

跨区块链的数据处理方法、装置、客户端、区块链系统

技术领域

[01] 本说明书实施例方案属于区块链数据处理的技术领域，尤其涉及一种跨区块链的数据处理方法、装置、客户端、区块链系统。

5 背景技术

[02] 区块链技术，也被称之为分布式账本技术，是一种去中性化的分布式数据库技术，其特点是去中心化、公开透明、不可篡改、可信任。区块链的每笔数据，都会广播全网

10 的区块链节点，每个节点都有全量的、一致的数据。随着区块链技术的发展和

应用，当前除全网公开的公有链之外，还存在私有链区、联盟链等，区块链系统间的跨链交互已

经成为技术或业务解决方案的重要问题。现有公开的跨链技术中可以使用中继链等技术

方案实现区块链间跨链交互。

[03] 区块链系统与区块链系统之间进行数据交互时通常需要区块链具备认证另外一条区块链上数据的能力，以保障区块链系统之间数据传输的安全性，防止区块链数据被篡改等。目前存在多种跨链网络实现，如 polkadot（波卡链，一种实现跨链通信的技术），

15 其设计了 Validator（验证者）角色，该角色会校验数据是否来自于区块链（使用区块链的数据认证逻辑），并且 Validators 会用 POS（Proof of Stake，权益证明，一种共识算法）算法共识验证结果，通过共识的数据可以确认来自于一条区块链，另外一条区块链即可使用该跨链数据。现有的中继链认证跨链数据时，跨链数据需均经过中继链的 POS 共识，中继链的共识算法吞吐量是跨链数据流量的上限，存在吞吐量上限瓶颈。同时，

20 区块链认证中继链上的数据时，需要验证多重签名，以 POS 为例，需要验证上百个签名，执行效率较低。

[04] 因此，业内亟需一种可以更加高效、高扩展性的认证跨链数据的解决方案。

发明内容

[05] 本说明书实施例目的在于提供一种跨区块链的数据处理方法、装置、客户端、区块链系统，可以更加高效的实现区块链之间的数据认证，扩展性更好，效率高，可以适用

25 更多的区块链应用场景。

- [06]本说明书实施例提供一种跨区块链的数据处理方法、装置、客户端、区块链系统是包括以下方式实现的：
- [07]一种跨区块链的数据处理方法，所述方法包括：
- [08]第一区块链向信任证明方发送跨链数据请求；
- 5 [09]信任证明方基于所述跨链数据请求获取第二区块链的跨链数据；
- [10]所述信任证明方对所述跨链数据进行验证，在所述验证通过后，使用采用的可信任执行环境策略对应的私钥对所述跨链数据进行签名；
- [11]信任证明方将签名后的请求结果返回给第一区块链；
- [12]所述第一区块链利用可信任执行环境策略对应的公钥验证所述请求结果，在验证通
- 10 过时确定所述请求结果中的跨链数据来自信任证明方且已由所述信任证明方完成验证。
- [13]一种跨区块链的数据处理方法，所述方法包括：
- [14]接收第一区块链发送跨链数据请求；
- [15]基于所述跨链数据请求获取第二区块链的跨链数据；
- [16]对所述跨链数据进行验证，在所述验证通过后，使用采用的可信任执行环境策略对
- 15 应的私钥对所述跨链数据进行签名；
- [17]将签名后的请求结果返回给第一区块链，所述请求结果包括所述跨链数据和所述签名。
- [18]一种跨区块链的数据处理的处理装置，所述装置包括：
- [19]请求接收模块，用于接收第一区块链发送跨链数据请求；
- 20 [20]数据获取模块，用于基于所述跨链数据请求获取第二区块链的跨链数据；
- [21]数据验证模块，用于对所述跨链数据进行验证，在所述验证通过后，使用采用的可信任执行环境策略对应的私钥对所述跨链数据进行签名；
- [22]证明返回模块，用于将签名后的请求结果返回给第一区块链，所述请求结果包括所述跨链数据和所述签名。
- 25 [23]一种跨区块链的数据处理的客户端，包括处理器以及用于存储处理器可执行指令的存储器，所述处理器执行所述指令时实现：

- [24]接收第一区块链发送跨链数据请求;
- [25]基于所述跨链数据请求获取第二区块链的跨链数据;
- [26]对所述跨链数据进行验证,在所述验证通过后,使用采用的可信任执行环境策略对应的私钥对所述跨链数据进行签名;
- 5 [27]将签名后的请求结果返回给第一区块链,所述请求结果包括所述跨链数据和所述签名。
- [28]一种区块链系统,所述系统包括本说明书任一实施例所述的装置或客户端。
- [29]一种跨区块链系统,包括至少两个区块链以及返回跨链数据的目标区块链对应的中继器,请求数据的区块链通过信对应的中继器获取目标区块链的跨链数据,所述中继器
- 10 包括本说明书实施例中任意一项所述的装置,或者本说明书中任意一项所述的方法实现。
- [30]本说明书实施例提供的一种跨区块链的数据处理方法、装置、客户端、区块链系统,可以基于 TEE 实现区块链间的数据交互,快速高效的实现跨区块链的数据认证,实现跨区块链数据处理的高扩展性,适应于更多的跨链数据交互场景。利用本说明书实施例方案,可以简易高效的提高跨链数据吞吐流量的上限,有效解决类似传统 POS 共识中
- 15 继使用场景有限、效率性能更较低问题。

附图说明

- [31]为了更清楚地说明本说明书实施例或现有技术中的技术方案,下面将对实施例或现有技术描述中所需要使用的附图作简单地介绍,显而易见地,下面描述中的附图仅仅是本说明书中记载的一些实施例,对于本领域普通技术人员来讲,在不付出创造性劳动性的前提下,还可以根据这些附图获得其他的附图。
- 20 [32]图 1 是本说明书提供的所述一种跨区块链的数据处理的流程及系统结构示意图;
- [33]图 2 是本说明书提供的所述一种跨区块链的数据处理方法实施例的流程示意图;
- [34]图 3 是本说明书提供的另一种跨区块链的数据处理方法实施例的流程示意图;
- [35]图 4 是应用本说明书实施例的一种跨区块链的数据处理方法的客户端的硬件结构框图;
- 25 [36]图 5 是本说明书提供的可以作为跨区块链中继器的一种跨区块链的数据处理的处理装置实施例的模块结构示意图;

[37]图 6 是本说明书提供的所述装置的另一种实施例的模块结构示意图;

[38]图 7 是本说明书提供的另一种跨区块链的数据处理的流程及系统结构示意图。

具体实施方式

[39]为了使本技术领域的人员更好地理解本说明书中的技术方案,下面将结合本说明书
5 实施例中的附图,对本说明书实施例中的技术方案进行清楚、完整地描述,显然,所描述的实施例仅仅是本说明书中的一部分实施例,而不是全部的实施例。基于本说明书中的一个或多个实施例,本领域普通技术人员在没有作出创造性劳动前提下所获得的所有其他实施例,都应当属于本申请保护的范围。

[40]区块链系统中的数据一般支持对第三方可证明。一般的,应用系统与区块链系统交互时,可以选择信任域内的区块链节点取数据,因为应用系统可以信任节点上的数据,区块链节点作为区块链网络中的一员,可以认证网络上的数据。应用系统可以部署近端的客户端,客户端可以设计实现区块链协议,可以认证区块链系统中的数据,因此应用系统可以使用客户端与区块链交互。本说明书实施例中区块链系统与区块链系统之间的交互,同样可以使用上述应用系统与区块链系统交互时的客户端方式实现。

15 [41]在本说明书实施例提供的方案中,实现跨链数据认证的客户端中可以使用 TEE (Trusted Execution Environment,可信任执行环境)的核心逻辑实现,使得该客户端自身可以认证区块链链上的数据。所述的 TEE 通常指在硬件设备中提供一个安全区域,确保敏感数据在隔离、可信的环境中存储、处理和保护。TEE 可以表示一个与硬件设备中其他模块隔离并能够执行可信应用程序(代码)的区域。TEE 可以包括 CPU 和其他
20 元件,如存储器和 IO。TEE 是与设备上的 Rich OS(通常是 Android 等)并存的运行环境,并且给 Rich OS(富操作系统)提供安全服务,它具有其自身的执行空间,比 Rich OS 的安全级别更高。TEE 所能访问的软硬件资源是与 Rich OS 分离的。TEE 提供了授权安全软件(可信应用,如本说明书实施例中的客户端的区块链验证应用)的安全执行环境,同时也保护了资源和数据的保密性、完整性和访问权限。为了保证 TEE 本身的
25 可信根,TEE 在安全启动过程中是要通过验证并且与 Rich OS 隔离的。

[42]支持 TEE 的硬件技术方案可以包括 AMD 的 PSP(Platform Security Processor)、ARM TrustZone(信任空间)技术(支持 TrustZone 的所有 ARM 处理器)、Intel x86-64 指令集等。本说明书提供的一种实施例方案中,可以选择 intel SGX(Intel Software Guard Extension,软件保护扩展策略),实现 TEE。Intel SGX 是因特尔芯片处理器开发商开

发的处理器技术，SGX 提供了一个被称为 enclave 的可信执行环境，它可以防止其他应用程序，包括操作系统、BIOS 系统等窥探和篡改受保护应用程序的状态及可信环境内的数据。一旦软件和数据位于 enclave 中，即便操作系统或者也无法影响 enclave 里面的代码和数据，Enclave 的安全边界只包含 CPU 和它自身同时。SGX 还可以引入

5 Attestation（证词）鉴证机制，可以在芯片中封装远程验证机制逻辑，由 Intel 或其代理响应请求，对程序及其结果是否可信进行认证。

[43]具体的，图 1 是本说明书提供的所述一种跨区块链的数据处理的方法流程示意图，也可以作为系统的构架示意图。虽然本说明书提供了如下述实施例或附图所示的方法操作步骤或装置、系统结构，但基于常规或者无需创造性的劳动在所述方法或装置中可以

10 包括更多或者部分合并后更少的操作步骤或模块单元。在逻辑性上不存在必要因果关系的步骤或结构中，这些步骤的执行顺序或装置的模块结构不限于本说明书实施例或附图所示的执行顺序或模块结构。所述的方法或模块结构的在实际中的装置、服务器或终端产品应用时，可以按照实施例或者附图所示的方法或模块结构进行顺序执行或者并行执行（例如并行处理器或者多线程处理的环境、甚至包括分布式处理、服务器集群的实施

15 环境）。

[44]具体的一种实施例如图 1 所示，可以选用 intel SGX 作为 TEE 中的核心逻辑，实现区块链系统的客户端，使得 TEE 程序（或称为 TEE 应用）自身可以认证区块链上的数据。客户端可以对外发布以下信息：

[45]D01、TEE 程序代码测量值，用于对外证明其会诚实得认证区块链上的数据；

20 [46]D02、TEE 中一对非对称密钥对的公钥。该公钥对应的私钥在 TEE 内部初始化时生成，外界不可读，用于 TEE 签名跨链数据，以对外证明数据来自于该可信任的 TEE 的客户端；

[47]D03、TEE 程序初始化时的参数，如 TEE 里客户端所连接的区块链信息；

[48]D04、TEE 以上三部分信息的远程证明信息。验证方可请求对应的远程证明方验证

25 该证明，以验证以上三部分信息是否来自于 TEE 的客户端。

[49]在图 1 中，区块链 A 通过 TEE 应用请求区块链 B 的数据，该 TEE 程序中可以包括链接至区块链的 SPV（Simplified Payment Verification，简单支付验证）客户端。SPV 简单支付验证，在 SPV 节点不保存全部的区块链数据，只保存区块头的信息，使得 SPV 客户端对整个存储的容量要求更小，因此 SPV 客户端一般也称为轻量级客户端或轻客

5 户端。需要跨链读取数据的区块链 A 验证并锚定以上 D01-D04 数据, 通过 TEE 远端证明信息验证以上部分数据是否来自于 TEE。区块链 A 通过对比 TEE 程序代码测量值, 判断该 TEE 是否所需的跨链 TEE。同时区块链 A 保存 TEE 中一对非对称密钥对的公钥, 该公钥用于验签来自 TEE 的跨链数据。区块链 A 还可以保存 TEE 程序初始化时的参数, 如 TEE 客户端所连接的区块链信息, 以知晓该 TEE 桥接哪条区块链。上述中所示的 TEE 程序 (TEE 应用) 中包含轻客户端, 具体实现上可以理解为 TEE 应用中包含有轻客户端程序模块, 这在本领域中属于技术人员可以在应用程序中实现的方式。

10 [50] 执行过程中, 需要跨链读取数据的区块链 A 请求 TEE 程序读取跨链数据。TEE 程序使用内置轻客户端 (如 SPV 客户端) 请求跨链目标区块链 B 并验证其数据。然后 TEE 程序可以基于 intel SGX 生成的使用私钥对轻客户端验证通过的请求结果进行签名, 返回给区块链 A。所述的请求结果可以包括所述跨链数据, 或者以及其他相关的数据信息。

15 [51] 区块链 A 可以使用前述中锚定的公钥验证从 TEE 程序获得的跨链数据, 验证跨链数据的确是已经经过可信的 TEE 程序完成验证。此时区块链 A 得到区块链 B 上的跨链数据, 并且该跨链数据是 TEE 使用内置客户端验证区块链 B 后再经过 TEE 上芯片自身的使用签名以保障该跨链数据是经过可信性的 TEE 验证后发送而来的, 请求认证的区块链 A 可以使用相应的公钥验证, 通过后即可使用该跨链数据, 快速高效的实现跨链区块链数据的认证处理。

20 [52] 上述实施例中, 一个 TEE 程序可以链接一个目标区块链, 请求认证的区块链可以通过 TEE 程序读取跨链数据。当存在多个区块链时, 可以为各个区块链相应的配置相应的 TEE 程序, 或者当新增加区块链时, 可以为其配置相应的 TEE 程序。这样, 基于本说明书提供的使用 TEE 作证明转换的跨链数据交互处理方式, 可以具有更好的扩展性, 有效提高跨链数据处理的吞吐量, 提高区块链系统之间的数据处理性能。

[53] 因此, 基于上述实施方案描述, 本说明书提供的一种跨区块链的数据处理方法的一种实施例中, 具体的如图 2 所示, 所述方法可以包括:

25 [54] S0: 第一区块链向信任证明方发送跨链数据请求;

[55] S2: 信任证明方基于所述跨链数据请求获取第二区块链的跨链数据;

[56] S4: 所述信任证明方对所述跨链数据进行验证, 在所述验证通过后, 使用采用的可信执行环境策略对应的私钥对所述跨链数据进行签名;

[57] S6: 信任证明方将签名后的请求结果返回给第一区块链;

[58]S8: 所述第一区块链利用可信任执行环境策略对应的公钥验证所述请求结果, 在验证通过时确定所述请求结果中的跨链数据来自信任证明方且已由所述信任证明方完成验证。

[59]上述实施例中, 所述的信任证明方的一种实现方式可以包括前述所述的 TEE 应用, 如采用程序代码实现的装置或终端设备或程序模块, 可以作为区块链间数据交互的中继器。所述的第一区块链可以为向信任证明方请求跨链数据的区块链, 所述的第二区块链可以为返回跨链数据的目标区块链。所述的第一区块链和第二区块链中的“第一”、“第二”描述并不限定具体指向固定的某一区块链, 不同的信任验证方中, 可以对应不同的区块链。

10 [60]所述的第一区块链向信任证明方发送跨链数据请求具体的可以是指第一区块链上的某个区块链节点发起的跨链数据请求, 相应的, 获取第二区块链上的跨链数据, 具体可以是通过第二区块链的某个区块链节点获取得到第二区块链的数据。

[61]需要说明的是, 上述实施例中所述的信任证明方采用的可信任执行环境策略可以包括 Intel SGX, 但本说明书不排除其他的可信任执行环境策略, 如 AMD 的 PSP (Platform Security Processor)、ARM TrustZone 技术 (支持 TrustZone 的所有 ARM 处理器)、Intel x86-64 指令集等。本说明书提供的一个实现方式中,

[62]S40: 所述的可信任执行环境策略包括 Intel SGX (英特尔的软件保护扩展策略)。

[63]信任证明方对获取的第二区块链数据进行验证时, 可以采用多种方式进行验证, 可以对区块链数据采用完整性的验证, 或者其他公有链、私有链以及采用合约协议约定的方式进行验证等。本说明书提供的一种实施方式如前述所述, 可以采用 SPV (简单支付验证), 具体的可以在所述信任验证方中设置有 SPV 轻客户端, 利用区块链头的数据实现对区块链数据的验证。SPV 简单支付验证, 在 SPV 节点不保存全部的区块链数据, 只保存区块头的数据, 使得 SPV 客户端对整个存储的容量要求更小, 因此 SPV 轻客户端一般也称为轻量级客户端, 实现快速的区块链数据验证。因此, 本说明书提供的另一个实施例中, 所述对所述跨链数据进行验证包括:

[64]S42: 采用 SPV (简单支付验证) 的方式对所述跨链数据进行验证。

[65]本说明书实施例提供的一种跨区块链的数据处理方法, 可以基于 TEE 区块链间的信任桥接器, 快速高效的实现跨区块链的数据认证, 实现跨区块链数据处理的高扩展性, 适应于更多的跨链数据交互场景, 简易高效的提高跨链数据吞吐流量的上限。有效解决

类似传统 POS 共识中继使用场景有限、效率性能更较低问题。

[66]上述实施例方案的实现中，可以一个 TEE 程序对应一个目标区块链，可以在一个 TEE 程序设置一个与所述目标区块链对应的 SPV 客户端。本说明书还提供另一种实施例，可以在一个 TEE 程序中设置多个 SPV 客户端，并存储各个 SPV 客户端相应的目标区块链的配置信息，实现一个 TEE 对应多个 SPV 客户端。当然，可以并列的设置多个 TEE 程序，例如一共 10 个目标区块链的情况下，可以设置三个 TEE 程序终端，如三个上述所述的处理装置或中继器，称为 T1、T2、T3，其分别各自设置有 3 个、3 个、4 个 SPV 客户端，每个 SPV 客户端对应一个目标区块链。因此，本说明书还提供另一种跨区块链的数据处理方法的实施例，在接收第一区块链发送跨链数据请求之后，所述方法还包括：

[67]根据所述跨链数据请求确定对应的 SPV 客户端（或称为简单支付验证的应用）；
[68]相应的，所述获取第二区块链的跨链数据以及对所述跨链数据进行验证包括：利用对应的 SPV 客户端（简单支付验证的应用）获取第二区块链的跨链数据以及对所述跨链数据进行验证。
[69]这样，利用本实施例方案，可以更加灵活进行跨区块链数据交互扩展，优化系统设计，提供更加合理的布局设置支持，提高跨链数据吞吐流量。

[70]图 7 是本说明书提供的另一种跨区块链的数据处理的流程及系统结构示意图。
[71]基于前述多侧区块链和信任证明方交互实施例方法的描述，本说明书还提供一种可以应用于所述信任证明方（区块链之间数据交互的中继器）一侧的方法实施例。具体的一种实施例如图 3 所示，可以包括：

[72]S200：接收第一区块链发送跨链数据请求；
[73]S202：基于所述跨链数据请求获取第二区块链的跨链数据；
[74]S204：对所述跨链数据进行验证，在所述验证通过后，使用采用的可信任执行环境策略对应的私钥对所述跨链数据进行签名；其中，所述可信任执行环境策略对应的公钥对外发布。

[75]S206：将签名后的请求结果返回给第一区块链，所述请求结果包括所述跨链数据和所述签名。

[76]所述第一区块链利用可信任执行环境策略对应的公钥验证所述请求结果，在验证通

过时确定所述请求结果中的跨链数据来自信任证明方且已由所述信任证明方完成验证。

[77]参照前述多侧交互相关实施例的描述，所述方法的另一个实施例中，

[78]S2040: 所述的可信任执行环境策略包括 Intel SGX (英特尔的软件保护扩展策略)。

[79]参照前述多侧交互相关实施例的描述，所述方法的另一个实施例中，所述对所述跨链数据进行验证包括：

[80]S2042: 采用 SPV (简单支付验证) 的方式对所述跨链数据进行验证。

[81]本说明书中上述方法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。相关之处参见方法实施例的部分说明即可。

- 10 [82]本申请实施例所提供的方法实施例可以在固定终端、移动终端、服务器或者类似的运算装置中执行。以运行在服务器上为例，图 4 是应用本说明书实施例的一种跨区块链的数据处理方法的客户端的硬件结构框图，当然，也可以应用在区块链的节点服务器中，其硬件结构可以与客户端硬件结构相同，或类似的硬件结构，或可以根据终端设备与服务
- 15 器设备的不同，有更多或更少的硬件结构。具体的，如图 4 所示，客户端 10 可以包括一个或多个（图中仅示出一个）处理器 102（处理器 102 可以包括但不限于微处理器 MCU 或可编程逻辑器件 FPGA 等的处理装置）、用于存储数据的存储器 104、以及用于通信功能的传输模块 106。本领域普通技术人员可以理解，图 4 所示的结构仅为示意，其并不对上述电子装置的结构造成限定。例如，服务器还可包括比图 4 中所示更多或者更少的组件，例如还可以包括其他的处理硬件，或者具有与图 4 所示不同的配置。
- 20 [83]存储器 104 可用于存储应用程序的软件程序以及模块，如本发明实施例中的一种跨区块链的数据处理方法对应的程序指令/模块，处理器 102 通过运行存储在存储器 104 内的软件程序以及模块，从而执行各种功能应用以及数据处理，即实现终端屏幕保险的投保、理赔、审查、赔付等处理。存储器 104 可包括高速随机存储器，还可包括非易失性存储器，如一个或者多个磁性存储装置、闪存、或者其他非易失性固态存储器。在
- 25 一些实例中，存储器 104 可进一步包括相对于处理器 102 远程设置的存储器，这些远程存储器可以通过网络连接至计算机终端。上述网络的实例包括但不限于互联网、企业内部网、局域网、移动通信网及其组合。

[84]传输模块 106 用于经由一个网络接收或者发送数据。上述的网络具体实例可包括计算机终端 10 的通信供应商提供的无线网络。在一个实例中，传输模块 106 包括一个网

网络适配器（Network Interface Controller, NIC），其可通过基站与其他网络设备相连从而可与互联网进行通讯。在一个实例中，传输模块 106 可以为射频（Radio Frequency, RF）模块，其用于通过无线方式与互联网进行通讯。

[85] 基于上述所述的跨区块链的数据处理方法，本说明书还提供一种检测车辆损伤的处理装置。所述的装置可以包括使用了本说明书实施例所述方法的系统（包括分布式系统）、软件（应用）、模块、组件、服务器、客户端等并结合必要的实施硬件的设备装置。基于同一创新构思，本说明书提供的一种实施例中的处理装置如下面的实施例所述。由于装置解决问题的实现方案与方法相似，因此本说明书实施例具体的处理装置的实施可以参见前述方法的实施，重复之处不再赘述。尽管以下实施例所描述的装置较佳地以软件来实现，但是硬件，或者软件和硬件的组合的实现也是可能并被构想的。具体的，如图 5 所示，图 5 是本说明书提供的可以作为跨区块链中继器的一种跨区块链的数据处理的数据处理装置实施例的模块结构示意图，所述装置可以为单独的客户端装置、服务器，或处理器/服务器集群，或分布式系统等方式实现，具体的可以包括：

[86] 请求接收模块 200，可以用于接收第一区块链发送跨链数据请求；

15 [87] 数据获取模块 202，可以用于基于所述跨链数据请求获取第二区块链的跨链数据；

[88] 数据验证模块 204，可以用于对所述跨链数据进行验证，在所述验证通过后，使用采用的可信任执行环境策略对应的私钥对所述跨链数据进行签名；其中，所述可信任执行环境策略对应的公钥对外发布；

[89] 证明返回模块 206，可以用于将签名后的请求结果返回给第一区块链，所述请求结果包括所述跨链数据和所述签名。

20

[90] 参照前述方法实施例所述，所述装置的另一个实施例中，所述数据验证模块 204 采用的可信任执行环境策略包括英特尔的软件保护扩展策略。

[91] 参照前述方法实施例所述，所述装置的另一个实施例中，所述数据验证模块 204 可以包括：

25 [92] 简单支付验证单元 2041，可以用于采用简单支付验证的方式对所述跨链数据进行验证。

[93] 图 6 是本说明书提供的所述装置的另一种实施例的模块结构示意图。

[94] 另一种实施例中，所述装置还包括：

[95]处理方确定模块，可以用于在接收第一区块链发送跨链数据请求之后，根据所述跨链数据请求确定对应的简单支付验证的应用；相应的，所述数据验证模块获取第二区块链的跨链数据以及对所述跨链数据进行验证包括；利用对应的简单支付验证的应用获取第二区块链的跨链数据以及对所述跨链数据进行验证。

- 5 [96]需要说明的是，本说明书实施例上述所述的装置，根据相关方法实施例的描述还可以包括其他的实施方式。具体的实现方式可以参照方法实施例的描述，在此不作一一赘述。

[97]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。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

[98]本说明书实施例提供的跨区块链的数据处理的处理方法和装置可以在计算机中由处理器执行相应的程序指令来实现，如使用 windows 操作系统的 c++语言在 PC 端实现，或其他例如 Linux、android、iOS 系统相对应的应用设计语言结合必要的硬件实现，或者基于分布式系统的多服务器处理，或者基于量子计算机的处理逻辑实现等。具体的，

15 本说明书提供一种实现上述方法或装置的客户端实施例中，所述客户端可以包括处理器以及用于存储处理器可执行指令的存储器，所述处理器执行所述指令时实现：

[99]接收第一区块链发送跨链数据请求；

[100] 基于所述跨链数据请求获取第二区块链的跨链数据；

[101] 对所述跨链数据进行验证，在所述验证通过后，使用采用的可信任执行环境策略对应的私钥对所述跨链数据进行签名；

20

[102] 将签名后的请求结果返回给第一区块链，所述请求结果包括所述跨链数据和所述签名。

[103] 如前述所述，所述客户端其他的实施例中，可信任执行环境策略包括英特尔的软件保护扩展策略。或者，所述处理器执行所述指令对所述跨链数据进行验证时，采用简单支付验证的方式对所述跨链数据进行验证。

25

[104] 上述所述的指令可以存储在多种计算机可读存储介质中。所述计算机可读存储介质可以包括用于存储信息的物理装置，可以将信息数字化后再以利用电、磁或者光学等方式的媒体加以存储。本实施例所述的计算机可读存储介质有可以包括：利用电能方式存储信息的装置如，各式存储器，如 RAM、ROM 等；利用磁能方式存储信息的装置

如，硬盘、软盘、磁带、磁芯存储器、磁泡存储器、U 盘；利用光学方式存储信息的装置如，CD 或 DVD。当然，还有其他方式的可读存储介质，例如量子存储器、石墨烯存储器等等。下述所述的装置或服务器或客户端或系统中的指令同上描述。

5 [105] 上述所述的装置或客户端可以部署在区块链系统中，可以实现区块链协议，可以认证区块链系统中的数据，与所连接的区块链交互的同时，可以将区块链数据根据其
他区块链的请求发送给相应的区块链，实现跨区块链之间的数据交互。因此，本说明书
还提供一种区块链系统，所述系统包括本说明书任一实施例所述的装置或客户端。

10 [106] 本说明书提供一个跨区块链系统实施例中，可以包括多个区块链，允许提供给
其他区块链数据的目标区块链可以设置有对应的中继器，所述的中继器可以如前述所述
的客户端或信任验证方或装置等。这样，在整个跨区块链系统中，可以实现高效、高扩
展的跨链数据交互。具体的，一种跨区块链系统，包括至少两个区块链以及返回跨链数
据的目标区块链对应的中继器，请求数据的区块链通过信对应的中继器获取目标区块链
的跨链数据，所述中继器包括本说明书实施例中任意一项所述的装置，或者本说明书中
任意一项所述的方法实现。

15 [107] 当区块链系统中有新的区块链接入，扩展跨区块链交互的设备性能时，则可以
通过本说明书实施例再增加一个中继器，高效便捷的实现扩展。因此，所述跨区块链系
统的另一个实施例中，当所述跨区块链系统接入新区块链，且所述新区块链允许与所述
跨区块链系统中的至少一条区块链进行跨链数据交互时，增加并配置所述新区块链的中
继器。

20 [108] 需要说明的是，本说明书实施例上述所述的装置、客户端、系统等，根据相关
方法或装置实施例的描述还可以包括其他的实施方式。具体的实现方式可以参照方法实
施例的描述，在此不作一一赘述。

25 [109] 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
硬件+程序类实施例而言，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。

[110] 上述对本说明书特定实施例进行了描述。其它实施例在所附权利要求书的范围
内。在一些情况下，在权利要求书中记载的动作或步骤可以按照不同于实施例中的顺序
来执行并且仍然可以实现期望的结果。另外，在附图中描绘的过程不一定要求示出的特

定顺序或者连续顺序才能实现期望的结果。在某些实施方式中，多任务处理和并行处理也是可以的或者可能是有利的。

[111] 本说明书实施例提供一种跨区块链的数据处理方法、装置、客户端、区块链系统，可以基于 TEE 实现区块链间的数据交互，快速高效的实现跨区块链的数据认证，
5 实现跨区块链数据处理的高扩展性，适应于更多的跨链数据交互场景。利用本说明书实施例方案，可以简易高效的提高跨链数据吞吐流量的上限，有效解决类似传统 POS 共识中继使用场景有限、效率性能更较低问题。

[112] 虽然本申请提供了如实施例或流程图所述的方法操作步骤，但基于常规或者无创造性的劳动可以包括更多或者更少的操作步骤。实施例中列举的步骤顺序仅仅为众多
10 步骤执行顺序中的一种方式，不代表唯一的执行顺序。在实际中的装置或客户端产品执行时，可以按照实施例或者附图所示的方法顺序执行或者并行执行（例如并行处理器或者多线程处理的环境）。

[113] 尽管本说明书实施例内容中提到利用 SPV 客户端对目标区块链数据进行验证、选取 Intel SGX 的 TEE 实现、中继器的交互等之类的数据获取、存储、交互、计算、判
15 断等操作和数据描述，但是，本说明书实施例并不局限于必须是符合行业通信标准、标准区块链协议、通信协议和标准网络模型/模板或本说明书实施例所描述的情况。某些行业标准或者使用自定义方式或实施例描述的实施例基础上略加修改后的实施方案也可以实现上述实施例相同、等同或相近、或变形后可预料的实施效果。应用这些修改或变形后的数据获取、存储、判断、处理方式等获取的实施例，仍然可以属于本说明书的可选
20 实施方案范围之内。

[114] 在 20 世纪 90 年代，对于一个技术的改进可以很明显地区分是硬件上的改进（例如，对二极管、晶体管、开关等电路结构的改进）还是软件上的改进（对于方法流程的改进）。然而，随着技术的发展，当今的很多方法流程的改进已经可以视为硬件电路结构的直接改进。设计人员几乎都通过将改进的方法流程编程到硬件电路中来得到相应的
25 硬件电路结构。因此，不能说一个方法流程的改进就不能用硬件实体模块来实现。例如，可编程逻辑器件（Programmable Logic Device, PLD）（例如现场可编程门阵列（Field Programmable Gate Array, FPGA））就是这样一种集成电路，其逻辑功能由用户对器件编程来确定。由设计人员自行编程来把一个数字系统“集成”在一片 PLD 上，而不需要请芯片制造厂商来设计和制作专用的集成电路芯片。而且，如今，取代手工地制作集成
30 电路芯片，这种编程也多半改用“逻辑编译器（logic compiler）”软件来实现，它与

- 程序开发撰写时所用的软件编译器相类似，而要编译之前的原始代码也得用特定的编程语言来撰写，此称之为硬件描述语言（Hardware Description Language，HDL），而 HDL 也并非仅有一种，而是有许多种，如 ABEL（Advanced Boolean Expression Language）、AHDL（Altera Hardware Description Language）、Confluence、CUPL（Cornell University Programming Language）、HDL、JHDL（Java Hardware Description Language）、Lava、Lola、MyHDL、PALASM、RHDL（Ruby Hardware Description Language）等，目前最普遍使用的是 VHDL（Very-High-Speed Integrated Circuit Hardware Description Language）与 Verilog。本领域技术人员也应该清楚，只需要将方法流程用上述几种硬件描述语言稍作逻辑编程并编程到集成电路中，就可以很容易得到实现该逻辑方法流程的硬件电路。
- 5 [115] 控制器可以按任何适当的方式实现，例如，控制器可以采取例如微处理器或处理器以及存储可由该（微）处理器执行的计算机可读程序代码（例如软件或固件）的计算机可读介质、逻辑门、开关、专用集成电路（Application Specific Integrated Circuit，ASIC）、可编程逻辑控制器和嵌入微控制器的形式，控制器的例子包括但不限于以下微控制器：ARC 625D、Atmel AT91SAM、Microchip PIC18F26K20 以及 Silicone Labs C8051F320，存储器控制器还可以被实现为存储器的控制逻辑的一部分。本领域技术人员也知道，除了以纯计算机可读程序代码方式实现控制器以外，完全可以通过将方法步骤进行逻辑编程来使得控制器以逻辑门、开关、专用集成电路、可编程逻辑控制器和嵌入微控制器等的形式来实现相同功能。因此这种控制器可以被认为是一种硬件部件，而对其内包括的用于实现各种功能的装置也可以视为硬件部件内的结构。或者甚至，可以
- 10 将用于实现各种功能的装置视为既可以是实现方法的软件模块又可以是硬件部件内的结构。
- [116] 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。具体的，计算机例如可以为个人计算机、膝上型计算机、车载人机交互设备、蜂窝电话、相机电话、
- 25 智能电话、个人数字助理、媒体播放器、导航设备、电子邮件设备、游戏控制台、平板计算机、可穿戴设备或者这些设备中的任何设备的组合。
- [117] 虽然本说明书实施例提供了如实施例或流程图所述的方法操作步骤，但基于常规或者无创造性的手段可以包括更多或者更少的操作步骤。实施例中列举的步骤顺序仅仅为众多步骤执行顺序中的一种方式，不代表唯一的执行顺序。在实际中的装置或终端
- 30 产品执行时，可以按照实施例或者附图所示的方法顺序执行或者并行执行（例如并行处

理器或者多线程处理的环境，甚至为分布式数据处理环境）。术语“包括”、“包含”或者其任何其他变体意在涵盖非排他性的包含，从而使得包括一系列要素的过程、方法、产品或者设备不仅包括那些要素，而且还包括没有明确列出的其他要素，或者是还包括为这种过程、方法、产品或者设备所固有的要素。在没有更多限制的情况下，并不排除在包括所述要素的过程、方法、产品或者设备中还存在另外的相同或等同要素。

[118] 为了描述的方便，描述以上装置时以功能分为各种模块分别描述。当然，在实施本说明书实施例时可以把各模块的功能在同一个或多个软件和/或硬件中实现，也可以将实现同一功能的模块由多个子模块或子单元的组合实现等。以上所描述的装置实施例仅仅是示意性的，例如，所述单元的划分，仅仅为一种逻辑功能划分，实际实现时可以有另外的划分方式，例如多个单元或组件可以结合或者可以集成到另一个系统，或一些特征可以忽略，或不执行。另一点，所显示或讨论的相互之间的耦合或直接耦合或通信连接可以是通过一些接口，装置或单元的间接耦合或通信连接，可以是电性，机械或其它的形式。

[119] 本领域技术人员也知道，除了以纯计算机可读程序代码方式实现控制器以外，完全可以通过将方法步骤进行逻辑编程来使得控制器以逻辑门、开关、专用集成电路、可编程逻辑控制器和嵌入微控制器等的形式来实现相同功能。因此这种控制器可以被认为是一种硬件部件，而对其内部包括的用于实现各种功能的装置也可以视为硬件部件内的结构。或者甚至，可以将用于实现各种功能的装置视为既可以是实现方法的软件模块又可以是硬件部件内的结构。

[120] 本发明是参照根据本发明实施例的方法、设备（系统）、和计算机程序产品的流程图和/或方框图来描述的。应理解可由计算机程序指令实现流程图和/或方框图中的每一流程和/或方框、以及流程图和/或方框图中的流程和/或方框的结合。可提供这些计算机程序指令到通用计算机、专用计算机、嵌入式处理机或其他可编程数据处理设备的处理器以产生一个机器，使得通过计算机或其他可编程数据处理设备的处理器执行的指令产生用于实现在流程图一个流程或多个流程和/或方框图一个方框或多个方框中指定的功能的装置。

[121] 这些计算机程序指令也可存储在能引导计算机或其他可编程数据处理设备以特定方式工作的计算机可读存储器中，使得存储在该计算机可读存储器中的指令产生包括指令装置的制造品，该指令装置实现在流程图一个流程或多个流程和/或方框图一个方框或多个方框中指定的功能。

- [122] 这些计算机程序指令也可装载到计算机或其他可编程数据处理设备上,使得在计算机或其他可编程设备上执行一系列操作步骤以产生计算机实现的处理,从而在计算机或其他可编程设备上执行的指令提供用于实现在流程图一个流程或多个流程和/或方框图一个方框或多个方框中指定的功能的步骤。
- 5 [123] 在一个典型的配置中,计算设备包括一个或多个处理器(CPU)、输入/输出接口、网络接口和内存。
- [124] 内存可能包括计算机可读介质中的非永久性存储器,随机存取存储器(RAM)和/或非易失性内存等形式,如只读存储器(ROM)或闪存(flash RAM)。内存是计算机可读介质的示例。
- 10 [125] 计算机可读介质包括永久性和非永久性、可移动和非可移动媒体可以由任何方法或技术来实现信息存储。信息可以是计算机可读指令、数据结构、程序的模块或其他数据。计算机的存储介质的例子包括,但不限于相变内存(PRAM)、静态随机存取存储器(SRAM)、动态随机存取存储器(DRAM)、其他类型的随机存取存储器(RAM)、只读存储器(ROM)、电可擦除可编程只读存储器(EEPROM)、快闪记忆体或其他内存技术、只读光盘只读存储器(CD-ROM)、数字多功能光盘(DVD)或其他光学存储、磁盒式磁带,磁带磁磁盘存储或其他磁性存储设备或任何其他非传输介质,可用于存储可以被计算设备访问的信息。按照本文中的界定,计算机可读介质不包括暂存电脑可读媒体(transitory media),如调制的数据信号和载波。
- 15 [126] 本领域技术人员应明白,本说明书的实施例可提供为方法、系统或计算机程序产品。因此,本说明书实施例可采用完全硬件实施例、完全软件实施例或结合软件和硬件方面的实施例的形式。而且,本说明书实施例可采用在一个或多个其中包含有计算机可用程序代码的计算机可用存储介质(包括但不限于磁盘存储器、CD-ROM、光学存储器等)上实施的计算机程序产品的形式。
- 20 [127] 本说明书实施例可以在由计算机执行的计算机可执行指令的一般上下文中描述,例如程序模块。一般地,程序模块包括执行特定任务或实现特定抽象数据类型的例程、程序、对象、组件、数据结构等等。也可以在分布式计算环境中实践本说明书实施例,在这些分布式计算环境中,由通过通信网络而被连接的远程处理设备来执行任务。在分布式计算环境中,程序模块可以位于包括存储设备在内的本地和远程计算机存储介质中。
- 25 [128] 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，对于系统实施例而言，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。在本说明书的描述中，参考术语“一个实施例”、“一些实施例”、“示例”、“具体示例”、或“一些示例”等的描述意指结合该实施例或示例描述的具体特征、结构、材料或者特点包含于本说明书实施例的至少一个实施例或示例中。在本说明书中，对上述术语的示意性表述不必须针对的是相同的实施例或示例。而且，描述的具体特征、结构、材料或者特点可以在任一个或多个实施例或示例中以合适的方式结合。此外，在不相互矛盾的情况下，本领域的技术人员可以将本说明书中描述的不同实施例或示例以及不同实施例或示例的特征进行结合和组合。
- 5
- 10 [129] 以上所述仅为本说明书实施例的实施例而已，并不用于限制本说明书实施例。对于本领域技术人员来说，本说明书实施例可以有各种更改和变化。凡在本说明书实施例的精神和原理之内所作的任何修改、等同替换、改进等，均应包含在本说明书实施例的权利要求范围之内。