1. 货币相关问题

问题 1: What is money?

Money is a **medium of exchange**, **unit of account**, and store of value used to facilitate transactions and economic activities.

问题 2: What are the three important functions of money?

The three important functions of money are:

1. Medium of Exchange: Used to buy goods and services.
2. Unit of Account: Provides a standard measure of value.
3. Store of Value: Retains value over time for future use.

问题 3: Do you think cryptocurrencies are good money?

Cryptocurrencies like Bitcoin are partially good money, serving as a medium of exchange and store of value, but their volatility and limited acceptance hinder their role as a stable unit of account. Stablecoins like USDT better fulfill money's functions due to price stability.

2. 比特币相关问题

问题 4: How do we secure our own accounts?

Bitcoin accounts are secured using **public-private key** cryptography, where the private key, kept secret, signs transactions to prove ownership, and losing it results in loss of funds.

问题 5: Who should keep the ledger?

In Bitcoin, the ledger is maintained by a **decentralized network** of nodes, primarily miners, who validate and record transactions in the blockchain without a single controlling entity.

问题 6: How should we trust the ledger maintainer?

Trust in Bitcoin's ledger is achieved through Proof of Work (**PoW**) consensus, where miners compete to solve costly computational puzzles, making manipulation economically impractical.

问题 7: How do we protect ourselves from attacks?

Bitcoin protects against attacks via Proof of Work, requiring attackers to control over 50% of mining power (51% attack), which is costly and impractical due to high energy demands.

问题 8: How does this help privacy?

Bitcoin's public-private key system provides pseudonymity(伪匿名性), linking transactions to addresses rather than real identities, offering some privacy unless linked to personal information.

问题 9: Does this conflict with transparency?

Bitcoin's pseudonymity does not conflict with transparency, as the blockchain is fully public, allowing anyone to verify transactions while protecting user identities.

问题 10: Why or why not?

Bitcoin balances privacy and transparency: pseudonymous addresses protect user identities (privacy), while the public blockchain ensures verifiable transactions (transparency).

问题 11: How does Zcash change this landscape?

Zcash enhances privacy over Bitcoin by using zero-knowledge proofs (zk-SNARKs) to hide transaction details (sender, receiver, amount) while allowing public verification of validity.

问题 12: What is the main difference between Zcash and Bitcoin?

The main difference is privacy: Bitcoin offers pseudonymity with transparent transactions, while Zcash uses zk-SNARKs for optional shielded transactions, hiding details.

问题 13: Proof of Work? How do you describe this in your own words?

Proof of Work (PoW) is a consensus mechanism where miners solve complex mathematical puzzles to validate transactions and secure the blockchain through computational effort.

问题 14: Why are miners willing to spend so much electricity in doing that?

Miners spend electricity on Proof of Work to earn block rewards (newly minted Bitcoins) and transaction fees, which incentivize their costly computational efforts.

问题 15: What is the difficulty of the genesis block?

The difficulty of Bitcoin's genesis block, created on January 3, 2009, was set to the minimum value of 1, as it was the first block.

问题 16: Can you recall two types of scaling solutions?

Two types of scaling solutions for Bitcoin are:

1. Layer 1 (On-Chain): Modifying the protocol, e.g., SegWit to increase block capacity.
2. Layer 2 (Off-Chain): Processing transactions off-chain, e.g., Lightning Network.

问题 17: What's SegWit? What's the original problem that it tried to solve?

SegWit (Segregated Witness) is a Bitcoin upgrade that separates signature data from transaction data, increasing block capacity and fixing transaction malleability to enable Layer 2 solutions like the Lightning Network.

问题 18: What's Lightning Network?

The Lightning Network is a Layer 2 scaling solution for Bitcoin, enabling fast, low-cost off-chain transactions via payment channels, settling final balances on-chain.

问题 19: What's Taproot?

Taproot is a 2021 Bitcoin soft fork introducing Schnorr signatures and MAST to enhance privacy and efficiency, making complex transactions appear like regular ones.

问题 20: What is "ordinals"? How does it change the Bitcoin ecosystem?

Ordinals inscribe data on individual satoshis, enabling NFTs and expanding Bitcoin’s use cases beyond payments to include digital collectibles and art.

3. 加密技术相关问题

问题 21: What is encryption?

Encryption is the process of converting plaintext into ciphertext using a key to protect data confidentiality, accessible only to authorized parties.

问题 22: Symmetric vs. Asymmetric?

Symmetric encryption uses one key for encryption and decryption (e.g., AES), fast but requires secure key sharing. Asymmetric encryption uses a public-private key pair (e.g., RSA), slower but secure for key exchange.

问题 23: What is a hash function?

A hash function is a one-way function mapping data to a fixed-size output (hash) to ensure data integrity, being deterministic, collision-resistant, and irreversible.

问题 24: Know some basic properties of cryptographic hash functions

Basic properties of cryptographic hash functions include:

1. Deterministic: Same input produces the same hash.
2. Collision Resistance: Hard to find two inputs with the same hash.
3. Preimage Resistance: Hard to reverse the hash to the original input.

问题 25: What’s its difference from encryption?

Hash functions are one-way, producing a fixed-size output to verify integrity, and are irreversible, while encryption is reversible to protect confidentiality using keys.

问题 26: How about digital signature?

A digital signature uses asymmetric cryptography (private key to sign, public key to verify) to ensure authenticity, integrity, and non-repudiation of messages or transactions.

问题 27: How digital signature works?

A digital signature works by hashing a message, signing the digest with the sender’s private key, and verifying it with the public key to ensure the message is unchanged and authentic.

问题 28: Why we need it? (or how is it better than our traditional wet ink signature?)

Digital signatures provide cryptographic proof of authenticity, integrity, and non-repudiation, making them harder to forge and more secure than wet ink signatures for digital transactions.

4. 以太坊相关问题

问题 29: What’s added in Ethereum such that it can achieve things that Bitcoin is incapable of?

Ethereum adds smart contracts, programmable code in a Turing-complete language, enabling complex applications like DeFi and DAOs beyond Bitcoin’s payment focus.

问题 30: What new feature we need to go beyond payment only?

Smart contracts are needed, allowing programmable, self-executing agreements for applications like lending, voting, and token creation.

问题 31: How do we safeguard the network while adding this new feature?

Ethereum safeguards smart contracts with the Gas mechanism, limiting computational resources, preventing infinite loops, and charging fees to deter malicious code.

问题 32: Can we predict whether a program will stop within a finite amount of time?

No, the halting problem proves it’s impossible to predict whether a program will stop in finite time for all cases, so Ethereum uses Gas limits.

问题 33: Can you program/code in Bitcoin?

Bitcoin supports limited programming via its scripting language (e.g., for multi-signature transactions), but it’s not Turing-complete, unlike Ethereum’s smart contracts.

问题 34: What’s not good?

Bitcoin’s scripting language is limited, not Turing-complete, and lacks flexibility for complex applications like DeFi or DAOs, restricting it to payments.

问题 35: What is the halting problem?

The halting problem proves no algorithm can determine whether any program will halt or run indefinitely for all inputs.

问题 36: And how does Ethereum deal with it?

Ethereum deals with the halting problem by enforcing Gas limits on smart contract execution, stopping programs that exceed allocated resources.

问题 37: What’s finality? 永久确认且不可逆的保证，通过共识机制实现。

Finality is the assurance that a blockchain transaction is permanently confirmed and cannot be reversed, achieved through consensus mechanisms.

问题 38: How’s finality different between PoW and PoS?

In PoW (e.g., Bitcoin), finality is probabilistic, requiring multiple confirmations to reduce reversal risk. In PoS (e.g., Ethereum post-Merge), finality is deterministic after validator attestations.

问题 39: Recall that one may launch 34% attack in PoS. What are the attacks and how can we counteract such attacks?

A 34% attack in PoS involves controlling one-third of validators to disrupt consensus (e.g., delay finality); countermeasures include slashing malicious validators’ stakes and social coordination.

问题 40: What is Gas in Ethereum? Why is it needed?

Gas is Ethereum’s transaction fee unit, needed to limit computational resources, prevent infinite loops (halting problem), and deter malicious code execution.

问题 41: What is "optimistic rollup"?

Optimistic Rollup is a Layer 2 scaling solution for Ethereum that processes transactions off-chain, assuming validity, and posts compressed data on-chain with fraud proofs.

问题 42: What does it mean by "optimistic" here?

"Optimistic" means transactions are assumed valid, processed off-chain, and only challenged via fraud proofs if disputes arise, reducing on-chain computation.

问题 43: How can we guarantee the validity of transactions submitted via rollup?

Validity is guaranteed by fraud proofs, allowing anyone to challenge invalid transactions within a dispute period, reverting them if proven invalid.

问题 44: What’re the differences between rollup and the Lightning Network?

Rollup processes general Ethereum transactions (e.g., smart contracts) off-chain with fraud or ZK proofs, while Lightning Network focuses on Bitcoin payment channels, settling balances on-chain.

问题 45: What happened in the 2016 DAO hack? What was the bug? What did the developers do to fix the bug?

The 2016 DAO hack exploited a reentrancy bug, allowing an attacker to drain 3.6 million ETH; developers hard-forked Ethereum to recover funds, creating Ethereum Classic.

问题 46: How did it happen?

The DAO hack exploited a reentrancy vulnerability, where a malicious contract repeatedly called the withdrawal function before the balance updated, draining funds.

问题 47: What is the vulnerability that results in the hack?

The vulnerability was reentrancy, allowing a malicious contract to repeatedly call back into the DAO’s withdrawal function before updating the balance.

问题 48: How could we solve it?

Reentrancy can be solved by using mutex locks, updating state before external calls, or using checked functions like `transfer` instead of `call`.

问题 49: What’s the consequence to Ethereum?

The DAO hack led to a hard fork, splitting Ethereum into Ethereum (ETH, reversing the hack) and Ethereum Classic (ETC, preserving the original chain).

问题 50: Ethereum shifted from PoW to PoS. What is PoS (in Ethereum)? Is it better than PoW?

PoS in Ethereum uses staked ETH for validator consensus, reducing energy use compared to PoW, which is more efficient but faces “nothing at stake” risks.

问题 51: What does it mean by "nothing at stake" in a PoS blockchain network like Ethereum?

“Nothing at stake” means PoS validators risk no loss for supporting invalid chains, potentially disrupting consensus, mitigated by slashing penalties.

问题 52: Is the shift from PoW to PoS a layer 1 or layer 2 scaling solution?

The PoW to PoS shift is a Layer 1 scaling solution, altering Ethereum’s core consensus protocol.

5. DAO 相关问题

问题 53: What is the full name of "DAO"?

DAO stands for Decentralized Autonomous Organization, a blockchain-based entity governed by smart contracts and community voting without central authority.

问题 54: What is the principal-agent problem? What are the agency costs?

The principal-agent problem occurs when agents prioritize self-interest over principals; agency costs are expenses to monitor and align interests.

问题 55: What is the agency conflict?

Agency conflict arises when agents favor their interests over principals.

问题 56: How do DAOs mitigate this agency conflict?

DAOs mitigate agency conflict by using transparent, automated smart contract governance and community voting to reduce centralized control and align interests.

问题 57: What are the new challenges?

New DAO challenges include low voter participation, smart contract vulnerabilities (e.g., DAO hack), and uncertain regulatory compliance.

问题 58: What are some common voting approaches?

Common DAO voting approaches include token-based (weighted by token holdings), quadratic (votes cost quadratically), and reputation-based voting.

问题 59: What’s quadratic voting?

Quadratic voting is a system where vote costs increase quadratically (e.g., 1 vote costs 1 token, 2 votes cost 4 tokens), balancing influence and reducing wealthy voter dominance.

问题 60: Does it prevent Sybil attack?

Quadratic voting partially mitigates Sybil attacks by increasing the cost of multiple identities, but additional measures like identity verification are needed.

6. Hyperledger Fabric 相关问题

问题 61: Why are Bitcoin/Ethereum-like blockchains not ideal for most business applications?

Bitcoin and Ethereum are public, permissionless blockchains with slow speeds, high costs, and public data, unsuitable for businesses needing privacy and compliance. Hyperledger Fabric, a permissioned blockchain, addresses these.

问题 62: How does frameworks like Fabric facilitate the adoption of blockchain into businesses?

Hyperledger Fabric provides modular, permissioned blockchain frameworks with privacy, scalability, and customizable consensus, enabling secure business adoption.

问题 63: What is a channel in a Fabric network?

A channel is a private subnet in Hyperledger Fabric with a separate ledger, allowing only channel members to access confidential transactions and data.

问题 64: What problem does it solve? (CA in Hyperledger Fabric)

The Certificate Authority (CA) in Hyperledger Fabric solves identity management by issuing digital certificates to authenticate nodes and users, ensuring authorized access.

问题 65: What benefit does it bring? (Channel in Hyperledger Fabric)

Channels provide data privacy by isolating transactions and ledgers, ensuring only authorized channel members can access sensitive business data.

问题 66: In Fabric, certificate authority is needed for identity management. Why is it important? Recall what’s man-in-the-middle attack?

Certificate Authority ensures trusted identities by issuing certificates, preventing man-in-the-middle attacks where attackers impersonate legitimate parties to intercept data(攻击者冒充合法方拦截数据).

问题 67: Do we need to worry a lot about Byzantine faults?

In Hyperledger Fabric, Byzantine faults are less concerning due to its permissioned nature with pre-approved nodes, though consensus like Raft addresses potential faults.

问题 68: What’s the other type of fault?

The other type is a crash fault, where a node fails or stops responding but does not act maliciously, unlike Byzantine faults.

问题 69: What’s Raft? Can it counteract Sybil attack?

Raft is a crash-fault-tolerant consensus algorithm in Hyperledger Fabric for ordering transactions. It doesn’t counteract Sybil attacks, as Fabric’s permissioned network prevents unauthorized identities.

问题 70: How does Raft ordering service work? At most how many failed nodes can it withstand?

Raft orders transactions via a leader-based consensus, other nodes sync with the leader, tolerating up to (N-1)/2 failed nodes, where N is the total number of nodes.

问题 71: Can Raft solve the Byzantine Generals Problem?

Raft cannot solve the Byzantine Generals Problem, as it assumes non-malicious crash faults, not malicious behavior.

7. 区块链应用相关问题

问题 72: Are all of these volatile(波动)? (Payment tokens)

Not all payment tokens are volatile; cryptocurrencies like Bitcoin are volatile, but stablecoins like USDT, pegged to assets, maintain stable value.

问题 73: We can categorise the token into different types?

Tokens include payment tokens (e.g., Bitcoin), utility tokens (e.g., governance), security tokens (investment assets), and non-fungible tokens (NFTs).

问题 74: What are stablecoins?

Stablecoins are cryptocurrencies pegged to stable assets (e.g., USD, gold) to minimize price volatility, used in trading, payments, and DeFi (e.g., USDT, USDC).

问题 75: What are the different types? What’s its benefit?

Stablecoin types include fiat-backed (e.g., USDT), crypto-backed (e.g., DAI), and algorithmic (e.g., UST). Benefits are price stability, liquidity, and fast cross-border transactions.
稳定币类型包括法币支持（例如 USDT）、加密支持（例如 DAI）和算法型（例如 UST）。

问题 76: What is the difference between NFT and other tokens like Bitcoin/Ether?

NFTs are unique, indivisible digital assets (e.g., art), while Bitcoin and Ether are fungible, interchangeable, and divisible tokens.

问题 77: What is so special about it?

NFTs are special for their uniqueness, verifiable blockchain ownership, and ability to represent digital or physical assets, enabling use cases like digital art and gaming.

问题 78: What is an SFT (ERC-1155)?

SFT (Semi-Fungible Token, ERC-1155) is an Ethereum standard combining fungible and non-fungible properties, managing multiple token types in one contract (e.g., game items).

问题 79: What’s its use case?

ERC-1155 is used in gaming (fungible currency and non-fungible items) and supply chains (fungible batches and unique assets).

问题 80: What is a rentable NFT (ERC-4907)?

A rentable NFT (ERC-4907) is an Ethereum standard allowing temporary transfer of NFT usage rights without transferring ownership, enabling rental markets.

问题 81: How does it help with the current landscape of NFTs?

ERC-4907 enhances NFT accessibility by allowing rentals (e.g., for gaming or virtual land), reducing costs and expanding use cases.
ERC-4907 通过允许租用 NFT（例如用于游戏或虚拟土地）增强可访问性，降低成本并扩展用例。

问题 82: Metaverse: the role of blockchain - data continuity; transaction.

Blockchain ensures data continuity (persistent identities and assets) and secure, transparent transactions in the metaverse.

---

1. 货币基础 (Money Fundamentals)

1.1 货币的定义与历史演变 (Definition and Historical Evolution of Money)

1.1.1 货币的本质 (What is Money?)

Money serves as a medium of exchange, store of value, and unit of account, simplifying transactions and facilitating economic activities. The document poses the question: "What comes to mind when you think of money?" to encourage reflection on the multiple functions of money.

- **交换媒介 (Medium of Exchange)**: Money solves the inefficiency problem of barter. For example, in ancient times, one might need to exchange eggs for cloth, but the needs of both parties might not match; money simplifies transactions as a universal medium.
- **价值储存 (Store of Value)**: Money
  - Example: Gold is often used as a store of value due to its scarcity and durability.
- **记账单位 (Unit of Account)**: Money provides a uniform standard for measuring value.
  - Example: The US dollar as a unit of account in international trade.

Supplementary: Economist Milton Friedman mentioned in "Money Mischief" that the form of money has evolved from stones and feathers to paper currency and electronic data, and may exist in the future as "computer bytes".

1.1.2 货币的功能与加密货币的表现 (Functions of Money and Performance of Cryptocurrencies)

The document cites Charles Wheelan's "Naked Economics" to evaluate the performance of cryptocurrencies in terms of money functions:

- **记账单位 (Unit of Account)**: Grade F (Fail). Due to severe price fluctuations, Bitcoin is not suitable as a stable measure of value.
    - Example: In 2021, Bitcoin's price soared from $30,000to$69,000, making it difficult to use for daily pricing.
- **价值储存 (Store of Value)**: Grade D (Poor). Volatility makes it unstable, but in some situations (such as government collapse), it can serve as a hedge.
    - Example: During Venezuela's hyperinflation, residents turned to Bitcoin to preserve wealth.
- **交换媒介 (Medium of Exchange)**:
    - Ordinary people: Grade C (Average). Due to long median transaction confirmation time and high fees, Bitcoin is inefficient for everyday transactions.
    - Unstable countries: Grade B+ (Good). In areas with failing governments, Bitcoin provides an alternative payment method.
    - Illegal transactions: Grade A+ (Excellent). Its anonymity makes it used for illegal activities such as drug trading.

Supplementary: The Central African Republic was the first country to adopt Bitcoin as legal tender, but the legal status of Bitcoin globally varies greatly, from completely legal to strictly prohibited.

1.2 比特币与区块链的起源 (Origin of Bitcoin and Blockchain)

1.2.1 比特币的提出 (Bitcoin's Proposal)

Bitcoin was proposed by the mysterious figure Satoshi Nakamoto in 2008, aiming to create a peer-to-peer electronic cash system that doesn't require a trusted third party. Its core features include:

- **防止双重支付 (Prevention of Double-Spending)**: Verifying transactions through a peer-to-peer network.
- **匿名性 (Anonymity)**: Participants don't need to reveal their real identities.
- **工作量证明 (Proof-of-Work)**: Generating new coins and maintaining network security through hash calculations (Hashcash).

Example: Satoshi Nakamoto published the Bitcoin white paper on November 1, 2008 via email.

1.2.2 比特币之前的尝试 (Attempts Before Bitcoin)

Bitcoin was not the first attempt at digital currency; the document mentions the following pioneers:

- **PayPal** (1998): An online payment platform, relying on centralized trust.
- **e-gold** (1996): A digital currency backed by gold, shut down due to regulatory issues.
- **b-money** (1998): Proposed by Wei Dai, emphasizing anonymity and decentralization, influenced by the Crypto-Anarchy ideology.
    - **密码朋克 (Cypherpunk)**: An ideology advocating for anarchism through cryptography, believing that encryption can render violence ineffective because participants cannot be tracked.

Supplementary: The domain name bitcoin.org was registered on August 18, 2008, and is hosted on Cloudflare's servers.

1.2.3 区块链1.0 (Blockchain 1.0)

Blockchain is the core technology of Bitcoin, known as "Blockchain 1.0," referring to cryptocurrency blockchains represented by Bitcoin. Its characteristics include distributed ledger and immutable records.

- **定义 (Definition)**: Blockchain is a chain of transactions linked by cryptography, maintained by nodes.
- **Example**: The Bitcoin network records each transaction through timestamps and hashes, forming an unchangeable chain.

Supplementary: Blockchain 1.0 mainly focuses on currency functionality; subsequent Blockchain 2.0 (smart contracts) and 3.0 (decentralized applications, DApps) expanded the application scenarios.

1.3 密码学基础 (A Taste of Cryptography)

Cryptography is the cornerstone of blockchain technology. The document introduces three core concepts: encryption, hash functions, and digital signatures.

1.3.1 加密 (Encryption)

Encryption converts plaintext into ciphertext to protect information security. It is divided into:

- **对称加密 (Symmetric Encryption)**:
  - Uses the same key for encryption and decryption.
  - Example: VPNs (such as ivpn.cityu.edu.hk at City University of Hong Kong) and HTTPS use symmetric encryption to protect data transmission.
- **公钥加密 (Public Key Encryption)**:
  - Uses a pair of keys: public key for encryption, private key for decryption.
  - Example: Alice encrypts a message with Bob's public key, and only Bob's private key can decrypt it.
  - **RSA加密 (RSA Encryption)**: A public key encryption algorithm based on the factorization of large prime numbers.
    - The document provides an RSA decryption exercise, requiring a private key to decrypt ciphertext.

Supplementary: Symmetric encryption is fast but requires a secure channel for key distribution; public key encryption is more secure but computationally complex, often used for key exchange.

1.3.2 哈希函数 (Hash Functions)

Hash functions map data of arbitrary length to values of fixed length (Hash/Digest/Fingerprint), used in blockchain for data integrity verification.

- **特性 (Features)**:
  - **单向性 (One-Way Function)**: It is impossible to derive the original input from the hash value.
  - **抗碰撞性 (Collision Resistance)**: The probability of different inputs producing the same hash is extremely low.
  - **确定性 (Deterministic)**: The same input always produces the same hash.
  - **敏感性 (Sensitivity)**: Small changes in input lead to significant changes in the hash.
- **示例 (Examples)**:
  - Password storage: Websites store hash values of user passwords (e.g., SHA256("123")).
  - **加盐密码 (Salted Passwords)**: Adding a random string (Salt) after the password to enhance security.

        - Example: SHA256("123"+"FDAF#!2") produces a different hash.
    - **比特币核心 (Bitcoin Core)**: Uses SHA256 to verify transaction and file integrity.
    - **BitTorrent**: Uses hashes to verify the integrity of downloaded files.

Exercise: The document asks whether "MyHash" is a valid hash function; due to its lack of one-way functionality and collision resistance, it does not meet cryptographic requirements.

1.3.3 数字签名 (Digital Signature)

Digital signatures are used to verify the authenticity and integrity of messages, superior to wet-ink signatures and electronic signatures.

- **机制 (Mechanism)**:
    - The sender signs the hash value of the message with their private key, and the recipient verifies it with the public key.
    - Example: Alice signs the message hash, and Bob verifies it using Alice's public key, ensuring the message has not been tampered with.
- **优势 (Advantages)**:
    - **不可伪造 (Unforgeable)**: Because only the sender holds the private key.
    - **可验证 (Verifiable)**: Recipients can confirm the validity of the signature.
- **应用 (Applications)**: Bitcoin transactions verify ownership through digital signatures.

Supplementary: Digital signatures are based on asymmetric encryption (such as RSA or ECDSA) and are widely used in software distribution and blockchain transactions.

1.3.4 加密与哈希函数的区别 (Differences Between Encryption and Hash Functions)

- **加密 (Encryption)**: A reversible process aimed at protecting data privacy, where ciphertext can be decrypted back into plaintext using a key.
- **哈希函数 (Hash Functions)**: An irreversible process that generates a fixed-length digest, used to verify data integrity, with no concept of decryption.

Example: Encryption protects bank account information, while hashing verifies whether a file has been tampered with.

1.4 区块链的重要性 (Why Blockchain Matters?)

Blockchain addresses the dependence on centralized institutions in traditional financial systems through decentralization, transparency, and immutability.

- **去中心化 (Decentralization)**: No single point of control, reducing trust costs.
- **透明性 (Transparency)**: Transaction records visible to all nodes.
- **安全性 (Security)**: Cryptography ensures data cannot be tampered with.

Example: The Bitcoin network can complete global transfers without banks, reducing fees and improving efficiency.

Supplementary: The applications of blockchain have expanded to supply chain management, medical records, voting systems, and other areas.

2. 比特币 (Bitcoin)

2.2 比特币的起源 (Origin of Bitcoin)

2.2.1 白皮书发布 (White Paper Publication)

Bitcoin was proposed by an anonymous person or team, **Satoshi Nakamoto**, on November 1, 2008, through a white paper titled "Bitcoin: A Peer-to-Peer Electronic Cash System." Its main features include:

- **Peer-to-Peer (P2P)**: Direct transactions without intermediaries.
- **No Trusted Third Party**: Not dependent on banks or governments.
- **Double-spending Prevention**: Using blockchain to ensure the same Bitcoin cannot be used repeatedly.
- **Anonymity**: Users don't need to reveal their real identities.
- **Proof-of-Work (PoW)**: Generating new Bitcoin through computation, protecting network security.
- **Explanation**: Bitcoin is like a decentralized bank where transactions are completed directly between users, recorded on the blockchain, preventing cheating.
- **Example**: When Bob sends Alice 1 Bitcoin, the blockchain record ensures Bob cannot use this 1 Bitcoin to buy something else.

2.2.2 P2P网络 (P2P Network)

Bitcoin is based on a **P2P Network**, similar to file-sharing systems (like BitTorrent). The characteristics of P2P networks include:

- Decentralization, with no single control point.
- Equal nodes, jointly maintaining the network.
- The document mentions **Napster** (an early P2P music sharing platform) to illustrate P2P technology applications before Bitcoin.
- **Explanation**: A P2P network is like a group of friends sharing files, without a central server, where everyone contributes.
- **Example**: In the Bitcoin network, thousands of nodes worldwide jointly verify transactions, preventing single points of failure.

2.5 比特币的技术原理 (Technical Principles of Bitcoin)

2.5.1 账户与密钥 (Accounts and Keys)

Bitcoin users manage accounts through **Public/Private Key Pairs**:

- **Private Key**: A secret number used to sign transactions and prove ownership.
- **Public Key**: Generated from the private key through elliptic curve cryptography, used to receive Bitcoin.
- **Bitcoin Address**: Generated by hashing the public key, 160 bits in length, including a checksum.
- Private keys cannot be reverse-engineered to derive public keys, providing high security (the document mentions the challenge being at the level of atoms in the universe, approximately 10^78-10^82).
- **Explanation**: A private key is like a bank card PIN, a public key is like an account address, and the address is a shortened version of the public key.
- **Example**: Alice's Bitcoin address is "1A1zP1eP5QGefi2DMPTfTL5SLmv7DivfNa," Bob sends Bitcoin through this address.

2.5.2 区块链与共识 (Blockchain and Consensus)

Bitcoin transactions are recorded on the **Blockchain**, similar to the oral ledger of **Rai Stone** from **Yap Island**:

- Transactions are ordered through a **Consensus Algorithm** to prevent tampering.
- **Byzantine Generals Problem (BGP)**: Solving the trust problem in distributed systems, ensuring the majority of nodes reach an agreement.
- **Proof-of-Work (PoW)**: Miners compete for the right to record by solving computational puzzles (Hashcash), with winners adding new blocks and receiving **Block Rewards**.
  - **Prevention of double-spending**: Ensuring transaction uniqueness through computation, preventing the same funds from being used multiple times.
  - **Network security**: Attackers would need to control over 50% of computing power (51% Attack) to tamper with the blockchain, which is extremely costly in high-power networks.
  - **Decentralization**: No need for a trusted third party; any node can participate in verification.
  - **Incentive mechanism**: Rewards miners, promoting network maintenance.
- **Explanation**: Blockchain is like a public ledger that everyone can view, with a consensus algorithm ensuring no one can cheat.
- **Example**: Miners solve mathematical puzzles, record Alice's payment to Bob, and receive a 12.5 Bitcoin reward.

2.5.3 双重支付问题 (Double-Spending Problem)

Double-spending is a challenge for digital currencies that Bitcoin solves through:

- **P2P Network**: All nodes verify transactions, preventing repeated payments.
- **Proof-of-Work**: Tampering with transactions requires recalculating all subsequent blocks, which is extremely costly.
- **Confirmation**: Transactions need to wait for 6 block confirmations (about 1 hour) to reduce the risk of tampering.
- **Explanation**: Double-spending is like using the same banknote to buy two things; Bitcoin prevents this problem through blockchain.
- **Example**: When Alice tries to use 1 Bitcoin to buy both coffee and a phone simultaneously, the network rejects the second transaction.

2.5.4 挖矿的方式 (Mining)

Mining is the practical operation of PoW in blockchain, using Bitcoin as an example, the process is as follows:

1. **Collecting transactions**: Miners collect unconfirmed transactions from the Mempool to form a Candidate Block.
2. **Constructing the block header**: The block header includes:
   - Previous Block Hash
   - Merkle Root of transactions
   - Timestamp
   - Nonce
   - Difficulty Target
3. **Computing the hash**: Miners use the SHA256 hash function to calculate the block header, generating a 256-bit hash value.

4. **Adjusting the Nonce**: The hash value must be less than the network-set difficulty target (e.g., starting with multiple "0"s). If not, miners adjust the Nonce and recalculate until a suitable hash is found.
5. **Broadcasting the block**: After finding a valid hash, miners broadcast the block to the network, and other nodes verify its legitimacy.
6. **Receiving rewards**: Upon verification, the block is added to the blockchain, and miners receive a **Block Reward** (newly generated Bitcoin) and **Transaction Fees**.

Example:

- The Bitcoin network requires the hash value to start with a certain number of "0"s (like 0000000000000000...). Miners may need to try billions of Nonce combinations, consuming significant computational resources.
- In 2025, the Bitcoin block reward will be 3.125 BTC (due to the Halving mechanism, where rewards are halved every four years).

2.5.5 最长链原则 (Longest Chain Rule)

When multiple miners in the Bitcoin network simultaneously mine new blocks (a fork), nodes select and follow the branch containing the most proof-of-work (the longest chain). This ensures the network eventually reaches consensus and prevents double-spending.

- **Explanation**: If there's a divergence in the blockchain, everyone follows the longest chain, as it represents the most computational investment.
- **Example**: When two blocks appear simultaneously in the network, nodes temporarily preserve both branches; once one branch generates new blocks, becoming the longest chain, nodes abandon the other branch.

2.5.6 51%攻击 (51% Attack)

Building a private chain:

- Attackers secretly mine in a private environment, generating an alternative chain (Fork), not broadcasting to the network.
- With control of the majority of computing power, the attacker's private chain may grow faster than the public chain (Main Chain).

If a miner or group of miners controls more than 50% of the total computing power (Hashrate) of the Bitcoin network, theoretically they could:

- Prevent new transactions from being confirmed.
- Double-spend their own Bitcoin (by transacting on different branches and making their branch the longest chain).
  However, the cost of launching a 51% attack is extremely high and would damage the value of Bitcoin, making it economically unfeasible for attackers in most cases.
- **Explanation**: If bad actors control more than half of the mining power, they could theoretically cheat, but the cost would be very high.
- **Example**: An attacker with 51% computing power could reverse their recent transaction and pay the same Bitcoin to someone else.

2.5.7 动态难度调整 (Difficulty Adjustment)

Bitcoin mining difficulty adjusts based on the total network computing power, with the goal of maintaining an average block time of about 10 minutes. Every 2016 blocks (about two weeks), the network checks the past block times; if the time is less than 10 minutes, the difficulty increases; if longer than 10 minutes, the difficulty decreases.

- **Explanation**: Mining difficulty automatically adjusts based on the number of miners and computing power, ensuring a new block is generated roughly every 10 minutes.
- **Example**: If many new miners join and speed up block generation, the difficulty will increase to slow mining down again.

2.5.8 每四年减半的机制 (Halving)

Approximately every four years (or precisely, every 210,000 blocks), the new Bitcoin reward that miners receive through mining is halved. This mechanism ensures that the total amount of Bitcoin will eventually approach the limit of 21 million, controlling the currency issuance rate and giving it a deflationary nature.

- **Explanation**: Every certain period, the Bitcoin reward miners receive for new blocks is halved, limiting the total amount of Bitcoin.
- **Example**: In 2009, 50 BTC/block; in 2012, 25 BTC; in 2020, 6.25 BTC; in 2024, 3.125 BTC.

2.5.9 矿工的激励机制 (Incentives of Miners)

There are two main motivations for miners to participate in mining:

1. **Block Reward**: Newly issued Bitcoin received after successfully mining a new block.
2. **Transaction Fees**: Fees paid by transactions included in the block.
   These rewards encourage miners to invest computational resources in maintaining network security and processing transactions.

- **Explanation**: Miners provide computing power to earn new Bitcoin and transaction fees.
- **Example**: A miner successfully packages a block containing 100 transactions, receiving both the block reward and the total fees of these 100 transactions.

2.5.10 未花费交易输出模型 (The UTXO Model, Unspent Transaction Output Model)

Bitcoin uses the UTXO model to track Bitcoin ownership rather than an account balance model like banks. Each Bitcoin is locked in a UTXO, and when transacting, users spend one or more UTXOs as inputs and generate new UTXOs as outputs (recipient's new unspent transaction output and change). This provides better privacy and prevents double-spending.

- **Explanation**: Bitcoin doesn't track how much money someone has (account balance) but tracks which transaction outputs haven't been spent (UTXO).
- **Example**: Alice receives 1 BTC from Bob, and this 1 BTC becomes a UTXO. When Alice wants to spend 0.5 BTC, she spends this 1 BTC UTXO as input, generating two new UTXOs as output: 0.5 BTC to the recipient and a 0.5 BTC change to herself.

2.6 比特币的升级与扩展 (Bitcoin Upgrades and Extensions)

2.6.1 Taproot升级 (Taproot Upgrade)

The **Taproot Upgrade** (implemented in 2021) is a **Soft Fork** of Bitcoin, implemented through **BIP 342** (Bitcoin Improvement Proposal):

- **TapScript**: Extends the script language, supporting **Schnorr Signatures** and **Merkleized Abstract Syntax Trees (MAST)**.
- Functions: Enhances transaction privacy, reduces complex contract costs, but is still not **Turing Complete**, unlike Ethereum that supports complex programs.

- **Explanation**: Taproot is like adding new features to Bitcoin, making transactions more private and efficient, but with limited programming capabilities.
- **Example**: Taproot makes multi-signature transactions look like regular transactions, hiding complex contract details.

2.6.2 隔离见证 (SegWit)

Segregated Witness (SegWit) (implemented in 2017) is a soft fork of Bitcoin that solves transaction capacity issues:

- Separates signature data (Witness Data) from transaction data, increasing effective block capacity.
- Fixes the **Transaction Malleability** problem, supporting subsequent extensions (like Lightning Network).
- **Bitcoin SV (Satoshi's Vision)** (a 2018 **Hard Fork**): Opposes SegWit, advocates for larger blocks (128MB), led by Craig Wright (who claims to be Satoshi Nakamoto).
- **Explanation**: SegWit is like storing transaction signatures separately, freeing up space for blocks to hold more transactions.
- **Example**: After SegWit, a 1MB block can accommodate more transactions, reducing transaction fees.

2.6.3 闪电网络 (Lightning Network)

The **Lightning Network** is a **Layer 2** extension protocol for Bitcoin, solving problems of slow transactions and high fees:

- **Payment Channel**: Two parties create a channel, recording multiple transactions, only going on-chain when opening (**Funding TX**) and closing (**Closing TX**) the channel.
- Advantages:
    - Fast: Transactions are almost instant, without waiting for block confirmations.
    - Low cost: In-channel transactions avoid on-chain fees.
- Disadvantages:
    - High cost of opening a channel.
    - Both parties need to be online, limited liquidity.
- **Layer 1 vs. Layer 2**:
    - **Layer 1**: The blockchain itself (like Taproot).
    - **Layer 2**: Extension protocols (like Lightning Network).
- **Explanation**: Lightning Network is like two people keeping records in a private ledger, only making them public at final settlement.
- **Example**: Alice and Bob transact daily coffee money through Lightning Network, requiring only two on-chain transactions.

2.6.4 交易确认时间 (Transaction Confirmation Time)

Bitcoin transactions need to wait for 6 block confirmations (about 1 hour), as explained in the white paper:

- If an attacker controls q% of the network's **Hashrate**, waiting for z blocks ensures the attack success probability is below 0.1%:
    - q=10%, z=5.
    - q=30%, z=24.

- The document references a probability table (at q=30%, the success rate with 6 blocks is about 13%), hence 6 blocks is an empirical value.
- **Explanation**: 6 block confirmations are like waiting for 6 locks, ensuring transaction security.
- **Example**: After Alice pays Bob, Bob waits for 6 block confirmations before shipping, reducing the risk of attack.

2.7 匿名性与假名性 (Anonymity and Pseudonymity)

Bitcoin is not completely anonymous but rather exhibits **Pseudonymity**:

- **Anonymity**: Completely hiding identity.
- **Pseudonymity**: Using pseudonyms (like Bitcoin addresses) that can be tracked through on-chain analysis.
- **Explanation**: Bitcoin addresses are like screen names that may be linked to real identities.
- **Example**: Alice pays with address "1A1zP...", but police trace her identity through exchange records.

2.8 比特币的价值来源 (Source of Bitcoin's Value)

Bitcoin's value comes from people's trust in it, similar to fiat currency:

- **Bryan Routledge**: Bitcoin has value because "people think it has value."
- **Medium of Exchange**: Bitcoin can be used for payments, but acceptance is limited.
- **Store of Value**: Due to high volatility, its storage function is relatively weak.
- **Explanation**: Bitcoin's value is like gold, supported by market confidence.
- **Example**: In 2021, Tesla accepted Bitcoin payments but cancelled due to volatility.

3. 以太坊 (Ethereum)

3.1 比特币的扩展与以太坊 (Bitcoin Extensions and Ethereum)

比特币有多种扩展协议，如彩色币和序数理论，但这些扩展受限于比特币本身的编程能力。以太坊则提供了更完整的编程平台。

- **彩色币 (Colored Coins)**: 通过"着色"比特币来代表特定资产或凭证，可用于产权登记等场景。
- **序数理论 (Ordinal Theory)**: 为每个聪（satoshi）分配唯一编号，使其具有非同质化特性，类似于NFT。
- **解释**: 这些扩展尝试在比特币基础上构建更复杂的应用，但受限于比特币脚本语言的局限性。
- **示例**: 某用户可能用1个彩色币代表其房产所有权；另一用户可能在特定聪上铭刻数字艺术作品。

3.2 以太坊概述 (Ethereum Overview)

3.2.1 以太坊简介 (Introduction to Ethereum)

Ethereum was proposed by Vitalik Buterin in 2013 and launched in 2015. It is a decentralized platform supporting smart contracts, based on blockchain technology.

- **Features**:
  - **Programmable Blockchain**: Supports Turing-complete programming languages (such as Solidity), allowing developers to create complex applications.
  - **Smart Contracts**: Automatically executed programs running on the Ethereum Virtual Machine (EVM).

    - **Consensus Mechanism**: Used Proof-of-Work (PoW) before "The Merge" in 2022, then transitioned to Proof-of-Stake (PoS).
- **Explanation**: Ethereum is like a globally shared computer where anyone can upload and run programs.
- **Example**: A company uses Ethereum smart contracts to automatically pay employee salaries without manual intervention.

3.2.2 与比特币的对比 (Comparison with Bitcoin)

- **Similarities**:
    - Both are decentralized blockchains using public/private key pairs to manage accounts.
    - Both are based on P2P networks and maintain ledgers through consensus algorithms.
- **Differences**:
    - **Programming capability**: Ethereum supports smart contracts, while Bitcoin is limited to simple scripts (not Turing-complete).
    - **Consensus mechanism**: Bitcoin always uses PoW, while Ethereum has transitioned to PoS.
    - **Usage**: Bitcoin primarily serves as a digital currency (medium of exchange) and store of value, while Ethereum supports DApps, NFTs, and more.
- **Explanation**: Bitcoin is like digital gold, while Ethereum is like a digital operating system.
- **Example**: Bitcoin is used for cross-border remittances, while Ethereum is used to create decentralized exchanges.

3.2.3 EOAs vs CAs (Externally Owned Accounts vs Contract Accounts)

- **EOAs (Externally Owned Accounts)**
    - **Definition**: Accounts controlled by users through private keys, representing individuals or entities in the Ethereum network.

      Creation: Generated by creating a pair of public and private keys, with the public key deriving the address (160-bit hash, starting with 0x).

      Control: The private key holder controls the account by signing transactions.

      Functions:
        - Send transactions (transfer ETH, call smart contracts).
        - Hold ETH and ERC-20 tokens.
        - Pay Gas fees (page 65: Gas limit issue).
- **CAs (Contract Accounts)**
    - **Definition**: Accounts controlled by smart contract code, stored on the Ethereum blockchain, executing predefined logic.

      Creation: Created by deploying smart contracts (including bytecode) through EOAs, generating contract addresses.

      Control: Controlled by smart contract code logic, no private keys, requiring calls from EOAs or other CAs.

      Functions:
        - Execute smart contract logic (such as DeFi, NFTs, DAOs).
        - Hold ETH and tokens.
        - Automatically respond to transactions or events.

3.3 以太坊的共识机制：权益证明（PoS

3.3.1 PoS简介 (Introduction to PoS)

Ethereum transitioned from PoW to PoS in 2022 through "The Merge," with validators participating in block creation by staking ETH.

- **Mechanism**: Validators stake at least 32 ETH, are randomly selected to create blocks, and receive transaction fees (Gas Fees) and block rewards.
- **Advantages**:
    - Low energy consumption: PoW requires extensive computation, PoS only requires staking.
    - Environmentally friendly: Reduces carbon footprint.
- **Explanation**: PoS is like shareholder voting, where those holding more ETH have a greater chance of recording transactions.
- **Example**: Alice stakes 32 ETH, becomes a validator, and receives a 0.1 ETH reward after validating transactions.

3.3.2 PoS的问题：无利害关系（Nothing at Stake） (PoS Problem: Nothing at Stake)

PoS faces the "Nothing at Stake" problem: validators might simultaneously validate blocks on multiple branches (forks), increasing the risk of double-spending.

- **Solutions**:
    - **Slashing**: When malicious behavior (such as double voting) is detected, a portion of the validator's ETH is destroyed, and they may be banned from further staking.
    - **Finality**:
        - **Probabilistic Finality**: In PoW, the more block confirmations, the harder it is to tamper with the blockchain.
        - **Cryptoeconomic Finality**: PoS ensures blocks are irreversible through economic incentives.
- **Explanation**: Nothing at Stake is like a cheater supporting two teams simultaneously; slashing is like confiscating their bet.
- **Example**: Bob attempts to validate blocks on two branches, is detected by the system, and loses 10 ETH.

3.3.3 PoS攻击场景 (PoS Attack Scenarios)

- **Attack 1: 33% attack**:
    - If attackers control 33% of staked ETH, they can potentially disrupt consensus through inactivity or double voting.
    - **Solution**: Increase the proportion of active validators, set stricter penalties.
- **Attack 2: 51% attack**:
    - Controlling 51% of staked ETH allows complete control of the network, but the cost is extremely high (requiring purchase of large amounts of ETH).
- **Attack 3: Sybil Attack**
    - Attackers generate multiple fake nodes or accounts, posing as legitimate participants in the network.
    - They participate in network voting, consensus, or data propagation, attempting to influence outcomes.
    - Decentralized systems rely on the honest behavior of most nodes; Sybil attacks undermine this trust through "strength in numbers."

- **Explanation**: A 33% attack is like minority shareholders causing disruption, while a 51% attack is like controlling shareholders manipulating a company.
- **Example**: An attacker controlling 33% of ETH refuses to validate new blocks, pausing the network until the slashing mechanism takes effect.

3.4 以太坊的扩展性解决方案 (Ethereum's Scalability Solutions)

3.4.1 扩展性问题与区块链三难困境 (Scalability Issues and the Blockchain Trilemma)

Ethereum faces scalability challenges: balancing decentralization, security, and scalability, known as the **Blockchain Trilemma**.

- **Problem**: High transaction volumes lead to network congestion, causing gas fees to skyrocket.
- **Explanation**: The trilemma is like cooking where you can only choose two out of fast, good, and cheap; you cannot have all three.
- **Example**: During the 2021 NFT boom, single transaction fees on Ethereum reached $100.

3.4.2 Layer 1扩展：链上解决方案 (Layer 1 Scaling: On-Chain Solutions)

- **The Merge**: Transition from PoW to PoS, reducing energy consumption and laying the foundation for future scaling.
- **Sharding**:
  - Dividing the blockchain database into multiple shards, each processing a portion of transactions, improving throughput.
  - **Proto-Danksharding**: Preliminary sharding plan, optimizing data storage, reducing Layer 2 costs.
- **Explanation**: Sharding is like dividing a large library into smaller libraries, allowing readers to borrow different books simultaneously.
- **Example**: After Ethereum sharding, Alice's NFT transaction and Bob's payment transaction are processed on different shards without interference.

3.4.3 Layer 2扩展：链下解决方案 (Layer 2 Scaling: Off-Chain Solutions)

Layer 2 processes transactions outside the main chain (Layer 1), reducing the burden on the main chain, and is divided into the following types:

3.4.3.1 Rollups

Rollups execute transactions off-chain, submitting compressed data to the main chain.

- **Types**:
  - **Optimistic Rollups**: Assume off-chain transactions are valid, setting a challenge period for disputes. Examples include Arbitrum.
  - **ZK-Rollups**: Use zero-knowledge proofs to verify transactions, more efficient but computationally complex.
- **Advantages**:
  - Lower transaction fees (Gas Fees).
  - Faster transaction speed.
  - Retain main chain security.
- **Disadvantages**:

    - Challenge periods may delay fund withdrawals.
    - Increased complexity makes development more difficult.
- **Comparison with Lightning Network**:
    - **Similarities**: Both are Layer 2 solutions, reducing main chain burden.
    - **Differences**: Lightning Network is based on payment channels, suitable for small payments; Rollups support complex smart contracts.
- **Explanation**: Rollups are like packaging many transactions into a small package, recording only the package information on the main chain.
- **Example**: Alice trades NFTs through Arbitrum with low fees, requiring only final result verification on the main chain.

3.4.3.2 Sidechains（侧链）

Sidechains are independent blockchains running parallel to Ethereum, interacting with the main chain through bridges.

- **Features**:
    - Faster block times and larger block sizes, improving throughput.
    - **Polygon PoS**: Ethereum's PoS sidechain, with an average block time of about 2 seconds.
- **Disadvantages**:
    - Reduced security: Not fully reliant on the main chain.
    - Decreased decentralization: Potential emergence of super nodes.
- **Explanation**: Sidechains are like auxiliary roads to a main road, faster but slightly less secure.
- **Example**: Bob quickly trades game tokens on the Polygon sidechain, with fees only 1/10 of the main chain.

3.5 智能合约与安全问题

3.5.1 智能合约特点 (Smart Contract Characteristics)

- **Immutable**: Code cannot be changed after deployment, unless using a proxy contract.
- **Storage required**: Full nodes need to store all smart contract code.
- **Gas limit**: Execution requires Gas payment; if Gas is insufficient, the transaction fails but Gas is not refunded.
- **Explanation**: Smart contracts are like vending machines, operating according to preset rules that cannot be changed midway.
- **Example**: Alice's smart contract did not execute due to insufficient Gas, resulting in a loss of 0.01 ETH.

3.5.2 安全漏洞 (Security Vulnerabilities)

- **Reentrancy Attack**: A contract is repeatedly called before payment completion, resulting in stolen funds.
    - **Example**: The 2016 DAO attack, where attackers stole 3.6 million ETH through a reentrancy vulnerability.
- **Storage Collision**:
    - Conflict between proxy contract and implementation contract storage layouts, causing data errors.
    - **Example**: The 2022 Aulius governance attack, where storage collision led to control of the governance mechanism.
- **Infinite loops**: Erroneous code could cause the blockchain to freeze, but Gas limits prevent this problem.

- **Explanation**: A reentrancy attack is like pressing the withdrawal button multiple times at an ATM before it dispenses cash; storage collision is like two books using the same shelf numbering.
- **Example**: An attacker exploits a reentrancy vulnerability to steal 10 ETH from Alice's contract.

3.6 以太坊与去中心化互联网

3.6.1 互联网的中心化问题 (Centralization Issues of the Internet)

The document quotes Leopold Kohr: "Whenever something is wrong, something is too big." The current internet is highly centralized, controlled by a few tech giants.

- **Example**: Twitter (now X) is operated by a single company, controlling user data and content moderation.
- **Explanation**: Centralization is like all data being stored on one company's servers, making it easily controlled or leaked.
- **Example**: Twitter's 2022 policy change restricting users from sharing others' real-time locations.

3.6.2 Tor与去中心化 (Tor and Decentralization)

Tor (The Onion Routing Project) protects privacy through multi-layer encryption (Onion Routing) but is not completely decentralized, relying on a limited number of relay nodes.

- **Problem**: Relay nodes can become bottlenecks, reducing the degree of decentralization.
- **Explanation**: Tor is like forwarding letters through multiple post offices to hide the sender, but the number of post offices is limited.
- **Example**: Alice uses Tor to access websites, with data encrypted and transmitted through multiple relays, protecting her privacy.

3.6.3 以太坊的去中心化潜力 (Ethereum's Decentralization Potential)

Ethereum provides decentralized alternatives through DApps, such as decentralized social media.

- **Example**: Lens Protocol, a decentralized social platform based on Ethereum, where users control their data.
- **Explanation**: Decentralized social media is like users storing their own posts without relying on platforms.
- **Example**: Bob publishes content on Lens, with data stored on Ethereum that platforms cannot delete.

4. DAO和私有链 (DAO and Private Chain)

4.1 传统组织中的代理问题 (Agency Problems in Traditional Organizations)

4.1.1 代理理论（Agency Theory）

Agency Theory examines the relationship between asset owners (principals) and those employed to manage these assets (agents). Due to misalignment of interests, agents may act against the principals' interests, leading to agency conflicts.

- **Definition**: Principals (such as shareholders) entrust their assets to agents (such as corporate executives) for management, but agents may pursue personal interests (such as high salaries or power), harming the principals' interests.
- **Example**: In the Luckin Coffee Scandal, executives falsified 2.2 billion RMB in sales data, exaggerated costs and expenses, ultimately resulting in the company paying $180 million in fines (document page 3). This demonstrates how agents can harm shareholder interests through fraudulent behavior.

4.1.2 四种代理成本（Four Types of Agency Costs）

1. **监控代理人（Monitoring Agents）**：
   - **Definition**: Principals need to spend resources to monitor agents to prevent self-serving behaviors. This includes audits, internal controls, etc.
   - **Case**: In the Enron Scandal (document page 7), CEO Jeff Skilling and former CEO Ken Lay deceived shareholders by hiding enormous debts, causing shareholders to lose $74 billion. Whistleblower Sherron Watkins exposed the problem, showing the consequences of insufficient monitoring.
   - **Explanation**: Monitoring costs are expensive because they require hiring auditors, establishing compliance systems, etc., but may still not completely prevent fraud.
2. **监控公司运营（Monitoring Firm Operations）**：
   - **Definition**: Principals need to obtain information about company operations to reduce information asymmetry. Information asymmetry means agents know more about the company than principals.
   - **Case**: Wells Fargo was fined $3 billion for opening millions of accounts without customer authorization (document page 8). Shareholders could not detect the problem in time due to lack of transparent information.
   - **Explanation**: Information asymmetry makes it difficult for shareholders to determine whether executive decisions are reasonable, increasing trust costs.
3. **过度开支（Excessive Expenses）**：
   - **Definition**: Agents may waste resources through high managerial perks or excessive executive compensation.
   - **Case**: Former WeWork CEO Adam Neumann was accused of treating employees improperly and squandering company funds (document page 9). For example, he used company funds to purchase a private jet, harming shareholder interests.
   - **Explanation**: Excessive expenses directly reduce company profits, lowering shareholder returns.
4. **未实现利润（Unrealized Profits）**：
   - **Definition**: The company misses profit opportunities due to suboptimal executive decisions by agents.
   - **Case**: Yahoo!'s multiple wrong decisions (document page 10), such as refusing to acquire Google for $1 million in 1998, rejecting Microsoft's$44.6 billion acquisition offer in 2008, and finally being acquired by Verizon for $5 billion in 2016. These decisions led to enormous value loss.
   - **Explanation**: Executive decision failures may stem from lack of ability or personal interest-driven motives, damaging the company's long-term value.

4.2 DAO的定义与优势

4.2.1 DAO的定义 (Definition of DAO)

A DAO is an organization that operates through blockchain technology, characterized by being decentralized and autonomous. It uses smart contracts to automatically execute rules, reducing dependence on traditional managers.

- **Definition**: A DAO is a blockchain-based organizational form where all rules and transaction records are stored on the blockchain, automatically executed through code, and members participate in decision-making through voting.
- **Example**: Uniswap is a DAO that allows users holding UNI tokens to vote on protocol upgrades and fund allocations (document pages 17-20).

- **Supplementary**: The "decentralization" of DAOs means there is no single controlling entity, with decisions made collectively by the community; "autonomy" means rules are automatically executed by code without human intervention.

4.2.2 DAO如何缓解代理问题 (How DAOs Mitigate Agency Problems)

DAOs reduce the agency costs of traditional organizations in the following ways (document pages 6, 11-12):

1. **减少对代理人的依赖 (Reducing Dependence on Agents)**:
   - DAOs transfer decision rights to investor-owners through smart contracts, reducing dependence on fund managers. For example, investment DAOs allow members to directly vote on fund usage, preventing managers from misappropriating funds.
   - **Example**: In traditional funds, managers might invest in low-return projects to receive kickbacks; in DAOs, investment decisions require community voting approval, increasing transparency.
2. **提高信息透明度 (Improving Information Availability)**:
   - The blockchain's public ledger records all transactions and decisions, reducing information asymmetry. Members can view fund flows and operational status in real-time.
   - **Example**: Uniswap's governance proposals (such as the "Uniswap Delegate Reward") are publicly available on-chain, and all UNI token holders can view and vote (document page 18).
   - **Explanation**: Transparency reduces agents' ability to hide information, making it easier for principals to monitor.
3. **新型治理结构 (Novel Governance Structure)**:
   - DAOs define governance rules through smart contracts, giving principals more choices and reducing dependence on agents. For example, members can directly propose and vote without going through executives.
   - **Supplementary**: Governance structures include the allocation of decision rights, incentive mechanisms, and accountability systems; DAOs ensure fair execution of these mechanisms through code.

4.3 DAO的治理机制 (Governance Mechanisms of DAOs)

4.3.1 治理的定义 (Definition of Governance)

- **Definition**: Governance is the means by which organizations coordinate decision rights, incentives, and accountability (document page 11). In DAOs, governance is achieved through blockchain and smart contracts, ensuring decisions are transparent and automated.
- **Example**: Uniswap's governance process includes proposal discussion (Governance Forum), preliminary voting (Snapshot), and formal on-chain voting (Governance Portal), ensuring community participation (document page 17).

4.3.2 治理的三个核心要素 (Three Core Elements of Governance)

1. **决策权 (Decision Rights)**:
   - **Definition**: Decision rights determine who has the authority to make key decisions. In DAOs, decision rights are typically decentralized, determined collectively by token holders.
   - **Example**: In Uniswap, users holding UNI tokens can vote on protocol upgrades, replacing the traditional model where CEOs make decisions.
   - **Supplementary**: The degree of decentralization of decision rights varies by DAO; some DAOs may still have core teams retaining partial control.

2. **问责制 (Accountability)**:
    - **Definition**: Accountability requires decision-makers to be responsible for the consequences of their actions. Blockchain transparency makes all actions traceable, enhancing accountability.
    - **Example**: Bitcoin's SegWit upgrade (document page 14) was implemented through community consensus, with all proposals and discussions public, making developers accountable to the community.
    - **Explanation**: Accountability ensures decision-makers cannot arbitrarily change rules or hide errors.
3. **激励 (Incentives)**:
    - **Definition**: Incentive mechanisms align the interests of principals and agents through rewards. DAOs incentivize members to participate in governance through token rewards.
    - **Example**: Uniswap's "Delegate Reward" proposal suggests rewarding delegates who actively participate in governance, encouraging more users to vote (document page 18).
    - **Supplementary**: Incentive alignment is key to resolving agency conflicts; DAOs achieve this goal through token economics.

4.4 链上与链下治理 (On-Chain and Off-Chain Governance)

4.4.1 链上治理 (On-Chain Governance)

On-chain governance refers to all governance activities (such as proposals and voting) being executed directly on the blockchain, with immutable records.

- **Definition**: On-chain governance automatically executes voting and decision-making through smart contracts, with results directly recorded on-chain.
- **Example**: Uniswap's final voting stage occurs on the blockchain, requiring a quorum of 40M UNI (document page 20).
- **Advantages**: Transparent, immutable, automated.
- **Disadvantages**: On-chain transactions require Gas fees, potentially limiting participation; high transparency may leak sensitive information.

4.4.2 链下治理 (Off-Chain Governance)

Off-chain governance refers to governance activities taking place outside the blockchain, such as forum discussions or Snapshot voting, with results potentially executed on-chain.

- **Definition**: Off-chain governance collects opinions through external platforms (such as forums or voting tools), reducing costs and complexity.
- **Examples**:
    - **Bitcoin's BIP Process** (document pages 14-15): Bitcoin Improvement Proposals (BIPs) are submitted and discussed via GitHub, with final implementation decided by miner and node consensus. For example, SegWit (Segregated Witness) was proposed through BIP 141, discussed in 2015, and activated in 2017.
    - **Snapshot** (document page 16): Snapshot is an off-chain voting interface where users connect their wallets to verify votes without on-chain transactions. Gnosis DAO uses Snapshot to configure voting rules, such as voting rights based on token holdings.
- **Supplementary**: SegWit is a protocol that optimizes Bitcoin transactions by separating signature data from transaction data, increasing block capacity. Snapshot links voting spaces through the Ethereum Name System (ENS), reducing Gas fees.
- **Advantages**: Low cost, flexible, suitable for preliminary discussions.

- **Disadvantages**: Results may be manipulated, less transparent than on-chain governance.

4.4.3 案例：Uniswap的治理流程

1. **请求评论 (Request for Comment, RFC)**:
   - Initial ideas are proposed in the Governance Forum, with 7 days of community discussion, considered off-chain governance.
   - Example: The "Uniswap Delegate Reward" proposal was initiated by Doo_StableLab, discussing reward mechanisms.
2. **温度检查 (Temperature Check)**:
   - Uses Snapshot for a 5-day off-chain vote, requiring a 10M UNI quorum, testing proposal feasibility.
3. **治理提案 (Governance Proposal)**:
   - After proposal iteration, it enters on-chain voting, requiring 2.5M UNI delegated support, a 40M UNI quorum, with a 7-day voting period.
   - Approved proposals have a 2-day timelock, ensuring review before execution.

- **Explanation**: Uniswap's hybrid governance balances efficiency and transparency, with off-chain discussions reducing costs and on-chain voting ensuring fairness.

4.5 DAO中的投票机制 (Voting Mechanisms in DAOs)

4.5.1 投票的重要性 (Importance of Voting)

Voting is the core of DAO governance, determining whether proposals pass. The document compares DAO voting with real-world voting (such as Hong Kong elections) (document page 21).

- **Real-world Voting**: Hong Kong elections require voters to show ID cards or specified documents (such as passports) to receive ballots, emphasizing identity verification.
- **DAO Voting**: Users verify their identity through crypto wallets, with voting rights typically tied to token holdings, without requiring traditional documentation.

4.5.2 投票类型 (Types of Voting)

The document introduces several types of voting in DAOs (document pages 22-23):

1. **单一投票 (Single Voting)**:
   - Each voter chooses one option, similar to the "one person, one vote" in traditional elections.
   - **Example**: PistachioDAO electing a governance leader, with voters choosing one person from Alice, Bob, Carol, and David.
2. **加权投票 (Weighted Voting)**:
   - Voting power is based on token holdings, with users holding more tokens having greater influence.
   - **Example**: In Uniswap's voting, users holding more UNI tokens have higher voting power.
3. **比例投票 (Proportional Voting)**:
   - Voters can allocate their voting weight across multiple options, similar to ranked-choice voting.
   - **Example**: PistachioDAO choosing a slogan, with voters allocating different weights to options like "In Pistachio we believe" and "In Pistachio Veritas."
4. **批准投票 (Approval Voting)**:

    - Voters can approve multiple options, ultimately selecting the option with the most approvals.
    - **Supplementary**: Approval voting is suitable for multi-candidate scenarios, reducing the "vote splitting" problem. For example, voters can simultaneously approve Alice and Bob as governance leaders.
- **Tools**: Snapshot supports various voting types, such as single, weighted, and proportional voting, commonly used in DAO off-chain governance.

4.6 DAO的挑战（新成本）

1. **技术复杂性 (Technical Complexity)**:
    - DAOs rely on blockchain and smart contracts, with high development and maintenance costs. Ordinary users may be unable to participate due to technical barriers.
    - **Example**: Uniswap's governance requires users to understand Snapshot and on-chain voting processes, potentially excluding members without technical backgrounds.
2. **治理效率低 (Low Governance Efficiency)**:
    - Decentralized decision-making requires multiple voting rounds, takes time, and may miss market opportunities.
    - **Example**: Uniswap proposals need to go through RFC, temperature check, and formal voting, taking several weeks.
3. **代币集中风险 (Token Concentration Risk)**:
    - Weighted voting may lead to a few large holders (Whales) controlling decisions, weakening decentralization.
    - **Example**: If a user holds a large amount of UNI tokens, they can unilaterally influence Uniswap governance.
4. **法律与监管 (Legal and Regulatory Issues)**:
    - The legal status of DAOs is unclear, potentially facing regulatory risks.
    - **Supplementary**: In 2023, the US SEC sued certain DAOs for unregistered securities offerings, highlighting regulatory challenges.

4.7 背景：从公共区块链到私有区块链

4.7.1 公共区块链与私有区块链的区别 (Differences Between Public and Private Blockchains)

- **Public Blockchain**: Anyone can join, data is completely open, typical examples are Bitcoin and Ethereum. Characterized by high degree of decentralization, but slower transaction speeds and lower privacy.
- **Private Blockchain**: Only authorized participants can join, data access is restricted, suitable for enterprise scenarios. Hyperledger Fabric is a typical private blockchain framework, emphasizing permission management and efficiency.
- **Transition in Course Materials**: The course materials transition from public blockchains (such as Bitcoin's on-chain/off-chain governance) to private blockchains, highlighting Hyperledger Fabric's applications in enterprises, such as reducing agency costs and supporting DAO governance.

4.7.2 Hyperledger Fabric简介

- **Definition**: Hyperledger Fabric is an open-source enterprise-grade blockchain framework hosted by the Linux Foundation, designed specifically for private and consortium chains, supporting modular architecture and permission management.

- **Characteristics**:
  - **Permissioned**: Only authenticated participants can access the network, suitable for scenarios requiring privacy and compliance.
  - **Modular**: Supports flexible component configuration, such as consensus mechanisms, storage methods, etc.
  - **High Performance**: Faster transaction speeds compared to public blockchains, suitable for high throughput scenarios.
  - **Privacy Protection**: Achieves data isolation through channels, ensuring sensitive information is only visible to specific participants.
- **Example**: In supply chain management, Hyperledger Fabric can be used to track every step of a product from production to sale, with only relevant parties (such as suppliers, logistics companies) able to access corresponding data.

4.8 Hyperledger Fabric的核心组件

4.8.1 架构概述 (Architecture Overview)

Hyperledger Fabric's architecture consists of multiple modular components that jointly support the operation of a distributed ledger. The main components are:

- **Nodes**: Participants in the network, divided into:
  - **Client Nodes**: Submit transaction proposals.
  - **Peer Nodes**: Store ledger data, execute chaincode, validate transactions.
  - **Orderer Nodes**: Responsible for transaction ordering and block generation, ensuring consistency.
- **Ledger**: Consists of the blockchain (storing transaction logs) and the world state (storing current data snapshots).
- **Channels**: Logically isolated sub-networks where only members within the channel can access data, similar to "private chat groups."
- **Chaincode**: Smart contracts running on the blockchain, defining business logic such as asset creation or transfer.
- **Membership Service Provider (MSP)**: Manages node identities and access permissions, ensuring only authorized users can operate.

Simple Explanation: A channel is like a meeting room that only specific members can enter, chaincode is the "rulebook" executed in the meeting room, and MSP is the "security" checking the identities of attendees.

4.8.2 通道 (Channels)

- **Definition**: Channels are the core mechanism for implementing data privacy in Hyperledger Fabric, with each channel having independent ledgers and chaincode.
- **Functions**:
  - Limiting data access: Only channel members can see transactions.
  - Supporting multi-party collaboration: Different channels can be used for different business scenarios.
- **Example**: In a pharmaceutical supply chain, manufacturers and hospitals can share drug production data within one channel, while pharmacies and hospitals share sales data within another channel, without interference.

4.8.3 链码 (Chaincode)

- **Definition**: Chaincode is a smart contract in Hyperledger Fabric, written in languages like Go or JavaScript, running on peer nodes.
- **Functions**:
  - Defining business logic: Such as creating assets, querying status, transferring ownership.
  - Isolated execution: Chaincode runs in Docker containers, preventing malicious code from affecting the network.
- **Course Example**: The code in the course demonstrates chaincode used for asset management:
  - `CreateAsset`: Create assets (with attributes like ID, color, size, owner, value).
  - `ReadAsset`: Query asset status.
  - `renderTemplate`: Display query results on web pages.
- **Additional Explanation**: The chaincode execution process includes:
  1. Client submits transaction proposal.
  2. Peer nodes simulate chaincode execution, generating read-write sets.
  3. Orderer nodes package transactions into blocks.
  4. Peer nodes verify and update the ledger.
- **Example**: In a used car trading platform, chaincode can define "vehicle transfer" logic, recording information such as vehicle ID, color, mileage, owner, etc., ensuring transactions are transparent and immutable.

4.8.4 成员服务提供者 (Membership Service Provider, MSP)

- **Definition**: MSP is responsible for identity management and access control, using Public Key Infrastructure (PKI) to assign digital certificates to nodes.
- **Functions**:
  - **Authentication**: Ensuring only authorized users can submit transactions.
  - **Access Control**: Defining who can read/write to the ledger or execute chaincode.
- **Simple Explanation**: MSP is like an "access control system," where only users with the correct "key" (certificate) can enter.
- **Example**: In financial scenarios, MSP ensures that only banks and regulatory agencies can access loan records, preventing unauthorized access.

4.9 交易流程 (Transaction Flow)

The transaction flow in Hyperledger Fabric includes the following steps:

1. **Proposal**: Client submits transaction proposals through SDK, including chaincode invocations and input parameters.
2. **Simulation**: Peer nodes execute chaincode, generating read-write sets, but do not update the ledger.
3. **Endorsement**: Peer nodes sign the read-write sets according to endorsement policies.
4. **Ordering**: Orderer nodes order transactions and package them into blocks.
5. **Validation and Commit**: Peer nodes validate transactions and update the ledger and world state.

Simple Explanation: The transaction flow is like sending a package: the customer places an order (proposal), the warehouse simulates packaging (simulation), the logistics company confirms (endorsement), the dispatch center arranges the route (ordering), and finally the package is delivered and recorded (validation and commit).

Example: In asset management, users submit "create asset" requests through web pages, chaincode validates inputs (such as asset ID, color), peer nodes simulate execution, orderer nodes generate blocks, and finally the ledger is updated.

4.10 Hyperledger Fabric与DAO的结合

- **DAO Governance Support**:
  - Hyperledger Fabric implements decentralized decision-making through chaincode, such as voting mechanisms.
  - Channels support multi-party collaboration, reducing agency conflicts.
  - MSP ensures only authorized members participate in governance, enhancing accountability.
- **Reducing Agency Costs**:
  - **Monitoring Agents**: Chaincode records all operations, preventing self-serving behaviors.
  - **Monitoring Firm Operations**: Ledger transparency reduces information asymmetry.
  - **Reducing Excessive Expenses**: Decentralized governance prevents executive misuse of resources.
  - **Avoiding Profit Loss (Unrealized Profits)**: Community voting optimizes decisions, reducing suboptimal choices.
- **Example**: A DAO uses Hyperledger Fabric to manage an investment fund, where investors vote on fund allocation through chaincode, all transactions are recorded on the ledger, and MSP restricts non-member access, ensuring transparency and fairness.

4.11 新挑战与成本 (New Challenges and Costs)

The course materials mention new costs introduced by DAOs and private blockchains, and Hyperledger Fabric is no exception:

- **Governance Complexity**: Multi-party collaboration may lead to low voting participation or control by a minority.
- **Technical Risks**: Chaincode vulnerabilities may result in data leakage or system crashes.
- **Legal Compliance**: Private blockchains need to meet enterprise regulatory requirements, such as data privacy regulations.
- **Additional Challenges**:
  - **Scalability**: Increasing channels and nodes may affect performance.
  - **Interoperability**: Integration with other blockchain systems may face technical barriers.

Example: In a supply chain DAO, low participation may result in critical proposals receiving no votes; chaincode vulnerabilities may expose sensitive data, requiring regular audits.

4.12 课件中的代码分析 (Analysis of Code in Course Materials)

The course materials showcase Hyperledger Fabric chaincode and web application code for asset management. Here's a detailed analysis:

- **Chaincode Functions**:

    - `CreateAsset`: Creates assets with parameters including ID, color, size, owner, value.
    - `ReadAsset`: Queries asset status, returning data in JSON format.
- **Web Application**:
    - `createHandler`: Loads the web page for creating assets (create.html).
    - `createResultHandler`: Processes user input, calls chaincode to create assets, displays results.
    - `renderTemplate`: Renders HTML templates, showing asset information.
- **Asset Structure**:

```
type Asset struct {
    AppraisedValue int    `json:"AppraisedValue"`
    Color          string `json:"Color"`
    ID             string `json:"ID"`
    Owner          string `json:"Owner"`
    Size           string `json:"Size"`
}
```

    - Fields are arranged alphabetically, ensuring JSON serialization consistency.
    - Can be extended to other scenarios, such as used car trading (adding a "mileage" field).

Example: In a used car trading platform, users input vehicle information (ID, color, mileage) through web pages, chaincode creates assets and stores them in the ledger, MSP verifies user identity, and results are displayed on the web page.

5. 行业应用、元宇宙与NFT

5.1 代币与加密货币概述 (Overview of Tokens and Cryptocurrencies)

5.1.1 代币与货币的区别 (Differences Between Tokens and Coins)

- **Coins**: Serve as a medium of exchange, used for payments or transactions. For example, Bitcoin is a typical cryptocurrency designed to serve as currency in a decentralized network.
- **Tokens**: Typically function as assets, representing specific uses or interests. Tokens operate on blockchains with diverse functions, such as accessing services or representing ownership.

Example: Bitcoin is a coin that can be directly used to purchase goods; ERC-20 tokens on Ethereum (such as Chainlink) are tokens used to pay for smart contract services.

5.1.2 代币分类 (Token Classification)

5.1.2.1 支付代币 (Payment Tokens)

- **Definition**: Cryptocurrencies used for payments, emphasizing privacy and decentralization. Bitcoin is the pioneer of payment tokens.
- **Characteristics**:
    - Serve as a medium of exchange, but with low recognition from governments and banks.
    - High price volatility, but gradually improving over time.
- **Example**: Using Bitcoin to purchase coffee at supporting merchants, but the price may change due to market fluctuations.

5.1.2.2 法定加密货币与稳定币 (Fiat Cryptocurrencies and Stablecoins)

- **Definition**: Stablecoins are cryptocurrencies pegged to fiat currencies, designed to reduce volatility. Fiat cryptocurrencies refer to digital currencies backed by governments (not detailed in the document, supplementary information follows).
- **Supplementary Information**: Stablecoins like USDT (Tether) are pegged 1:1 with the US dollar, widely used in trading to avoid price fluctuations. Fiat cryptocurrencies like China's digital yuan (e-CNY) are issued by central banks.
- **Characteristics**: Stablecoins have low volatility, suitable for daily transactions; fiat cryptocurrencies are regulated by governments with clear legal status.
- **Example**: Using USDT to purchase other tokens on cryptocurrency exchanges, with stable prices; using digital yuan for online payments.

5.1.2.3 证券代币 (Security Tokens)

- **Definition**: Tokens representing traditional financial assets (such as stocks, bonds), regulated by securities laws (mentioned but not explained in the document).
- **Supplementary Information**: Security tokens record ownership through blockchain, increasing transaction transparency and efficiency, but must comply with strict regulatory requirements.
- **Characteristics**: Can represent company shares, real estate, and other assets, with high liquidity but strict regulation.
- **Example**: A company issues security tokens, allowing investors to purchase company shares through blockchain, similar to stocks but stored in digital form.

5.1.2.4 实用代币 (Utility Tokens)

- **Definition**: Tokens used to access specific platforms or services (mentioned but not detailed in the document).
- **Supplementary Information**: Utility tokens grant holders the right to use products or services, typically not intended as investment tools.
- **Characteristics**: Strong functionality, usually do not represent asset ownership.
- **Example**: Golem tokens (GNT) on Ethereum used to pay for decentralized computing services.

5.1.2.5 非同质化代币 (Non-Fungible Tokens, NFTs)

- **Definition**: Tokens representing unique digital assets (such as artwork, virtual real estate) that cannot be interchanged and are recorded on the blockchain.
- **Characteristics**: Each NFT has uniqueness, with ownership recorded on blockchain, commonly used for digital collectibles or game assets.
- **Example**: Purchasing a piece of digital art (such as Beeple's NFT work), with ownership recorded on the Ethereum blockchain.

5.2 加密货币相关犯罪 (Cryptocurrency-Related Crimes)

5.2.1 规模与影响 (Scale and Impact)

- **Research Data** (Foley et al., 2019):
  - About 25% of Bitcoin users are involved in illegal activities.
  - Approximately $76 billion in illegal transactions annually (46% of Bitcoin transactions), close to the size of US and European drug markets.

- **Trend**: The proportion of illegal transactions has decreased with increased mainstream interest and the emergence of more opaque cryptocurrencies (such as Monero).

5.2.2 案例 (Cases)

- **2016 Bitcoin Theft**: The US Department of Justice recovered $3.6 billion in stolen Bitcoin, the largest financial seizure in history.
- **2024 Ethereum Attack**: Two brothers stole $25 million in 12 seconds by attacking the Ethereum blockchain, involving wire fraud and money laundering.

Example: Hackers exploit cryptocurrency anonymity to purchase illegal goods on the dark web, while law enforcement agencies track fund flows through blockchain analysis.

5.3 全球加密货币监管 (Global Cryptocurrency Regulation)

The document shows the global regulatory status of cryptocurrencies (as of November 2021):

- **Absolute Ban**: Some countries (such as China) prohibit all cryptocurrency transactions.
- **Implicit Ban**: No explicit legislation prohibiting cryptocurrencies, but indirect restrictions through limiting banking services, etc.
- **Regulated**: Many countries (such as the United States) regulate cryptocurrencies through taxation, Anti-Money Laundering, and Anti-Terrorism Financing laws.
- **El Salvador**: The first country in the world to adopt Bitcoin as legal tender.

Supplementary Information: As of 2025, the regulatory environment continues to evolve. For example, China maintains a strict ban, while the European Union regulates the market through MiCA (Markets in Crypto-Assets Regulation).

Example: In the United States, crypto transactions must be reported to the IRS for taxation; in China, individuals holding cryptocurrencies may face legal risks.

5.5 Zcash 与零知识证明 (Zcash and Zero-Knowledge Proof)

Zcash is a privacy-focused payment token; the document introduces its features and zero-knowledge proof.

5.5.1 Zcash 特点 (Zcash Features)

- **Based on Bitcoin**: Zcash inherits Bitcoin's block reward and fixed total supply.
- **Privacy**: Implements selective disclosure through encryption technology, different from Bitcoin's complete transparency.
- **Address Types**:
  - **Shielded Address (z-)**: Transaction information is not visible, protecting privacy.
  - **Transparent Address (t-)**: Similar to Bitcoin, transactions are public.
- **Transaction Types**: Supports four transaction combinations (shielded to shielded, shielded to transparent, etc.).

Example: Alice uses Zcash's shielded address to transfer funds to Bob; the transaction amount and addresses are not visible to the public, but Bob can verify receipt of the funds.

5.5.2 零知识证明 (Zero-Knowledge Proof)

- **Definition**: A cryptographic technique allowing one party (the prover) to prove to another party (the verifier) that certain information is true without revealing the specific information.
- **Simple Explanation**: Like proving you know a password without saying the password itself.
- **Example (provided in the document)**:
    - Alice proves to color-blind Bob that two pens have different colors (one red, one green), without telling the specific colors.
    - Method: Bob randomly swaps the pens, Alice correctly states each time whether they were swapped, proving she can distinguish colors.
- **Application in Zcash**: Zero-knowledge proofs ensure transactions are valid (correct amounts, no double-spending) while hiding transaction details.

Supplementary Information: Zcash uses zk-SNARKs (Zero-Knowledge Succinct Non-Interactive Arguments of Knowledge), which are computationally efficient and suitable for blockchain privacy protection.

5.6 加密货币相关犯罪 (Cryptocurrency-Related Crimes)

5.6.1 规模与影响 (Scale and Impact)

- **Research Data** (Foley et al., 2019):
    - About 25% of Bitcoin users are involved in illegal activities.
    - Approximately $76 billion in illegal transactions annually (46% of Bitcoin transactions), close to the size of US and European drug markets.
- **Trend**: The proportion of illegal transactions has decreased with increased mainstream interest and the emergence of more opaque cryptocurrencies (such as Monero).

5.6.2 案例 (Cases)

- **2016 Bitcoin Theft**: The US Department of Justice recovered $3.6 billion in stolen Bitcoin, the largest financial seizure in history.
- **2024 Ethereum Attack**: Two brothers stole $25 million in 12 seconds by attacking the Ethereum blockchain, involving wire fraud and money laundering.

Example: Hackers exploit cryptocurrency anonymity to purchase illegal goods on the dark web, while law enforcement agencies track fund flows through blockchain analysis.

5.7 最大可提取价值 (MEV Exploits)

The document introduces MEV (Maximal Extractable Value), which is the practice of extracting additional profits by reordering blockchain transactions.

5.7.1 基本概念 (Basic Concepts)

- **Definition**: MEV is the additional value that validators (or miners) can obtain by manipulating transaction order, inserting, or excluding transactions.
- **Origin**: Originated from miner behavior in Proof of Work (PoW).
- **Realized Economic Value**: The actual MEV profits obtained.
- **Mechanism**: Validators prioritize transactions based on gas fees to profit.

Example: Miners place high gas fee transactions at the beginning of blocks, prioritizing them to earn more fees.

5.7.2 套利 (Arbitrage)

- **Definition**: Profiting from price differences in different markets, considered "benign MEV."
- **Example (Stock Arbitrage)**:
  - Stock price is $100 at exchange A,$99 at exchange B.
  - Buy 1000 shares at B $(99/share), sell at A($100/share), net profit $980(after$20 in fees).
- **Blockchain Arbitrage**: Exploiting price differences in Automated Market Makers (AMMs).
  - **Constant Product Market Maker (CPMM)**:
    - Formula: (R_A \times R_B = K) (K is constant).
    - Initial: Pool has 1000 A and 1000 B tokens, K = 1,000,000, A:B price is 1:1.
    - Market price: 1.1 A = 1 B.
    - Arbitrageurs deposit B tokens, withdraw A tokens, adjusting the pool price to market level, earning the difference.

Example: An arbitrageur discovers that token A's price in a Uniswap pool is lower than the market, buys A and sells it on another exchange for profit.

5.7.3 前置交易 (Front-Running)

- **Definition**: Profiting by inserting one's own transaction before another transaction based on advance knowledge, considered malicious MEV.
- **Example (Stock Market)**:
  - A broker learns that a client will purchase 500,000 shares (price $100).
  - The broker first buys 5,000 shares $(100/share), pushing the price to$101.
  - The client's order pushes the price to $110, the broker sells 5,000 shares, making$50,000.
- **Blockchain Front-Running**:
  - MEV bots (like Jaredfromsubway.eth) use high gas fees to execute transactions first, earning millions of dollars.

Example: A user submits a large buy order on Uniswap, MEV bots buy tokens first to drive up the price, then sell for profit.

5.8 DeFi概述 (DeFi Overview)

去中心化金融(DeFi)是基于区块链技术的金融系统，通过智能合约提供金融服务，无需传统中介机构如银行或经纪商。

- **核心特点**:
  - **去中心化**: 没有中央机构控制，由分布式网络集体维护。
  - **开放性**: 任何人无需身份验证即可参与，只需加密钱包。
  - **透明性**: 所有交易记录公开可见，代码通常开源。
  - **可组合性**: DeFi协议像"乐高积木"，可组合创建新服务。
  - **自动化**: 智能合约自动执行金融操作，减少人工干预。
- **主要应用**:
  - 去中心化交易所(DEX)
  - 借贷平台

  - 流动性挖矿
  - 质押
  - 稳定币
- **风险**:
  - 智能合约漏洞可能导致资金被盗
  - 加密资产价格波动可能引起清算
  - 监管不确定性
  - 用户操作错误可能导致资金永久丢失
- **实例**: Uniswap(去中心化交易所)、MakerDAO(稳定币发行)、Aave(借贷平台)
- **解释**: DeFi像"无人银行"，所有金融服务通过自动运行的代码(智能合约)完成，无需银行或公司介入。
- **示例**: 用户在Aave平台存入ETH获得利息，或抵押ETH借出其他代币，整个过程通过智能合约自动执行，无需银行批准。

5.9 元宇宙概述 (Overview of Metaverse)

5.9.1 元宇宙的定义 (Definition of Metaverse)

The document quotes Matthew Ball's (2022) definition, describing the metaverse as:

- **Definition**: A massive, interoperable network of real-time rendered 3D virtual worlds that users can experience synchronously and persistently, with a sense of presence. It maintains continuity of data, including identity, history, entitlements, objects, communications, and payments.
- **Simple Explanation**: The metaverse is a virtual, real-world-like digital space where users can interact, trade, and live as they would in reality, with data remaining consistent across different platforms.
- **Example**: In the metaverse game Decentraland, users can purchase virtual land, build houses, and use the same digital identity across different virtual worlds.

5.9.2 元宇宙与远程工作 (Metaverse and Work From Home)

The document mentions that the metaverse can enhance the remote work (Work From Home, WFH) experience:

- **Virtual Meetings**: Users can display work content through shared virtual screens, enhancing interactivity.
- **Team Connection**: Users interact with colleagues in virtual spaces, such as through virtual gestures (like thumbs-up) to establish connections, providing an immersive experience.
- **Example**: Using Oculus Workrooms, team members can present PowerPoint slides in a virtual meeting room and interact in real-time, as if they were in the same office.

5.10 元宇宙中的经济活动 (Economic Activities in the Metaverse)

5.10.1 经济活动的必然性 (Inevitability of Economic Activities)

The document points out that economic activities in the metaverse are inevitable, not limited to financial payments but also including the trading of digital assets.

- **Simple Explanation**: The metaverse, like the real world, requires economic behaviors such as buying, selling, and leasing, involving virtual goods and services.
- **Example**: On the Roblox platform, users purchase virtual Gucci bags (worth approximately $4,115) to decorate their virtual characters.

5.10.2 区块链的作用 (Role of Blockchain)

Blockchain provides a trust-less environment in the metaverse:

- **Immutable Ledger**: Blockchain records all transactions, ensuring transparency and immutability.
- **Trust-less Environment**: No intermediaries needed; users can trade directly, reducing fraud risk.
- **Example**: In Decentraland, purchased virtual land is recorded on the Ethereum blockchain, and ownership can be verified by anyone.

Additional Information: Blockchain, through distributed ledger technology, ensures data security and decentralization, making it suitable for cross-platform needs in the metaverse.

5.11 NFTs 和 SFTs 在元宇宙中的作用 (The Role of NFTs and SFTs in the Metaverse)

5.11.1 非同质化代币 (Non-Fungible Tokens, NFTs)

- **Definition**: NFTs are tokens representing unique digital assets (such as artwork, virtual real estate) that cannot be interchanged and are recorded on the blockchain.
- **Role in the Metaverse**: NFTs serve as carriers of digital assets, supporting ownership and transactions in virtual worlds.
- **Example**: In The Sandbox, users can purchase a virtual painting (NFT) that can be displayed or sold across different metaverse platforms.

5.11.2 半同质化代币 (Semi-Fungible Tokens, SFTs)

- **Definition** (not detailed in the document, supplementary information follows): SFTs combine the characteristics of fungible tokens and non-fungible tokens. Initially fungible (interchangeable), they can become non-fungible (unique) under specific conditions.
- **Simple Explanation**: SFTs are like "multi-purpose tickets" that are interchangeable at first but become unique souvenirs after use.
- **Role in the Metaverse**: SFTs support flexible asset management, such as virtual event tickets or game items.
- **Example**: In a metaverse concert, SFTs are sold as tickets (fungible) but transform into unique digital collectibles (non-fungible) after use.

5.11.3 跨链互操作性 (Cross-Chain Interoperability)

- **Definition** (mentioned but not explained in the document): The ability of different blockchain networks to interact with data and assets.
- **Simple Explanation**: Allowing NFTs or SFTs to circulate between different metaverses or blockchains, such as transferring from Ethereum to Solana.
- **Additional Information**: Cross-chain bridges like Wormhole enable asset transfers between chains, enhancing interoperability in the metaverse.
- **Example**: Virtual land purchased as an NFT in Decentraland can be transferred to The Sandbox via a cross-chain bridge.

5.12 GameFi 与玩赚模式 (GameFi and Play-to-Earn Model)

5.12.1 GameFi 定义 (Definition of GameFi)

- **Definition**: GameFi combines gaming and decentralized finance (DeFi), using blockchain technology to achieve true ownership and financialization of in-game assets.
- **Simple Explanation**: Players earn money in games and own virtual items that can be traded or used for financial activities.
- **Characteristics**:
  - **True Ownership**: Players own in-game assets (such as NFT equipment) recorded on the blockchain.
  - **Reward System**: Earn tokens through game activities, tradable on decentralized exchanges (DEX).
  - **Play-to-Earn (P2E) Model**: Players receive economic returns through gameplay.
- **Example**: In Axie Infinity, players breed virtual pets (NFTs), earn tokens (SLP) through battles, and can trade them on Uniswap.

5.12.2 DeFi 整合 (DeFi Integration)

GameFi seamlessly integrates DeFi mechanisms, including:

- **Staking**: Players lock tokens to receive rewards.
  - **Example**: In The Sandbox, staking SAND tokens to earn returns from virtual land.
- **Farming**: Providing liquidity to pools to earn returns.
  - **Example**: Players deposit game tokens into DEX pools to earn transaction fees.
- **Lending**: Using in-game NFTs as collateral for loans.
  - **Example**: Players collateralize Axie Infinity NFT pets to borrow stablecoins from DeFi platforms.

Additional Information: DeFi mechanisms are automated and transparent through smart contracts, enhancing the financial capabilities of GameFi.

5.13 元宇宙的挑战与未来 (Challenges and Future of the Metaverse)

5.13.1 技术与监管挑战 (Technical and Regulatory Challenges)

- **Technical Challenges**: The metaverse requires powerful computing capabilities, real-time rendering, and low-latency network support; current technology is not yet fully mature.
- **Regulatory Challenges**: The legal status of virtual assets and compliance issues in cross-border transactions (such as anti-money laundering) still need clarification.
- **Example**: The EU's MiCA regulation (implemented in 2024) requires metaverse platforms to comply with crypto asset regulations.

5.13.2 分散的元宇宙 (Disjoint Metaverse)

- **Issue**: The document suggests that the metaverse may become fragmented due to different technical standards, with companies like Apple, Huawei, and Microsoft each developing independent metaverses.
- **Additional Information**: Lack of unified standards could lead to fragmented user experiences; cross-chain interoperability and open protocols (such as the Metaverse Standards Forum) are potential solutions.
- **Example**: Virtual clothing purchased in Apple's metaverse may not be usable in Microsoft's metaverse.

5.13.3 虚拟与现实的连接 (Connection Between Virtual and Reality)

- **Issue**: How do activities and assets in the metaverse connect with the real world?

- **Additional Information**: Real-world laws (such as property law) and economic systems need to interface with the virtual world; ownership disputes over NFTs may require court decisions.
- **Example**: A virtual Gucci bag in Roblox may be worth more than a physical bag, but its legal ownership remains unclear.