验方的另一可信第三方;
- [176] 所述装置还包括:
- [177] 校验单元, 将所述执行结果发送给所述另一可信第三方, 以由所述另一可信第三方的合约引擎, 针对所述执行结果进行校验; 如果所述执行结果校验通过, 将所述执行结果发布至所述区块链中的所有节点设备进行共识处理, 并在共识通过后, 将所述执行结果存储至所述区块链。
- 20 [178] 可选的, 所述执行结果为所述可信第三方基于所述合约源码的执行结果生成的结果集; 其中, 所述结果集包括执行合约源码过程中产生的读写集和执行合约源码后得到的执行结果;
- 25 [179] 针对所述执行结果的校验过程, 包括:
- [180] 所述另一可信第三方的合约引擎从所述读写集中读取输入数据, 将读取到的输入数据加载到所述合约源码, 并模拟执行所述合约源码;

- [181] 确定模拟执行所述合约源码产生的执行结果，与所述结果集中的执行结果是否一致；
- [182] 如果是，确定所述执行结果校验通过。
- [183] 可选的，所述执行结果为所述可信第三方基于所述合约源码的执行结果生成的
- 5 结果集；其中，所述结果集包括执行合约源码过程中产生的读写集和执行合约源码后得到的执行结果；
- [184] 针对所述执行结果的校验过程，包括：
- [185] 所述另一可信第三方的合约引擎从所述读写集中读取输入数据，将读取到的输入数据加载到所述合约源码，并模拟执行所述合约源码；
- 10 [186] 确定模拟执行所述合约源码产生的读写集，与所述结果集中的读写集是否一致；
- [187] 如果是，确定所述执行结果校验通过。
- [188] 可选的，所述区块链为联盟链。
- [189] 请参见图 6，为本说明书一实施例提供的区块链的智能合约执行装置的模块图，所述装置对应了图 3 所示实施例，所述区块链对接有可信第三方，所述可信第三方中部
- 15 署有执行智能合约的合约引擎，所述装置包括：
- [190] 接收单元 410，所述可信第三方的合约引擎接收所述区块链中的节点设备发送的合约源码；其中，所述节点设备为所述区块链中响应客户端发起的用于调用目标智能合约的目标交易请求的节点，所述合约源码为所述目标智能合约的合约源码；
- [191] 执行单元 420，执行所述合约源码并向所述节点设备，并向所述节点设备返回执
- 20 行结果。
- [192] 可选的，所述交易包含有输入参数；
- [193] 所述接收单元 410 中接收所述区块链中的节点设备发送的合约源码，具体包括：
- [194] 接收所述区块链中的节点设备发送的合约源码以及所述输入参数；
- [195] 所述执行单元 420 中执行所述合约源码，具体包括：
- 25 [196] 向所述节点设备获取与所述输入参数对应的输入数据，将所述输入数据加载到所述合约源码，并执行所述合约源码。
- [197] 可选的，所述装置还包括：

- [198] 校验单元，在接收到所述节点设备发送的针对其它可信第三方的合约引擎的结果集进行的校验请求时，对所述结果集进行校验；其中，所述结果集包括执行合约源码过程中产生的读写集和执行合约源码后得到的执行结果；
- [199] 共识单元，如果所述结果集校验通过，将所述结果集中的执行结果发布至所述
- 5 区块链中的所有节点设备进行共识处理，并在共识通过后，将所述结果集中的执行结果存储至所述区块链。
- [200] 可选的，针对所述第二执行结果的校验过程，包括：
- [201] 从所述读写集中读取输入数据，将读取到的输入数据加载到所述合约源码，并模拟执行所述合约源码；
- 10 [202] 确定模拟执行所述合约源码产生的执行结果，与所述结果集中的执行结果是否一致；
- [203] 如果是，确定所述执行结果校验通过。
- [204] 可选的，针对所述第二执行结果的校验过程，包括：
- [205] 从所述读写集中读取输入数据，将读取到的输入数据加载到所述合约源码，并
- 15 模拟执行所述合约源码；
- [206] 确定模拟执行所述合约源码产生的读写集，与所述结果集中的读写集是否一致；
- [207] 如果是，确定所述执行结果校验通过。
- [208] 可选的，所述区块链为联盟链。
- [209] 上述实施例阐明的系统、装置、模块或单元，具体可以由计算机芯片或实体实
- 20 现，或者由具有某种功能的产品来实现。一种典型的实现设备为计算机，计算机的具体形式可以是个人计算机、膝上型计算机、蜂窝电话、相机电话、智能电话、个人数字助理、媒体播放器、导航设备、电子邮件收发设备、游戏控制台、平板计算机、可穿戴设备或者这些设备中的任意几种设备的组合。
- [210] 上述装置中各个单元的功能和作用的实现过程具体详见上述方法中对应步骤的
- 25 实现过程，在此不再赘述。
- [211] 对于装置实施例而言，由于其基本对应于方法实施例，所以相关之处参见方法实施例的部分说明即可。以上所描述的装置实施例仅仅是示意性的，其中所述作为分离部件说明的单元可以是或者也可以不是物理上分开的，作为单元显示的部件可以是或者

也可以不是物理单元，即可以位于一个地方，或者也可以分布到多个网络单元上。可以根据实际的需要选择其中的部分或者全部模块来实现本说明书方案的目的。本领域普通技术人员在不付出创造性劳动的情况下，即可以理解并实施。

[212] 以上图 5 描述了区块链的智能合约执行装置的内部功能模块和结构示意图，其实
5 质上的执行主体可以为一种电子设备，包括：

[213] 处理器；

[214] 用于存储处理器可执行指令的存储器；

[215] 其中，所述处理器被配置为：

[216] 区块链中的节点设备接收客户端发起的用于调用目标智能合约的目标交易；其
10 中，所述区块链对接有可信第三方，所述可信第三方中部署有执行智能合约的合约引擎；

[217] 响应于所述目标交易，从所述区块链的分布式数据库中获取所述目标智能合约的合约源码；

[218] 将所述合约源码发送给所述可信第三方，以使所述可信第三方的合约引擎执行所述合约源码，并向所述节点设备返回执行结果。

15 [219] 以上图 6 描述了基于区块链的物流保价装置的内部功能模块和结构示意图，其实质上的执行主体可以为一种电子设备，包括：

[220] 处理器；

[221] 用于存储处理器可执行指令的存储器；

[222] 其中，所述处理器被配置为：

20 [223] 所述可信第三方的合约引擎接收所述区块链中的节点设备发送的合约源码；其中，所述节点设备为所述区块链中响应客户端发起的用于调用目标智能合约的目标交易请求的节点，所述合约源码为所述目标智能合约的合约源码；所述区块链对接有可信第三方，所述可信第三方中部署有执行智能合约的合约引擎；

[224] 执行所述合约源码并向所述节点设备，并向所述节点设备返回执行结果。

25 [225] 在上述电子设备的实施例，应理解，该处理器可以是中央处理单元（英文：Central Processing Unit，简称：CPU），还可以是其他通用处理器、数字信号处理器（英文：Digital Signal Processor，简称：DSP）、专用集成电路（英文：Application Specific

- Integrated Circuit, 简称: ASIC) 等。通用处理器可以是微处理器或者该处理器也可以是任何常规的处理器等, 而前述的存储器可以是只读存储器(英文: read-only memory, 缩写: ROM)、随机存取存储器(英文: random access memory, 简称: RAM)、快闪存储器、硬盘或者固态硬盘。结合本发明实施例所公开的方法的步骤可以直接体现为硬件
- 5 件处理器执行完成, 或者用处理器中的硬件及软件模块组合执行完成。

[226] 本说明书中的各个实施例均采用递进的方式描述, 各个实施例之间相同相似的部分互相参见即可, 每个实施例重点说明的都是与其他实施例的不同之处。尤其, 对于电子设备实施例而言, 由于其基本相似于方法实施例, 所以描述的比较简单, 相关之处参见方法实施例的部分说明即可。

- 10 [227] 本领域技术人员在考虑说明书及实践这里公开的发明后, 将容易想到本说明书的其它实施方案。本说明书旨在涵盖本说明书的任何变型、用途或者适应性变化, 这些变型、用途或者适应性变化遵循本说明书的一般性原理并包括本说明书未公开的本技术领域中的公知常识或惯用技术手段。说明书和实施例仅被视为示例性的, 本说明书的真正范围和精神由下面的权利要求指出。
- 15 [228] 应当理解的是, 本说明书并不局限于上面已经描述并在附图中示出的精确结构, 并且可以在不脱离其范围进行各种修改和改变。本说明书的范围仅由所附的权利要求来限制。