单位代码	10445
学　　号	2017309035
分 类 号	TP311
学习方式	全日制

山东师范大学

硕士学位论文

（专业学位）

论文题目　**基于区块链的数字资产存证系统设计与实现**

专业学位名称　工程硕士

方向领域名称　计算机技术

申请人姓名　张亚伟

指导教师　张问银 教授　李　晔 研究员

论文提交时间　2019 年 6 月 10 日

单位代码	10445
学　　号	2017309035
分类号	TP311
学习方式	全日制

山东师范大学

硕 士 学 位 论 文

（专 业 学 位）

论文题目　**基于区块链的数字资产存证系统设计与实现**

专业学位名称：工程硕士

方向领域名称：计算机技术

申 请 人 姓 名：张亚伟

指 导 教 师：张问银 教授 李晔 研究员

论文提交时间：2019 年 6 月 10 日

目　录

摘要.....I
ABSTRACT.....II
第 1 章 绪论.....1
1.1 引言.....1
1.2 课题研究背景和意义.....1
1.3 国内外研究现状.....2
1.4 课题设计目标.....3
1.5 论文内容安排.....3
1.6 本章小结.....4
第 2 章 相关概念和技术.....5
2.1 区块链的基本概念.....5
2.2 数字资产存证应用.....6
2.2.1 数字资产的含义.....6
2.2.2 数字资产存证和电子数据存证的关系.....7
2.3 IPFS.....7
2.3.1 IPFS 的基本概念.....7
2.3.2 IPFS 的工作原理.....8
2.4 区块链核心技术.....9
2.4.1 P2P 网络.....9
2.4.2 共识机制.....10
2.4.2.1 共识机制的作用.....10
2.4.2.2 PoW 共识的基本流程.....10
2.4.2.3 PBFT 共识的基本流程.....11
2.4.2.4 RAFT 共识的基本流程.....12
2.4.3 智能合约.....13
2.4.3.1 智能合约的概念.....13
2.4.3.2 智能合约的请求调用流程.....14
2.4.4 密码学相关技术.....15
2.4.4.1 非对称加密.....15
2.4.4.2 哈希算法.....15
2.4.4.3 数字签名.....16
2.4.4.4 Merkle Tree.....17

2.5 本章小结.....18
第 3 章 基于区块链的数字资产存证系统需求分析.....19
3.1 系统功能需求分析.....19
3.2 系统主要功能需求.....20
3.2 系统非功能需求分析.....23
3.2.1 系统环境需求分析.....23
3.2.2 系统质量需求分析.....23
3.2.3 系统性能需求分析.....23
3.2.3 系统可支持性需求分析.....24
3.3 本章小结.....24
第 4 章 基于区块链的数字资产存证系统概要设计.....25
4.1 存证系统网络拓扑图.....25
4.2 改进的区块链框图.....25
4.3 数字资产存证系统概要设计.....26
4.3.1 系统底层架构设计.....26
4.3.2 系统总体框架设计.....27
4.3.3 系统总体功能模块.....28
4.4 本章小结.....29
第 5 章 基于区块链的数字资产存证系统详细设计与实现.....30
5.1 资产数据存证系统详细流程图.....30
5.2 区块链存证系统主要功能模块设计与实现.....31
5.2.1 用户管理子系统的设计与实现.....31
5.2.2 文件传输子系统的设计与实现.....32
5.2.2.1 文件加密功能模块的设计与实现.....32
5.2.2.2 文件授权功能模块的设计与实现.....33
5.2.3 数字资产保全子系统的设计与实现.....34
5.2.3.1 智能合约的设计与实现.....34
5.2.3.3 区块生成的设计与实现.....36
5.2.3.3 区块共识的设计与实现.....37
5.2.4 数字资产管理子系统的设计与实现.....39
5.3 本章小结.....40
第 6 章 基于区块链的数字资产存证系统测试.....41
6.1 系统测试环境介绍.....41
6.2 系统测试方案.....42

6.3 系统主要功能测试.....43
6.3.1 文件加密功能测试.....43
6.3.2 文件上传功能测试.....44
6.3.3 文件下载功能测试.....45
6.3.4 数字资产保全功能测试.....46
6.4 本章小结.....49
总结与展望.....50
参考文献.....51
攻读硕士学位期间的主要成果.....53
致谢.....54

摘要

随着实体经济逐步向数字化经济转型升级，大量由个人或企业创造的数字资产开始在网络中流通，方便数据分享的同时也衍生出了很多棘手的问题。像用户隐私泄露、版权纠纷、数据盗用等安全事件时有发生，严重侵扰了人们的正常工作和生活。另外，数字资产天然具有易变性、脆弱性的特点，一旦出现单点故障或病毒感染等状况，很有可能会使其原始状态遭到破坏，从而使其失去应有的价值。而且，一般用户在取证、存证、鉴证等方面意识薄弱，会给不法分子留下可乘之机。而普通存证又存在存证机构少、防篡改能力弱、证据效力低等缺陷，亟需一种新的存证技术取而代之。

本文针在对数字资产的特点做了深入分析和研究之后，基于区块链设计并实现了一种数字资产存证系统。该系统利用区块链去中心、免信任、安全透明、防篡改、集体维护以及可追溯性等特点，结合分布式数据存储、P2P 通信、智能合约、共识机制、密码学、Token 激励等技术手段，把图片、音视频、电子邮箱、电子合同等这一类数字资产以哈希指纹的形式固定下来。另外，IPFS 系统实现了对数字资产的分片和哈希化，以弥补区块链天然存在的存储缺陷；Merkle 树保障了交易数据的完整性和有效性；时间戳技术和链式结构做到了可对历史数据进行回溯。智能合约负责自动执行业务逻辑和程序代码，以削弱人为因素的影响；非对称加密用于加密和授权；PoA 共识算法对事务和区块生成进行验证，以维护整个区块链网络的稳定运行。

经过实验验证，本系统能够实现文件加密、文件上传、文件分片、文件下载、文件查看、证据固定、文件授权、文件比对等功能需求，在完成以上工作的过程中还对智能合约进行了编写、编译和部署和调用。存证用户在 Web 前端可通过 Web3.js 封装的 JSON-RPC API 与 Kovan 测试网络进行交互，在完成以上操作之后即可授权司法鉴定机构获取区块链上的存证数据，通过与原始文件进行对比出具相应的司法鉴定报告。本文为企业或个人数字资产的存证和保全提供了很好的借鉴意义。

关键词：区块链；数字资产；存证；IPFS；哈希指纹

中图法分类号：TP311

ABSTRACT

As the real economy gradually transforms into a digital economy, a large number of digital assets created by individuals or enterprises begin to circulate in the network, which facilitates data sharing and also has many difficult problems. Security incidents such as user privacy disclosure, copyright disputes, data theft, etc. occur frequently, seriously harassing people's normal work and life. In addition, digital assets are naturally characterized by volatility and vulnerability. Once a single point of failure or virus infection occurs, it is likely to destroy its original state, thus losing its value. Moreover, the general user's awareness of evidence collection, deposit verification, and attestation is weak, leaving the criminals with an opportunity. However, the common deposit certificate has the defects of fewer depository institutions, weak anti-tampering ability, and low evidence validity. It is necessary to replace it with a new depository technology.

After in-depth analysis and research on the characteristics of digital assets, this paper designs and implements a digital asset depositing system based on blockchain. The system utilizes blockchain decentralization, trust-free, secure and transparent, tamper-proof, collective maintenance and traceability, combined with distributed data storage, P2P communication, smart contract, consensus mechanism, cryptography, Token incentives and other technical means. Digital assets such as pictures, audio and video, e-mail, and electronic contracts are fixed in the form of hash fingerprints. In addition, the IPFS system implements fragmentation and hashing of digital assets to compensate for the natural storage defects of the blockchain; the Merkle tree guarantees the integrity and validity of the transaction data; the timestamp technology and the chain structure do Backtracking historical data. Smart contracts are responsible for automating business logic and program code to reduce the impact of human factors; asymmetric encryption for encryption and authorization; and PoA consensus algorithm for verifying transaction and block generation to maintain stable operation of the entire blockchain network.

After experimental verification, the system can realize the function requirements of file encryption, file uploading, file fragmentation, file downloading, file viewing, evidence fixation, file authorization, file comparison, etc. In the process of completing the above work, the smart contract is also carried out. Write, compile, deploy, and call. The certificate user can interact with the Kovan test network through the JSON-RPC API encapsulated in the Web3.js on the web front end. After completing the above operations, the judicial authentication

authority can be authorized to obtain the certificate data on the blockchain, and the original document is used. Compare the corresponding judicial appraisal report. This paper provides a good reference for the deposit and preservation of corporate or personal digital assets.

Keywords：Blockchain; Digital assets; Proof of existence; IPFS; Hash fingerprint
Classification: TP311

第 1 章 绪论

1.1 引言

现如今，人类社会正经历着从物理社会向数字社会过渡的转变，数字经济相比于实体经济开始处于略微的优势。在互联网蓬勃发展的这几十年中，信息的传递、万物的互联、便携的移动终端设备把一切和数字化都关联了起来。但在目前这种互不信任、虚拟的网络环境下，各种网络安全事故层出不穷，已经严重影响到人们的正常生活。数字资产是价值传递的本源，很有必要选择一种新的技术架构来保障其在存证方面具有切实可信的法律效率。时下最火的区块链技术可有效保障数字资产在证据固定和保全的安全性，为法院、司法机关、仲裁、检察院、公证处等权威机构直接提供有效证据，提高取证、鉴证的效率[1]。

1.2 课题研究背景和意义

随着大数据时代的到来以及实体经济正逐步向数字经济转变，数字资产俨然已成为人们生活、工作和学习过程中不可或缺的信息载体，典型的代表包括电子文档、电子合同、图片、音频视频、网页、学位证书等。但由于数字资产的数字性和易改无痕性等特点，使互联网版权、互联网合同、电子合同等在内的纠纷案件逐渐增多，相关机构在进行定案量刑的过程中都会涉及到电子证据的提取和存证工作。

2013 年 1 月 1 日，《中华人民共和国民事诉讼法》正式把电子证据纳入法定证据的范畴[2]。但由于像数字资产等一类的电子证据极易发生丢失、缺损或泄露等情况，从而丧失了原有的法律效力，这为相关权威机构证据的提取和判定也增添了很大的难度。鉴于此，亟需一种可信的存证技术来保障数据的完整性不被破坏，快速推进司法区块链在我国的建设。传统存证把需要固定的电子证据存放在中心化的服务器或备份到移动硬盘等电子介质[3]。这就容易因其数据中心掌控在少数管理人员手中，他们负责数据的增添、修改、删除等操作，完全有能力对真实的电子数据进行篡改或独占。除此之外，像单点故障、DDoS 攻击等外部威胁也会使电子证据的功能发生改变，成为不具备法律效力的一般电子数据。因此，开发一种能很好保护数据安全的存证系统是当务之急，新兴的区块链技术也许是解决此类痛点问题的锦囊妙药。

2008 年，一位化名为中本聪（Satoshi Nakamoto）的日裔美国人发表了一篇名为《比特币：一种点对点电子现金系统》的论文[4]。该论文一经问世便引起了学术界巨大的轰动和广泛的探讨。在当时金融危机大爆发的背景下，比特币的诞生解决了金融行业一直

以来存在的“双花”问题，同时也为拜占庭将军问题提供了有效的解决方案。2009 年，创世纪块的诞生标志着比特币系统正式开始运行，而其底层核心技术也被单独抽取出来作为一个新概念（即区块链）广为流传。在接下来的时间里，政府、企业、高校、研究机构等都开始纷纷布局区块链技术，区块链在不仅适合于互联网金融行业，在医疗数据管理、保险、数字知识产权保护等其他领域也有不错的应用案例。

1.3 国内外研究现状

区块链存证主要应用于版权保护[5]、防伪公正[6-7]、数据存储[8-11]、数字资产管理（DAM）[12-15]等场景，以下分别从国内和国外两个角度出发，阐述国内外对区块链存证相关领域的研究现状。目前国内对区块链存证的研究主要集中在版权保护等方面。安妮股份开发并落地了基于 FISCO BCOS 的版权存证平台。链动时代实现了不动产的链上登记功能。阿里云和法大大联合推出基于区块链技术的电子邮件存证产品[16]。该产品利用区块链技术去中心化、链上数据不可篡改、多点维护等特性将电子邮件的元数据（包括数据指纹）同步存储于第三方权威机构，以备司法鉴定机构出证鉴定报告使用。针对普通合同管理系统存在合同防伪码有可能会被恶意篡改、删除和覆盖等问题，高提出基于区块链防伪平台的合同管理系统[17]，该系统由多个认证节点参与数据认证，以保障合同的不可伪造性。朱研究了区块链技术在电商存证鉴证中的应用[18]，通过把电子数据的特征数据、Hash 值和用户账户等关键信息存放到区块链上，实现对重要证据文件的固定，需要时可用来检验文件的一致性。雷从时间戳取证到区块链取证角度出发，对网络著作权纠纷中电子存证的抗辩事由与司法审查进行了详细地分析[19]。李等人结合法官审查电子数据真实性关注的角度，探讨了基于区块链存证的电子数据真实性[20]。徐研究和实现了基于区块链的云取证系统，借助区块链弥补传统电子取证技术和云取证技术的缺陷[21]。杭州、北京两地互联网法院率先采用区块链取证存证技术，对因著作权归属问题引起的侵权纠纷事件进行受理[22]。

在美国，每年仅版权保护就产生超过 1.126 万亿美元的收入，而且版权的注册过程十分缓慢，主要是让大型企业受益[23]。Stem 是音乐领域的一个区块连项目，旨在为那些热爱音乐人提供一个方便发布原创作品的平台，整个过程没有前期成本，确保他们能直接获得报酬。Custos 使用数字水印技术和比特币区块链对数字媒体文件（例如：视频、电子书等）的版权进行保护。Binded 对上传至私人版权库的图片进行 Hash 计算，生成的数据指纹（Hash）会被添加到比特币区块链中，然后上传者可以把收到的版权证书当作证明，以备不时之需。Ascribe 背后的理念是让每个人对自己的作品有更多的控制权创作者可以打破对内容、发行和版权的限制。Chronicled 是第一家利用物联网、人工智能和区块链技术为端到端智能供应链提供解决方案的公司，它提供的服务包括：1）使用数字身份去关联客户的物理产品；2）跟踪用户数据和产品的历史和来源；3）使用智

能合约自动化业务流程。Civic 用来为企业或个人提供控制和保护身份的工具，例如 Civic 安全 ID 等，使用户可以安全访问网站和服务。

一方面，区块链存证相关的研究主要集中在版权保护、公正防伪、审计追踪、数据存储、身份认证和存证鉴证等方面。另一方面，基于区块链的存证应用还存在很多问题：1）区块链存证在国内还不成熟，大多企业仍旧延续传统的中心化管理模式[24]；2）区块链还有很多性能上的问题需要解决，例如吞吐量、延展性等。3）缺乏统一的行业标准，大多数应用尚在探索阶段，理论远比实际应用多。在前人工作基础之上，本文进一步研究基于区块链的存证应用。其中，存证对象面向企业所有的资产信息；各大权威机构参与认证，以保障整个网络的稳定运行。

1.4 课题设计目标

本文针对普通存证存在的存证机构少、防篡改能力弱、证据效力低等问题，结合区块链去中心化、去信任化、不可篡改、集体维护、可追溯性等特点，实现一个组织可动态加入、身份得到验证、重要功能可以实现的基于区块链的可信数字资产存证系统。该系统服务于数字资产从产生、存储再到利用的整个过程。该平台综合运用了多种技术，包括使用 IPFS 提取和挂载数据，利用区块链的链式结构存储和保护交易数据，web.js 负责连接客户端和区块链网络，是两者之间进行通信和交互；智能合约提供业务上的逻辑和程序代码的支持，调用它可自动化处理一些相关的事务，返回结果由 PoA 共识机制筛选出的记账节点负责打包入链；非对加密技术对用户身份进行签名和授权，保证链上数据的存储安全和传输安全。

1.5 论文内容安排

本文的内容安排如下：

第一章是绪论。包括课题研究背景和意义、相关国内外现状、系统设计目标等。

第二章介绍了相关概念和技术。相关概念包括区块链、数字资产、数字资产存证和电子数据存证的异同、IPFS；涉及到的区块链核心技术分别是 P2P 网络、共识机制、智能合约以及密码算法。密码算法又包括非对称加密、Hash 算法、数字签名以及 Merkle 树等。

第三章是数字资产存证系统的需求分析。分别从功能需求分析和非功能需求分析两个方面出发对系统的功能实现和性能要求进行了阐述。

第四章是基于区块链的数字资产存证系统概要设计。涵盖了系统底层架构设计、系统整体架构设计以及系统总体功能结构设计。

第五章给出了基于区块链的数字资产存证系统的详细设计与实现。对系统的整体运行流程进行了设计，然后对不同功能模块进行了设计与实现。

第六章是系统测试。对需求阶段提到的功能代码实现之后再进行测试，验证系统的各项功能指标都可以按照预期顺利完成。

最后对论文的内容进行了总结和展望。

1.6 本章小结

本章从理论基础出发先后介绍了本课题的研究背景和意义、国内外研究现状、设计目标以及内容安排等。目前，人们的数字资产保全意识还很薄弱，而数字资产作为价值的存储介质又无时无刻都在面临着被篡改的风险。区块链作为一门能够促使人类进行大规模协作的新技术，完美地契合了普通存证一直以来存在的痛点问题。

第 2 章 相关概念和技术

2.1 区块链的基本概念

区块链（Blockchain）本质上一种去中心化、公开透明、难以篡改的文档型数据库，是一种促进人类大规模协作的技术手段，解决了多点之间相互信任以及利益分配的问题[25]。区块链技术并非是一门新技术，而是分布式数据存储、P2P 通信、密码学、智能合约以及共识机制等计算机技术的综合集成创新[26]，其创新之处是添加了 Token 激励的经济因素，使得整个区块链系统得以正常运行。区块链技术俨然已成为各个行业互相角逐的重要技能，目前已在金融、教育、存证、溯源等领域大放异彩。

图 2-1 展示了区块链的数据结构，此账本由两部分组成，分别是数据“区块（Block）”和“链（Chain）”。区块又包括“区块头”和“区块体”两部分，其中区块头负责实现区块链的重要功能，每一个区块都包含有上一区块的 Hash 值，然后与本区块 Hash 值串联在一起形成一个“链条”。区块体是存放交易数据的地方，这些交易数据被默克尔（Merkle）树组织在一起，从叶子节点开始逐层往上哈希，并最终形成 Tx_Root 用于对整棵树进行签名，通过这种方式，可有效鉴别区块数据的真伪。最终经过检验并确认的区块会被盖上时间戳，方便后续对历史数据进行追溯。

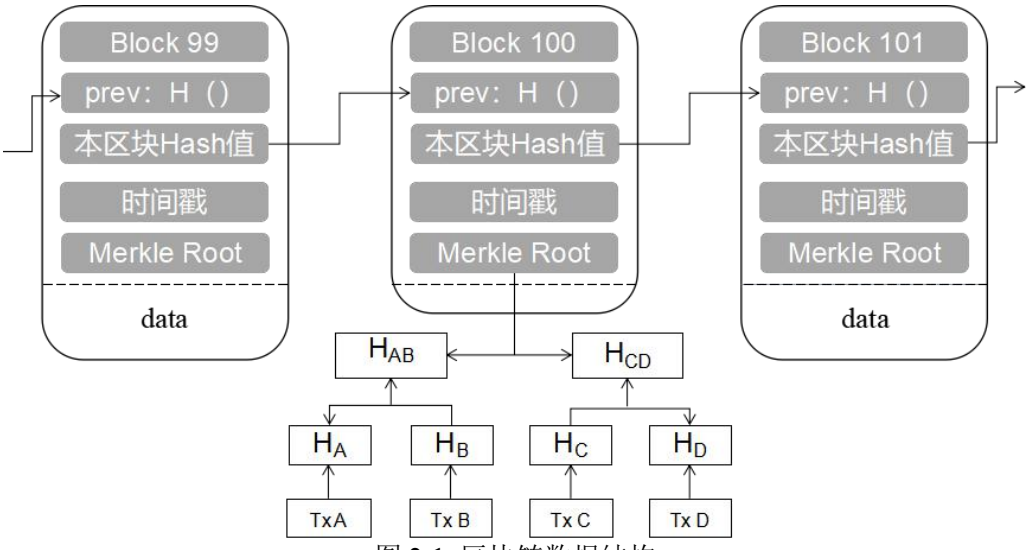

图 2-1 区块链数据结构

目前一些价值局域网已经在逐步形成，主流的区块链平台如比特币、以太坊[27]、Fabric[28]等，已应用于数字资产、贸易金融、股权债券、供应链溯源、联合征信、公示公证、物联网共享、数据安全等领域，并形成各自的公有链、联盟链或私有链，构建了相关行业的信用价值网络。以比特币为主的加密货币是区块链技术应用的原型，也是基于区块链的第一个应用，更是目前为止最成功的一个应用。但因其只能实现简单的挖矿、

转账和查询操作，因此应用领域有限。以太坊挣脱了加密货币的枷锁，它在原有区块链的基础上又新添加了一个虚拟机，为编译好的智能合约提供运行环境，以完成更复杂的指令，延伸区块链的功能[29]。Fabric 是 Hyperledger 的一个分支项目，也是企业级联盟链的代表。图 2 展示了主流区块链平台特点。

表 2-1 主流区块链平台

平台	模型	内置代币	共识机制	图灵完备	执行环境	账户设计
比特币	公有链	有	PoW	否	内置脚本引擎	UTXO
以太坊	公有链	有	PoW+PoS	是	EVM	账户模型
Fabric	联盟链	无	PBFT	是	Docker	账户模型

根据组成节点类型或者权限的不同，区块链可划分为公有链（Public blockchains）、联盟链（Consortium blockchains）和私有链（Private blockchains）。公有链顾名思义是一种开放性的区块链网路，所有节点均可自由加入和退出，因此具有很好的开发性和透明性，但同样也面临着监管难、耗损高、处理速度慢等缺陷。公有链的典型代表有以太坊、EOS[30]等，它们使用的共识算法分别是 PoS-PoW 混合（即 PoA）、DPoS，以太坊是一种去中心化应用（DApp）开发平台，于 2014 年由 Vitalik Buterin 提出；EOS 被定位为区块链操作系统，由 BM 创造，其前身是 Bitshares 和 Steemit。联盟链的权限所属是企业与企业（B-B）之间，节点及节点发出的消息必须经过验证才能给予通过。Fabric 是目前最受欢迎的企业级联盟链，它前期使用的共识算法是 PBFT，目前已发展到类 BFT 和 Kafka。私有链的访问权限完全归属于个人或一个组织内部，因此它的隐私性较好。私有链所选择的共识算法包括 RAFT 和 Paxos 等，前者是后者的简化版。

2.2 数字资产存证应用

2.2.1 数字资产的含义

数字资产（Digital Assets）是指企业或机构在生产、管理或经营过程中累积的具有一定价值的各种数字化内容和信息[31]，例如：图像、音视频文件、网页截屏、电子邮件、积分等。这些数字化内容和信息贯穿于企业的整个运作过程，有些甚至被当作产品用于出售。数字资产的这些特殊性决定了其安全必定要受到保护。近年来版权纠纷、数据被盗、恶意篡改等事件给企业造成了很大的困扰，究其原因，与其存证意识淡薄是密不可分的。而普通存证只是将数字资产以备份的形式存放在中心化的服务器，这种方式具有存证机构少、防篡改能力弱、证据效力低等缺点。按照法律规定，单一或部分电子证据远远不能支撑整个案件的判定[32]，所以亟需一种新的存证方式打破传统壁垒。基于区块链的可信数字资产存证系统对以上问题的解决提供了一种思路，该系统的底层架构整合

了哈希加密、共识机制以及去中心化存储等多种技术，其中共识机制和中心化存储是普通存证区别于区块链存证最明显的标志。

2.2.2 数字资产存证和电子数据存证的关系

电子数据（Electronic data）是我们日常生活和工作中最常接触的一种数据存在形式，它是指借助于计算机应用、通信和现代信息技术等手段形成的包括文字、图像、电子视频等电子信息资料[33]。数字资产和电子数据的相同点是：两者都是经由现代化电子设备采集、存储、共享以及接收的材料；不同的是数字资产是具有价值的，例如加密货币、电子合同、电子邮箱等。而电子数据则不一定全部都有价值，但却涵盖了所有的数字资产，它们是包含与被包含的关系。

数字资产存证只针对于有价值的数据，因此其存证的内容往往是用户特别关心的证据信息。电子数据存证则表现的更加宽泛，适合所有包括有价和无价的数字信息。数字资产和电子数据经过技术加工和权威机构验证便可成为具有法律效力的电子证据，这是为了进一步的证据保全[34]。数字资产保全和电子数据保全的原理相同，即对在一个时间点内生成的文件内容进行加密固化，生成与之相对应的唯一 Hash 值（数字指纹），该 Hash 值可确保数据的完整性、有效性和真实性，为将来可能产生的争议和纠纷等诉讼案件提供强有力的原始证据，有效保障用户的法律权益和经济权益不受侵害。

2.3 IPFS

2.3.1 IPFS 的基本概念

IPFS（InterPlanetary File System，星际文件系统）是一种永久的、共享文件的、分布式的网络传输协议，最初由 Juan Benet 设计，为 Protocal Labs 所拥有[35]。另外，IPFS 还是 DHT（Distributed Hash Table，分布式哈希表）、Git 版本控制系统和 Bittorrent 的综合集成创新，而这些互联网技术在各自领域都能表现出了良好的性能。IPFS 基于内容的寻址方式可有效缓解传统信息系统过度负载的情况，使访问分散到网络中的其他参与节点，以减少对文件托管中心的依赖。IPFS 本质上是一个用于检索和共享 IPFS 对象的 P2P 系统，IPFS 对象是包含数据（Data）和链接（Link）的数据结构，而链接又包括 Link 的名字（Name）、Link 指向的 IPFS 对象的 Hash 以及 Size。所有 IPFS 对象的集合构成了密码认证的 Merkle DAG（Directed Acyclic Graph，有向无环图）[36]。值得注意的是，区块链也是一种天然的有向无环图，其后续区块总是保存着前一区块的 Hash 值，通过这种方式使前后呼应，达到一种不可轻易攻破的状态。以太坊区块链使用另一种更为复杂的 Merkle-Patricia 树结构，区块链在 IPFS 中的建模如图 2-2 所示。

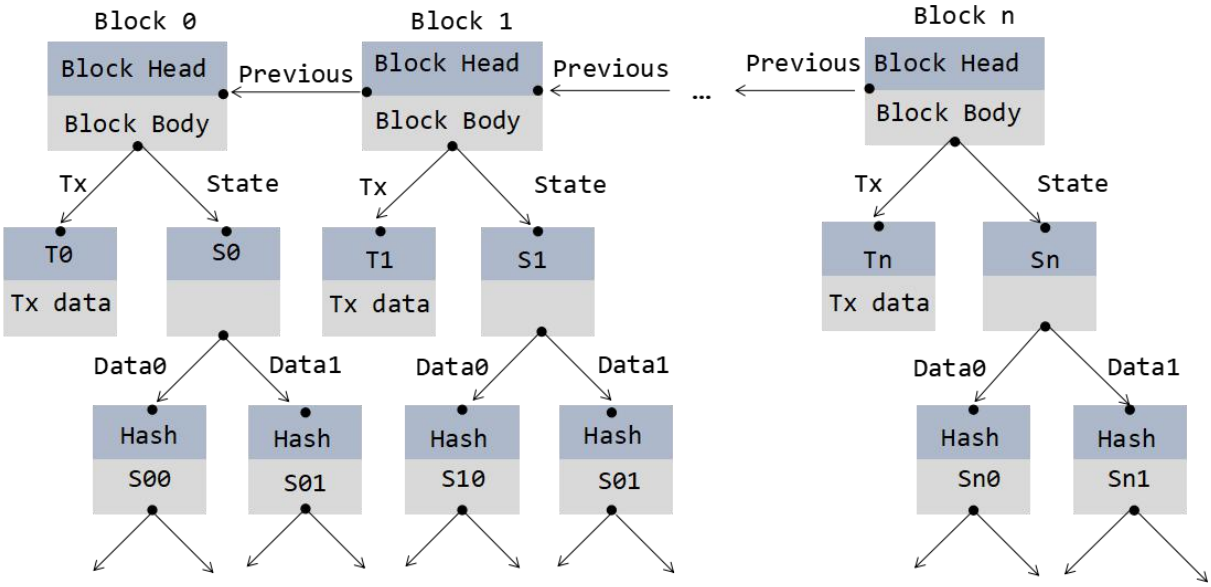

图 2-2 区块链在 IPFS 中的模型

IPFS 被称为是下一代 Web 协议，而目前我们所使用的互联网协议是 HTTP 协议（超文本传输协议）。该协议发展至今，其地位已很难撼动，它遵循着一种固定的 Client-Server（客户端/服务器）模型。该模型按照服务器的 IP 地址创建一个标识符（URL），在网络环境良好的情况下（不发生离线、单点故障[36]、DDoS 攻击等）向其他客户端提供所有负载。这种基于位置（IP）寻址的方式受制于目标服务器，很容易导致服务中断、数据丢失或被篡改等问题的发生。综上，亟需一种新的数据传输方式，来弱化甚至是取代传统中心化管理模式的统治地位。IPFS 协议或许是扭转局面最好的选择，未来我们将以“ipfs://文件 Hash”（目前是 http://127.0.0.1:8080/ipfs/文件 Hash）的方式访问文件，同时伴以数字钱包插件留作交易使用，这可看作是未来浏览器的雏形。

2.3.2 IPFS 的工作原理

IPFS 的工作原理如下：

（1）上传至 IPFS 系统的每个文件以及文件碎片（块）会被赋予一个称之为加密哈希（Crytographic hash）的唯一指纹。

（2）IPFS 通过网络溯源每个文件的历史版本，移除重复文件，减少冗余。

（3）所有参与节点只存储和它相关的内容，包括一些索引信息，有助于厘清每个节点都存储了什么。

（4）查找文件的时候，只需通过文件的哈希值，便能追踪到存储文件的节点及唯一指纹映射的内容。

（5）使用 IPNS（分布式命名系统，Decentralized naming system）为每个文件提供一种可读的（Human-readable）命名规范[37]。

2.4 区块链核心技术

2.4.1 P2P 网络

P2P 网络（Peer-to-Peer networking，对等网络）是区块链技术中又一个比较重要的关键技术，它使用 UDP（User Datagram Protocol）协议来促成节点之间的通信[]。相比于传统面向连接的 TCP 协议，UDP 协议是一种无须连接的协议，即节点之间不需要经过中间实体便可实现点对点通信，这是 P2P 通信技术最通俗的解释。详细的 P2P 网络通信原理如下：网络中的所有参与节点共享各自所拥有的部分硬件资源（包括存储空间、计算能力、打印机等），当某个节点需要访问时便可通过共享资源提供的服务和内容实现，从而绕开了中心化服务器的控制，实现了客户机即是服务器、服务器即客户机的双向效应。图 2-3、图 2-4 分别展示了中心服务器模式和 P2P 网络模式，可清晰地观察到两者的区别：一个是中心化的管理模式，另外一个是去中心化的管理模式。

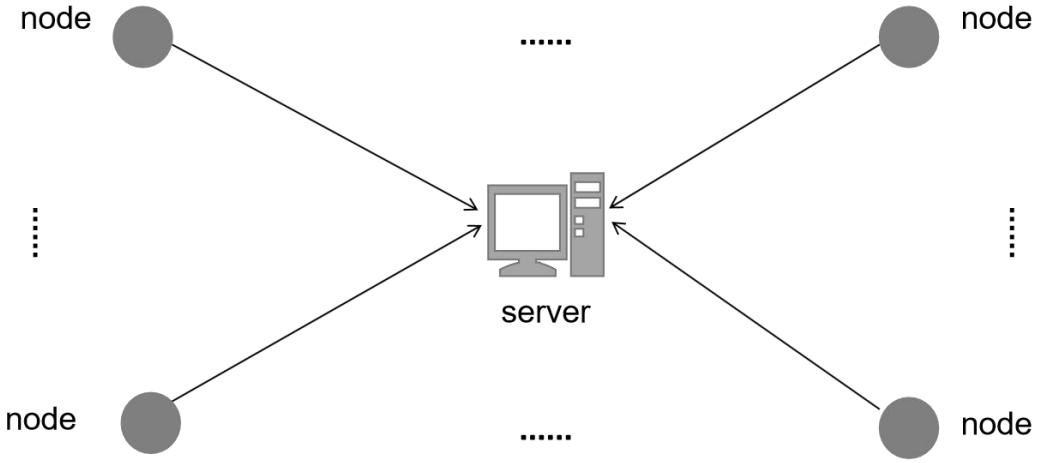

图 2-3 中心服务器模式

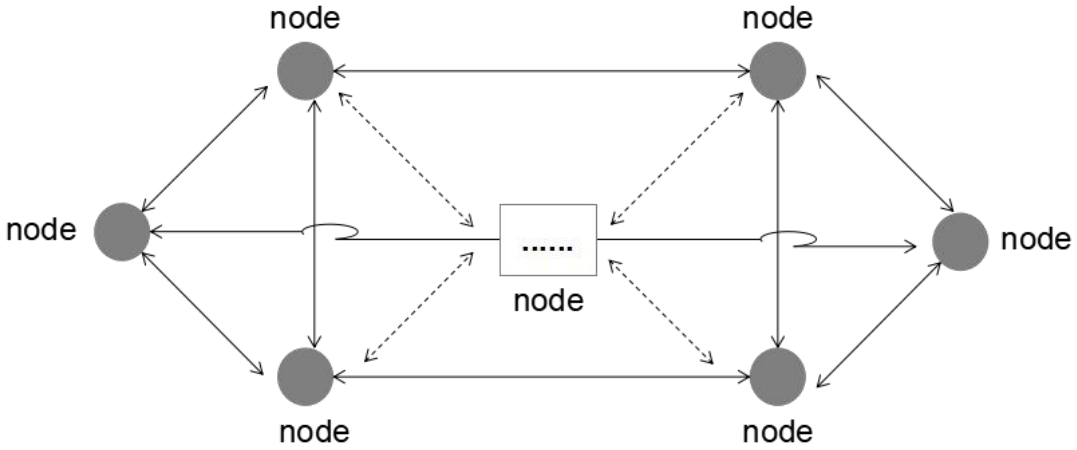

图 2-4 P2P 网络模式

P2P 网络除了去中心化之外，还具有可扩展性、健壮性、高性价比、隐私保护和负载均衡等特点，从而使其能够在更多领域中应用和推广。区块链技术也正是利用 P2P 网

络的这些优势完美地与其他节点进行数据同步，进而建立一种分布式的、永不宕机的世界性计算机。

2.4.2 共识机制

共识机制（Consensus Mechanism）的由来和分布式系统有着紧密的联系，是为了解决分布式网络中节点之间信息不对称、单点故障、数据丢失的问题，具有可插拔、可验证以及防攻击性等特点[39]。目前广为人们熟知并应用的共识算法包括 Clique PoA（授权证明）、PoW（工作量证明）、PoS（权益证明，Proof-of-Stake）、DPoS（股份授权证明，Delegated Proof-of-Stake）、PBFT（实用拜占庭容错）、IBFT、PAXOS、RAFT 等，其中，前三者基于证明，而后三者基于投票。基于证明的共识机制是目前公链最为常用的一种机制，一般具有以下共性：证明通过广播实现，所有节点参与验证（验证易但求解难）；如果证明不通过，就放弃原始证明；对证明无误的交易进行确认。基于投票的共识机制所表现出的特点包括：封闭的投票范围、被联编的投票者、孤立的投票事项以及投票消耗流量与安全的正比例关系[40]。未来，共识机制的演进趋势将会向从多到少、从链到图、从确定到随机的方向发展。从多到少代表的是参与节点的数量；从链到图是指共识的呈现方式；从确定到随机是指参与共识节点的选择机制。

2.4.2.1 共识机制的作用

通过共识算法筛选出记账节点，保证同一个区块只能有一个节点产生并广播，其他人只能充当验证和同步副本作用，而不能对其进行任何的更改。因为系统会自动比较，会认为相同数量最多的区块是原始区块，少部分和别人数量不一样的区块是被篡改过的区块。在这种情况下，如果想要篡改某个区块上的数据，需要同时做到以下两点：第一，需要重算当前区块及之后的所有区块。第二，需要将自己的区块链同步到整个区块链网络上的大部分节点。但是因为区块链上的节点会源源不断的产生，而且整个区块链网络上有成千成万的节点，算到最后一个区块并且在下一个区块产生之前修改整个网络中的大部分节点，这要求你的节点的算力远超其他节点的算力之和。

2.4.2.2 PoW 共识的基本流程

PoW（Proof-of-Work，工作量证明）早在比特币出现之前就已经存在，区块链所使用的 PoW 共识机制正是借鉴于此。区块链的 PoW 共识机制包含有挖矿者（Miner）和验证者两个角色。对于挖矿者，它需要完成数学难题的解答，即通过不断变换随机数（Nounce）更改区块头，然后对区块头进行双哈希计算，求得的最终结果如果比目标值（Target value）小则挖矿成功。对于验证者，它的任务是检验矿工是否真的求解出了难题。图 2-5 显示了 PoW 共识的基本流程，其中的难度值（Difficulty value）是限制出块

速度的一个可调节变量，例如比特币的出块速度被控制在每 10 分钟一个区块[41]。值得注意的是，上一区块 Hash、Merkle 根 Hash、难度值、时间戳以及版本都是固定的，只有随机数是可以变换的。通过这种方式，矿工竞相寻找能够使目标值满足前多少位为 0 的随机数，一旦经计算得到的结果小于目标值则表明挖矿成功，否则继续更改随机数直至符合条件为止。挖到矿的节点会把区块头广播至区块链网络的其他节点，其他的节点会再次对该区块头做一次 SHA(SHA(记账节点挖到的区块头))，如果结果一样，则证明该区块确实是有效的，否则抛弃这个区块进行下一轮的记账权争夺。

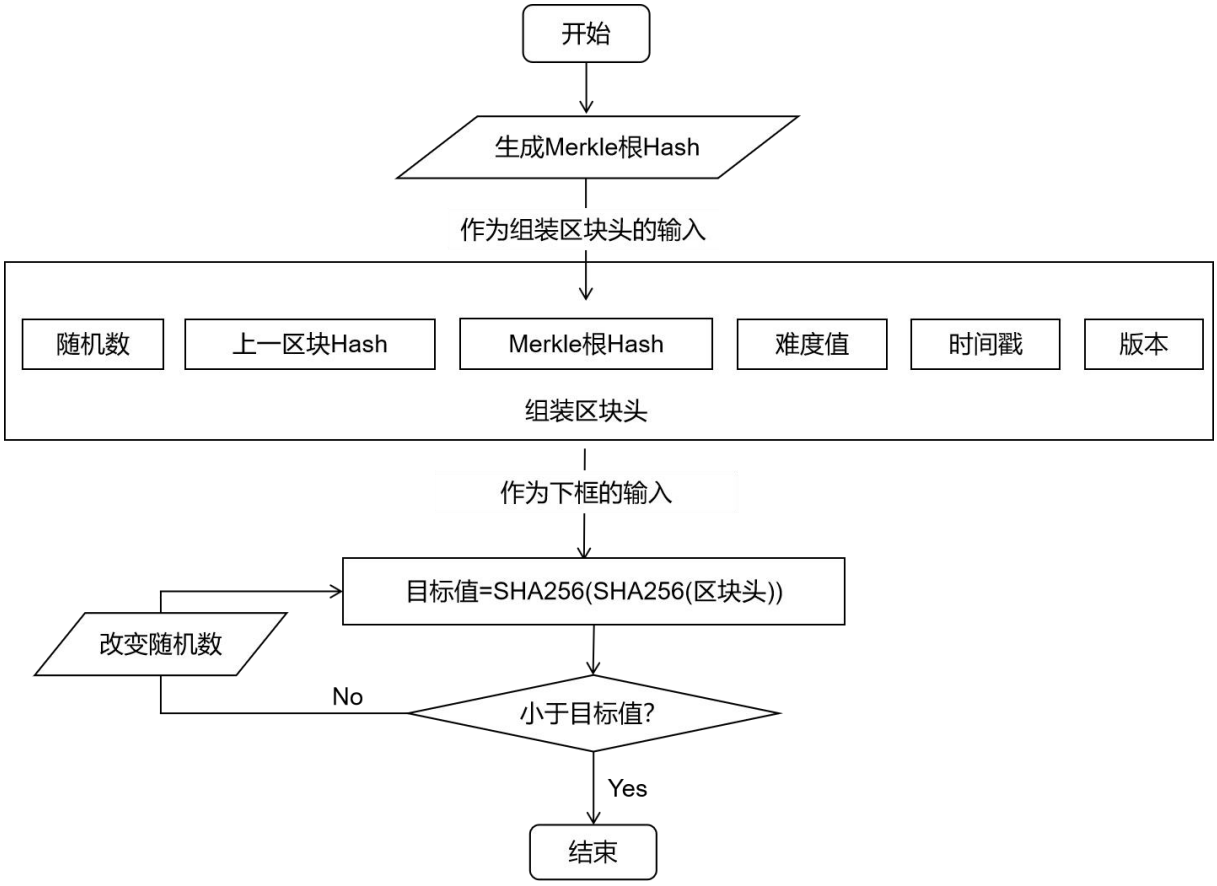

图 2-5 PoW 共识的基本流程

2.4.2.3 PBFT 共识的基本流程

PBFT（Practical Byzantine Fault Tolerance，实用拜占庭容错）属于 BFT（拜占庭容错）算法的一种，由 Castro 和 Liskov 于 1999 年提出，是 Fabric 0.6 版本所采用的共识算法[42]。和 PoW 共识算法所不同的是 PBFT 共识算法无须通过代币奖励的方式来维持整个区块链网络的稳定运行，而且它还有一个比较显著的特点：具有很高的容错性，即网络中有 3f+1 个节点，其中 f 代表失效节点的个数，在这种情况下网络仍然可以正常运行，换句话说，该网络可以容忍 f/3f+1 的出错率[43]。图 2-6 展示了 PBFT 共识的基本流程。其中，预准备（Pre-prepare）、准备（Prepare）和提交（Commit）是确保共识结果

能够达到一致性标准的三个阶段，C 代表客户端，Ⅰ、Ⅱ、Ⅲ、Ⅳ为服务端，而且Ⅳ为已经宕机的服务端。具体的 PBFT 算法步骤如下：

（1）选择一个服务端（这里是 I）作为请求的接收方，该服务端也被称为主节点，其他服务端充作备份节点。

（2）客户端向 I 发出请求，I 转而将请求广播给Ⅱ、Ⅲ、Ⅳ等其他备份节点。

（3）Ⅱ、Ⅲ、Ⅳ收到 I 转发的请求后分别按照Ⅱ->I、Ⅲ、Ⅳ，Ⅲ->I、Ⅱ、Ⅳ的路径继续广播，而Ⅳ因为宕机的原因，所以不能转播。

（4）若 I、Ⅱ、Ⅲ、Ⅳ在准备阶段都能收到超过一定数量相同的请求则转而进入到提交阶段，提交阶段的 I、Ⅱ、Ⅲ、Ⅳ如果也能收到超过一定数量相同的请求便把结果反馈给客户端。

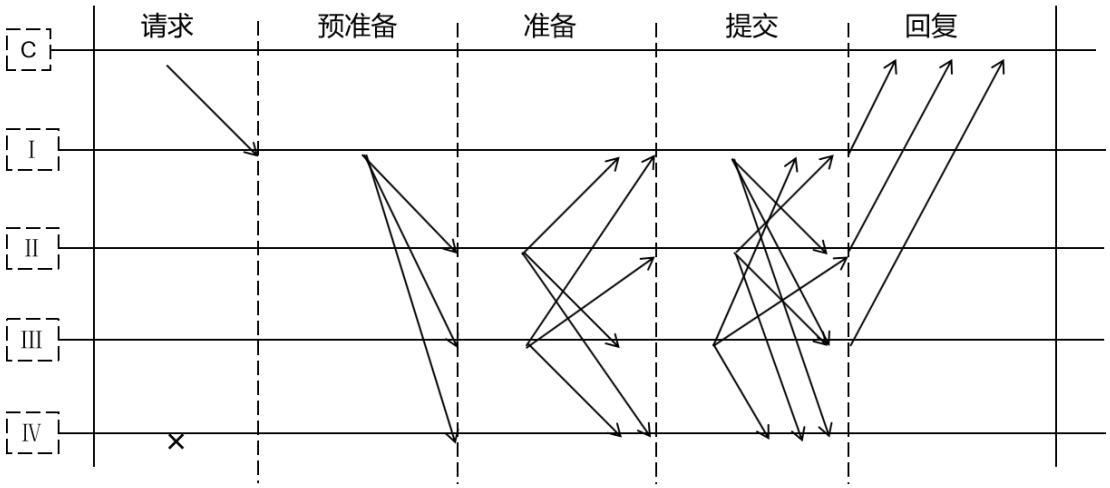

图 2-6 PBFT 共识的基本流程

2.4.2.4 RAFT 共识的基本流程

相比于 Paxos 共识算法十分的复杂，RAFT 共识算法就显得非常的易于理解，图 2-7 展示了 RAFT 共识的基本流程。Follower（追随者）、Candidate（候选者）、Leader（领导者）是网络集群中服务器所显示的不同状态，同一台服务器可以在这三种状态中灵活转变。通常来说，Follower 所代表的是一般情况下的服务器状态，其主要作用除了参与 Leader 的竞选之外，还会同步备份由 Leader 发送的日志。Candidate 是 Follower 在投票选主的过程中所处的中间服务器状态，一旦 Candidate 得到超过 n/2 以上的票数，则会成功当选为新一期的 Leader。具体的 RAFT 算法步骤如下：

（1）初始化每个 Follower 节点，Follower 节点都有成为候选人的资格，一旦开始选主，所有 Follower 进入 Candidate 状态。

（2）统计每个候选节点的得票数，遵从“少数服从多数”的原则，只有票数超过一半以上的候选节点才能成为最终的 Leader 节点。

（3）成为 Leader 的节点会周期性地向 Follower 节点发送心跳，确认 Leader 节点是“活着”的同时将日志文件同步复制给集群中的备份节点。

（4）备份节点对日志内容进行确认，无误之后保存到本地。

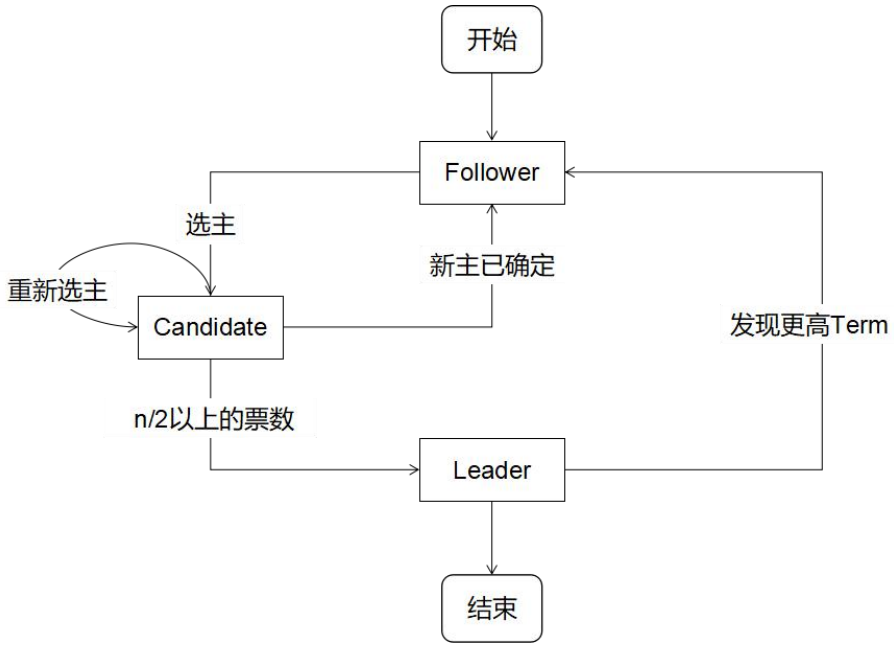

图 2-7 RAFT 共识的基本流程

2.4.3 智能合约

2.4.3.1 智能合约的概念

智能合约（Smart Contracts）的概念首先由尼克·萨博于 1995 年提出，但因为缺乏可信的执行环境而长期处于搁置状态，直到 2014 年 Vitalik Buterin 发明了以太坊才为人所知。它是一种旨在以信息化方式传播、验证或执行合同的计算机协议，能够使设备在不受干扰的环境下自动执行各种交易并解决其身份认证问题[44]。目前智能合约的应用十分广泛，涉及到的领域包括金融、医疗保健、物联网、供应链等，例如：DTCC 与 IBM、Aconi 及 R3CEV 合作，利用智能合约促进信用违约互换[45]；PokitDok 和 Intel 公司推出的 Dokchain 在确认交易双方身份后，他们便可按照预先达成的合约快速交易[46]。图 2-8 展示了智能合约的模型，该模型展示了从合约状态到合约值、再从合约值到数据上链的整个流程。

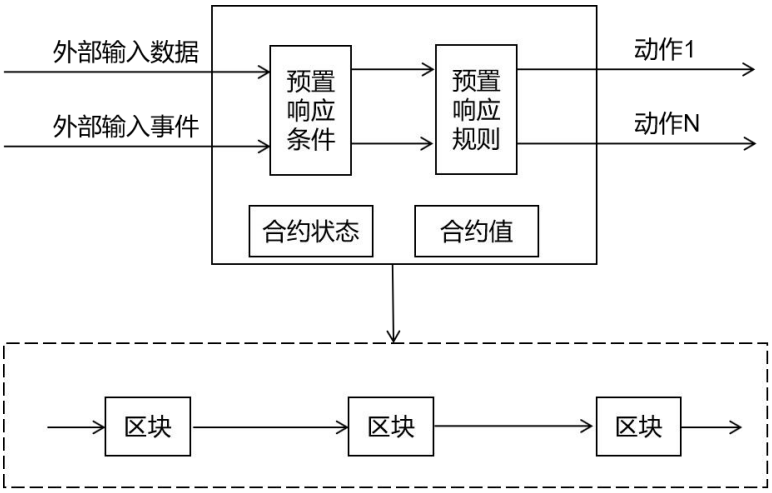

图 2-8 智能合约模型

2.4.3.2 智能合约的请求调用流程

Solidity 是智能合约的开发语言，编写好的.sol 文件经过编译便以字节码的形式部署在区块链上，并由以太坊沙盒 EVM（以太坊虚拟机，Ethereum Virtual Machine，类似于 JVM）执行。Remix web-IDE 是智能合约的在线编译器，也可以对它进行部署，前提是像 Metamask 等钱包账户里必须有足够的余额，而且区块链网络环境已完成搭建或使用现成的由区块链开发平台提供的测试网络。图 2-9 展示了智能合约的请求调用流程，这是开发去中心化应用程序所必须完成的一个过程。

（1） 把智能合约（.sol 格式）编译成虚拟机能识别的字节码（Bytecode）形式；

（2） 将编译好的智能合约部署在区块链网络上；

（3） 用户在系统界面通过 Web3.js 库封装的 JSON-RPC API 调用已完成部署的智能合约。

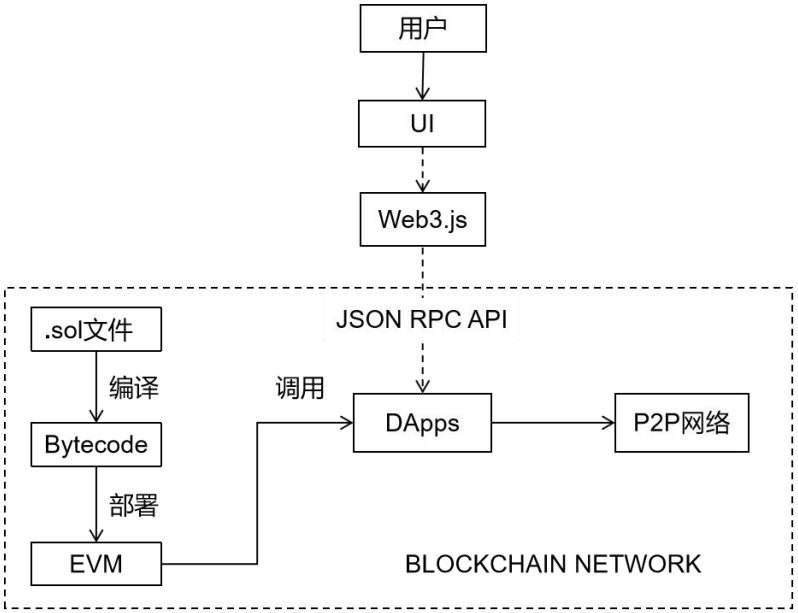

图 2-9 智能合约的请求调用流程

2.4.4 密码学相关技术

2.4.4.1 非对称加密

1976 年 Diffe 和 Hellman 两人共同发表了一篇名为《密码学的新方向》的论文，该论文一经刊出便在学术界引起了巨大的轰动，奠定了未来几十年密码学飞速发展的基础，也对后来比特币和区块链技术的诞生奠定了基础。数字加密技术、椭圆曲线加密算法以及哈希计算都属于密码学的范畴，而数字加密技术又包括对称加密（symmetric encryption）和非对称加密（asymmetric encryption）两种技术。相比于对称加密使用同一种密钥加密和解密，非对称加密技术则是选择不同的密钥参与数字加密。这两种不同的密钥分别是公钥和私钥，具有的特性是：公钥加密的数据只有私钥才能解开，同样私钥加密的数据也只有公钥才能解开；公钥是可以公开的，而私钥只能由用户本人保管。区块链同样也是采用非对称加密技术来保障数据的完整性和不可篡改性，例如比特币系统通过 Secp256k1 这种椭圆曲线加密算法对 256 位随机数（私钥）进行数学计算，最终得到相对应的非对称特殊值（公钥），并基于公钥转换生成比特币交易所用的地址[47]。图 2-10 展示了比特币中公钥、私钥以及地址的关系：

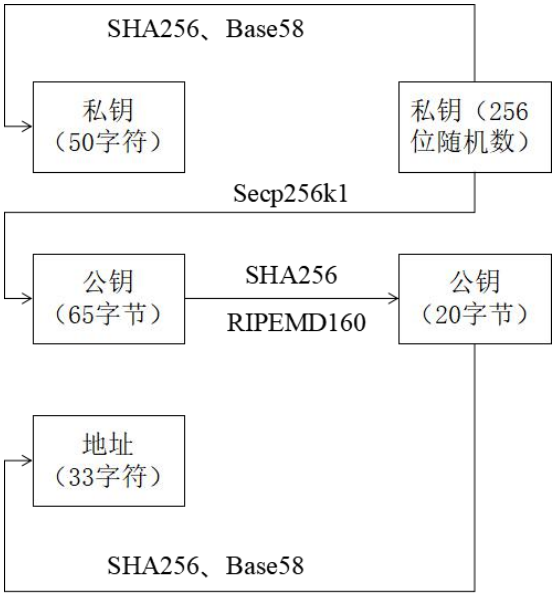

图 2-10 公钥、私钥和地址之间的关系

2.4.4.2 哈希算法

哈希算法（Hash Algorithm）又称散列算法、杂凑算法，是一种基于任意文件创造小数字（指纹）的方法，即以较短的信息概括文件所有的内容，是保证文件唯一性的方法或手段[48]。哈希算法是一种无处不在的安全性保证，从密码存储到文件验证系统，到处都能看到它们的身影，它的基本原理是：使用一个输入有效得到一个与之相对应的、

唯一确定的、固定长度的字符串。也就是说，即使是一字符的改变也会导致“雪崩效应”，输出不同的结果。这种确定性算法是哈希的精髓所在，但也面临着碰撞、生日攻击、量子攻击等诸多难题。所谓碰撞（Collisions）是指不同的输入会得到同样的输出，这是哈希算法绝不允许的操作；生日攻击只针对于输出长度短、操作简单的哈希算法，例如 MD5 哈希。量子计算机依靠其超快的计算速度，被认为是未来最有可能破解哈希算法的工具[49]。值得一提的是，比特币使用的是双重 SHA-256 算法，而以太坊则使用经过 KECCAK256 修改后的 SHA3 算法。因为之前的 SHA1、SHA2 算法已不太安全，以太坊 2.0 分片技术采用的是效率更高的 BLAKE2b 哈希算法。总而言之，哈希算法未来要继续提高自身的复杂性和抗攻击性，以应对各种挑战。

2.4.4.3 数字签名

数字签名（Digital Signature）是一种认证机制，它使得消息的发送方对该消息添加一个起签名作用的码字，消息的接受方在收到消息后会对上面的码字进行确认，以核实该消息所属确实是消息的发送方。值得注意的是，这与传统意义上的用笔签名不同，数字签使用的是非对称加密中的私钥。数字签名是区块链去信任化特性的技术支持，有了签名功能比特币等数字资产才能在匿名条件下进行交易。另外，多重签名是一种更为复杂的技术，它是数字签名的升级版，其原理可简答概述为：对一笔数字资产的使用需要加盖多个签章才能生效。总的来说，数字签名为我们提供了几个比较关键的功能和性质：完备性、可验证性和不可伪造性。完备性可理解为一个私钥对应一个签名；可验证性提供了消息发送方身份的认证功能；不可伪造性涵盖了不可篡改性和不可否认性两方面：不可篡改性保证了签发内容的安全，不可否认的意思是指签发人不能否认自己签过名的消息。图 2-11 展示了数字签名的详细流程，即：

（1）发送者通过哈希算法生成电子合同的摘要（哈希值），然后使用自己的私钥对该 Hash 值进行数字签名；

（2）发送方使用接受者的公钥对电子合同进行加密；

（3）加密密文和签名密文打包发送给接受者；

（4）接收方使用发送方公钥对签名密文进行解密，得到一个 Hash 值Ⅰ；

（5）接受方使用自己的私钥对加密密文进行解密，得到原始数据 Hello!，对该数据进行 Hash 运算，得到一个 Hash 值Ⅱ；

（6）对 HashⅠ和 HashⅡ进行比较，如果这两个值相同则证明电子合同是完整的，否则电子合同很有可能已被篡改。

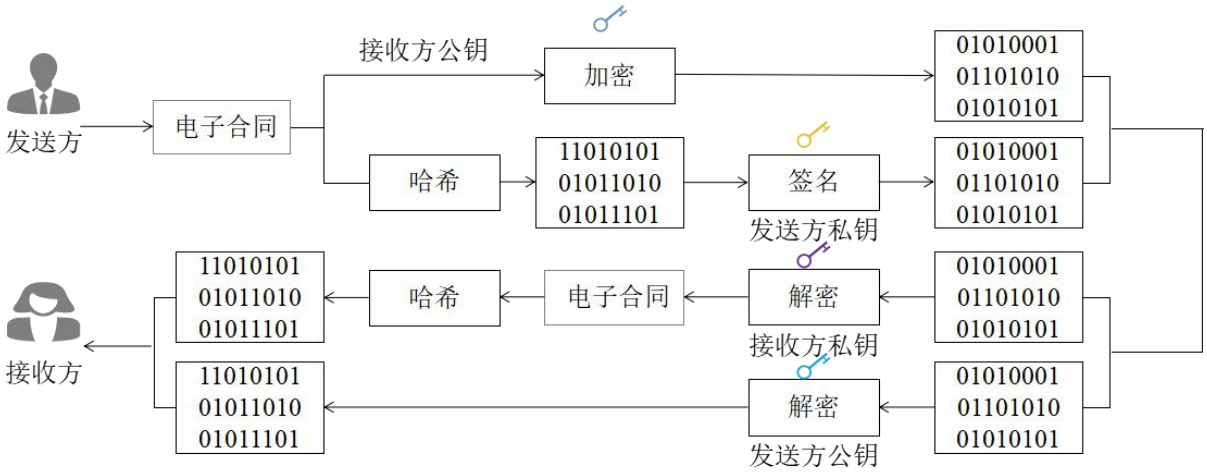

图 2-11 安全认证流程

2.4.4.4 Merkle Tree

默克尔树（Merkle Tree）也被叫作哈希树（Hash Tree），由著名计算机科学家 Ralph Merkle 提出，是一种用于存储 Hash 值的二叉树或多叉树，因为它的叶子节点、中间节点和根节点都是由 Hash 值组成。图 2-12 展示了 Merkle Tree 的基本构造，从中我们可以观察到：Merkle Tree 的叶子节点（N0、N1、N2、N3）分别是数据块（Data Blocks）经过 Hash 运算之后得到的一个 Hash 值，而这些数据块可以是交易信息、文本文档，也可以是诸如图片、音频、视频等此类的大文件，但需要排除区块链这种特殊的分布式记账簿，因为它的存储空间有限，大概每个区块只能存放 1MB 的数据内容。N4、N5 是由相邻叶子节点 N0、N1 和 N2、N3 组成的字符串进行 Hash 运算之后得到的中间节点。同理，N6 的顶端 Hash 值是由 N4、N5 对应字符串组合到一块 Hash 得到的根节点。

默克尔树的特殊结构决定了其防篡改能力非常强，任意一个数据区块中的内容发生变化，哪怕是一字节的修改，也会导致最终 Top Hash 的不同。区块链正是利用默克尔树这种根 Hash 的唯一确定性来保障链上数据的安全和可靠。除此之外，默克尔树在区块链中还有简单支付验证（Simple Pay Verify, SPV）的作用。相比于需要矿工去完成的交易验证，SPV 的任务是对每笔交易是否存在进行确认，过程如下：仍然使用图 2-7 所示例子，假如我们需要证明 Data 1 确实存在，依据默克尔树逐层往上计算的特点，只需要知道邻节点 N0 和 N5 的 Hash 值，然后从下往上依次散列便可得到根 Hash，再与区块头中的 Hash 进行对比，如果相同则代表交易确实存在，否则数据的安全性和完整性已遭到破坏。除了以比特币、以太坊为代表的区块链会应用到 Merkle tree，其他领域像零知识证明、P2P 下载也会用到。

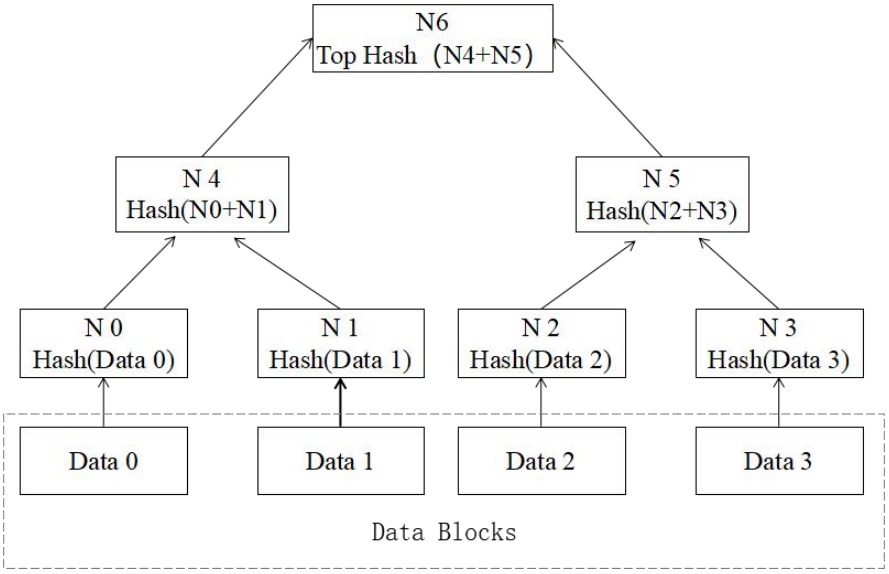

图 2-12 默克尔树

2.5 本章小结

本章节首先从区块链的概念、数据结构、分类以及主流平台这四个角度出发，对区块链技术进行了概括。然后解析了电子数据和数字资产两者之间存在的差异、IPFS 的含义。紧接着依次阐述了 P2P 网络、共识机制、智能合约、密码学等区块链的几个核心技术。非对称加密、Hash 运算、数字签名以及 Merkle 树都是密码学的范畴，用来保障数字资产的存储安全和传输安全。下一节将给出区块链存证系统的需求分析。

第 3 章 基于区块链的数字资产存证系统需求分析

基于区块链的数字资产存证系统主要任务是结合分布式账本技术（DLT，Distributed ledger technology）、IPFS 分布式文件分拣协议、RSA 公钥加密算法、PoA 共识算法等其他技术，为存证用户提供一个可信的、具备数字资产存证与鉴权能力的应用平台。该平台的服务对象可以是企业，也可以是个人，服务内容包括文件加密、文件上传、文件查看、文件下载、证据保全、文件授权、核实真伪以及可信凭证等。

3.1 系统功能需求分析

该系统针对日益严峻的版权纠纷、隐私泄露、电信诈骗、证据保全意识淡薄、互联网法院执法过程中的公信力不足、取证不及时等痛点问题，为用户提供以下功能：①提供操作友好的用户界面（UI），包括但不局限于：可查询、可下载、可固定、可验证（是否被篡改）、可展示等功能；②将需要存证的数字资产上传至 IPFS 分布式文件系统，该系统会向 Web 前端返回一个基于存证对象生成的唯一 Hash 值，以备上链；③存证用户不仅能够从数字资产存证平台查询到相关的数字资产信息，还可以从 IPFS 所提供的性能最优节点下载文件分片并还原[50]；④存证用户若要鉴权，可授权第三方发证机构查看链上信息，并与原始数据进行比较，如果结果一致，则开具鉴定报告。

用例图（Use Case Diagram）是 UML（Unified Modeling Language，统一建模语言）中较为重要和常用的一种图，常被用于系统的需求分析阶段。图 3.1 展示了基于区块链的数字资产存证系统用例图，其中参与者（Actor）由存证用户和机构用户组成，针对存证用户，系统提供的可见外部功能包括用户注册、用户登录、文件加密、文件上传、证据固定、文件查看、文件下载、文件授权等；针对机构用户，系统提供的可见外部功能包括：机构注册、机构登录、证据校验、证据同步以及开具鉴定报告等。

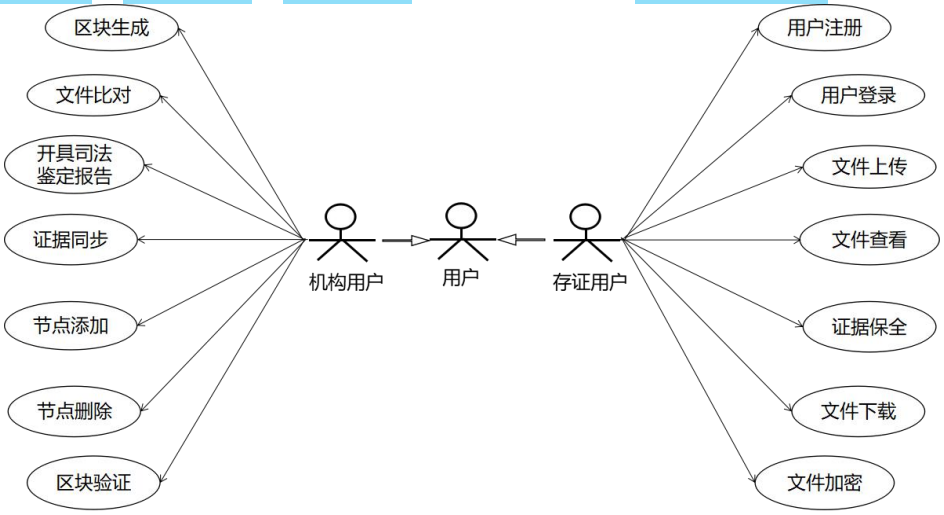

图 3-1 数字资产存证系统用例图

3.2 系统主要功能需求

用户注册功能需求如表 3-1 所示。本存证应用服务的对象是需要存证的用户，存证用户在注册时可直接将自己的信息捆绑到能够唯一标识身份的 DID(Decentralized Identifier，分布式身份标识）上[51]，然后将其作为交易数据打包存放至区块链中。当用户想要存证的时候通过区块链查找到其 DID，然后实现自主身份认证，减少了普通应用程序频繁注册的麻烦，同时也可避免忘记密码的苦恼。

表 3-1 用户注册

名称（标识符）	用户注册（User-Register）
功能描述	将能够唯一代表用户身份的分布式身份标识存放至区块链上
对应角色	存证用户
输入	用户名、DID
输出	存证用户注册信息

用户登录功能需求如表 3-2 所示。用户登录需提供唯一代表其身份的用户标识，如果已经注册，则区块链会自主认证用户身份，认证通过就可以使用系统提供的数字资产存证服务。

表 3-2 用户登录

名称（标识符）	用户登录（User-Login）
功能描述	登录存证系统
对应角色	存证用户
输入	DID
输出	存证系统 UI 界面

文件加密功能需求如表 3-3 所示。文件加密是针对原始文件上传至 IPFS 系统之后，其哈希指纹有可能会遭到泄露而设置的功能模块，否则一旦有人得到文件的 Hash 指纹，文件的内容就会暴露出来，最终会给用户造成很大的损失。而加密过后的文件即使被他人窃取到也于事无补，这时候的文件再上传至 IPFS 系统就显得格外安全。

表 3-3 文件加密

名称（标识符）	文件加密（File-Encry）
功能描述	对原始文件进行加密
对应角色	存证用户
输入	原始文件
输出	加密文件

感觉没必要

文件上传功能需求如表 3-4 所示。文件上传是数字资产上链保全的第一步，也是把文件进行哈希化的重要一步。之所以让文件先通过 IPFS 存储，是因为 IPFS 是一种分布式文件存储协议，其基于文件内容的寻址方式可替代传统以文件位置作为寻址方式的 HTTP 协议。另外，区块链仅能存储比较小的数据，类似于图片、文档等大文件尚无很好的应对之策。IPFS 的出现可提高区块链的存储性能，帮助其在链外解决大数据不能在链上保全的问题。

表 3-4 文件上传

名称（标识符）	文件上传（File-Upload）
功能描述	把文件分割存储在不同的节点，同时获得文件的哈希指纹
对应角色	存证用户
输入	数字资产
输出	文件的 Hash 值

文件查询功能需求如表 3-5 所示。在 IPFS 系统的客户端输入文件哈希指纹，即可查询到该文件。

表 3-5 文件查询

名称（标识符）	文件查询（File-Query）
功能描述	查看已经分片的文件
对应角色	存证用户、司法鉴定中心
输入	文件 Hash 值
输出	原始文件

文件下载功能需求如表 3-6 所示。文件下载和文件查询的原理很相似，都是基于文件的内容（即哈希指纹）在 IPFS 系统中寻找到存放文件分片的节点，然后对这些文件分片进行还原。相较于视频类等传统以中心化模式工作的 App 来说，以 IPFS 和 BitTorrent 为代表的分布式文件存储和查找协议占据着明显的效率和安全优势。

表 3-6 文件下载

名称（标识符）	文件下载（File-Download）
功能描述	下载分片文件并还原，为后续鉴权做准备。
对应角色	存证用户
输入	文件 Hash 值
输出	原始文件

数字资产保全功能需求如表 3-7 所示。数字资产经 IPFS 系统哈希化保全在区块链

中，由区块链的时间戳、Merkle 根、共识算法、智能合约等技术共同保障已存证数据的安全性、完整性和有效性。经联盟节点验证的区块被同步存储于本地区块链副本中，即使部分节点出现宕机或其他状况，也不会造成数据丢失的情况，更不会影响到区块链网络的稳定运行。

表 3-7 数字资产保全

名称（标识符）	证据保全（DA-P）
功能描述	对元数据、Hash 值以及授权信息等证据进行保全
对应角色	存证用户
输入	元数据、Hash 值、授权信息等
输出	交易信息

文件比对功能需求如表 3-8 所示。文件比对服务的提供者是联盟节点，这些节点由权威机构组成，任何结构想要加入，需要征得其他授权节点一半以上的允许。司法鉴定机构为用户提供文件比对的服务，一旦比对结果证实原始文件的所属者确实为存证用户所有，且未经篡改，即可向用户或其他执法部门出具司法鉴定报告。

表 3-8 文件比对

名称（标识符）	完整性检验（File-Compare）
功能描述	对原始文件和链上数据进行比较
对应角色	司法鉴定中心
输入	链上数据和原始文件
输出	司法鉴定报告

文件授权功能需求如表 3-9 所示。司法鉴定机构在对文件进行比对时需要经过存证用户的授权，只有这样该机构才能对原始文件进行哈希计算，得到的 Hash 值与链上记录的数据进行比较。

表 3-9 文件授权

名称（标识符）	文件授权（File-Authorization）
功能描述	授权第三方用户进行查看
对应角色	存证用户、联盟节点
输入	对方公钥签名、授权者私钥签名的文件
输出	原始文件

节点管理功能需求如表 3-10 所示。以太坊测试网 Kovan 使用的是授权证明共识机制（PoA），该共识机制的特点是能够实时调整节点的加入和角色的转变，并对各联盟

成员的节点性能进行预判，共同决定授权节点的添加或删除。另外，普通节点只有验证的功能，挖矿和投票都由授权节点来完成，大大减少了算力的浪费，但挖矿是没有奖励的，仅仅是授权节点的义务劳动。

表 3-10 节点管理

名称（标识符）	节点管理（Node-Mana）
功能描述	实时调整节点的加入和角色的转变，对节点性能进行预判
对应角色	所有共识节点
输入	授权节点投票
输出	投票结果

3.3 系统非功能需求分析

3.3.1 系统环境需求分析

基于区块链的数字资产存证系统对运行环境的需求包括：可使用云平台提供的虚拟机（VMware Workstation 15 Pro）、分布式文件存储和查询系统（IPFS 或者 Swarm）等资源。底层区块链网络的授权节点不能低于 1 个，总节点个数不能低于 4 个。可使用以太坊提供的测试网络（Kovan 或者是 Rinkeby）。因 PoA 共识算法无挖矿奖励，因此负责区块打包的矿工仅仅是将自身荣誉与身份联系在一起，通过为存证用户提供服务得到一部分佣金，以此作为维护区块链网络稳定运行的动力。

3.3.2 系统质量需求分析

表 3-11 展示了系统的质量需求分析，包括可用性和可靠性两个方面，并分别展示了其对应的详细要求。

表 3-11 系统质量需求分析

主要质量指标	详细要求
可用性	该系统可为用户提供人性化的界面、帮助以及具有介绍性和说明书性质的文档。
可靠性	当用户在没上传的文件就提交的情况下会显示“请上传文件”；系统能够在 7×24h 内安全运行，年宕机（非计划）时间不得超过 8 小时；快速部署，灵活操作。

3.3.3 系统性能需求分析

表 3-12 展示了系统的性能需求分析，包括吞吐率、冗余性以及延展性，并分别展示了其对应的详细要求。

表 3-12 系统质量需求分析

主要性能指标	详细要求
吞吐率	能够准确把状态机中的交易数据存储到区块链中，合理占用内存，出块时间固定在达到商用的时间段内，系统吞吐量高。
冗余性	数字资产经过 IPFS 系统处理会生成唯一上链的 Hash 值，因此系统是低冗余的。
延展性	采用统一数据模型技术，方便系统的扩展。由于统一数据模型结构统一，可以统一记账，并实现数据的万能导入，方便第三方数据的集成。

3.3.4 系统可支持性需求分析

表 3-14 展示了系统的可支持性需求分析，包括实现、封装以及授权，并分别展示了其对应的详细要求。

表 3-13 系统可支持性需求分析

主要性能指标	详细要求
实现	采用以太坊+PoA 成熟技术，和以太坊同步升级，保护用户的投资，内置浏览器，通用智能合约，确保系统快速上线。懂 Go 语言就可以开发应用，降低区块链开发成本。
封装	采用智能合约开发框架，经过多个项目的验证和考验，并对合约进行封装，确保合约的安全。
授权	只有经过用户授权的第三方机构才能查看文件原始信息

3.4 本章小结

本章节从功能性需求分析和非功能性需求分析两个方面出发，对系统的需求分析做了详细的阐述。其中，功能性需求和具体的业务有关，例如文件加密、文件上传、文件分片、数字资产保全、文件查询、文件下载、文件对比、授权访问等。而非功能需求主要和系统所表现的性能有关，包括系统的环境需求、质量需求、性能需求以及可支持性需求等，旨在为用户提供良好的使用体验。

第 4 章 基于区块链的数字资产存证系统概要设计

4.1 存证系统网络拓扑图

图 4-1 展示了基于区块链的数字资产存证系统网络拓扑图。该网络拓扑图清晰地给出了存证的具体流程：首先由存证用户通过存证平台将数字资产的哈希指纹保全到区块链中，区块链网络中的其他节点负责同步区块数据。当存证用户因版权与他人产生纠纷时，即可授权司法鉴定机构下载原始文件，通过与已存证数据进行特征比对，生成司法鉴定报告，并及时向存证用户和执法部门提供证据。

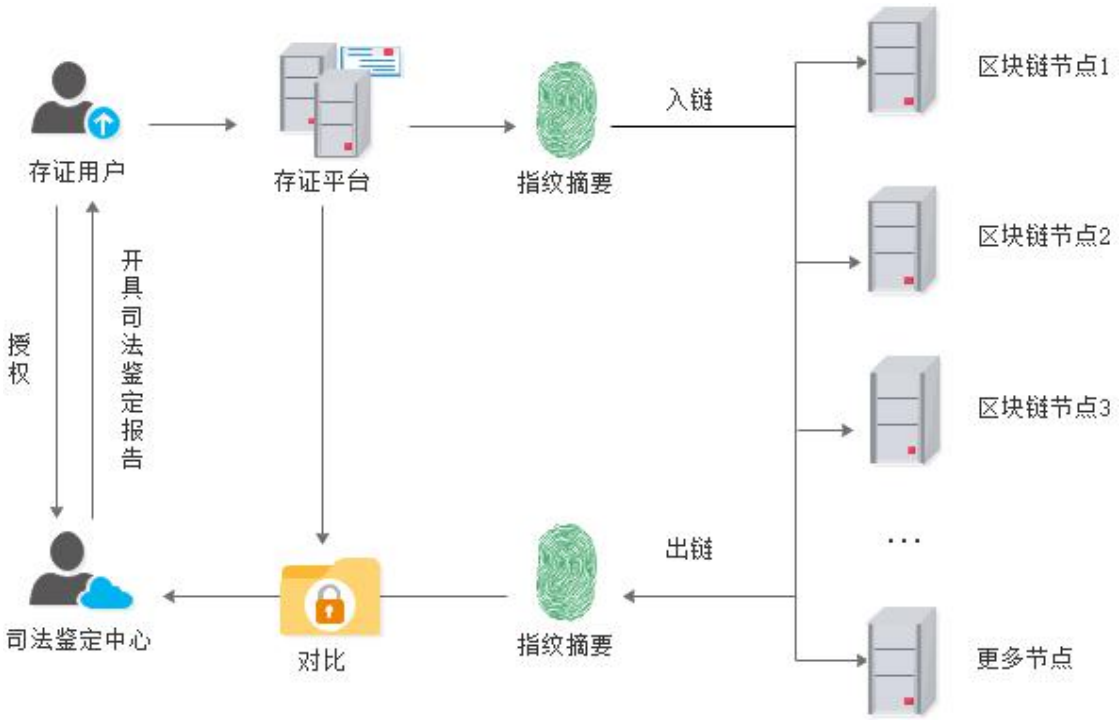

图 4-1 系统网络拓扑图

4.2 改进的区块链框图

区块链作为一种特殊的分布式共享记账方式，存在天然的存储缺陷，它只能存储简单的文字记录，每个区块也仅能存放不超过 1MB 的文字记录，这对其在实际场景中的应用和推广产生了很大的阻力[52]。结合区块链的特点以及 IPFS 在文件分拣和点对点传输方面具有的优势，给出了如图 4-3 所示、经过改进的区块链框图。该框图最大的特点是将所占字节比较大的超文本文件转化为哈希指纹，而 Hash 值是一段固定长度的字符串，所占字节也很小，所以非常适合上链存储。

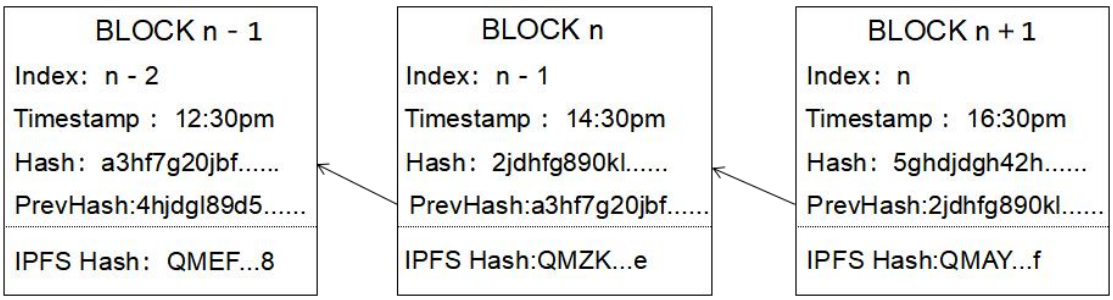

图 4-2 改进的区块链框图

4.3 数字资产存证系统概要设计

需求分析之后就可以开始有针对性地对系统结构进行概要设计，这是系统实现的关键一步，也是实现详细设计的基础，其目的是建立合理的结构体系，以减轻后续环节的工作量。另外，概要设计实现的是系统的底层架构设计、总体结构设计、功能模块划分、用户界面设计、数据库设计等，重在强调系统包含的模块数量、各个模块之间的接口以及模块本身所具有的功能。

4.3.1 系统底层架构设计

基于区块链的可信数字资产存证系统底层架构由六层组成，自下而上分别是数据层、网络层、共识层、激励层、合约层、应用层。

（1）第一层是数据层。数据层封装了可信数字资产存证的所有信息，包括元数据、账户信息和数据指纹信息和交易信息等。

（2）第二层是网络层。网络层封装了 P2P 组网机制、数据传播机制和数据验证机制等，以实现不同节点之间区块数据的同步与验证。

（3）第三层是共识层。基于区块链的应用本质是一种由多个节点同时运行的 DApp（去中心化应用），单个节点生成的结果需要经过全网节点确认才能被打包进链[53]。基于区块链的可信数字资产存证系统同样符合去中心化应用的特点，并选择改进的 PoA 机制作为共识标准。

（4）第四层是激励层，激励层包括 Token 发行机制和 Token 分配机制，通过经济手段，奖励那些参与记账的节点，维护网络的正常运行。

（5）第五层是合约层，该层封装了资产固定、身份认证、审计追踪等，使一些指令自动执行，实现我们想要的功能。

（6）最后一层是应用层，这一层被用来对外提供基于区块链的各种应用，包括账户注册和登录、账户管理、数字资产的存证和鉴权等。

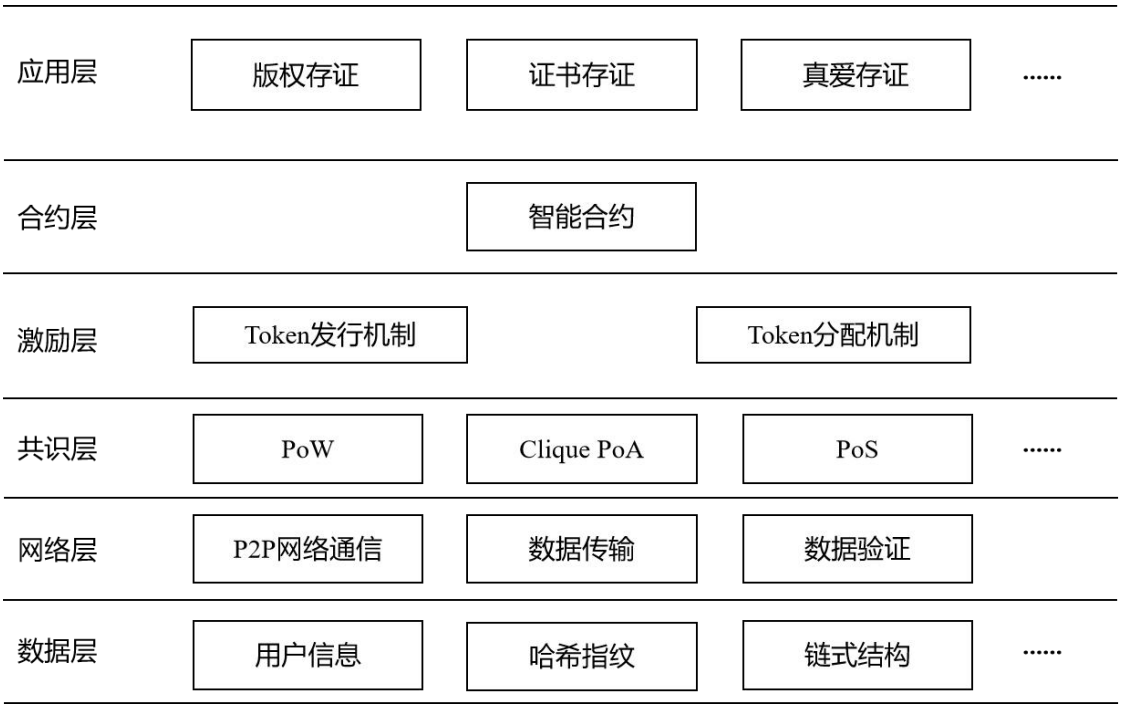

图 4-3 系统底层架构设计

4.3.2 系统总体框架设计

图 4-3 展示了数字资产存证应用的总体框架设计，最上层显示了存证平台的服务对象，包括企业用户、个人用户和执法部门。类似于金融、传媒、电商和保险等这些常与数字资产打交道的企业尤其重视数字资产的存证。

存证对象面向各类数字资产，包括图片、音频视频、电子合同、电子发票、电子邮件、版权等。之所以提供基于区块链的数字资产存证应用，是因为当下信息安全问题严重阻碍了社会的发展，各种网络安全事件层出不穷，给用户造成很大的损失。

IPFS 是一种新的网络协议，旨在取代传统的 HTTP 协议，发挥自身在数据存储和传输上的优势。存证用户将加密文件上传至 IPFS 系统，会得到一个基于加密文件生成的 Hash 值。该 Hash 值是唯一确定的，且具有不可逆性，恰好可以作为证据固定到区块链中，然后再利用区块链去中心化、防篡改、历史数据可回溯、集体维护等特性保证链上数据的安全性、完整性和一致性。

本存证应用的底层区块链是一种授权联盟链，包括凭证管理中心（CA）、司法鉴定机构、审计、公证以及仲裁都是区块链网络的组成部分。PoA 是该联盟链选择使用的共识算法，属于授权类共识机制的范畴。挖矿、区块生成以及区块确认都由投票选出的授权节点完成，节省了挖矿时算力的浪费。经过最终确认的区块被添加到主链，其他参与节点完成同步操作。

最底层是基础服务层，服务类别包括文件分片、证据保全、取证和鉴证等。

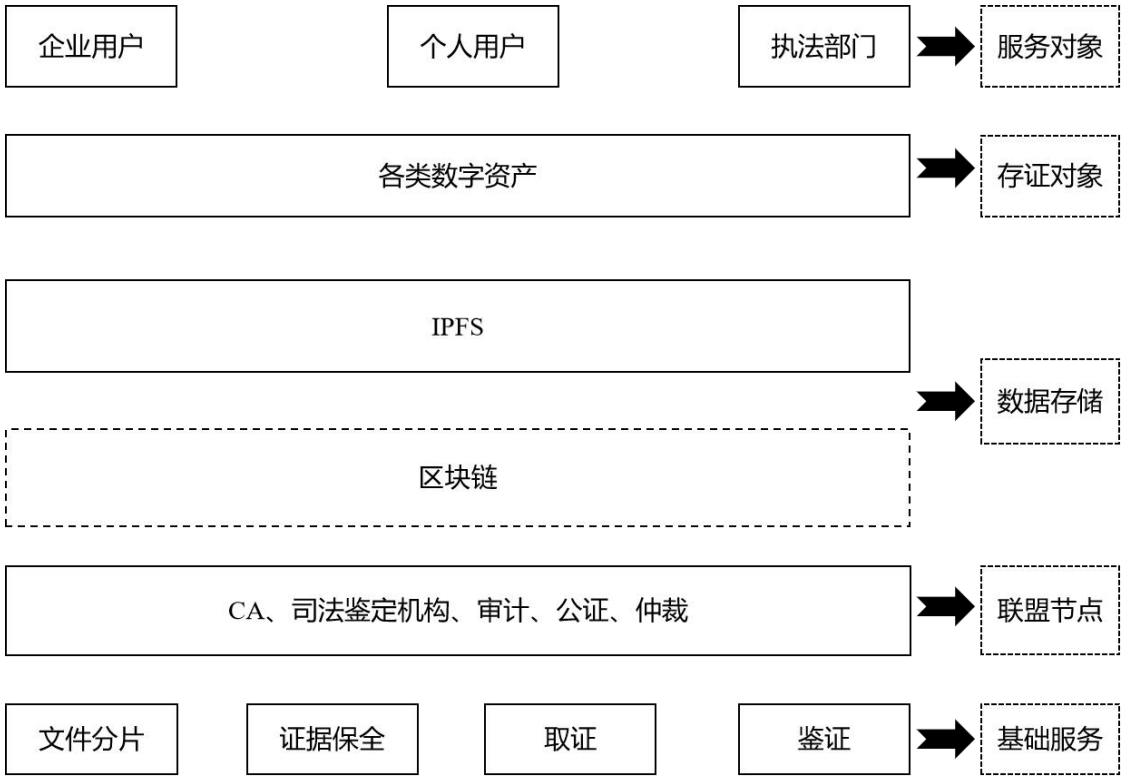

图 4-4 系统总体架构设计

4.3.3 系统总体功能模块

图 4-5 展示了数字资产存证系统的总体功能模块。其中用户管理模块包括用户登录、用户注册两个部分，服务的对象包括存证用户和联盟用户，存证用户是一般用户，只有存证方面的需求，而联盟用户是第三方权威机构，既要负责区块的产生和验证，又要服务于一般用户的存证需求，为他们开具司法鉴定报告。节点管理模块又分为节点删除和节点添加两个子功能模块，本存证应用使用的联盟链是一种授权区块链，节点的加入和删除都要经过授权节点的投票。授权访问模块选择 RSA 非对称加密算法或 AES 对称加密算法对文件进行加密，保证在文件传输的过程中只有目标用户能看查看。区块链系统主要实现的功能是对存证文件进行保全，整个过程需要调用智能合约。文件管理包括文件加密、文件上传、文件查看、文件下载、文件比对、出具司法鉴定报告等功能模块。

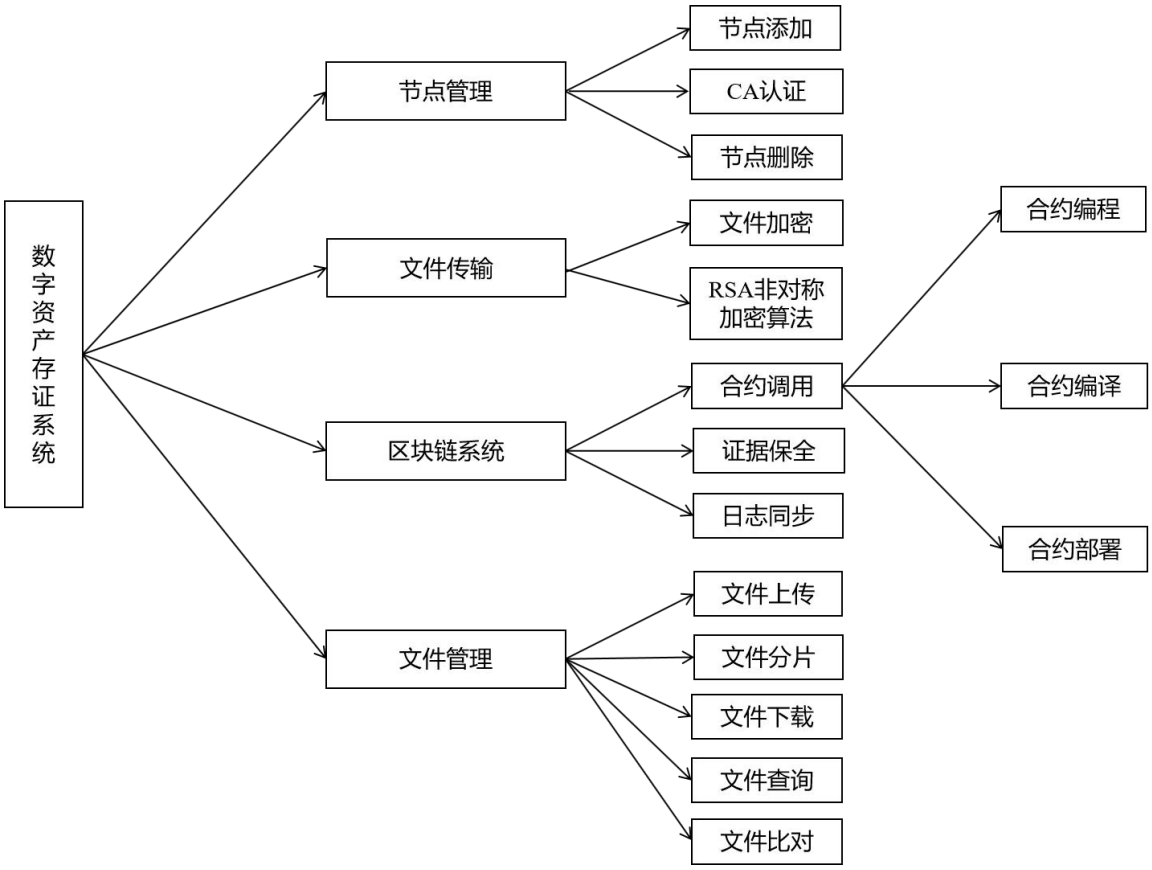

图 4-5 系统总体功能模块结构设计

4.4 本章小结

本章节从三个方面对存证系统进行了简要的设计。首先，对区块链网络的搭建和运行原理进行了概括性的阐述，其中，Truffle 框架是用来为区块链开发提供通用的模板，简化项目的开发流程，方便智能合约的编写、编译和部署。然后，给出了系统底层框架设计，即利用区块链存储数字资产的关键数据（Hash 值）。区块链网络六个层次的功能各不相同，但却是相辅相成、相互作用的关系。紧接着，设计了系统总体框架，并介绍了该框架要表达的内容。最后对系统总体功能模块进行了划分。

第 5 章 基于区块链的数字资产存证系统详细设计与实现

5.1 资产数据存证系统详细流程图

图 5-1 展示了资产数据存证系统详细设计的流程图，具体运行流程如下：

（1）验证节点对提交节点进行身份认证，并预执行交易，但交易信息不会被写入账本，执行结果会被返回给提交节点。认证通过的即可登录存证平台。

（2）使用 RSA 非对称加密算法对原始文件进行加密，为后续授权访问做准备。

（3）选择 IPFS 系统中性能最优的若干个节点，即时常活跃、很少出问题的节点。

（4）向选中节点分发文件分片，IPFS 系统会返回相应的 Hash 值。

（5）通过调用智能合约把文件的元数据（或者称为特征数据）和加密后进过 Hash 函数计算得到的指纹数据存放入区块链中。

（6）存证用户需要鉴权的时候，从 IPFS 系统下载文件分片并还原。

（7）将授权信息写入区块链，以监督被授权者任务完成情况。

（8）通过 PoA 共识算法筛选出用于记账的主节点，由它负责区块的生成和交易数据的打包工作。

（9）备份节点负责同步主节点广播的区块数据。

（10）司法鉴定机构被授权访问原始文件，并对其进行相同的哈希运算。

（11）将通过哈希得到的值与区块链中存储的指纹数据进行对比，比对成功则证明文件未被篡改。

（12）司法鉴定机构会依据鉴定结果在线出示司法鉴定报告。

（13）司法鉴定报告用于在线裁决和保险理赔等，以期解决纠纷。

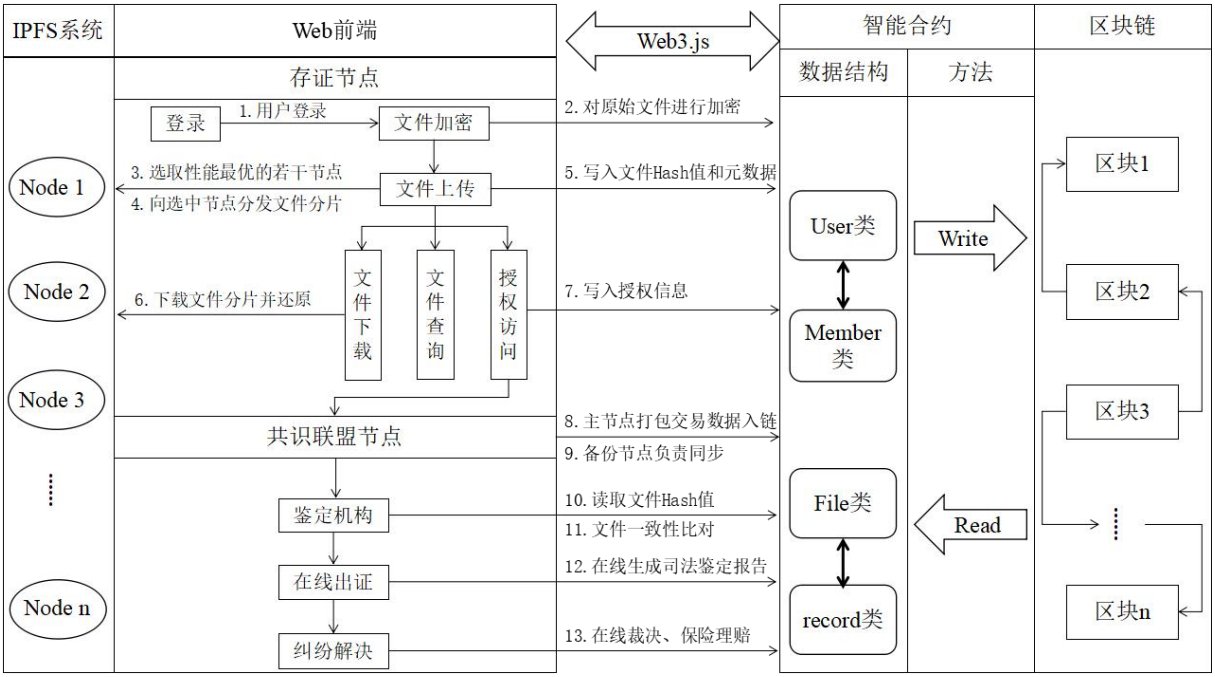

图 5-1 资产数据存证系统详细流程图

5.2 区块链存证系统主要功能模块设计与实现

基于区块链的数字资产存证系统主要涵盖了以下几个重要功能模块：文件加密模块、文件上传模块、证据保全模块、区块共识模块以及授权访问模块。文件加密模块主要是为了防止证据尚未保全之前就被攻击，即保障链下数据的安全。文件上传模块的作用是把原始文件分割存放在多个性能良好的节点中，与此同时，IPFS 系统根据文件内容生成唯一确定的 Hash 值返回给 Web 前端，为后续证据上链做准备。证据保全模块利用区块链存储，以达到数字资产难以篡改的目的。区块共识模块应用于交易数据打包阶段，旨在对广播的交易内容和打包好的区块进行验证，验证通过的区块会被除记账节点以外的其他节点同步记录，当共识网络中的某个节点想要查看数据时只需要检索区块链即可。授权鉴权模块是区块链存证的最后一个环节，需要司法鉴定机构通过文件的结果出具相应的司法鉴定报告，由用户保管或直接交由法院处理。

5.2.1 用户管理子系统的设计与实现

用户管理子系统主要是对用户身份进行确认，通过验证的用户方可继续接下来的存证操作。本文选择将用户的注册信息和能够代表其唯一身份的 DID 共同存储于区块链这一特殊的分布式账本中，实现用户身份的自主验证。

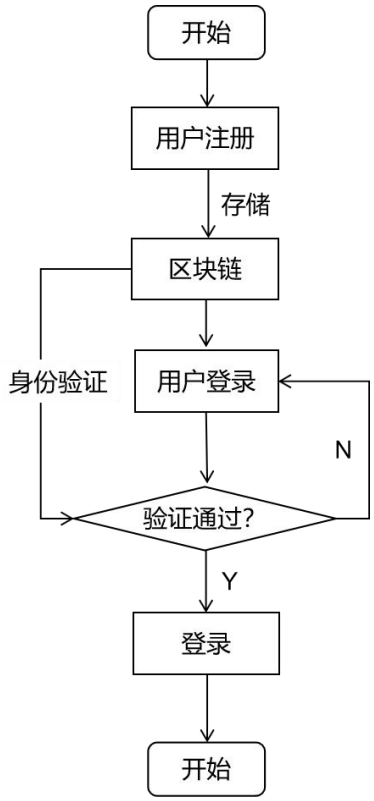

图 5-2 用户注册流程图

5.2.2 文件传输子系统的设计与实现

文件传输子系统包括文件加密功能模块和授权访问功能模块两个部分。其中，前者是为了防止任何获取到哈希指纹的人都有可能从 IPFS 系统中提取出原始文件。为了防止以上事件的发生，就必须在文件被分片之前对它实施安全处理。在这种情况下即使有人得到了文件的 Hash 值，也难以提取到有用的价值。后者被设计的初衷是让存证用户授权第三方查看文件信息，而其他用户是没有权限查看的。通过这种方式，能够很好的保护自己的隐私不被泄露，同时也可以安全地与他人分享。

5.2.2.1 文件加密功能模块的设计与实现

使用本存证应用的一般用户需要首先对上传的文件进行加密操作，该过程使用的是 RSA 公开密钥算法。我们熟知以比特币为主的加密货币大都使用 ECC（Elliptic Curves Cryptography，椭圆曲线加密）公开密钥加密算法。以上两种算法都属于非对称加密的范畴，都会生成公钥和私钥两种密钥，由公钥加密的数据只有私钥才能解密，而由私钥加密的数据也只有公钥才能解密。用户在存证的过程中根据实际情况也可自由选择是否加密或者不加密两种状态，系统默认的是加密，否则需要手动修改为不加密。

文件加密的详细流程如图 5-3 所示：

（1）由存证用户选择对数字资产加密与否；

（2）如果不加密，直接将原始文件的 Hash 值记录到账本中；

（3）如果加密，使用 GPG 提供的 RSA 公开密钥加密算法对数字资产进行加密；

（4）将加密文件上传至 IPFS 系统，由它把文件分割存放在参与的多个节点中，并向 Web 前端返回基于文件内容生成的哈希值；

（5）将上述得到的哈希指纹保全在区块链这种高强度防止数据被修改的文档型记账本中。

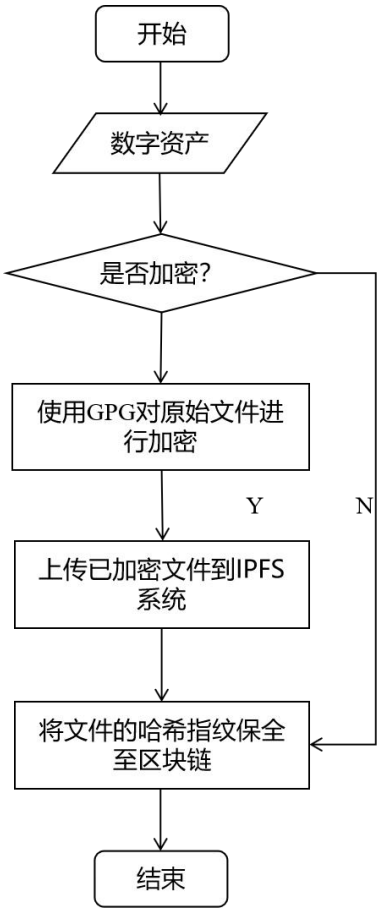

图 5-3 文件加密流程

5.2.2.2 文件授权功能模块的设计与实现

文件授权是对文件加密功能的延伸，其主要作用是保证数据传输的安全，同时只允许授权节点访问文件内容，其他节点未经许可是不能查看的。文件授权的详细流程如图 5-4 所示：

（1）存证用户从 CA 中心获取被授权者的公钥（以 pubkey.asc 的加密形式存在）；

（2）存证用户使用被授权者的公钥对存证文件进行加密；

（3）被授权者接收到已加密的存证文件；

（4）被授权者使用自己的私钥对问价进行解密；

（5）被授权者可以查看原始文件内容。

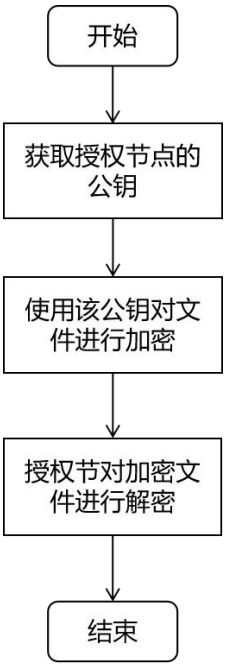

图 5-4 文件授权的详细流程

5.2.3 数字资产保全子系统的设计与实现

数字资产保全子系统是区块链存证应用最关键的一个环节，该环节用于实现文件的上链操作，包括智能合约的调用、区块的生成以及区块的共识三个组成部分[54]。由于区块链的存储限制，证据保全的对象并不是数字资产本身，而是经 IPFS 系统哈希化之后得到的 Hash 值。该 Hash 值会被反馈给 Web 前端，然后被当作一次交易打包成一个区块，在经过其它节点验证之后永久地固定在区块链中。值得注意的是，区块链存证应用需要节点开启 Web3.js 封装的 JSON.RPC API 调用才能够实现与区块链进行交互。

5.2.3.1 智能合约的设计与实现

运行在 EVM 中的智能合约是资产数据上链保全的先决条件，通过其自动执行的特点，可减少第三方服务机构内部管理人员等因素干扰，使得信息记录更加安全。Web 前端通过 web3.js 调用预先部署在区块链网络上的存储合约，将需要存证的数字资产的 Hash 值固定在区块链上，以达到链上数据不可篡改的目的，为后续证据的出示和鉴权提供依据。Solidity 是最常用的描述智能合约的语言，除此之外，LLL、Serpent 也可以作为描述智能合约的语言，但普及度不高。本文设计的数字资产存证应用使用到的智能合约主要有两种：一种是用于资产数据登记，即将数字资产的 Hash 指纹提交至区块链系统，另一种则是用于数据确权（Proof of Ownership），即证明上述已登记的资产数据在某一时间点确实存在，而且它归于某个主体所有。

运行在 EVM 中的智能合约从生成到被调用的整个流程如图 5-5 所示：

（1）加载 Web3，为 Web 前端与区块链网络的交互提供 JSON.RPC API；

（2）用 Solidity 语言对能实现哈希指纹上链等功能的智能合约进行编程；

（3）选择是否使用在线编译的方式编译智能合约，一种选择是 Remix ide 提供的在线编译，另外一种使在 Geth 客户端中安装 Solc 编译器；

（4）编译创建的字节码被当作交易数据向空地址发送；

（5）智能合约被部署在区块链上的 EVM 中，生成合约地址，等待外部事务调用。

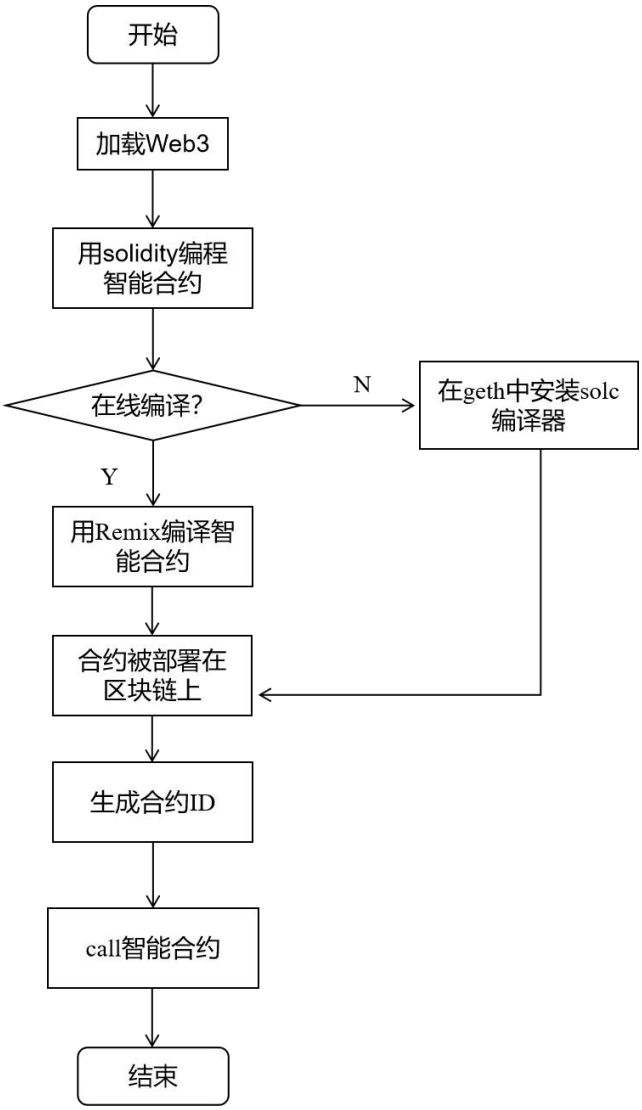

图 5-5 智能合约的运行流程

存储合约的源代码如下所示：

```
pragma solidity ^0.4.17;   //^0.4.17 表示所用 solidity 语言的版本号
contract StoreHash { //合约名为 StoreHash.sol
string ipfsHash;
//将经由 IPFS 系统返回的哈希指纹记录到区块链中
function sendHash(string x) public {//检索保全在区块链中的哈希指纹
    ipfsHash = x;
    }
function getHash() public view returns (string x) {
    return ipfsHash;
    }
}
```

智能合约编写好之后需要在 Remix 在线编译器中进行编译（Compile），编译的目的是使智能合约转化为 EVM 能够执行的字节码（Bytecode），然后生成一个 ABI（应用系统二进制接口，Application Binary Interface）。最后通过有账户余额的 Metamask 钱包将编译好的智能合约部署在 Kovan 上，该区块链测试网络会返回智能合约的地址供开发人员调用。

5.2.3.2 区块生成的设计与实现

区块是哈希指纹、数字签名信息等资产数据存放的地方，包括区块头和区块体两部分，区块的数据结构如表 5-1 所示：

表 5-1 生成区块的信息

字段	数据类型	描述
Index	int	区块的高度
Timestamp	string	区块的准确生成时间
IpfsHash	int	记录到区块链上的资产数据
Hash	string	本区块的 Hash 值
PrevHash	string	父区块（前一区块）的 Hash 值

在资产数据被打包进区块之前需要被 Hash 化，节省空间的同时保护资产数据的完整性。图 5-6 展示了区块生成的整个流程，详细执行步骤如下：

（1）用户从客户端发起一笔交易，该交易会被广播到网络中的其他参与节点，并缓存到各自的交易池（TxPool）中；

（2）每个节点都会对交易池中交易数据的签名进行有效性验证，如果有效则调用虚拟机执行交易，此时会返回一个交易列表和交易回执，否则抛弃这笔交易；

（3）接下来是组装区块，由通过 PoW 共识机制筛选出的主节点进行交易的排序和打包，没有记账权的等待下一轮挖矿开启；

（4）区块被成功挖到并打包好所有的交易之后，由主节点将该事件进行广播；

（5）收到广播的节点会对区块数据进行同步；

（6）把前一区块的 Hash 值存放到本区块头中，形成一个由散列值构成的链式结构，即区块链。

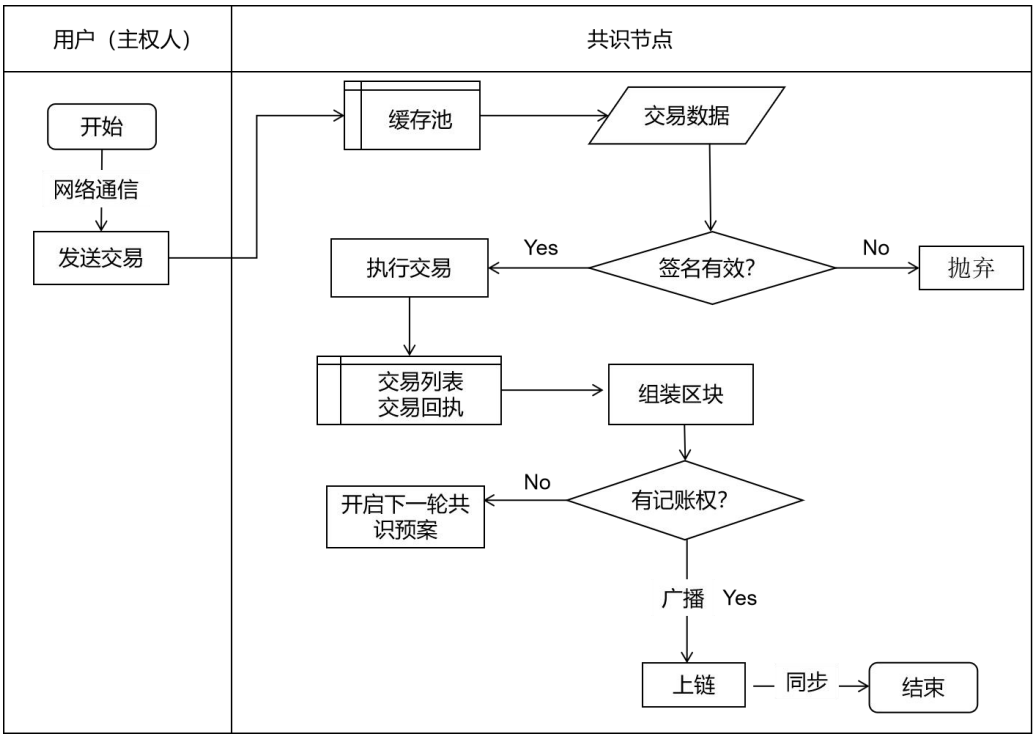

图 5-6 区块生成流程图

5.2.3.3 区块共识的设计与实现

区块共识的核心作用是产生区块和验证区块，目前有多重共识机制可以选择，但挑选记账节点的方式各有不同。例如：对于 PoW 共识机制来讲，拥有算力高的节点明显占据着有力的地位，但也因此损耗了宝贵的电力资源。表 5-2 展示了在 PoW 和 PoS 共识算法下生成的区块存在着一些差异，而这种差异主要体现在区块头结构。

表 5-2 PoW 和 PoS 区块头结构的差异

序号	字段	PoW	PoA
1	Coinbase	挖矿奖励地址	被提名为 signer 节点的地址
2	Nonce	随机数	提名分类（add or delete）
3	Extra	其他数据	signers 集合
4	Difficulty	挖矿难度	优先级

PoA（Proof of Authority，授权证明机制）是以太坊测试网 Rinkeby 正在使用的共识机制，该共识机制的原理也十分简单，即由已授权节点(signers)负责区块的挖掘和验证工作。普通节点如果想要加入联盟网络，必须得到所有授权节点投票的一半以上才可以，而授权节点如果在区块验证中存在恶意行为，也会被以投票（如果超 50%）的方式从联盟链中剔除。图 5-7 展示了 PoA 共识算法的工作流程。

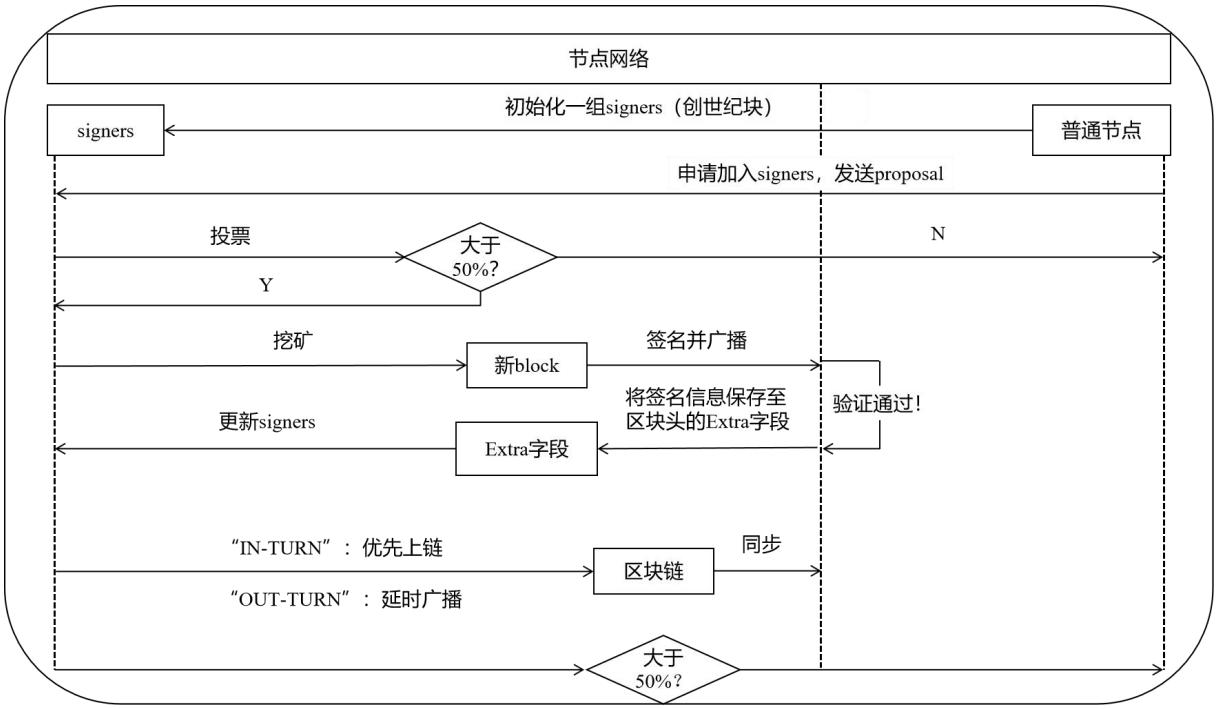

图 5-7 PoA 共识算法的工作流程

① 在 Genesis block 中初始化一组授权节点，并将其地址保存在 Extra 字段；

② 授权节点对挖矿生成的区块进行签名并广播，并将其签名保存在 Extra 字段；

③ 更新 Extra 字段，会显示所有授权节点的地址；

④ 处在“IN-TURN”的授权节点在某一区块高度优先上链；

⑤ 如果有新的 signer 加入，需经已授权 signers 至少（SIGNER_COUNT / 2) + 1）的投票数；相反，剔除旧的授权节点也要按照投票原则进行。

授权节点参与投票的核心代码如下：

```
//存放地址和布尔值的键值对映射
proposals map[common.Address] bool
//签名者的地址
signer common.Address
// 按时间排列的投票名单
Votes []*Vote `json:"votes"`
// 当前投票结果，避免重新计算
Tally map[common.Address]Tally `json:"tally"`
```

5.2.4 数字资产管理子系统的设计与实现

数字资产管理子系统的功能包括：加密文件上传、获取加密文件的哈希指纹、根据哈希指纹反向查看或下载加密文件以及文件对比等。数字资产管理子系统各功能模块的运行流程如图 5-8 所示：

（1）存证用户把需要存证的文件进行上传；

（2）文件被 IPFS 系统分割存储到多个性能优越的节点中；

（3）返回文件内容的哈希指纹，和存证用户的签名信息一起固定到区块链上；

（4）区块链网络上的授权节点对验证通过的区块数据进行同步；

（5）上传者通过生成的哈希指纹能够查看或下载文件；

（6）司法鉴定机构为存证用户提供法律援助，检索已存证的区块链数据；

（7）存证用户授权联盟链上的某一节点查看文件原始数据；

（8）被授权者对比原始数据的哈希指纹和区块链上已存证的历史历史数据；

（9）比对成功则证明该存证文件未被篡改，司法鉴定机构可据此可出具司法鉴定报告，一旦存证用户遇到版权纠纷、数字资产造到篡改等事件的发生，该报告就可作为证据交由互联网法院等权威机构审定并做出最终裁决。

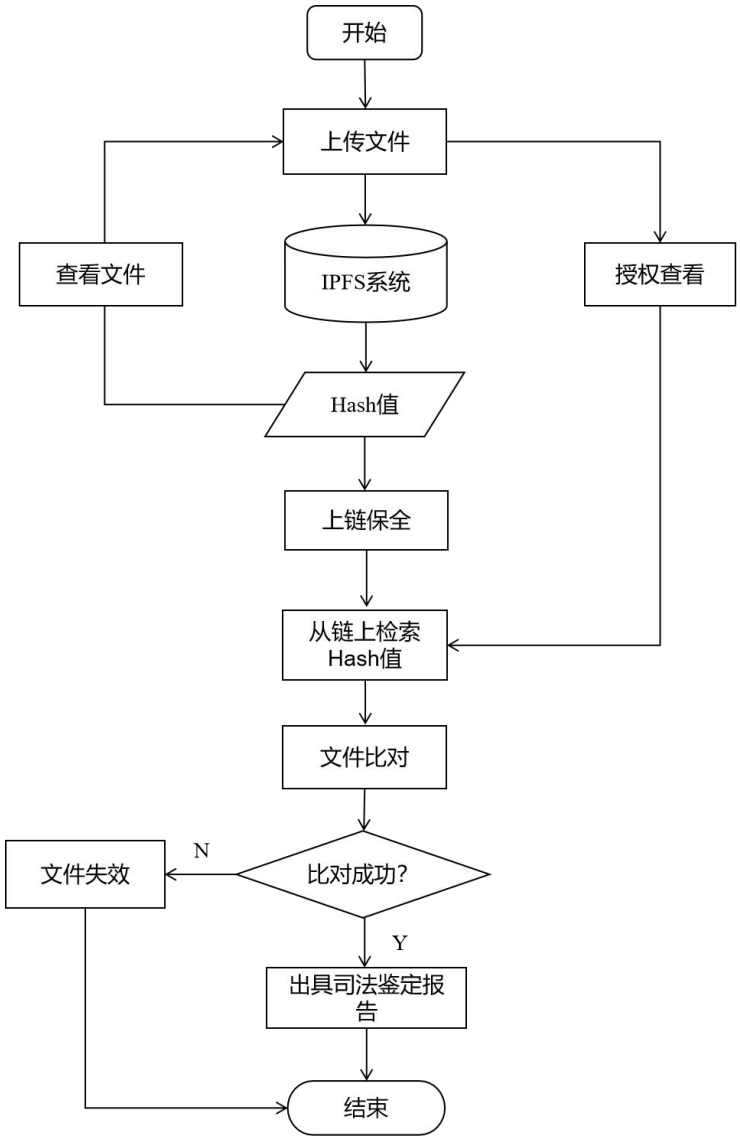

图 5-8 数字资产管理子系统运行流程

实现文件上传、获取哈希指纹、通过哈希指纹查看文件的核心代码如下：

```
//文件上传
this.saveTextBlobOnIpfs(ipfsContent).then((hash) => {console.log(hash); //返回 Hash 值
//获取 Hash 值
ipfs.cat(this.state.strHash).then((stream) => {console.log(stream);
```

5.3 本章小结

本章详细设计了数字资产存证系统的流程图，阐述不同功能模块之间的联系。然后从数字资产传输子系统、数字资产保全子系统、数字资产管理子系统三个角度出发，对系统的各项功能逐一设计。数字资产传输子系统包括文件加密和授权访问两个功能模块；数字资产保全子系统包括证据保全、合约调用、区块生成、区块共识四个功能模块；数字资产管理子系统包括文件上传、文件授权、文件比对三个功能模块。

第 6 章 基于区块链的数字资产存证系统测试

系统测试是检验系统能否按照设计的那样正常运行，如果能够正常运行则测试成功，整个系统也就完成，否则肯定是哪个环节出现了错误。系统测试是必不可少的环节，因为通过不断的测试能够优化系统的设计方案，弥补代码出现的漏洞。

6.1 系统测试环境介绍

对本数字资产存证系统进行测试的环境如下：

1） 操作系统：Ubuntu 18.04.1 LTS ；

2） 以太坊测试网：Kovan、Rinkeby ；

3） 区块链开发框架：Truffle（v5.0.2）；

4） 前端开发框架：React（一种用于构建 UI 的 Javascript 库） ；

5） 编程语言：Solidity（v0.4.24）、Go（v1.10.3）；

6） 以太坊钱包：Metamask（v5.3.1）；

7） 浏览器：火狐浏览器 ；

8） Solidity 在线编译器：Remix（v0.7.6） ；

9） 代码编辑器：Sublime Text3 ；

10） IPFS 环境：go-ipfs（v0.4.18）。

此外，系统的部分软件测试环境还包括：Npm 的版本为 5.6.0，NodeJS 的版本为 9.10.1，python 的版本为 3.6.7。

本文选择以太坊测试网 Rinkeby 来进行测试，Rinkeby 测试测试网使用的是 PoA 共识机制，区块链网络中的所有授权节点参与到区块的生成和验证中来，记账权的归属由对区块签名后的排序决定。存证用户往区块链中发送交易数据，同一时间段的所有事务都被记账节点打包进一个区块，经广播、确认后便可加入到区块链中。图 6-1 展示了系统测试阶段的实验结构示意图。

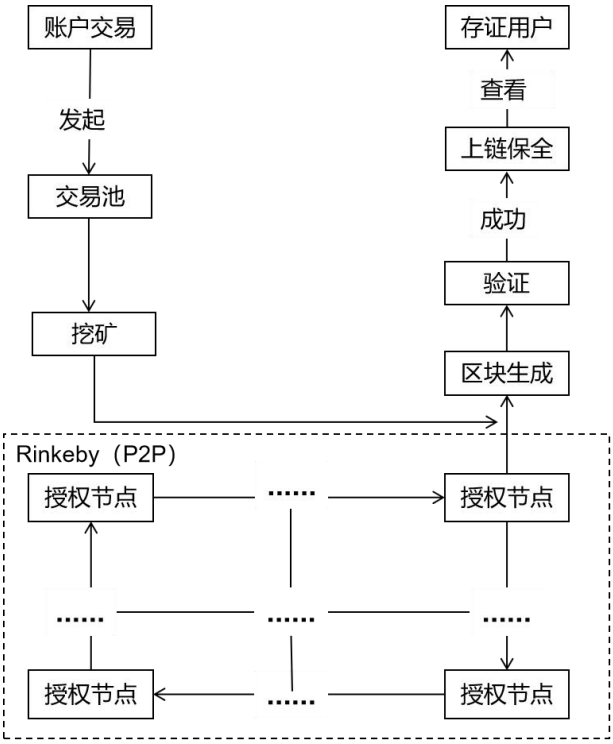

图 6-1 实验模型结构示意图

存证用户发起账户交易，将数字资产的哈希指纹当作交易数据保全到区块链上。起初，交易数据在交易池中等待被打包进一个区块，授权节点通过 PoA 共识机制筛选出记账节点打包交易数据并验证区块和事务，验证通过的区块即可永久保存在区块链中以备存证用户查询和鉴权。司法鉴定机构作为联盟链中的一个授权节点，如果接收到存证用户的存证需求，也可为他们提供法律上的援助，在需要时开具互联网法院等执法部门认可的司法鉴定报告。

6.2 系统测试方案

本系统的测试方案如下：

（1）首先对从本地选择的文件进行加密（也可不加密），观察是否生成加密文件。

（2）配置 IPFS 环境并初始化 IPFS 节点，完成之后启动守护程序将节点服务器连接到网络。执行文件上传操作，获得文件的 Hash 值；然后转而依据该 Hash 值从多个节点查询或下载文件。

（3）在火狐浏览器中安装 MetaMask 钱包插件。MetaMask 钱包是一款在 Firefox 或 Chrome 扩展程序中运行的插件（以太坊）钱包，特点是无需下载、轻量级。

（4）在 Remix 中编程、编译并部署一个存储合约到区块链上，一旦部署成功，Metamask 钱包中的账户余额就会减少，还会得到该合约的地址，而且在 Etherscan 中可查找到详细的交易信息。接下来就是调用已部署好的存储合约，将数字资产保全至区块链中，如果交易成功，钱包余额同样会因 Gas 消耗而减少，也同样会产生交易详情。

（5）搭建区块链网络环境，这里选择使用 Rinkeby，它是一种以太坊测试网，使用的共识机制是 PoA，该共识算法相较于 PoW 具有区块生成时间受控制、节点加入需经许可、减少算力消耗的优势。

6.3 系统主要功能测试

6.3.1 文件加密功能测试

对资产数据进行客户端加密使用的是 RSA 公开密钥加密算法，该算法可成私钥和公钥两种密钥，是典型的非对称加密算法。存证用户使用司法鉴定中心的公钥对存证文件进行加密，即直接授权联盟链中的某一司法鉴定机构对存证文件进行访问，区块链系统中的其他节点是没有查看权限的。表 6-2 是文件加密的测试用例和预期结果。图 6-4 展示了数字资产存证系统的文件加密功能页面。

表 6-2 文件加密的测试用例和结果

模块名	文件加密模块
操作人员	存证用户
测试类型	功能测试
前置条件	用户有数字资产存证需求
测试目的	检验文件是否可以被正常加密
操作步骤	1. 在本地选择需要上传的文件； 2. 点击“加密”按钮对文件进行加密。
预期结果	用户的数字资产被加密
测试文件	1. 1 寸照片.jpg 2. 电子合同.txt 3. 青岛之行.mp4 4. 电子发票.pdf
实际结果	1. 会显示“已加密！”字样 2. 返回数字资产的加密文件

图 6-2 文件加密功能系统页面

6.3.2 文件上传功能测试

文件上传是数字资产上链存证的第一步，由于区块链存在天然的存储缺陷，不可能将图片、文档、文本、视频等源信息直接存放入区块链。但为了能够实现数字资产的上链保全，可以考虑将原始文件的特征数据（即元数据）和指纹摘要（即 Hash 值）捆绑在一起作为 Merkle 树的一部分间接实现存证数据的上链保全。表 6-3 展示了文件上传测试用例和结果。

表 6-3 文件上传的测试用例和结果

模块名	文件上传模块
操作人员	存证用户
测试类型	功能测试
前置条件	文件已经被加密
测试目的	加密文件是否可以被正常提交到 IPFS 系统，并得到相应的哈希指纹。
操作步骤	1. 在本地选择需要上传的文件 2. 点击“上传”按钮
预期结果	在 IPFS 系统中可以查找到上传的文件
测试文件	1. 1 寸照片.jpg 2. 电子合同.txt 3. 青岛之行.mp4 4. 电子发票.pdf
实际结果	1. 文件被成功上传至 IPFS 系统 2. 可以看到返回的 Hash 值

图 6-5 展示了数字资产上传功能系统页面，值得注意的是上传到 IPFS 的不是原始数字资产本身，而是经过客户端加密的文件，如电子发票.pdf.gpg 是已经加密的文件。

图 6-3 数字资产上传功能系统页面

6.3.3 文件下载功能测试

表 6-4 文件下载的测试用例和结果

模块名	文件下载模块
操作人员	存证用户
测试类型	功能测试
前置条件	存证文件已上传至 IPFS 系统
测试目的	观察基于文件的哈希指纹下载并还原存放于多个节点的文件分片
操作步骤	1. 在 Hash 列表中选中要下载的对象； 2. 在弹出的对话框中选择“下载”按钮。
预期结果	可以快速下载已上传的文件
测试文件	1. 1 寸照片.jpg 2. 电子合同.txt 3. 青岛之行.mp4 4. 电子发票.pdf
实际结果	文件被完整地下载下来

图 6-5 展示了数字资产存证系统的文件下载功能页面:

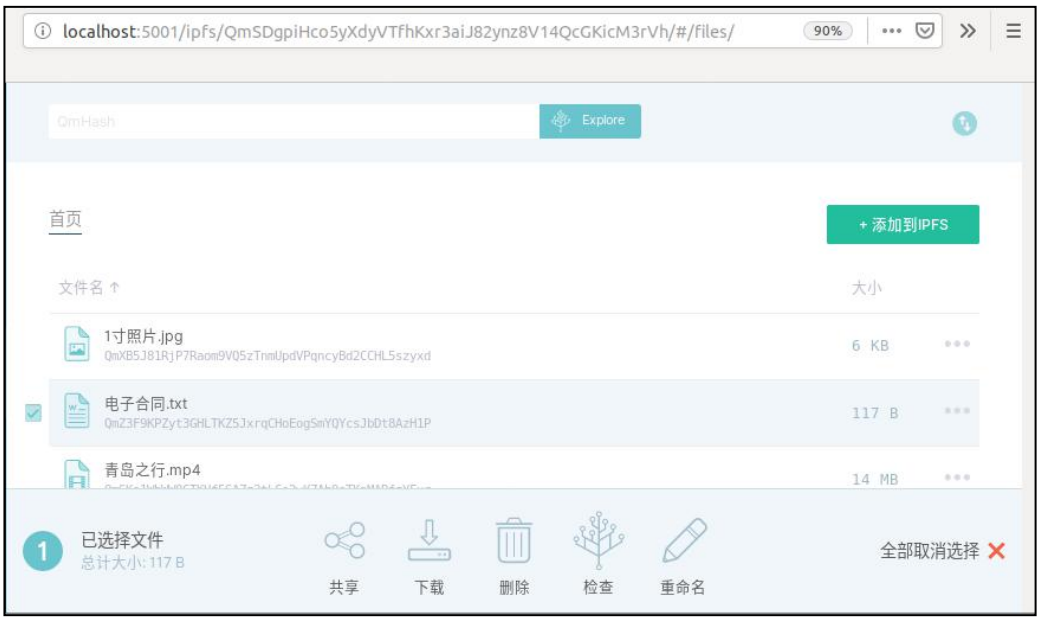

图 6-4 文件下载功能页面

6.3.4 数字资产保全功能测试

表 6-5 展示了数字资产保全的测试用例和结果。

表 6-5 数字资产保全的测试用例和结果

模块名	数字资产保全模块
操作人员	存证人员
测试类型	功能测试
前置条件	文件已转化为哈希指纹
测试目的	将存证数据固定到区块链中，以备后续取证、鉴证使用。
操作步骤	1. 将智能合约部署区块链上； 2. 提取智能合约的地址和 ABI 接口； 3. 点击“加载交易信息”按钮，会显示相关交易信息。
预期结果	1. 文件的哈希指纹被保全在区块链中 2. 在 Etherscan 中可以查看到详细的交易信息和区块信息
测试文件	电子发票.pdf
实际结果	1. 存证文件被成功固定到区块链中 2. Metamask 钱包账户的余额减少 3. UI 中可查看到 Gas 消耗和区块的高度

IPFS 系统对存证文件进行分片之后会得到基于文件内容生成的 Hash 值，该 Hash 值被存放在 Hash 列表中以待上链。整个过程需要 Remix 在线编译器、Metamask 钱包等工具的参与。Remix 是智能合约的集成开发环境，提供合约的编写、编译、部署和调用等功能；而 Metamask 数字钱包提供了带有 Ether 余额的账户和 Koven 测试网。图 6-5 展示了智能合约被顺利地部署在 Koven 测试网上，并返回了合约的地址以及相关函数（sendHash 和 getHash）。图 6-6 展示了指纹上链后反馈的详细交易信息。

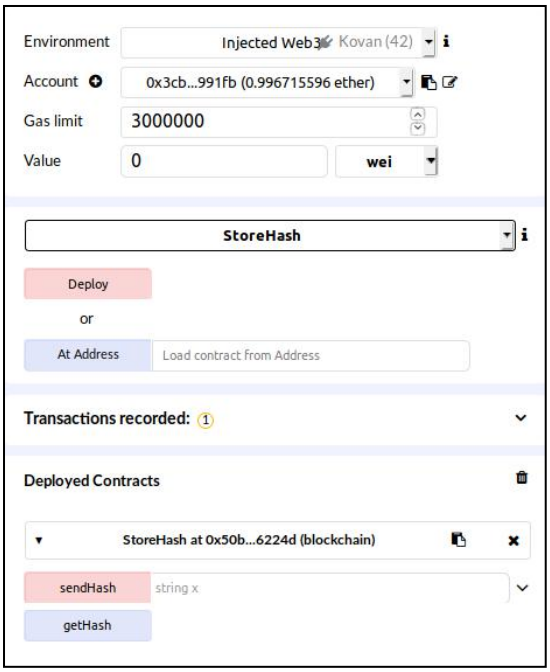

图 6-5 智能合约的部署

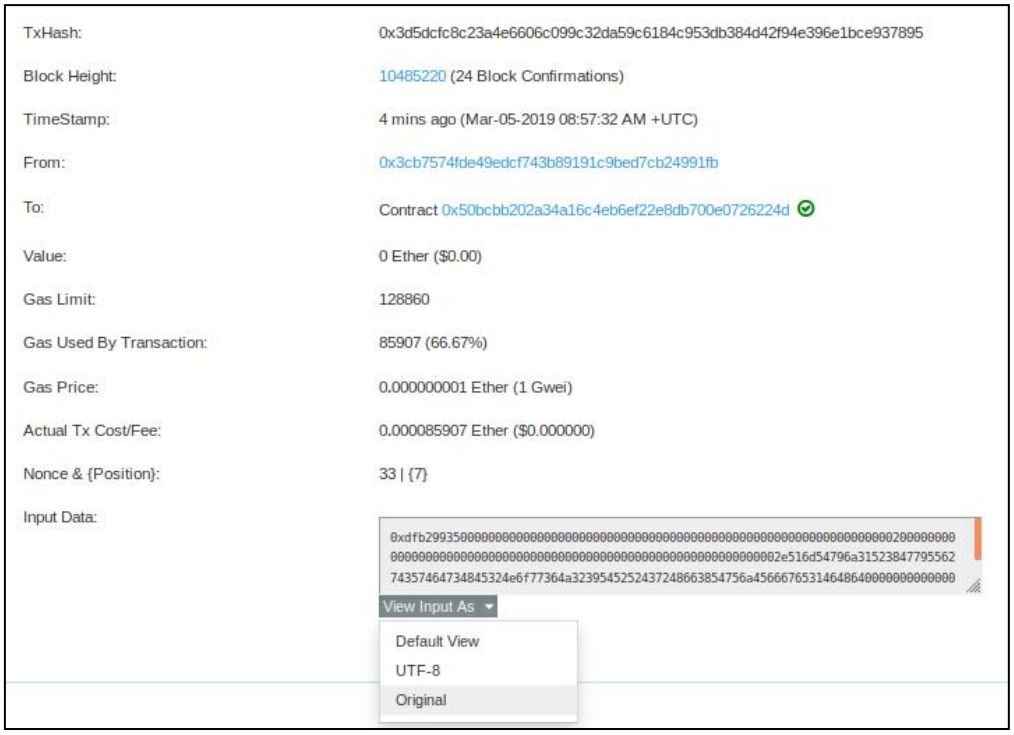

图 6-6 详细交易信息

图 6-7 展示了区块的生成信息，包括生成的区块高度、时间戳、本区块 Hash、前一区块 Hash、叔块 Hash、挖矿者的账户、难度值、区块大小、Gas 消耗、最大 Gas 消耗限制、随机数、挖矿奖励以及额外的交易数据等。

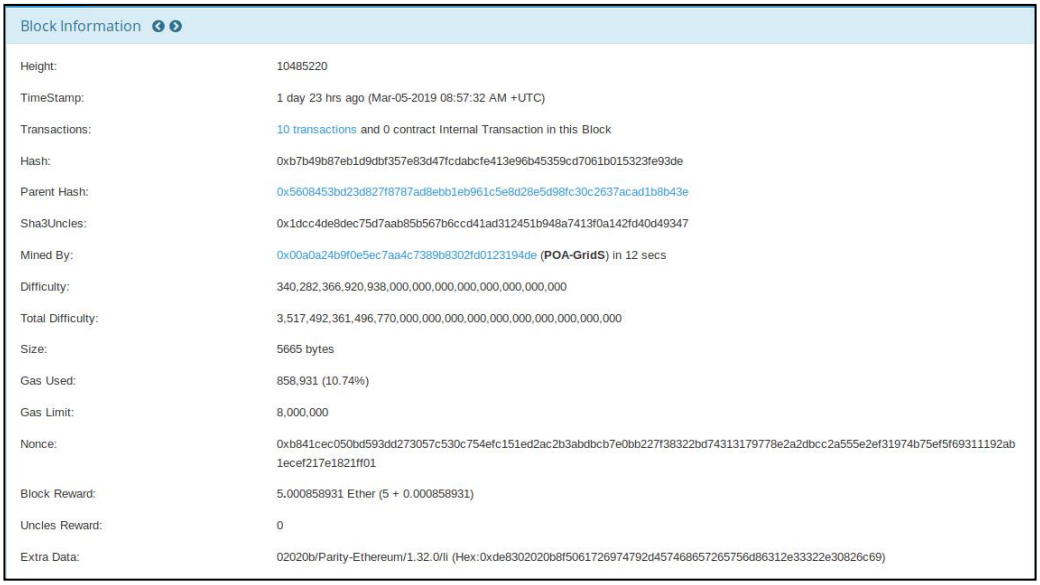

图 6-7 区块的生成详细信息

图 6-8 展示了数字资产保全功能系统页面，交易确认之后点击加载交易清单，会显示存证文件的哈希指纹、已部署智能合约的地址、区块高度、油耗（即此次交易消耗的 Gas）等信息。至此，数字资产上链保全成功完成预期测试效果。

图 6-8 数字资产保全功能页面

文件比对是司法鉴定机构开具鉴定报告、执法部门进行确权比较关键的一步，关乎到存证用户的合法权益能否得到保障。通过存证检索接口，司法鉴定机构从区块链中检索出已存证加密文件的 Hash 指纹，检索到的数字签名被用于身份验证，证明该存证数据确实是由存证用户本人上传。接下来需要做的工作是：存证用户授权司法鉴定机构访问数字资产的原始数据，并对它进行哈希计算，把得到的结果与链上的存证数据进行对

比，如果相同，则证明用户的数字资产并未被他人篡改。最后，由司法鉴定中心依据比对结果，出具具有权威性质的司法鉴定报告，用户拿着这份报告便可对侵害自身合法权益的行为进行警告或向互联网法院等其他执法机构诉讼。

6.4 本章小结

本章节对系统的主要功能进行了测试，测试的目的是检验系统的核心功能是否可以正常运行。测试之前需要搭建区块链的网络环境，用到的硬件、软件环境在 6.1 章节已经给出，而且详细构造了实验模型结构示意图。环境搭建好之后设计了详细的系统测试方案，清晰地展现出需要完成的工作和系统功能测试的先后顺序，方便实际测试时能够保证有条不紊地进行。接下来对各个功能模块分别进行了测试，并给出了系统界面，结果显示各项功能指标都能按照预期顺利完成。

第 7 章 总结与展望

数字资产作为企业或个人珍贵的数字财富，具有很高的经济和收藏价值，但因其数字化的特点，所以会被不法分子利用、篡改和牟利。更令人担心的是，一旦侵权事件发生，大多数人不懂得如何保全证据，而传统存证方式又存在公信力弱、存证机构少以及安全系数低等问题。本系统设计的初衷便是为了解决以上痛点问题：首先，介绍了本课题的研究背景和意义，讨伦了相关课题的国内外研究现状，并进一步阐述了系统的设计目标以及组织安排。然后，依次给出了区块链、数字资产和 IPFS 的相关概念，并对 P2P 网络、共识机制、智能合约以及密码学等区块链核心技术进行了详细地解说。紧接着，分别从功能性需求分析和非功能需求分析两个方面对系统的相关需求进行了描述。在此基础之上，对区块链存证系统进行了概要设计，设计的内容包括系统底层架构、系统整体架构以及系统功能架构等。在以上工作都完成之后，便对系统进行了详细的设计，设计的内容包括存证系统的流程图以及各功能模块的流程图，并给出了功能实现的伪代码，总体上实现了基于区块链的数字资产存证系统。最后，对系统所展示的功能和性能进行了测试，结果显示存证文件上传、存证文件下载、存证文件上链以及存证文件授权等功能都能够顺利地运行。

限于自身能力的限制以及区块链技术尚不成熟，本文设计并实现的区块链存证系统还有许多需要改进和优化的地方。随着业务需求的不断扩充，会相应地出现一些性能上的问题，比如如何处理大量的交易，区块的出块速度应该如何控制才能防止分叉的发生等等。另外，智能合约的安全问题更是重中之重，及时阻断合约的执行会避免进一步的损失，不失为一种有效的应对措施，也是未来工作需要投入更多精力去研究的地方。随着时间的推移以及用户法律意识的不断增强，对存证应用的需求也会增多，因此还需要在原先提供的功能基础之上再新添加若干个其他的功能，以使系统功能更加的丰富，更加贴近人性化的服务要求。

参考文献

[1]李兆森等. 基于区块链的电子数据存证应用研究[J].软件,2017,38(08):63-67.
[2]陈希林. 基于 Hadoop 的电子证据保全平台的研究与实现[D].中国人民公安大学,2017.
[3]刘敖迪等.区块链技术及其在信息安全领域的研究进展[J].软件学报,2018,29(07):2092-2115.
[4]Satoshi N. Bitcoin:a peer-to-peer electronic cash system [EB/OL].https://bitcoin.org/bitcoin.pdf.
[5]李绍民等.区块链多媒体数据版权保护方法研究[J].科技资讯,2015,13(35):13+15.
[6]刘家稷,等.使用双区块链的防伪溯源系统[J].信息安全学报,2018,3(03):17-29.
[7 安瑞,何德彪,张韵茹,等.基于区块链技术的防伪系统的设计与实现[J].密码学报,2017,4(2):199-208.
[8]梅颖.安全存储医疗记录的区块链方法研究[J].江西师范大学学报,2017,41(05):484-490.
[9]梁志勇.基于区块链的高校文件存储系统的探究[J].电脑与电信,2018(03):53-54+63.
[10]李亚楠.基于区块链的数据存储应用研究[D].北京交通大学,2018.
[11]Sharples M, Domingue J. The Blockchain and Kudos: A Distributed System for Educational Record, Reputation and Reward[C]. Springer, Cham, 2016:490-496.
[12]吕坤等.基于区块链的数字资产交易系统设计与实现[J].软件导刊,2018,17(07):209-213.
[13]余其凤等.区块链技术在图书馆数字资产管理中的应用探讨[J].数字图书馆论坛,2018(07):30-36.
[14]郑阳等.区块链技术在数字知识资产管理中的应用[J].出版科学,2018,26(03):97-104.
[15] Zhu Y, Qin Y, Zhou Z, et al. Digital Asset Management with Distributed Permission over Blockchain and Attribute-Based Access Control[C]. IEEE Computer Society, 2018:193-200.
[16]刘阳明.阿里云推出区块链技术邮箱存证产品[J].计算机与网络,2016,42(20):19.
[17]兴业银行股份有限公司西宁分行课题组,高海明.基于区块链技术的合同防伪[J].青海金融,2018(03):55-60.
[18]朱兴雄.区块链技术在电商存证鉴证中的应用[J].现代商业,2018(01):35-36.
[19]雷蕾.从时间戳到区块链:网络著作权纠纷中电子存证的抗辩事由与司法审查[J].出版广角,2018(15):10-14.
[20]李静彧等. 基于区块链存证的电子数据真实性探讨[J].软件,2018,39(06):109-112.
[21]徐蕾. 基于区块链的云取证系统研究与实现[D].西南科技大学,2017.
[22]童丰.公证介入区块链技术司法运用体系初探——从杭州互联网法院区块链存证第一案谈起[J].中国公证,2018(09):60-64+1.
[23]Zhu Y, Qin Y, Zhou Z, et al. Digital Asset Management with Distributed Permission over Blockchain and Attribute-Based Access Control[C]. IEEE Computer Society, 2018:193-200.
[24]韦宇.企业如何有效地进行数字资产管理[J].科教文汇(下旬刊),2018(06):78-79.
[25]袁勇等.区块链技术发展现状与展望[J].自动化学报,2016,42(04):481-494.
[26]We Are Social.2015 年全球数字、社交和移动调查报告 [EB/OL].http://www.199it.com/archives/324011.html.
[27]Vitalik Buterin.Ethereum:A Next-Generation Smart Contract a-nd Decentralized Application Platform [EB/OL].http://ethfans.org/posts/ethereum-whitepaper.

[28] Hyperledger Fabric 白皮书 [EB/OL]. http://hyperledger-fabric.readthedocs.ioen/latest/whatis.html.
[29]邹均等.区块链核心技术与应用[M].北京：机械工业出版社，2018：185-217.
[30]EOS 白皮书[EB/OL].http://github.com/EOSIO/Documentation/blob/master/TechnicalWhitePaper.md.
[31]张俊等.基于区块链的电网大数据数字资产管理架构[J].电力信息与通信技术,2018,16(08):1-7.
[32]喻恒彦. 基于区块链技术的存证系统设计与实现[D].成都理工大学,2018.
[33]冒小乐等.基于区块链的电子数据存证的设计与实现[J/OL].中兴通讯技术:1-12[2019-01-17].http://kns.cnki.net/kcms/detail/34.1228.TN.20181121.1324.002.html.
[34]何泾沙等. 针对集中式电子数据保全系统的数据指纹提取方法及系统:CN106254341A[P]. 2016.
[35]Benet J. IPFS - Content Addressed, Versioned, P2P File System[J]. Eprint Arxiv, 2014.
[36]邓鹏等.Namenode 单点故障解决方案研究[J].计算机工程,2012,38(21):40-44.
[37]殷龙等. 基于 IPFS 的分布式数据共享系统的研究[J]. 物联网技术, 2016, 6(6):60-62.
[38]熊丽兵.精通以太坊智能合约开发[M].北京：电子工业出版社，2018：2-239.
[39]Ongaro,Ousterhout.In Search of an Understandable Consensus Algorithm[EB/OL].https://raft.github.io/raft.pdf.
[40]Dinh T T A , Liu R , Zhang M , et al. Untangling Blockchain: A Data Processing View of Blockchain Systems[J]. IEEE Transactions on Knowledge & Data Engineering, 2017, PP(99):1-1.
[41]Bigi G , Bracciali A , Meacci G , et al. Validation of Decentralised Smart Contracts Through Game Theory and Formal Methods[J]. 2015.
[42]韩璇等.区块链安全问题:研究现状与展望[J].自动化学报,2019,45(01):206-225.
[43]高丽芬等.区块链共识机制之拜占庭算法[J].数字通信世界,2019(01):43+49.
[44]陈伟利等.区块链数据分析：现状、趋势与挑战[J/OL].计算机研究与发展,2018(09)[2018-09-30].ht-tp://kns.cnki.net/kcms/detail/11.1777.TP.20180921.1203.002.html.
[45]Sharples M, Domingue J. The Blockchain and Kudos: A Distributed System for Educational Record, Reputation and Reward[C]. Springer, Cham, 2016:490-496.
[46]Christidis K , Devetsikiotis M . Blockchains and Smart Contracts for the Internet of Things[J]. IEEE Access, 2016, 4:2292-2303.
[47]程丽辰等. 区块链技术及其安全问题[J]. 信息通信技术, 2017(3):39-45.
[48]郭珊珊. 供应链的可信溯源查询在区块链上的实现[D].大连海事大学,2017.
[49]张首晟.量子计算、人工智能与区块链[J].政策,2018(08):34-36.
[50]徐蕾. 基于区块链的云取证系统研究与实现[D].西南科技大学,2017.
[51]张圣垚. 基于区块链的电子病历系统的设计与实现[D].哈尔滨工业大学,2018.
[52]G. Wood. (2013). Ethereum: A Secure Decentralised Generalised Transaction Ledger. [Online]. Available: http://gavwood.com/paper.pdf.
[53]Wu Z , Meng Z , Gray J . IoT-based Techniques for Online M2M-Interactive Itemised Data Registration and Offline Information Traceability in a Digital Manufacturing System[J]. IEEE Transactions on Industrial Informatics, 2017:1-1.
[54] Preden J S , Tammemae K , Jantsch A , et al. The Benefits of Self-Awareness and Attention in Fog and Mist Computing[J]. Computer, 2015, 48(7):37-45.

攻读硕士学位期间的主要成果

[1]张亚伟, 张问银, 王九如, 赵伟. 基于区块链的数字资产管理系统框架设计与分析[J]. 计算机科学与应用,2019, 9(1): 28-37. DOI: 10.12677/csa.2019.91004

[2]赵伟, 张问银, 王九如, 张亚伟. 基于区块链的企业管理系统框架设计与分析[J]. 网络与信息安全学报, 2019, 5(2): 20-29.

参与科研项目

[1] 基于用户交互特性的社会网络情感分析技术研究,国家自然科学基金,项目编号(61273148).

[2] 基于模糊理论的网络舆情分析、评价与对策研究,国家社会科学基金,项目编号(12BXW040).

[3] 云取证关键技术研究,山东省自然科学基金,项目编号(2011ZRB03002).

[4] 基于浅层语义的网络舆情信息分析关键技术研究,山东省自然科学基金(2012ZRB01195).

[5] 面向公共安全的网络舆情信息分析预警及信息源追踪系统研究,山东省科技发展计划(2013GGX10102).

[6] 基于大数据分析的云取证系统研究与实现,山东省科技发展计划项目,项目编号(2014GGB03002).

致谢

时间荏苒，光阴似箭。两年的硕士研究生生涯即将结束，我的毕业论文也在规定的时间内圆满完成。总的来说，论文创作的整个过程并不是一帆丰顺的，既有汲取知识养分、恍然大悟的喜悦，同时也掺杂了许多酸楚与挫折。区块链毕竟是一门刚兴起不久的新技术，自 2017 年 9 月份入学以来，我花费了将近一年的时间去学习和研究它的理论知识，包括阅读大量的相关文献、参加大大小小区块链相关会议十几场以及上机操作并调试了各种典型案例的源代码（比如 Metacoin、CryptoKittes、宠物商店、CryptoZombie 以及积分管理系统等）。研二上半年的时间基本是为毕业论文做准备，学习新东西的同时也夯实了以前所学的知识。

其实我写这篇论文的灵感来自于第七届“中国软件杯”大学生软件设计大赛，当时的赛题名为可信数字资产存证应用。坦白说，我对这个赛题非常的感兴趣，但它限制了区块链的开源平台只能选择 Hyperledger Fabric，而遗憾的是，我从一开始都在和以太坊打交道，所以便放弃了这次项目实践的机会。但当时我就想到能否使用以太坊这个区块链开发平台来实现存证应用，于是便有了本文的论题。

有句俗话说得好，“只要人人都能献出一点爱，世界将会变成美好的人间”，感恩是永久不变的话题，是为了向曾经帮助过自己的人表达最真挚的谢意。我的论文之所以能够如期完成，正是因为有了他人的帮助。在此，首先我要感谢的人是我自己，感谢那个一路走来不轻言放弃的自己；感谢那个即使伤痕累累也要奋发图强的自己；更感谢那个虽处处忍让但坚持己见的自己。然后我要感谢的是临沂大学信息科学与工程学院的张问银教授以及王九如副教授，他们知识渊博而又慈祥可亲，总是悉心指导着我，让我懂得不管是在学习还是在工作中只有脚踏实地才能厚积薄发。再次我要感谢的是山东师范大学信息科学与工程学院信息系统与网络信息安全省高校重点实验室的各位老师和同学们，我们一起度过了一个美好的、值得回忆的研一时光。同时更要感谢与我同门的赵伟、李蓓和田宗晴同学，正是因为有了他们的关怀和陪伴，我才能够坚持到现在。最后特别要感谢我的家人、亲戚和朋友，每次当我感觉学习枯燥乏味的时候他们总会安慰我，给我加油打气，他们永远是我坚强的后盾。

回忆过往，因为过往值得回忆。那些在我最困难的时候帮助过我的人；那些在我人生道路上阻碍我前行的事；那些在我学习过程中遇到的难题和挫折。这些我都会诚然接受，并将其当做日后工作和生活的参照。感谢的话不是一两句话就可以表达彻底的，但只要感情真挚、心存善念，我相信你们也会在自己的人生道路上闪闪发亮！