软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2017,28(6):1474−1487 [doi: 10.13328/j.cnki.jos.005232]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel: +86-10-62562563

基于区块链的应用系统开发方法研究*

蔡维德[1], 郁 莲[2], 王 荣[1], 刘 娜[1,3], 邓恩艳[4]

[1](软件开发环境国家重点实验室(北京航空航天大学) 数字社会与区块链实验室,北京 100191)

[2](北京大学 软件与微电子学院,北京 102600)

[3](沈阳工程学院 信息学院,辽宁 沈阳 110136)

[4](北京天德科技有限公司,北京 100089)

通讯作者: 蔡维德, E-mail: tsai7@yahoo.com, 郁莲, E-mail: lianyu@ss.pku.edu.cn

摘 要: 从区块链的技术层面及应用层面分析其特征,并给出区块链的分类.挖掘区块链的设计需求,针对区块链的一致性和可扩展性的应用需求进行深入分析.对区块链的应用系统开发方法及区块链建模进行研究,提出了账户区块链(account blockchain,简称 ABC)和交易区块链(trading blockchain,简称 TBC)的双链设计模型.对智能合约进行深入剖析,提出了链上代码并行执行模型应用原则.最后,对区块链应用技术进行总结和展望.

关键词: 区块链;ABC/TBC;链上代码;区块链应用需求

中图法分类号: TP311

中文引用格式: 蔡维德,郁莲,王荣,刘娜,邓恩艳.基于区块链的应用系统开发方法研究.软件学报,2017,28(6):1474−1487. http://www. jos.org.cn/1000-9825/5232.htm

英文引用格式: Tsai WT, Yu L, Wang R, Liu N, Deng EY. Blockchain application development techniques. Ruan Jian Xue Bao/ Journal of Software, 2017,28(6):1474−1487 (in Chinese). http://www.jos.org.cn/1000-9825/5232.htm

Blockchain Application Development Techniques

TSAI Wei-Tek[1], YU Lian[2], WANG Rong[1], LIU Na[1,3], DENG En-Yan[4]

[1](Digital Society & Blockchain Laboratory, State Key Laboratory of Software Development Environment (Beihang University), Beijing 100191, China)

[2](School of Software and Microelectronics, Peking University, Beijing 102600, China)

[3](Information School, Shenyang Institute of Engineering, Shenyang 110136, China)

[4](Tiande Technologies, Beijing 100089, China)

Abstract: This paper presents a blockchain definition independent of any digital currency, and describes its characteristics including consensus protocols, design patterns, scalability, databases, and chaincode. The paper then presents a permissioned blockchain, called Beihangchain, with its unique consensus algorithms, interfaces, and design. It also proposes ABC (account blockchain) and TBC (trading blockchain), to be used for a variety of applications including copyright protection and digital payment. Finally, this paper analyzes chaincode requirements and provides guidelines for effective chaincode.

Key words: Blockchain; ABC/TBC; chaincode; blockchain application requirements

1 区块链简介

区块链(blockchain)是由多独立节点参与的分布式数据库系统,也可以理解为分布式账簿(distributed ledger

* 基金项目: 国家自然科学基金(61690200, 60973001)
Foundation item: National Natural Science Foundation of China (61690200, 60973001)
收稿时间: 2016-11-03; 修改时间: 2016-12-14; 采用时间: 2017-01-09; jos 在线出版时间: 2017-02-20
CNKI 网络优先出版: 2017-02-22 10:47:37, http://www.cnki.net/kcms/detail/11.2560.TP.20170222.1047.004.html

technology,简称 DLT),由这些节点共同维护.它的特点是不易篡改、很难伪造、可追溯.区块链记录所有发生交易的信息,过程高效透明,数据高度安全.凡是需要公正、公平、诚实的应用领域,都可以应用区块链技术.

区块链把数据分成不同的区块,每个区块通过特定的信息链接到上一区块的后面,前后顺连,呈现一套完整的数据.每个区块的块头(block header)包含前一个区块的哈希值(previous block Hash),该值是对前区块的块头进行哈希函数计算(Hash function)而得到.区块之间都会由这样的哈希值与先前的区块环环相扣形成一个链条,如图 1 所示.

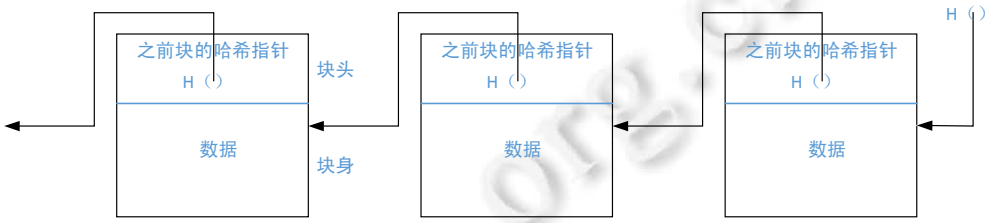

Fig.1 Blockchain schematic

图 1 区块链示意图

从技术层面上看,区块链的核心要素包含以下 3 个方面.

(1) 块链结构:每一区块有时间戳;都使用前一区块的哈希加密信息;对每个交易进行验证;

(2) 多独立拷贝存储:每个节点都存储同样信息,享有同样权利;独立作业;互相怀疑,互相监督;

(3) 拜占庭容错:容忍少于三分之一的节点恶意作弊或被黑客攻击,保证系统仍然能够正常工作.

要素(1)指出,区块链是一个“账簿”;要素(2)指出,区块链是一个“分布式账簿”;而要素(3)指出,区块链是一个“一致性的同步分布式账簿”.

区块链可选择不同的加密方法,如 RSA、中国的国密算法[1]、Ed25519[2]等的签名算法.根据区块链自身特有的安全、极难篡改的特性,在金融领域外的很多应用场景中,使用签名、解签能够达到足够高的安全级别.

各个节点在独立作业的同时存储着同样的信息,并且拥有同样的权利.如果这一点不能保障的话,就不可称为区块链.例如,若链上的某个节点有特殊的权利,甚至可以改变链上数据,这样的链就远离了区块链的真意.与现有的分布式存储方式不同,区块链分布式账本是同步的,而不是在一个账本形成之后,再制成多个备份.

拜占庭将军模型[3]的共识算法有串行与并行两种.

- 拜占庭将军模型于 1982 年出自 SRI Int’l,实用拜占庭共识协议 PBFT(practical byzantine fault tolerance)[4]是经过多年研究,由 MIT 在 1999 年提出实用协议.交易与投票是串行的,建块过程要经过 3 次投票;
- 并发拜占庭共识协议 CBFT(concurrent byzantine fault tolerance)是由作者的团队于 2015 年提出,此项算法将交易与投票并行进行.

以比特币的区块链为代表的第 1 代区块链并未使用拜占庭将军算法.比特币和以太坊的公有区块链使用 PoW(51%的投票)[5,6].作为第 2 代区块链的代表,以太坊的私有链选用了 PBFT.作为第 3 代区块链的代表,北航链使用的是 CBFT,提高了性能.

从应用层面,区块链具有以下重要特征.

(1) 极难篡改性.一旦数据进入了区块链,即使是系统内部的工作人员,在区块链中也无法做任何更改.因此,区块链系统是可以被信赖的.这种极难更改的特点不是来自使用某种操作,而是由于区块链自身的机制;

(2) 链上代码(chaincode).区块链载入的合同或法律文件为可执行的程序,在条件都满足时,会让法律事务自动生成,这就是所谓的“链上代码”,在以太坊里也称为“智能和约”[7];

(3) 参与交易的每个人都拥有完整的数据,每个人可以用自己的数据来做决定;

(4) 每个人都有完整的历史数据,因此很难被其他人欺骗.区块链可以为相互不信任的人建立一种信任网络,每个人可以进行交易而不需要知道对方是什么人,因为每个人都有完整的数据,每个人都信任自己的数据,而且知道自己的数据是有共识的;

(5) 区块链的架构是分享的、分布式的、重复的,就地取材;

(6) 区块链与传统的计算架构不一样.最简单的方式是把原来的主-客架构变成多个主人,这在传统上被认为是不明智的方法.由于计算机硬件、通信及存储的成本降低,同时,社会对安全性和防篡改性的需求提高,原来不明智的方式变成了最先进的方式.

根据不同的应用领域特点,可以选择不同类型的区块链.一般分为公有链和许可链.

1) 公有链:所有节点中立、开放,都可以投票、记账、建块.因为全网都需要投票,所以交易速度非常慢.所有节点都可以参与投票,需要一个激励机制,即挖矿机制.许多公有链是基于无政府主义理念而设计的,所以采用点对点或 P2P 网络以躲避监管.两年前,比特币网络交易速度是每秒 7 笔[8],现在降到每秒 1.1 笔[9],而且越来越下降;

2) 许可链[10]:只有被许可的节点才能参与投票、记账、建块,包含私有链、联盟链、企业链等所有非公有链.交易速度比较快;不需要挖矿,交易成本低;可成为监管利器.北航链、天德链、Hyperledger[11]都是许可链,许可链会成为商业应用领域的主流[12].

公有链与许可链的技术需求和架构差异巨大,面临的问题也不一样.

2 区块链应用系统的需求与架构设计

2.1 区块链应用系统的需求

2.1.1 一致性需求

在分布式环境下,数据为保证一致性需要使用一致性协议.公有链主要使用工作证明 PoW(proof of work)[5]和股权证明 PoS(proof of stake)[6]机制;而许可链中主要使用 PBFT 和 CBFT.一般而言,区块链系统越高速越好,但是共识的代价昂贵,许多计算力及节点通信都花在共识机制上.例如,PBFT 需要 3 轮投票,每轮都采用广播式通信方式.每次通信都需要签名、解签,再加上每笔交易都要签名和解签,因而,80%的计算力都花在共识处理上.使用不同的共识算法会产生全然不同的区块链架构和流程,面临的研究问题也不同.PoW(公有链)面临的问题是速度和可扩展性,PBFT(许可链)面临的问题是并发.PoW 依靠节点的计算力来完成共识,PBFT 却不需要.

2.1.2 软件设计需求

区块链不同于传统数据库.使用区块链开发应用系统与传统系统比较会有很多差异.例如,应用区块链技术开发银行系统,可以省去很多中间环节,简化流程、节约成本.传统的软件架构也会发生变化.例如,IBM 把传统的 MVC(model,view,control)设计模式(design pattern)变为 MVBC(model,view,blockchain,control)[13].

设计区块链应用系统还有一个新问题,即,可以把功能放在应用系统上,或是放到区块链上用链上代码执行.很多人都推崇链上代码,可是链上代码执行会消耗大量的计算力.链上代码需要建块,而建块是一项昂贵的运营流程,需要执行共识协议.建议大部分的功能应该是在应用系统里,只有少数功能放在链上代码里.

2.1.3 可扩展性需求

可扩展性一直是区块链系统的一个挑战,从第一代比特币区块链到第二代以太坊区块链都面临严峻的问题.虽然有各式各样的解决方案,但是每种方案都有它的缺陷.例如,以太坊提出可无限扩张白皮书里提到的方案,经过了两年仍然不能实现.有些方案放弃区块链的定义来解决可扩展性需求,例如 BigchainDB[14],RSCoin[10,15,16].他们放弃区块链多拷贝的需求来提高交易速度.这些偏离传统区块链定义的系统是否能被接受还有待观察.一般来说,这样的系统因为放弃了多拷贝的需求,所以需要在其他方面补足来增加安全性.

北航链的可扩展性分为 3 步(参考第 4.1 节及文献[17]):(1) 使用 CBFT 并行的算法做拜占庭将军投票,从而提高建块速度;(2) 提出 ABC,TBC 双链架构[17],保护隐私、并行计算、节省计算力、简化应用架构;(3) 利用 ABC,TBC 双链的特点,使一条链可以在运行时分裂成两条链,有两套不同的硬件分别执行这两条链,以提高速

度.有这 3 个机制,既可以使用原始定义的区块链,又可以有高速以及可扩展性.

2.1.4 数据库需求

区块链虽然被称为分布式数据库,但是它的作业和传统数据库大不相同.不但与关系(relational)型数据库不一样,也与对象(object)数据库、NoSQL 数据库或时间(temporal)数据库不一样.高速区块链与低速区块链是截然不同的:在低速环境下,交易是用串行的方法来处理,所以低速区块链的一致性问题不大;而高速环境下,交易和建块是并行的,所以一致性是一个新问题.因为传统数据库是以个别交易,而区块链是以建块来维持一致性.

区块链一致性问题与传统数据库一致性问题不一样,例如在区块链里,每秒可以有上万次交易,而每秒都可以有多块被建立,所以每块也可以有上万次交易.这些交易中,可能有很多交易与同一个数据有关联.例如在央视微电影项目中,几秒钟之内会有上万人点播同一个视频,所以在一块里面,可能就要对同一个视频有上千个点播.如果使用传统数据库,每次点播都是一个写(write),而在同一个交易里面不可有一个以上的 write 在同一个数据上.可是在央视微电影平台上,必须允许同时在一个块中有上千个 write 作用在同一个数据上.

2.1.5 链上代码需求

链上代码原被称为求问题与传统(smart contract),给人们的印象是既智能又受法律保护的合约.但事实上两者都不是.传统的智能合约没有匹配的法律框架,不是有效力的法律合同.智能合约的参与者亦没有相关的法律条文及框架来保护.如果加上法律框架的支撑,链上代码可以成为合约.

链上代码的执行与建块息息相关,所以它的执行模型与建块流程相互影响,以至于链上代码在理论上变成一个很难的问题.问题难处在于:每次建块时,需要寻找必须要启动的链上代码,而且在一些链上代码系统里,那些代码必须完成执行之后才能建块.如果涉及的数据很多,而且链上代码很复杂,这将造成链上代码与建块冲突.虽然理论上链上代码是一个很难的问题,但是在实际系统中仍然可用.第 3.3 节将讨论一些实际解决方案.

2.2 北航链的体系架构

北航链是北京航空航天大学与北京大学联合开发的许可链,其设计初衷是为公信和金融服务,北航链摒弃了 P2P 网络[18,19]和挖矿机制[5],以可扩展性为第一目标[20],并且重视速度优化.为了确保系统安全,北航链加入了节点信用制度.这是首次采用信誉机制(reputation system)来识别作弊节点,一旦发现节点的作弊行为,立即将其排除在投票节点之外.

在北航链的设计中,拜占庭式投票和数据采集可同时进行,加快了信息处理速度,具有独特的建块过程.此外,对每个交易进行投票.为了确保安全,也对块的投票结果投票,判断是否有叛徒节点.由于有 4 轮投票,将产生更多的信息(每轮产生 $O(N^2)$个消息);北航链采用并发操作,所以速度快.另外,北航链还设计了一整套可扩展的机制,例如 ABC(account blockchain),TBC(trading blockchain)双链架构及区块链云架构.可扩展机制使得区块链具有高吞吐性、低延迟性以及高隐私性.有了这样的机制,当工作量请求增加的时候,只要增加机器就能够处理,从而实现负载均衡.图 2 是北航链架构图.

- 存储层:存储层包括操作系统和数据库服务;
- 基础区块链层:传输服务将缓存中的交易放入桶中;块服务为每个桶中的交易创建位图;Round Robin 使用循环法选择线索,创建并向所有其他节点发送块,进一步执行信誉计算;同步器广播本地区块链的长度,接收遗漏的块,并存储接收的块;ABC(帐户区块链)同步区块链,以确保不同节点的一致状态,创建帐户索引以加速查询,并提供帐户公私密钥服务;对于链上代码交易,TBC(交易区块链)首先执行链上代码,然后将结果放入桶中,对于非链上代码交易,直接放入桶中,并准备创建块;
- 缓存层:用于缓存内存中的临时信息,包括从用户和链上代码接收的新交易;那些块尚未传输到磁盘;并且支持系统运行的临时数据存储;
- API 层:提供了外部和内部 API 接口.内部 API 用于节点之间的内部通信,例如投票、广播块;外部 API 用于外部用户,例如接受新交易和查询操作;
- 链上代码层:提供与合同相关的服务.链上代码根据领域特定要求编写,由所有利益相关者进行合法正确性验证,然后部署在区块链系统中执行.该层具有 3 个功能:与用户的交互(编辑)、流程执行引擎和支

持帐户管理、状态存储和发送交易的合同服务;

- 应用层:此层有应用程序,例如银行系统、计算法律系统、信用认证系统和供应链系统.在设计区块链时,节点越多,系统越安全,但是共识起来会更慢,所耗的计算力越大.

Fig.2 Beihang chain architecture

图 2 北航链架构图

2.3 区块链接口设计

OBCC(open blockchain connector)是一套区块链的统一接口,提供应用方便高效地使用区块链的功能,包括将用户数据存入区块链、查询用户需要的信息,如图 3 所示.

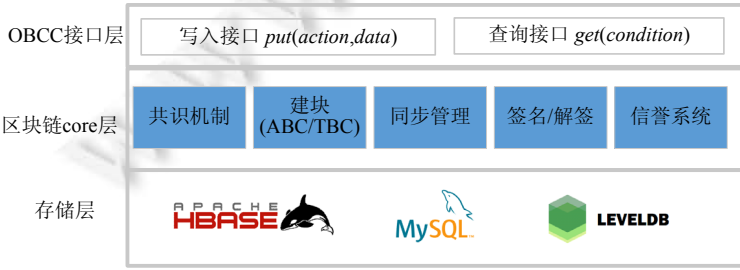

Fig.3 OBCC interface design

图 3 OBCC 接口设计

写入区块链的接口定义为 *put*(*action*,*data*),其中,

- 参数 *action* 表明用户的数据处理意图,可以是 create,insert, update 或 delete.注意:区块链是不能更改已经存入区块链的数据,这里的 update 和 delete 不是像数据库那样对数据执行 update 或 delete,而是在区块链上记录下对数据所发生过的操作,即,作为一笔新交易记录在案;
- 参数 *data* 是用户的数据,根据不同的应用领域,格式和内容会不同.

区块链查询接口定义为 *get*(*condition*),其中,参数 *condition* 表明用户的查询条件,可以是块的哈希值或交易的哈希值,也可以与应用有关的关键字等.倒排索引、大数据分析技术的使用,使得用户可以快速高效地获取有价值的查询结果.

OBCC 提供一个工具包,用户可以把它导入到自己的软件项目工程里,编程开发时,像是调用本地函数或方法一样使用区块链的功能接口.当用户程序需要调用区块链的功能时,由 OBCC 客户端代理将请求广播到各个区块链节点 OBCC 服务器端代理,该代理负责调用区块链的相关功能进行处理,最终存入区块链或查询到信息并返回.如图 4 所示.

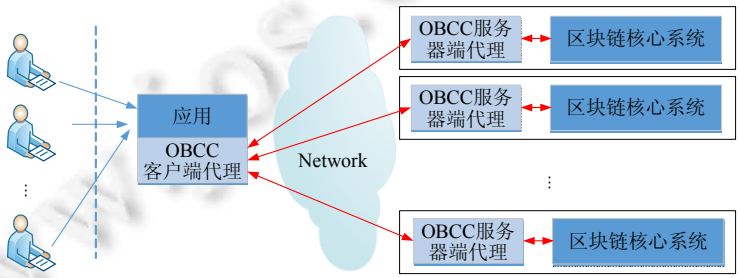

Fig.4 Blockchain application development based on OBCC

图 4 基于 OBCC 的区块链应用系统开发

本文实现了 Java 版的区块链连接器——JBCC,已经支持多个区块链的应用系统的开发,包括央视微电影管理平台、高校学籍及档案管理系统、金融跨国支付系统、银行信用卡消费管理系统、跨行业积分跟踪管理系统.基于 OBCC 的区块链的应用系统开发,具有开发周期短、可扩展性高、运行速度快的特点.

3 区块链应用开发方法研究

3.1 区块链双链设计研究

目前,区块链应用中通常只有一条区块链,把账目、合约、交易等全部放在这条区块链上.比如,日本银行所做的 POC[21]以及欧洲央行等模型仍使用旧式的通用账本架构[22].2015 年 5 月,欧洲银行联盟(Euro banking association,简称 EBA)提出的通用一条链概念.凡加入的机构,都要将自己内部的账户信息与其他加入的机构共享(没有隐私).所有参与的机构都在该链上作为一个节点参与投票,维持账目的一致性.这个通用一条链的设计不符合实际金融需求,因为没有保护隐私.这种架构造成系统上存在大量不同的数据,也违背了软件工程原则.这种设计扩展性差、吞吐量低.随着业务增多、节点的增加,通信量会非常庞大,延迟越来越高,所以性能会更低.

本文提出了一种新的架构是所有参与的机构分享元数据(metadata)及协议(protocols),但不分享数据(data 就是账户).所有参与的单位都可以与其他单位互相交易,而保证隐私性.根据这个概念,至少有下面两类区块链,如图 5 所示.

1) ABC 账户区块链(account blockchain):ABC 仅存储账户信息和交易后的信息,但不执行交易;
2) TBC 交易区块链(trading blockchain):TBC 仅存储对交易有用的信息并且执行相关交易.

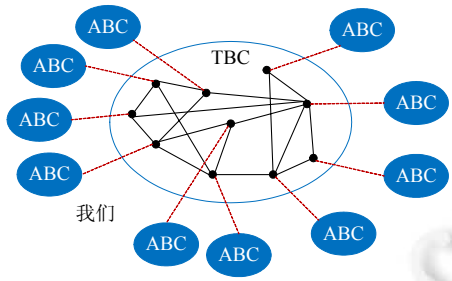

Fig.5 ABC and TBC architecture

图 5 ABC 和 TBC 架构图

ABC 负责查询、保存账目、建块.比如,ABC 存储金融机构或家庭帐户信息,一条链内账户信息是共享的,这样使得账户信息很难被篡改.同时,ABC 也提供可扩展性,即:当区块链处理大小超过限制,可以被分割成多个子 ABC,由不同机器上托管以保持平衡的工作量.一个区块链(链 1)(如图 6 所示,块 1、块 2、块 3)可以分成两个区块链,第 1 条(链 2)为块 1、块 2、块 3、块 4A,第 2 条(链 3)为块 1、块 2、块 3、块 4B,而这两个区块链都符合区块链的定义.

1) 块子链:链 2、链 3 都是块子链.每块有时间戳;都使用前块的哈希加密信息,每个交易都被验证;
2) 多独立的拷贝:链 2、链 3 的每个节点都存着同样信息,而且独立作业,互相怀疑,互相监督;
3) 拜占庭将军问题投票:链 2、链 3 都使用拜占庭将军问题投票,容忍少于三分之一的节点恶意作弊或被黑客攻击.

链 2、链 3 块子链,它们都是独立的区块链,可由不同的硬件或者服务器来支持.这样的区块链具有云计算、负载均衡的能力.

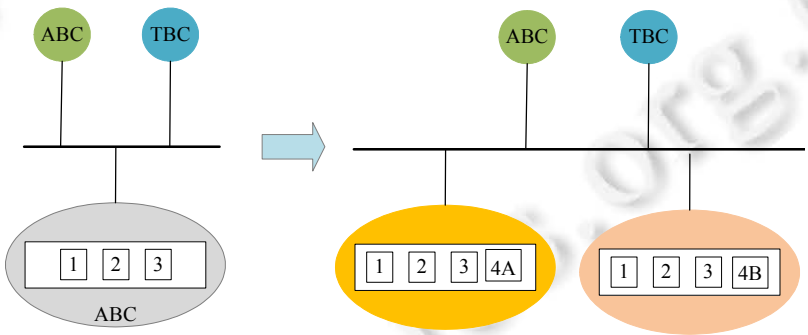

Fig.6 Scalability architecture

图 6 可扩展性架构图

TBC 负责建块、执行交易.TBC 仅仅是用作交易和结算的通道(或场所),它不保存交易双方账户信息,而存储在 TBC 的数据也被加密,使得只有参与机构可以看到数据.采用 ABC 和 TBC 双链架构,每个机构都可以拥有自己的账户区块链.只有当需要交易的信息时,才必须共享到交易区块链上.

这种机制的意味着在交易后,银行或者机构可以给予访问区块链权限,而底层的客户端的数据只能由相关银行和监管机构可以看到.在央视区块链应用系统中,ABC,TBC 概念被大量使用,另外,在国外的金融机构也使用北航链所设计的这种架构.

以下举例分析使用单链架构时计算能力需求.上交所 2014 年有 9 737.52 万个账户,2016 年 5 月,一天交易量 1 218 万笔,平均每秒 846 次.中国至少有 110 家券商、20 家银行,还有证监所和交易中心,至少需要有 150(n)个计算节点,中国工商银行有 4.65 亿个人客户和 509 万公司客户.20 家银行等至少约有 30 亿账户.

- R:每个节点每次建块投票需要通信 4 次;
- T:设交易所一天交易时间为 4 小时.

每次建块投票需要 $R\times150\times150(n^2)$=9 万的交换消息.假设每秒建一块,每天要建 $60\times60\times T$=1.44 万块.一天区块链要 1.44 万×9 万=12.96 亿建块交换消息.每个消息要签名和解签,因为是金融应用,还需要对消息本身进行加密和解密,所以每个消息需要 4 次加密计算.因而,每个节点每天要处理 12.96 亿建块交换消息加上 12.96×4=51.84 亿建块加密计算.也就是说,每个节点每秒 9 万建块交换消息加上 4×9 万=36 万建块加密计算.

每天按 1 000 万笔交易、每秒平均 $1000/(T\times60\times60)$=694 次交易、每次区块链交易每个节点查询需 30 亿(账户)次计算,每个节点每天需 30 亿×1000 万=3 万兆=3×10^{16} 查询计算.也就是说,每秒每个节点平均 2×10^{12} 查询计算.每次交易要签名和解签,还要加密和解密,每个节点都要处理每次交易,所以每天要处理 1000 万×4 交易加密计算,或是每秒每个节点处理 694×4=2777 次交易加密计算.

综上所述,平均每个节点每秒要处理 9 万建块交换消息、36 万建块加密计算、694 次交易、2×10^{12} 交易查询计算,再加上 2777 次交易加密计算.

采用双链架构时,每个组织至少有一条 ABC 链,大型组织可由多条 ABC 链.ABC 可用于账户消息记录、查询,TBC 可用于交易消息记录、查询.假设每条 ABC 链有 10 节点,1 个节点 1 台机器,每个组织有一条 ABC 链,所以 150 条 ABC 链. 假设每秒建一块,每条 ABC 链平均处理 0.2 亿(30 亿/150)账户.

每条 ABC 链每个节点每秒需要 $10\times10\times R$=400 的建块交换消息,加上 1 600(400×4)次建块加密计算.每条 ABC 链(10 个节点)每秒所有节点一共要处理 4 000(400×10)建块交换消息,16 000 次建块加密计算.一条 ABC 链所有节点的工作量远少过于通用一条链一个节点的工作量.

每条 ABC 链每天平均要处理 26.6 万(1000 万×4/150)交易(每笔交易至少要 2 个账户,假设复杂情况下有更多账户,如 4 个账户,所有交易平均分在 150 条 ABC 链),每秒每个节点平均处理 18(26.6 万/(60×60×4))次交易,每秒每个节点处理 72(=18×4)次加密计算.每条 ABC 链每秒所有节点一共要处理 180 次交易,720 次交易加密计算.一条 ABC 链所有节点的工作量远少过于通用一条链一个节点的工作量.

每条 ABC 链每个节点每秒处理 18 次交易.所以每个节点每秒需要查询 0.2 亿×18=3.6×10^8 次交易查询计算.每条 ABC 链每秒所有节点一共要处理 3.6×10^9 次交易查询计算.一条 ABC 链所有节点的工作量远少过于通用一条链一个节点的工作量.

一般情况下,每次交易不超过 10 个组织参与,例如交易所、银行、清算所、结算所等.依据熊猫模型[23],每条 ABC 链可以和多个组织交易,而不相干的组织无需参与其中交易.所以每条 ABC 链参与若干条 TBC 链,而不需要参与所有 TBC 链.因此,TBC 数量和账户是可以变动的,如果有清算所的架构,大约 10 条 TBC 链就够(一个清算所可以支持许多组织),假设最坏情形需要 20 条 TBC 链,每条 TBC 链有 30 节点参与共识计算.每个节点在一个时间上有一万个账户(旧的账户可以继续留在 TBC 历史记录中,但是不能交易),这表示有一万个账户同时在一条 TBC 上交易,20 条 TBC 链表示有 20 万个账户可以在那个时间可以交易.

每条 TBC 链每秒需要 $30\times30\times R$=3600 建块交换消息,加上 14 400(3600×4)次建块加密计算.这远少于通用一条链一个节点的工作量.每条 TBC 链(30 个节点)每秒所有节点一共要处理 10.8(3600×30)万建块交换消息,43.2(10.8 万×4)万次建块加密计算,所以每条 TBC 链所有节点的工作量只稍大于通用一条链的一个节点的工作量.

每天 1 000 万交易,每条 TBC 链每天平均要处理 200 万(1000 万×4/20)交易消息,平均每秒每个节点处理 139(200 万/(60×60×T)次交易,每秒每个节点处理 556 次交易加密计算.少于通用一条链一个节点的工作量.

每个 TBC 节点只有一万个账户,远少过 ABC 链(0.2 亿)和通用一条链(30 亿)的账户,所以每个 TBC 交易查询工作量也会远少过 ABC 链和通用一条链交易查询工作量.

讨论见表 1.

总结:通过采用双链架构,交易查询速度可以提升,成本也极大地降低;大量计算也可以并发,也保护隐私.

Table 1 Comparision between a general blockchain and ABC/TBC (per second/per node)

表 1 通用一条链和 ABC/TBC 比较(每秒、每节点)

	通用一条链	ABC	TBC
特性	一条链 150 节点,每个节点计算力要强大,不保护隐私	150 条 ABC 链、每条链 10 节点,保护隐私	20 条 TBC 链、每条链 30 节点,保护隐私
每节点账户 (所有节点账户)	30 亿户 (30 亿×150=4500 亿)	0.2 亿户 (0.2 亿×10×150=300 亿)	一万户 (1 万×30×20=600 万)
建块	9 万次建块消息, 36 万次建块加密	400 次建块消息, 1 600 次建块加密	3 600 次建块消息, 14 400 次建块加密
交易处理	700 次交易,2 777 次交易加密计算	18 次交易,72 次交易加密计算	139 次交易,556 次交易加密计算
交易查询	3×10^{16} 查询计算	3.6×10^{8} 查询计算	1.39×10^{6} 查询计算
运行	一条链运行	150 条 ABC 链可以并行运行, 也可以和多条 TBC 链并行运行	20 条 TBC 链可以并行运行, 也可以和多条 ABC 链并行运行

3.2 基于法律的区块链应用开发技术

传统的应用需求分析及建模通常包含功能、性能、安全、界面等方面,而区块链应用需求分析及建模还需要考虑法律,因为很多区块链应用与法律有关系.例如,使用区块链来存储电子证物,在中国必须要满足下面 3 个条件.

1) 及时性:数据必须是及时收集的;
2) 过程性:过程的数据必须被记录;
3) 不可篡改性:所收集及存储的数据必须证明没有被篡改过.

以央视微电影信息综合管理平台的需求为例:

在这个系统中,有 3 种角色,包括用户、发行商和广告商;有这些信息,包括用户信息(姓名、账户、IP 地址)、发行商信息(姓名、账户、IP 地址、视频内容)及视频(题目、存储地址、特性、观看次数、上交时间等).

如果用传统设计方法,图 7 所示的是一个相当好的解决方案.但是如果这是一个法律合规性的应用,图 7 就违反了很多原则.

1) 及时性:监视代理数据库所记载的数据是经过控制中心所送来的,如果控制中心是延迟发送,这套系统就不是及时性系统,不具法律效应;
2) 过程性:视频的发送是不经过控制中心,也没有被记载在监视数据库中;这套系统所记载的数据只有用户点击视频的记录,并没有完整的记录;
3) 不可篡改性:收集的数据存在监视代理的数据库里,而这个数据可以被监视代理内部人更改,或是外面黑客攻击更改,以造成不正确的信息转发给发行商和广告商.

所以,根据上面 3 个法律原则,图 7 的设计是不够成为电子证物的系统.区块链可以用来存储电子证物,但不是随意更改原有系统就可以成为电子证物的系统.例如,如果有人把图 7 监视代理的数据库变成区块链,也把发行商的数据库也变成区块链,上面及时性和过程性的问题都仍然存在.

因此,需要一种新的设计方法.首先考虑哪些信息需要存储在区块链上,图 8 给出一种解决方案.这个解决方案中共有 4 个区块链,其中 3 个是 ABC,一个是 TBC.3 个 ABC 是用户区块链、发行商区块链、广告商区块链,分别存储用户、发行商及广告商的数据,确保这些数据不会被篡改;TBC 是交易区块链,当用户点击时,一次交易完成.下面分析电子证物的需求.

1) 及时性:*A* 用户点击的时候,该点击信息先送到交易区块链,交易区块链把这些信息记录加上时间戳并把它分送到发行商及广告商区块链.点击的信息被及时的收集没有被延迟.*B* 视频数据库发送视频时,也是先送到交易区块链,再转给用户,因而交易区块链也收到及时的信息.所以,这套系统符合及时性;
2) 过程性:这套系统收集开始及结束的过程,就是用户点击以及视频播放.所以,这套系统符合过程性;
3) 不可篡改性:这套系统所收集的数据都存入区块链,以至于很难被篡改.系统管理人员及外面的黑客

都很难篡改系统所存的数据,因为区块链是多备份的,即使部分节点被攻破,数据仍然很难被篡改.而且这套系统的交易区块链所存的信息又复制到发行商区块链及广告商区块链,一份信息可能被好几个区块链分别存储,容错性更强.所以,这套系统符合不可篡改性.

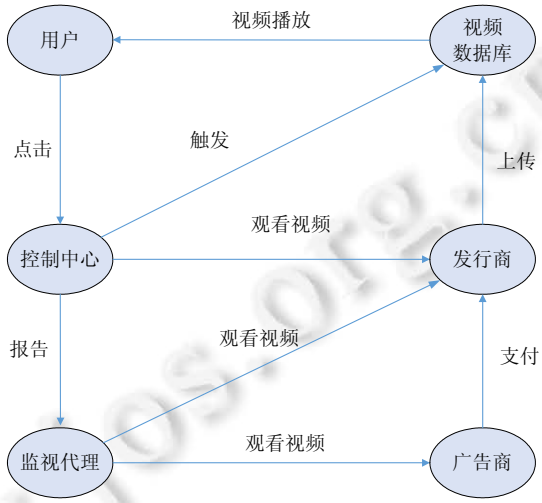

Fig.7 Traditional database design

图 7 传统设计

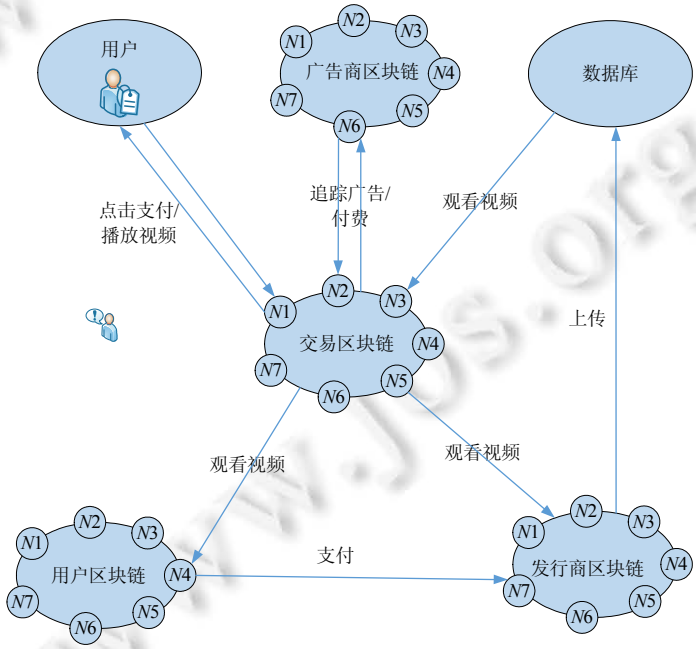

Fig.8 CCTV blockchain design

图 8 央视区块链流程图

3.3 链上代码设计研究

目前,区块链系统里使用的智能合约,意图建立一种无法被人为篡改和操控的升级版的代码合同,并声称将

用于金融、贸易、物联网、登记等领域.但是 The DAO 事件使人们看到该“智能合约”的非智能、非合同的本质特征.智能合约非法律所界定的有效力的合同,“智能合约”的假设是代码执行在区块链上,就成为法律上的合约.数据放在区块链上是难以被篡改的,这显然是一个不成立的假设.要成为合约需要多方法律上的验证.仅仅使用难以篡改的数据或数据库,离法律意义上的合约差得甚远.

在传统的可靠数据库管理系统(DBMS)中,事务(transaction)[24−26]应该具有的 4 个特性 ACID(ISO/IEC 10026-1:1992):原子性(atomicity)、一致性(consistency)、隔离性(isolation)、持久性(durability).CAP 理论指出,分布式计算系统不可能同时确保一致性(consistency)、可用性(availability)和分区容忍性(partition)[27,28].

在区块链数据库中,是不遵循 ACID 原则的.区块链用分布式账本保证数据的一致性,由建块来维持一致性,每块包含许多交易.区块链的事务方式和传统数据库事务方式并不相同,见表 2.

Table 2 Comparision between traditional database transaction and blockchain transanction

表 2 传统数据库事务方式和区块链的事务方式比较

	传统数据库	区块链
事务单位	每次交易都单独处理	事务处理是由建块来完成,每块包含多个交易,多交易一起处理
事务处理延时	因为交易单独处理,所以事务处理的延迟可能是两个交易有冲突,才会产生延迟,例如两个 Write 或者是一个 Read 和一个 Write	在 PBFT 中,每块可以有多个交易,而交易之间可能会有冲突,这个冲突有 3 种解决方法:1) 在应用上解决;2) 在建块时解决;3) 用链上代码解决.在 CBFT 中,同一时间可能有多块同时被建造以及投票,因而上面所述的问题更加复杂
事务处理算法	锁、乐观并发控制(optimistic concurrency control)[29]、多版本并发控制(MVCC[30])	PBFT 和 CBFT.每个区块链节点都有自己的数据库,而每个数据库都有自己的事务管理方法.所以,PBFT 和 CBFT 可以使用数据库自己的事务管理方法存块中数据
处理系统	DBMS(数据库管理系统)	每个区块链节点都有自己的数据库管理系统

由此可见,传统的数据处理方式和区块链的数据处理方式是不一样的,其中一个最大影响就是链上代码的执行,例如以太坊链上代码,每次链上代码都必须在建块前完成,以便把链上代码所改变的数据放在账目里.以太坊的链上代码执行模型[31]如图 9 所示.

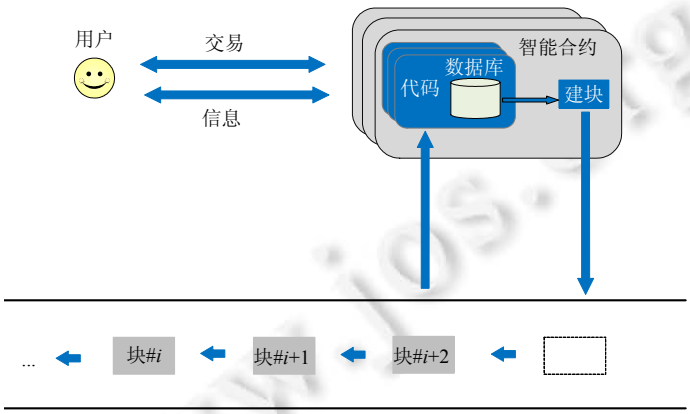

Fig.9 Execution model of ethereum chaincode

图 9 以太坊链上代码执行模型

但是,这样的设计并不适合金融需求.

1) 有的链上代码需要很长时间才能完成,例如某个股票链上代码,需要涨到某个价位才执行买卖.而这样的链上代码可能需要几个礼拜甚至几个月才能完成;

2) 在区块链上,许多账户都可以有链上代码,每次建块的时候,都要查询哪些链上代码需要执行.金融链一秒钟需要建立许多块,因而每次建块的时候,即使查询某个账户上的链上代码需要执行都要耗费多时,然后执行链上代码[32,33].例如:在中国工商银行有几亿的账户,即使 1%的账户有链上代码,每次建

块时候都要查询几百万的账户;

3) 在一个实际的金融系统,建块必须是高速的.建块高速,建的块也多,而每次建块都要查询链上代码并且加以执行,执行完下一块才能开始建[34].这就变成很难的任务.

金融链上代码需求:

1) 链上代码必须考虑长时间的合约,因为这是市场的需求;
2) 建块必须是高速的,因为这也是市场的需求;
3) 链上代码必须经过软件工程分析、测试、验证通过,例如使用形式化方法、仿真等方式;
4) 链上代码必须经过法律上的验证.

根据以上的需求,本文提出下列原则:

- 原则 A:一个功能可以写在应用系统中或者链上代码里,在绝大多数的环境,尽量在应用系统中解决;
- 原则 B:绝大多数的链上代码都应该不需要查询就能够被发现需要执行;
- 原则 C:每个账户的链上代码都只能够处理自己的账户或是少数相关的账户;
- 原则 D:大多数的链上代码都能够迅速地完成;
- 原则 E:长时间的链上代码,可以存中间的数据在中间的区块中.

所以,链上代码要有局部性(partial)、地方性(localized)、限时性(limited time span)、针对性(specific)才能实用.例如,这段代码只与某个账号有关系、每天下午 5 点执行、仅限制某些交易.在央视的微电影系统里,所有的点击、交易都不使用链上代码,但是分账使用链上代码.因为大部分的点击不需要分账,大部分情况下,链上代码不会被执行,所以符合原则 A.每次分账都是因为点击才产生的,因而每次建块不需要查询哪一个账户的链上代码需要执行,只有被点击的账户需要查询链上代码是否需要执行,这就符合原则 B.分账符合原则 C,也符合原则 D.原则 E 指出,长时间的链上代码要有一个复杂的机制来处理,例如,另外有一个区块链来存链上代码出的中间数据.

4 总结与展望

本文从区块链构成要素、应用特征开始,讨论区块链需求,包括一致性需求、软件设计需求、可扩展性需求、数据库需求和链上代码需求.在此基础上设计了北航链的体系架构.首次提出了开放式区块链连接器 OBCC,并实现了 Java 版的区块链连接器——JBCC.该 JBCC 已经支持多个区块链的应用系统的开发,具有开发周期短、可扩展性高、运行速度快的特点.为使区块链能够应用于金融业,介绍了由作者提出的双链模型,即 ABC 和 TBC.满足通信量庞大,快速响应、账户信息隐私的需要.本文讨论了链上代码面临法律合法性、代码可信性、执行安全性和外部信息共识等许多问题.

References:

[1] 国家密码管理局公告(第 23 号).2012. http://www.oscca.gov.cn/News/201204/News_1229.htm
[2] Ed25519: High-speed high-security signatures. http://ed25519.cr.yp.to
[3] Lamport L, Shostak R, Pease M. The Byzantine generals problem. ACM Trans. on Programming Languages and Systems (TOPLAS), 1982,4(3):382−401. [doi: 10.1145/357172.357176]
[4] Castro M, Liskov B. Practical Byzantine fault tolerance. In: Proc. of the USENIX Association. 1999. 173−186.
[5] Nakamoto S. Bitcoin: A Peer-to-Peer Electronic Cash System. 2008.
[6] King S, Nadal S. PPCoin: Peer-to-Peer Crypto-Currency with Proof-of-Stake. 2012.
[7] Ethereum Whitepaper. https://github.com/ethereum/wiki/wiki/White-Paper
[8] Bitcoin traffic bulletin. http://hashingit.com/analysis/34-bitcoin-traffic-bulletin
[9] Bitcoin traffic bulletin (redux). http://hashingit.com/analysis/44-bitcoin-traffic-bulletin-redux
[10] 蔡维德,赵梓皓,张弛,郁莲.英国央行数字货币 RSCoin 探讨.金融电子化,2016,(10):78−81.
[11] Hyperledger whitepaper-wg. https://wiki.hyperledger.org/groups/whitepaper/whitepaper-wg

[12] UK Government Chief Scientific Adviser. Distributed ledger technology: Beyond block chain, UK government office for science. 2016. https://www.gov.uk/government/uploads/system/uploads/attachment_data/file/492972/gs-16-1-distributed-ledger-technology.pdf

[13] Veena P, Ahluwalia G, Panikkar S. An Economy of Things—A Visionary Architecture and Monetization of Devices with Blockchain. IBM Inter Connect, 2016.

[14] McConaghy T, Marques R, Müller A, De Jonghe D, McConaghy TT, McMullen G, Henderson R, Bellemare S, Granzotto A. BigchainDB: A Scalable Blockchain Database. 2016.

[15] Danezis G, Meiklejohn S. Centrally banked cryptocurrencies. arXiv preprint arXiv:1505.06895, 2015.

[16] 蔡维德,罗佳.浅析私有区块链技术.2016. http://sanwen8.cn/p/1efjDL3.html

[17] Tsai WT, Blower R, Zhu Y, Yu L. A system view of financial blockchains. Service-oriented System Engineering, 2016. 450−457. [doi: 10.1109/SOSE.2016.66]

[18] Schollmeier R. A definition of peer-to-peer networking for the classification of peer-to-peer architectures and applications. In: Proc. of the 1st Int'l Conf. on P2P'01. IEEE, 2001. 101−102. [doi: 10.1109/P2P.2001.990434]

[19] Bandara HMND, Jayasumana AP. Collaborative applications over peer-to-peer systems—Challenges and solutions. Peer-to-Peer Networking and Applications, 2013,6(3):257−276. [doi: 10.1007/s12083-012-0157-3]

[20] 蔡维德.以系统观念来看区块链.2016. http://finance.sina.com.cn/money/bank/bank_hydt/2016-08-23/doc-ifxvcsrm2277962.shtml

[21] Nomura Research Institute, Ltd. NRI to conduct proof of concept for applying blockchain technology to banking. 2015. https://www.nri.com/global/news/2015/151216_1.aspx.2015

[22] Pinna A, Ruttenberg W. Distributed Ledger Technologies in Securities Post-Trading. 2016.

[23] 郁莲,邓恩艳.区块链技术.中国计算机学会通讯,2017,13(5):10−15.

[24] Haerder T, Reuter A. Principles of transaction-oriented database recovery. ACM Computing Surveys (CSUR), 1983,15(4):287−317. [doi: 10.1145/289.291]

[25] Gray J. The transaction concept: Virtues and limitations, 1981,81:144−154.

[26] Gray J, Reuter A. Transaction Processing: Concepts and Techniques. Elsevier, 1992.

[27] Brewer E. Towards robust towards robust distributed systems. In: Proc. of the 19th ACM Symp. on Principles of Distributed Computing. 2000. [doi: 10.1145/343477.343502]

[28] Gilbert S, Lynch N. Brewer's conjecture and the feasibility of consistent. In: Proc. of the Available, Partition-Tolerant Web Services. 2015.

[29] Kung HT, Robinson JT. On optimistic methods for concurrency control. ACM Trans. on Database Systems, 1981,6(2):213−226. [doi: 10.1145/319566.319567]

[30] Bernstein PA, Goodman N. Multiversion concurrency control—Theory and algorithms. ACM Trans. on Database Systems 1983, 8(4):465−483. [doi: 10.1145/319996.319998]

[31] Delmolino K, Arnett M, Kosba A, Shi E. Step by step towards creating a safe smart contract: lessons and insights from a cryptocurrency lab. Berlin, Heidelberg: Springer-Verlag, 2016. 79−94. [doi: 10.1007/978-3-662-53357-4_6]

[32] Thomas S, Schwartz E. Smart oracles: A simple, powerful approach to smart contracts. 2014. https://github.com/codius/codius/wiki/Smart-Oracles:-A-Simple,-Powerful-Approach-to-Smart-Contracts

[33] Hazard J, Hardjono T. CommonAccord: Towards a foundation for smart contracts in future blockchains. In: Proc. of the Blockchains and the Web: W3C Workshop on Distributed Ledgers on the Web. 2016.

[34] Yu L, Tsai WT, Li GN, Yao YF, Hu CJ. Smart-Contract Execution with Concurrent Block Building. In: Proc. of the 10th IEEE Int'l Symp. on Service-Oriented System Engineering, 2016.

蔡维德(1958－),男,四川庐县人,博士,教授,博士生导师,主要研究领域为区块链技术,软件工程,分布式系统,云计算与大数据.

郁莲(1963－),女,博士,副教授,主要研究领域为分布式计算,形式化方法,区块链技术,软件分析与验证.

王荣(1988－),男,硕士,主要研究领域为区块链,机器学习.

刘娜(1984－),女,讲师,主要研究领域为区块链,移动数据库.

邓恩艳(1972－),女,主要研究领域为区块链,软件工程.