



(12)发明专利申请

(10)申请公布号 CN 109857727 A

(43)申请公布日 2019.06.07

(21)申请号 201910157804.5

(22)申请日 2019.03.02

(71)申请人 杭州复杂美科技有限公司
地址 310000 浙江省杭州市西湖区文三路
90号东部软件园6号楼7层702室

(72)发明人 马登极 王志文 吴思进

(51)Int.Cl.
G06F 16/21(2019.01)
G06F 16/23(2019.01)

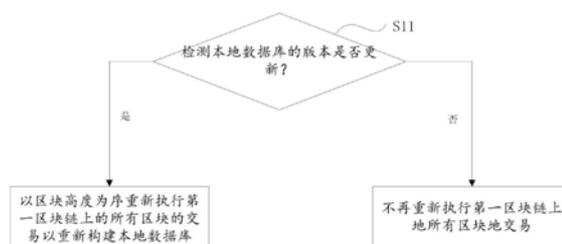
权利要求书1页 说明书4页 附图2页

(54)发明名称

一种本地数据库重构方法、设备和存储介质

(57)摘要

本发明提供一种本地数据库重构方法、设备和存储介质,该方法包括:检测本地数据库的版本是否更新:是,则以区块高度为序重新执行第一区块链上的所有区块的交易以重新构建本地数据库。本发明通过为本地数据库配置版本号,检测数据库版本号是否有更新,如果有更新,则按照区块高度从低到高的顺序重新执行区块中的交易以重构本地数据库,从而保证本地数据库中均为当前区块链版本的实际结果同时大大提高了重新构建数据库的效率。



1. 一种本地数据库重构方法,其特征在于,包括:
检测本地数据库的版本是否更新:
是,则以区块高度为序重新执行第一区块链上的区块的交易以重新构建本地数据库。
2. 根据权利要求1所述的方法,其特征在于,所述重新计算第一区块链上的区块的交易以重新构建本地数据库包括:
删除本地数据库,按照区块高度从低到高的顺序获取并执行各所述区块的交易;
将计算执行各所述交易后生成的若干数据信息存储在本地数据库中以重新构建所述本地数据库。
3. 根据权利要求2所述的方法,其特征在于,还包括:
若执行各所述交易过程中发生中断,系统重启后从中断的区块高度后继续执行后续各所述区块。
4. 根据权利要求2或3所述的方法,其特征在于,在所述重新计算第一区块链上的区块的交易之前还包括:
获取本地数据库中有更新的合约名称;
所述重新计算第一区块链上的区块的交易以重新构建本地数据库,包括:
删除本地数据库中存储的关于所述合约名称对应合约的若干数据信息,根据所述合约名称获取并执行与对应合约相关的区块交易以重新构建本地数据库。
5. 一种设备,其特征在于,所述设备包括:
一个或多个处理器;
存储器,用于存储一个或多个程序,
当所述一个或多个程序被所述一个或多个处理器执行时,使得所述一个或多个处理器执行如权利要求1-4中任一项所述的方法。
6. 一种存储有计算机程序的存储介质,其特征在于,该程序被处理器执行时实现如权利要求1-4中任一项所述的方法。

一种本地数据库重构方法、设备和存储介质

技术领域

[0001] 本申请涉及区块链技术领域,具体涉及一种本地数据库重构方法、设备和存储介质。

背景技术

[0002] 随着区块链版本的迭代,本地数据库和状态数据库也应该不断有更新,然而区块链节点分布众多,版本往往不会及时更新,然而本地数据库不进行共识,如果版本更新不及时也不会影响数据存入,然而在升级版本后,本地数据库会有和当前版本所记录数据结构不一致的情况,现有更新本地数据库的方案为删掉本地数据库和状态数据库,重新从区块高度为0开始同步数据。现有的方案缺陷在于:由于区块高度动辄几百万高度,还会随时增长,删除数据库后重新在系统中同步数据的方法耗时很长,效率低下。

发明内容

[0003] 鉴于现有技术中的上述缺陷或不足,期望提供一种效率高的,仅需要重构本地数据库的重构方法、设备和存储介质。

[0004] 第一方面,本发明提供一种本地数据库的重构方法,包括:

[0005] 检测本地数据库的版本是否更新:

[0006] 是,则以区块高度为序重新执行第一区块链上的所有区块的交易以重新构建本地数据库。

[0007] 进一步,重新计算第一区块链上的所有区块的交易以重新构建本地数据库包括:

[0008] 删除本地数据库,并按照区块高度从低到高的顺序获取并执行各区块的交易;

[0009] 将计算执行各交易后生成的若干数据信息存储在本地数据库中以重新构建本地数据库。

[0010] 进一步,上述方法还包括:

[0011] 若执行各交易过程中发生中断,系统重启后从中断的区块高度后继续执行后续各区块。

[0012] 第二方面,本发明还提供一种设备,包括一个或多个处理器和存储器,其中存储器包含可由该一个或多个处理器执行的指令以使得该一个或多个处理器执行根据本发明各实施例提供的本地数据库重构方法。

[0013] 第三方面,本发明还提供一种存储有计算机程序的存储介质,该计算机程序使计算机执行根据本发明各实施例提供的本地数据库重构方法。

[0014] 本发明诸多实施例提供的一种本地数据库重构方法、设备和存储介质通过为本地数据库配置版本号,检测数据库版本号是否有更新,如果有更新,则按照区块高度从低到高的顺序重新执行区块中的交易以重构本地数据库,大大提高了重构数据库的效率。进一步,本发明还可以只针对其中某一种合约更新时针对该合约存储的数据进行重新构建,对其他没有更新的合约数据则不做处理,有针对性的进行数据库更行,进一步减少了重构数据库

的时间。

附图说明

[0015] 通过阅读参照以下附图所作的对非限制性实施例所作的详细描述,本申请的其它特征、目的和优点将会变得更明显:

[0016] 图1为本发明一实施例提供的一种本地数据库重构方法的流程图。

[0017] 图2为图1所述方法的一优选实施方式的流程图。

[0018] 图3为图1或图2所示方法的一优选实施方式的流程图。

[0019] 图4为本发明一实施例提供的一种设备的结构示意图。

具体实施方式

[0020] 下面结合附图和实施例对本申请作进一步的详细说明。可以理解的是,此处所描述的具体实施例仅仅用于解释相关发明,而非对该发明的限定。另外还需要说明的是,为了便于描述,附图中仅示出了与发明相关的部分。

[0021] 需要说明的是,在不冲突的情况下,本申请中的实施例及实施例中的特征可以相互组合。下面将参考附图并结合实施例来详细说明本申请。

[0022] 本发明提供一种本地数据库重构的技术方案。在本发明中,区块链数据库包括本地数据库和状态数据库,状态数据库中存储交易的关键信息,通过区块链共识实现更新,在区块高度正常增长的情况下,状态数据库存储的是当前版本的实际结果,一般不需要重新构建;本地数据库用于存储记录一些交易的辅助信息,用来辅助用户查询需要的信息,如供列表查询用户的资产,该用户的买单时间,挂单时间等相关信息,这些信息不需要通过共识,可以直接记录在本地数据库即可;本地数据库以key-value的形式对数据信息进行存储,查询时通过提供key值进行数据库查询。

[0023] 本发明的技术方案中,为本地数据库配置文件配置版本号,区块链版本升级时(如优化了某项功能或者增加了某项功能等),会有相应的key值更新或增加,当有key更新或者增加时,就更新配置文件版本号。

[0024] 图1为本发明一实施例提供的一种本地数据库重构方法的流程图。

[0025] 如图1所示,在本实施例中,本发明提供一种本地数据库重构方法,包括:

[0026] S11、检测本地数据库的版本是否更新:

[0027] 是,则以区块高度为序重新执行第一区块链上的所有区块的交易以重新构建本地数据库。

[0028] 具体地,系统重新启动后,会检测判断配置文件当前的版本号b与存储在本地数据库中的版本号a(数据库中有专门的key值记录版本号信息,重启不消失)是否有更新,若有,则更新本地数据库中的版本号a为当前的版本号b,并删除本地数据库存储的第一区块链上的交易键值信息,并按照区块高度从低到高的顺序重新执行第一区块链上各区块的交易,并将执行交易后生成的若干辅助信息重新存储到本地数据库。

[0029] 进一步,上面所述的本地数据库的版本更新,可以是配置文件的版本号增加,也可以是按照其他定义方式进行改变版本号,只要是符合节点认定规则能实现新旧版本的判定即可。

[0030] 在更多实施例中,本地数据库是否重新构建,即是否重新执行区块链上的交易以重新生成本地数据库,与数据库代码有更新没有关系,如果本地数据库版本不变,即使代码改变,系统重新启动也不会重新执行所有区块;即使代码不改变,若本地数据库版本改变,系统重启后也会重新执行所有区块。

[0031] 上述实施例通过检测本地数据库的版本是否有更新以对本地数据库重新构建,实现了单独更新本地数据库,不需要从0开始重新从网络同步数据,也不需要更新状态数据库,大大节省了版本升级后重新构建数据库的时间。

[0032] 图2为图1所述方法的一优选实施方式的流程图。如图2所示,在一优选实施例中,上述方法还包括:

[0033] S12、若执行各交易过程中发生中断,系统重启后从中断的区块高度后继续执行后续各区块。

[0034] 具体地,系统在重新执行第一区块链上区块中的交易时,会在执行完某个区块交易的区块高度信息存储到本地数据库中,若在执行过程因为某些原因(如网络中断,停电等情况)发生中断,则系统重新启动后,则将从存储在本地数据库中的最后一个区块高度开始继续执行后面区块高度的区块交易,直到达到状态数据库中记录的区块高度。

[0035] 图3为图1或图2所示方法的一优选实施方式的流程图。如图3所示,在一优选实施例中,重新计算第一区块链上的区块的交易之前还包括:

[0036] S111、获取本地数据库中有更新的合约名称;

[0037] 重新计算第一区块链上的区块的交易以重新构建本地数据库,包括:

[0038] S112、删除本地数据库中存储的关于合约名称对应合约的若干数据信息,根据合约名称获取并执行与对应合约相关的区块交易以重新构建本地数据库。

[0039] 具体地,系统重启后,检测本地数据库配置文件版本更新时判断是否有合约的更新,若有,则获取更新的合约名称,根据合约名称删除本地数据库中存储的该合约名称对应的合约的相关数据,然后从区块高度为0的区块开始执行与该合约有关的交易,将执行后的若干交易数据存储到本地数据库中。

[0040] 上述实施例中通过仅构建某一种或几种更新合约的本地数据库,对于其他未更新的合约数据则不做处理,这样进一步提高了重新构建数据库的速度。

[0041] 图4为本发明一实施例提供的一种设备的结构示意图。

[0042] 如图4所示,作为另一方面,本申请还提供了一种设备300,包括一个或多个中央处理单元(CPU)301,其可以根据存储在只读存储器(ROM)302中的程序或者从存储部分308加载到随机访问存储器(RAM)303中的程序而执行各种适当的动作和处理。在RAM303中,还存储有设备300操作所需的各种程序和数据。CPU301、ROM302以及RAM303通过总线304彼此相连。输入/输出(I/O)接口305也连接至总线304。

[0043] 以下部件连接至I/O接口305:包括键盘、鼠标等的输入部分306;包括诸如阴极射线管(CRT)、液晶显示器(LCD)等以及扬声器等的输出部分307;包括硬盘等的存储部分308;以及包括诸如LAN卡、调制解调器等的网络接口卡的通信部分309。通信部分309经由诸如因特网的网络执行通信处理。驱动器310也根据需要连接至I/O接口305。可拆卸介质311,诸如磁盘、光盘、磁光盘、半导体存储器等等,根据需要安装在驱动器310上,以便于从其上读出的计算机程序根据需要被安装入存储部分308。

[0044] 特别地,根据本公开的实施例,上述任一实施例描述的本地数据库重构方法可以被实现为计算机软件程序。例如,本公开的实施例包括一种计算机程序产品,其包括有形地包含在机器可读介质上的计算机程序,计算机程序包含用于执行本地数据库重构方法的程序代码。在这样的实施例中,该计算机程序可以通过通信部分309从网络上被下载和安装,和/或从可拆卸介质311被安装。

[0045] 作为又一方面,本申请还提供了一种计算机可读存储介质,该计算机可读存储介质可以是上述实施例的装置中所包含的计算机可读存储介质;也可以是单独存在,未装配入设备中的计算机可读存储介质。计算机可读存储介质存储有一个或者一个以上程序,该程序被一个或者一个以上的处理器用来执行描述于本申请的本地数据库重构方法。

[0046] 附图中的流程图和框图,图示了按照本发明各种实施例的系统、方法和计算机程序产品的可能实现的体系架构、功能和操作。在这点上,流程图或框图中的每个方框可以代表一个模块、程序段、或代码的一部分,该模块、程序段、或代码的一部分包含一个或多个用于实现规定的逻辑功能的可执行指令。也应当注意,在有些作为替换的实现中,方框中所标注的功能也可以以不同于附图中所标注的顺序发生。例如,两个接连地表示的方框实际上可以基本并行地执行,它们有时也可以按相反的顺序执行,这根据所涉及的功能而定。也要注意的是,框图和/或流程图中的每个方框、以及框图和/或流程图中的方框的组合,可以通过执行规定的功能或操作的专用的基于硬件的系统来实现,或者可以通过专用硬件与计算机指令的组合来实现。

[0047] 描述于本申请实施例中涉及到的单元或模块可以通过软件的方式实现,也可以通过硬件的方式来实现。所描述的单元或模块也可以设置在处理器中,例如,各单元可以是设置在计算机或移动智能设备中的软件程序,也可以是单独配置的硬件装置。其中,这些单元或模块的名称在某种情况下并不构成对该单元或模块本身的限定。

[0048] 以上描述仅为本申请的较佳实施例以及对所运用技术原理的说明。本领域技术人员应当理解,本申请中所涉及的发明范围,并不限于上述技术特征的特定组合而成的技术方案,同时也应涵盖在不脱离本申请构思的情况下,由上述技术特征或其等同特征进行任意组合而形成的其它技术方案。例如上述特征与本申请中公开的(但不限于)具有类似功能的技术特征进行互相替换而形成的技术方案。

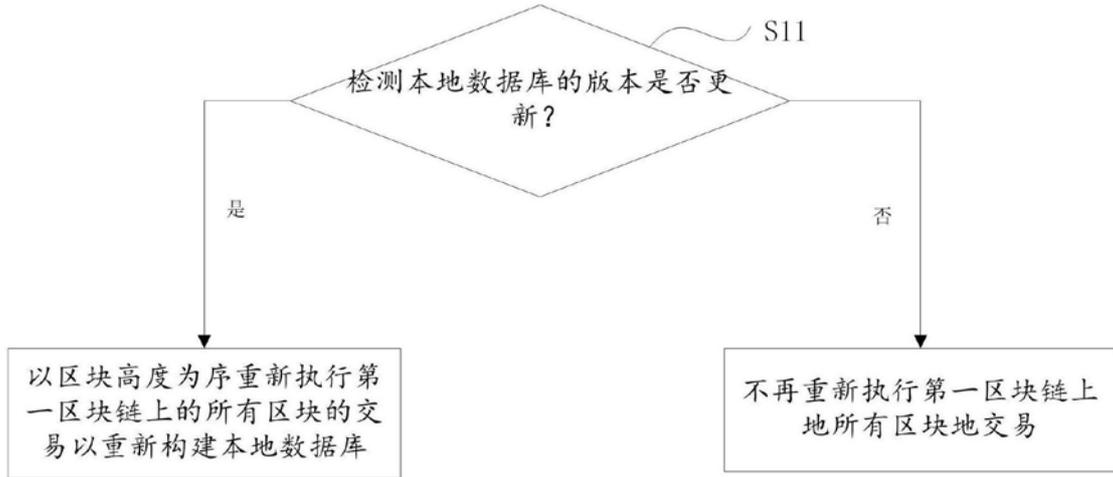


图1

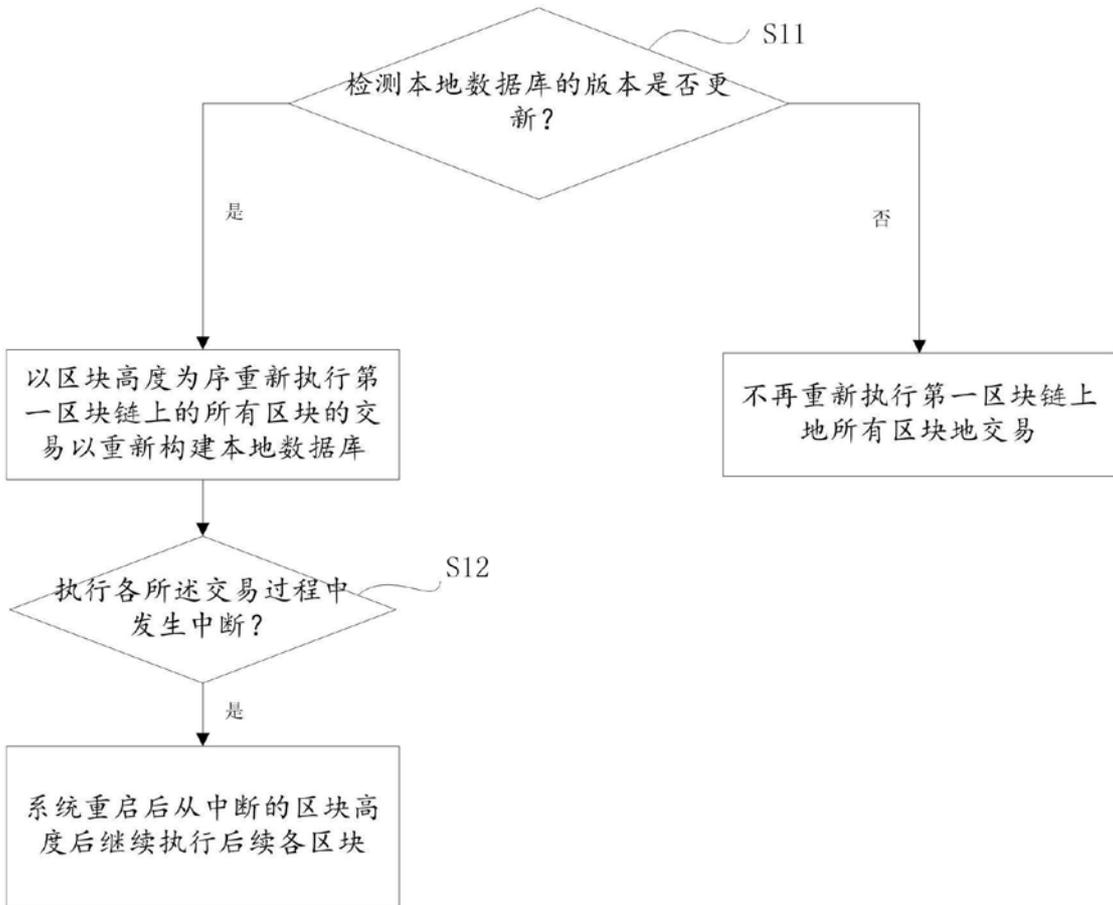


图2

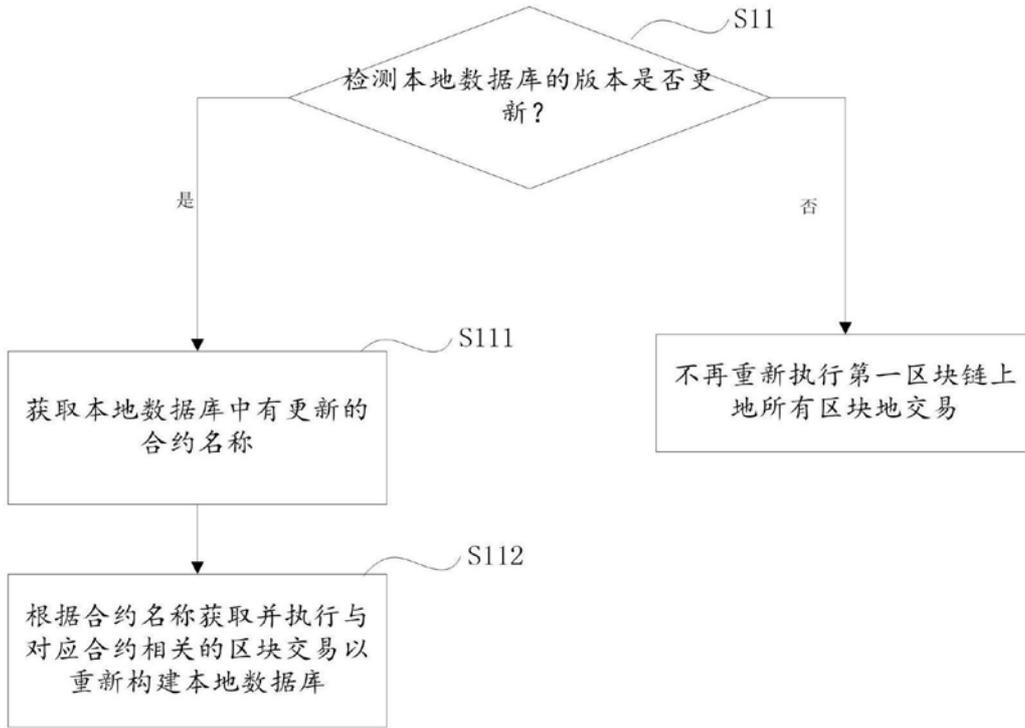


图3

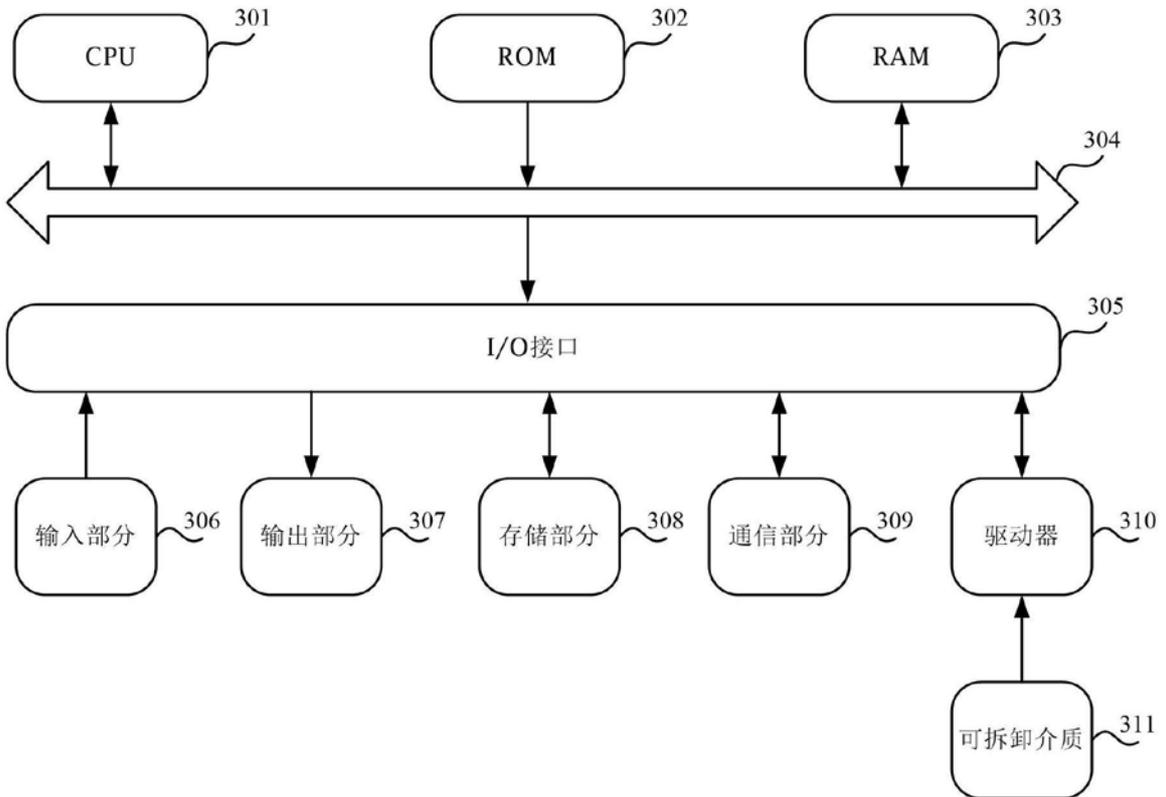


图4