

Blockchain Technology: A Review

Seoyoung Kim ^{*} [†]

Associate Professor of Finance
Santa Clara University

September 2026

For Oxford Research Encyclopedias: Business & Management

Article Summary:

Blockchain technology is often framed either as a revolutionary replacement for existing financial infrastructure or as an inefficient ideology in search of a problem. This article advances a more precise and analytically grounded perspective. At its core, a public blockchain represents an expansion of digital recordkeeping systems, introducing public, permissionless ledgers alongside existing private, permissioned architectures. Rather than advocating broad adoption, the article clarifies the conditions under which blockchain-based systems can be meaningful and incrementally value-adding—and when they are not.

The article examines the core technical and economic foundations of blockchain systems, including cryptographic verification, decentralized consensus in adversarial environments, programmable execution through smart contracts, composability and interoperability through interface standards, and incentive-based leaderless participation. Together, these components enable shared recordkeeping and rule enforcement without

^{*} Email: srkim@scu.edu // Address: 500 El Camino Real; Santa Clara, CA 95053

[†] I thank M. Apu II for his invaluable contributions and support.

assigning authority to a centralized recordkeeper, while introducing tradeoffs in governance and operational efficiency, including replicated data storage and redundant validation effort.

Where trusted intermediaries and legal enforcement are effective, centralized architectures often remain superior. The comparative advantage of public, permissionless systems instead arises in environments characterized by fragmented trust, concerns about opaque intermediation, risks of discretionary abuse, or high coordination costs. By clarifying these boundary conditions, the article provides a framework for evaluating blockchain use cases in financial markets—tokenization and decentralized financial intermediation—grounded in technical architecture and institutional integration.

Keywords: blockchain, distributed ledger systems, protocol governance, incentive alignment, smart contracts, decentralized finance (DeFi), tokenization

JEL Codes: G23, G28, D47, D85

Table of Contents

1. Introduction	3
2. Origins and Evolutionary Path	9
3. Conceptual Foundations and Core Technical Architecture	12
a. Cryptographic Primitives.....	13
b. Consensus Mechanisms.....	14
c. Smart Contracts and Interface Standards.....	16
d. Incentive Alignment Mechanisms and Economic Design	18
4. Applications Across Sectors.....	19
a. Decentralized Financial Intermediation.....	20
b. Digital Assets and Tokenization	27
c. Limits of Applicability.....	28
5. Legal, Governance, and Institutional Considerations.....	30
a. Legal Classification and Regulatory Uncertainty	30
b. Governance and Accountability in Decentralized Systems	33

For institutional participants, these forms of concentration—and the resulting uncertainty regarding who exercises effective influence and how conflicts are resolved—can pose a significant barrier to adoption. This distinction between leaderless execution and leaderless control—two concepts that are often conflated in popular discourse—raises

important questions about transparency, legitimacy, and responsibility, particularly where protocol changes can materially affect asset functionality, compliance obligations, or economic outcomes. This governance ambiguity helps explain why, in current institutional practice, certain applications—such as tokenized securities—have been implemented with blockchains functioning as duplicative or shadow recordkeeping systems rather than as determinative legal ledgers.	38
c. Institutional Readiness and Operational Integration	38
6. Concluding Remarks	40
References	41
Table 1. Institutional Boundary Conditions Favoring Centralized versus Public-Blockchain Systems	48
Table 2. Sources and Implications of Concentration in Nominally Leaderless Systems.....	49
Appendix A. Glossary	50
Append-only Ledger.....	50
Application Layer	50
Application-Layer Risk.....	50
Composability	50
Consensus Mechanism	50
Cryptographic Primitives.....	51
Cryptographic Verification	51

Execution Environment.....	51
Incentive Alignment.....	51
Interface Standards.....	51
Permissioned System	52
Permissionless System	52
Protocol-Defined Coordination.....	52
Protocol Layer	52
Public Blockchain	52
Smart Contract.....	53
Trustless	53
Validator.....	53
Appendix B. Digital Signatures and Public-Private Key Authorization (Stylized Example)	54

1. Introduction

With technological advances that have improved the accuracy, security, and accessibility of digital records, reliable recordkeeping has become an indispensable mechanism for establishing and enforcing property rights in modern markets.¹ Notably, the determination of legal ownership and access rights has increasingly transitioned from physical possession—such as

¹ See, for instance, Demsetz, Harold (1967), “*Toward a Theory of Property Rights.*”

cash and bearer instruments—to digital records maintained by authorized institutions such as banks, transfer agents, and county recorders. At its most fundamental level, public blockchains—or simply blockchains²—constitute an expansion of digital recordkeeping solutions, adding public, permissionless options alongside existing private, permissioned systems.

Although the term “blockchain technology” is relatively recent, its conceptual and structural underpinnings draw on decades of research in mathematics and computer science. These foundational works include developments in fault-tolerant distributed systems that balance liveness and safety, gossip protocols for propagating information across large and decentralized networks, and cryptographic primitives—such as hash functions, public-key cryptography, and digital signatures—that enable reliable verification of transaction authenticity. Ultimately, the seemingly rapid emergence of Bitcoin and the overnight proliferation of additional blockchain systems do not constitute a sudden technological rupture, but instead reflect the culmination of extensive efforts by scholars and practitioners across multiple disciplines.

In this article, I evaluate blockchain technology as an institutional-design alternative rather than as a context-free replacement for centralized infrastructure. That is, public blockchains substitute protocol-based enforcement for certain forms of institutional

² For clarity, “blockchain” refers to blockchain-based distributed ledgers, as distinct from other distributed architectures such as directed acyclic graph (DAG)–based designs. In common usage, the unqualified term “blockchain” refers to public, permissionless systems, whereas private implementations are typically referenced with qualifiers such as “private,” “permissioned,” or “enterprise.” A glossary of selected technical terms is provided in **Appendix A**.

discretion—cryptographic verification, consensus rules, and economic incentives perform functions conventionally assigned to authorized recordkeepers and organizational hierarchies. Whether blockchains add incremental value depends on the institutional setting, and centralized systems generally remain superior where trusted intermediaries, legal enforcement, and coordinated decision-making operate effectively.

Accordingly, I focus on blockchain’s technical foundations, protocol-level design (including governance and incentive mechanisms), and real-world applications, emphasizing how the appropriateness of blockchain-based systems hinges critically on the specific problem at hand. My objective is not to advocate for the general adoption of public blockchains, but rather to clarify the boundary conditions under which blockchain-based solutions may be meaningful and incrementally value-adding. More broadly, my analysis situates blockchain technology within the context of financial infrastructure and the design space spanning distributed and centralized systems, highlighting the complex tradeoffs among security, scalability, and decentralization.³

Specifically, natural questions arise as to how leaderless, decentralized systems continue to process transactions, maintain security, and coordinate technical updates without a centralized authority to oversee the process. That is, modern financial infrastructure typically relies on trusted intermediaries—such as Bank of America—to process transactions and maintain reliable records, with public confidence grounded in an institution’s adherence to

³ The challenge of balancing these competing interests, known as the “blockchain trilemma,” was popularized by Vitalik Buterin in earlier blog posts and technical discussions addressing potential solutions. See, for instance, Buterin, Vitalik (2017), “*Sharding FAQ*,” accessed at https://vitalik.eth.limo/general/2017/12/31/sharding_faq.html.

careful internal controls, regulatory oversight, and legal accountability. Public blockchains, by contrast, are designed around a different class of internal controls known as consensus mechanisms. These carefully designed mathematical protocols determine how agreement on the state of the ledger is reached and aim to provide trustless validity in a distributed environment where participants can read from and write to the ledger without vetted permission from a central authority.⁴

As with any set of rules designed to achieve consensus, coordination complexity increases with the number of disparate participants, much like the difference between organizing a nationwide vote and a single elected official making a unilateral decision. Because not all participants in a permissionless system can be assumed to act honestly, a fault-tolerant consensus mechanism must balance liveness and safety. Permitting all transactions indiscriminately maximizes liveness but sacrifices safety, while prohibiting all transactions ensures safety at the expense of liveness. Blockchain systems therefore draw on fault-tolerant distributed systems design to navigate these tradeoffs, which in turn implicate the broader tension among security, scalability, and decentralization.

Furthermore, since consensus mechanisms are designed primarily to ensure safety, blockchain systems must also incorporate governance mechanisms that operate under incomplete contracting to ensure that sufficient participation is sustained in equilibrium. That is, beyond protocol-level correctness, the system must also draw on principles from

⁴ I use the term “trustless” to refer to the ability of participants to establish the validity of records and rule compliance through cryptographic proofs and protocol-enforced mechanisms, without reliance on institutional discretion or identity-based trust. I do not imply the absence of all forms of trust, nor do I preclude reliance on off-chain enforcement or verification of records where real-world actions are involved.

organizational economics to induce a large mass of leaderless, voluntary participants—who are free to enter and exit the system at will—to supply the ongoing effort required to verify and process transactions and to propose necessary technical updates to the system itself.

These participants, commonly referred to as miners or transaction validators, are compensated for their work with the blockchain's native token, which is required to transact within the system. The usefulness of a blockchain for recordkeeping and transaction processing underpins demand for the native token and thereby its monetary value. Thus, the incentive to maintain the value of their token holdings motivates validators to continue supplying effort and technical contributors to continue developing updates, analogous to how property values incentivize homeowners to invest in maintaining the safety and upkeep of their neighborhoods. The specifics of a blockchain's monetary policy—including token issuance, supply constraints, and validator compensation—therefore constitute a central governance mechanism influencing incentive alignment and participation over time.

Finally, natural questions arise as to whether the benefits of public blockchains justify their costs. Permissionless, distributed recordkeeping systems entail significant inefficiencies relative to centralized, permissioned systems, including widespread replication of the ledger across validating nodes, repeated transaction execution and verification, and constrained throughput. Where trusted intermediaries and legal enforcement are available—as in much of the modern financial infrastructure of developed economies—centralized architectures typically remain more efficient and operationally effective. Indeed, modern centralized financial institutions emerged as responses to the inefficiencies of earlier, more decentralized financial

arrangements—which may be described as “primitive DeFi”⁵—that relied on bilateral trust, fragmented recordkeeping, and limited enforcement. Accordingly, permissionless, distributed systems do not represent a superior replacement but rather a costly alternative to much of the institutional financial infrastructure in developed economies today.⁶

The comparative advantages of public, permissionless systems instead arise in environments characterized by limited trust in centralized authorities or heightened concerns about manipulation, opaque intermediation, or the improper exercise of discretionary control. In certain sub-markets and institutional settings—such as trading venues susceptible to frontrunning, censorship, or undisclosed preferential access—the absence of a central operator with unilateral, unchecked authority can itself be value-adding. By substituting institutional discretion with open-source,⁷ protocol-enforced rules, public blockchains can mitigate specific forms of abuse that are otherwise difficult to detect or contest, even as they introduce meaningful efficiency tradeoffs. At the same time, these systems do not eliminate all sources of

⁵ I use the term “primitive DeFi” here as an analytical analogy rather than a historical label. It refers to premodern and early modern financial arrangements characterized by decentralized coordination and limited institutional enforcement, which motivated the subsequent emergence of centralized intermediaries. A related framing appears in Kim, Seoyoung (2023), *“DeFi For Dummies,”* which contrasts contemporary blockchain-based decentralized finance (“modern DeFi”) with earlier decentralized financial practices that lacked scalable enforcement and standardized recordkeeping.

⁶ My analysis in this paper abstracts from settings involving severe institutional breakdown—such as the collapse of sovereign governance or monetary authority—and instead examines the marginal value of public blockchains within otherwise well-functioning legal and financial systems.

⁷ The economic significance of open-source protocols lies in the public verifiability and contestability of protocol logic, whereby market participants, auditors, and regulators can directly inspect, verify, and monitor the rules governing system operation rather than relying on high-level descriptions of black-box implementations. In practice, most major public blockchain codebases are maintained on collaborative platforms such as GitHub (see “GitHub Docs,” GitHub, accessed at <https://docs.github.com/en>).

trust: while blockchains can enforce rules governing records and execution, they cannot independently verify the truth of the off-chain events those records are meant to reflect.

In the sections that follow, I explore the technical foundations, incentive structures, applications, and institutional constraints through which public blockchains attempt to resolve—and may instead reproduce—problems of trust, coordination, and governance.

2. Origins and Evolutionary Path

The first viable and widely adopted implementation of a public blockchain emerged with the deployment of the Bitcoin blockchain in 2009.^{8 9} The central innovation of Bitcoin was not simply the introduction of a shared public ledger, but the integration of cryptographic primitives with a robust consensus mechanism capable of achieving agreement over the state of that ledger without delegating control to an authorized intermediary.

Thus, Bitcoin was conceived as a response to an institutional design problem—namely, how to reduce reliance on trust-based, institution-driven models of recordkeeping and value transfer. As articulated in its original whitepaper, the objective was to create “an electronic payment system based on cryptographic proof instead of trust, allowing any two willing parties to transact with each other without the need for a trusted third party.”¹⁰ This architectural

⁸ The Bitcoin genesis block, Block 0, was created on January 3, 2009. See details of the genesis block accessed at <https://www.blockchain.com/explorer/blocks/btc/0>.

⁹ Earlier design proposals explored decentralized digital currency and cryptographic recordkeeping concepts but did not culminate in a deployed, permissionless blockchain system with sustained operation and adoption. See, for instance, Dai, Wei (1998), “*b-money*,” accessed at <https://www.weidai.com/bmoney.txt>, and Szabo, Nick (2005), “*Bit gold*,” accessed at <https://unenumerated.blogspot.com/2005/12/bit-gold.html>.

¹⁰ See Nakamoto, Satoshi (2008), “*Bitcoin: A Peer-to-Peer Electronic Cash System*,” accessed at <https://bitcoin.org/bitcoin.pdf>.

choice fundamentally reframed the problem of trust in digital recordkeeping systems, shifting it from trust-based reliance on institutional enforcement to trustless validation through transparent protocol-enforced rules.

Although Bitcoin established the feasibility of decentralized, trustless integrity, its design prioritized security and robustness at the expense of computational efficiency and functional flexibility.¹¹ Subsequent blockchain systems sought to expand the range and scope of state-contingent interactions that could be coordinated on-chain. A landmark development in this evolutionary path was the deployment of the Ethereum blockchain in 2015,¹² which extended the blockchain paradigm toward more expressive execution environments.¹³ By introducing a general-purpose, programmable smart-contract layer at the protocol level, Ethereum enabled developers to encode more complex forms of conditional logic, natively supporting a wider

¹¹ The Bitcoin protocol's functional simplicity and limited expressiveness are often regarded as features rather than flaws, reflecting an emphasis on security, predictability, and minimization of attack surfaces. See, for instance, Lightspark Team (2015), "*Bitcoin Script: The Programming Language of Money*," accessed at <https://www.lightspark.com/glossary/bitcoin-script>, and Blatchford, Jack & Matt Allen (2025), "*The Bitcoin Renaissance Series 2.0: A Technical Deep Dive*," accessed at https://fcatalyst.com/bin-public/060_www_fidelity_com/external_fcat/documents/Bitcoin-Renaissance-2.0-A-Technical-Deep-Dive.pdf.

¹² The Ethereum genesis block, Block 0, was created on July 30, 2015. See details of the genesis block accessed at <https://etherscan.io/block/0>.

¹³ Vitalik Buterin, a founding developer of Ethereum, described the project's origins as an explicit response to limitations encountered in his earlier experience as a Bitcoin developer, particularly his work on "colored coins"—an ad hoc, overlay-based approach aimed at recording ownership claims to securities and other real-world assets on the Bitcoin blockchain. In the course of these efforts, Buterin encountered significant technical frictions, stating:

"[w]hat I soon realized, however, was the sheer difficulty that many of these [Bitcoin-based] projects were facing, and the often ugly technological hacks that were required to make them work. [...] Each individual project was attempting to implement its own blockchain or meta-layer on top of Bitcoin, and considerable effort was being duplicated and interoperability lost as a result. [...] In short, Ethereum is a next-generation cryptographic ledger that intends to support numerous advanced features, [...] which allows you to code the features yourself through a mechanism known as [smart] 'contracts'."

See Buterin, Vitalik (2014), "*Ethereum: Now Going Public*," Ethereum Foundation Blog, accessed at <https://blog.ethereum.org/2014/01/23/ethereum-now-going-public/>. For more details on the use cases and technical implementation of colored coins, see, for instance, Rosenfeld, Meni (2012), "*Overview of Colored Coins*," accessed at <https://bitcoil.co.il/BitcoinX.pdf>.

range of applications than basic value transfer—akin to the move from linear, notepad-style digital records to spreadsheet-based recordkeeping, which expanded expressiveness and composability while introducing new classes of logic- and design-related risk, even as the integrity of the underlying storage layer remained unchanged.¹⁴

As a result, the evolution from Bitcoin to Ethereum illustrates how blockchain architectures have developed through iterative responses to both technical constraints and institutional design challenges. Likewise, other blockchain systems reflect different architectural priorities through alternative design choices at the protocol level. For example, Stellar—deployed in 2015¹⁵—was designed with a set of native operations intended to facilitate regulatory compliance and financial-institution use cases, including built-in support for asset issuance, account authorization, and transaction controls relevant to anti-money laundering (AML) requirements.¹⁶ By contrast, Ethereum largely defers such additive features to customizable application-layer contracts and widely adopted interface standards, rather than embedding them directly in the base protocol.

¹⁴ These expanded execution capabilities introduce additional sources of risk at the application layer, where vulnerabilities may arise from contract logic and implementation choices, distinct from the security properties of the underlying blockchain protocol.

¹⁵ I reference Stellar’s deployment date as 2015 rather than 2014 to reflect the launch of its current protocol and network state. Specifically, the Stellar ledger was reinitiated on September 30, 2015 following consensus failures in its initial 2014 deployment, which was based on a protocol forked from the Ripple codebase. See, for instance, Higgins, Stan (2014), “*Stellar Network Fork Prompts Concerns Over Ripple Consensus Protocol*,” CoinDesk, accessed at <https://www.coindesk.com/markets/2014/12/09/stellar-network-fork-prompts-concerns-over-ripple-consensus-protocol/>. See also the reinitiated ledger beginning with Ledger 2, accessed at <https://stellar.expert/explorer/public/ledger/2>.

¹⁶ See, for instance, Stellar Development Foundation (2017), “*Using Stellar for icos*,” Stellar Blog, accessed at <https://stellar.org/blog/foundation-news/using-stellar-for-ico>. As described therein: “Stellar’s primary goal is to facilitate issuing and trading tokens, especially those tied to legal commitments by known organizations, such as claims on real-world assets or fiat currency... [including] built-in token capabilities that can support multi-signature authorizations, generate dividends, limit who can hold tokens, and more.”

Overall, these differing approaches illustrate how public blockchains occupy distinct points in the design space, trading off flexibility, standardization, and regulatory alignment in service of particular use cases rather than converging on a single, dominant architecture. In the next section, I examine these architectural components in greater detail, outlining the core technical and economic building blocks that give rise to these tradeoffs.

3. Conceptual Foundations and Core Technical Architecture

By integrating cryptographic techniques with decentralized consensus mechanisms, blockchain systems enable the maintenance of a shared, append-only ledger whose integrity does not depend on any single institution. Rather than relying on a trusted intermediary to validate transactions and update ownership records, blockchains distribute these functions across a network of participants who collectively enforce protocol rules. This architectural design fundamentally alters how trust is established in digital systems, substituting institutional authority with cryptographic verification and protocol-defined coordination.

Conceptually, blockchain architectures combine several core components: cryptographic primitives that secure data and authenticate participants, consensus mechanisms that coordinate agreement over system state, execution environments that support programmable logic, and economic design features that govern participation, ongoing development, and resource allocation. While implementations vary across systems, these elements constitute the shared foundation underlying blockchain networks.

In this section, I explain how trustless verification is established using *cryptographic primitives* (Section 3.a), describe how agreement on the valid state of the ledger is achieved

through *consensus mechanisms* (Section 3.b), discuss how *smart contracts and interface standards* define system capabilities and enable coordination across applications (Section 3.c), and analyze how *incentive alignment and economic design* sustain leaderless participation and long-run system functionality (Section 3.d).

a. Cryptographic Primitives

Cryptographic primitives provide the foundational security guarantees underlying blockchain systems. In particular, hash functions and digital signatures play a central role by enabling verifiable commitments and authorization in an adversarial environment, without reliance on institutional identity checks or centralized verification. Together, these mechanisms enable data integrity and secure access to accounts while limiting information disclosure, thereby substituting cryptographic proof for institutional attestation.

Using cryptographically derived public–private key pairs, digital signatures enable participants to credibly demonstrate control over accounts and authorize transactions without relying on centralized identity verification or disclosing private credentials. By separating authorization from identity and password-based checks, signature schemes enable pseudonymous participation—where only the public key and transaction request are revealed—while allowing transaction validity to be verified cryptographically rather than institutionally.¹⁷

¹⁷ In **Appendix B**, I provide a stylized example of a public–private key pair and the corresponding digital signature process to illustrate how valid transaction requests are confirmed and unauthorized attempts are rejected.

While digital signatures establish ownership and control, hash functions enable verifiable, tamper-evident commitments to recorded transaction data that ensure ledger integrity. Hash functions map arbitrary-length inputs to fixed-length outputs,¹⁸ commonly referred to as hash values. Because even a small change in the input produces a materially different output, any attempts to alter historical transaction records are readily detectable through a hash mismatch. Hash functions therefore support the trustless integrity of an append-only ledger without requiring centralized oversight or institutional enforcement.

Together, these tools support trustless validity and non-repudiation in a decentralized environment. Importantly, the cryptography used in blockchain systems is not novel in isolation; rather, its significance lies in the combination of well-established cryptographic techniques with distributed coordination mechanisms to authorize valid transactions, reject invalid requests, and secure shared state without centralized oversight.

b. Consensus Mechanisms

Consensus mechanisms determine how blockchain networks agree on the state of the ledger in the presence of distributed participants and potential adversaries. In public, permissionless blockchains, consensus protocols must operate without assuming trusted identities or permissioned control, making them especially demanding from a design perspective.

Bitcoin's Proof-of-Work (PoW) mechanism was the first widely deployed, operational solution to this problem, requiring participants to solve a computationally intensive puzzle in

¹⁸ Unlike encryption schemes, hash functions are one-way mappings: given a hash value, it is computationally infeasible to recover the original input. As a result, hash functions are used to commit to data—and to sequences of data—ensuring data integrity rather than confidentiality.

order to append new blocks of transaction data to the blockchain. Two defining characteristics of PoW are the brute-force work required to produce a valid block and the relative ease with which other network participants can verify that block once it is proposed.¹⁹ As a result, historical blocks become increasingly difficult to alter, since doing so would require re-solving the PoW puzzle for the modified block and all subsequent blocks, entailing substantial cumulative computational work to restore full ledger consistency.

While PoW offers strong security properties, it is inherently resource-intensive and constrains transaction throughput. Subsequent systems have explored alternative approaches—most notably Proof of Stake (PoS)—which replace energy-intensive computation with stake-based validation rights to propose and secure new blocks. PoS mechanisms can be more resource-efficient and support higher throughput, but they introduce greater mathematical and incentive-design complexity than brute-force PoW mechanisms, particularly in balancing safety and liveness in adversarial environments.

Across consensus designs, blockchain systems face persistent tradeoffs among security, scalability, and decentralization. Improving transaction throughput or reducing latency often requires concessions along other dimensions, such as increased reliance on privileged validators or layered architectures. As a result, some networks have adopted multi-layer designs, in which a base “Layer 1” protocol prioritizes security and decentralization, while “Layer 2” protocols

¹⁹ The underlying logic of PoW mirrors many familiar asymmetric-verification problems. For example, guessing an unknown phone number requires brute-force trial and error, whereas verifying a provided number is quick and straightforward. PoW mechanisms similarly rely on costly block production paired with inexpensive public verification.

process higher-frequency transactions off the main chain and periodically bundle transaction data into compressed commitments submitted to the base layer.²⁰

Thus, consensus mechanisms address the core problem of agreement in decentralized systems: how a network of unvetted participants can converge on a shared ledger state without centralized oversight. However, consensus alone only determines which transactions are accepted and in what order—not what those transactions do, how they are composed to enable interoperability across applications, or why participants continue to supply effort over time. These additional dimensions are addressed through execution environments, interface standards, and incentive mechanisms, which jointly shape how blockchain systems support complex applications and maintain operational viability over time.

c. Smart Contracts and Interface Standards

Early blockchain systems were designed primarily as specialized recordkeeping systems, with intentionally limited scripting capabilities centered on recording ownership and transferences of ownership. The introduction of programmable smart contracts marked a significant expansion of blockchain functionality—with the deployment of the Ethereum blockchain in 2015, blockchains evolved toward more expressive execution environments capable of natively supporting more complex conditional logic and interoperable applications.

Smart contracts are deterministic software programs enabling developers to encode rules and store data directly in on-chain applications that execute automatically when

²⁰ Examples include base-layer protocols such as Bitcoin and Ethereum, and Layer 2 systems such as the Lightning network or roll-up based systems.

predefined conditions are met. However, with so many developers working independently across the world, interoperability across applications is not automatic but must be deliberately engineered. To facilitate interoperability and promote widespread adoption, each blockchain ecosystem maintains a system of interface standards and pre-vetted smart-contract libraries, particularly for common use cases such as token issuance and asset transfers.²¹ Similar to how standardized specifications for credit-card issuers enable merchants to accept different cards without maintaining a multitude of issuer-specific hardware, agreed-upon interface standards for common on-chain applications mitigate the need for developers to write extensive custom code to ensure compatibility across applications and tooling.²² Furthermore, interface standards streamline the development process by facilitating composability, enabling developers to selectively assemble reusable application components into customized solutions—much like modular LEGO bricks that allow users to combine pieces in a wide variety of configurations.

²¹ On Ethereum, for instance, interface standards are developed through the *Ethereum Improvement Proposal* (EIP) process, with EIP-1 establishing the purpose, classification, and submission guidelines for proposals. A specialized class of EIPs—known as *Ethereum Requests for Comment* (ERCs)—governs interface standards for common applications built on Ethereum. Notable examples include ERC-20 (EIP-20), which defines a standard interface for fungible tokens, and ERC-721 (EIP-721), which defines a standard interface for non-fungible tokens. See Becze, Martin, and Hudson Jameson (2015), “EIP-1: EIP Purpose and Guidelines,” Ethereum Improvement Proposals, accessed at <https://eips.ethereum.org/EIPS/eip-1>; Buterin, Vitalik, and Fabian Vogelsteller (2015), “ERC-20: Token Standard,” Ethereum Improvement Proposals, accessed at <https://eips.ethereum.org/EIPS/eip-20/>; and Entriiken, William, Jacob Evans, Nastassia Sachs, and Dieter Shirley (2018), “ERC-721: Non-Fungible Token Standard,” Ethereum Improvement Proposals, accessed at <https://eips.ethereum.org/EIPS/eip-721>.

²² In this respect, a blockchain network’s process for adopting interface standards parallels the role of the International Organization for Standardization (ISO), which convenes experts across the world to propose global standards for widely used products and services. For instance, ISO/IEC 7810 specifies the physical characteristics of identification and payment cards, including the ID-1 format used for credit cards, which standardizes dimensions and thickness, among other attributes, to ensure global interoperability. See International Organization for Standardization (2019), “ISO/IEC 7810:2019(en) Identification Cards — Physical Characteristics,” accessed at <https://www.iso.org/standard/70483.html/> and <https://www.iso.org/obp/ui/en/#iso:std:iso-iec:7810:ed-4:v1:en>.

In addition to its benefits, however, increased programmability introduces new risks, including application-level vulnerabilities, unintended interactions among contracts, and challenges related to the immutability of deployed code. As a result, smart contract design highlights a recurring tension between flexibility and reliability that recurs throughout blockchain system architecture.

d. Incentive Alignment Mechanisms and Economic Design

Blockchain protocols also embed organizational and economic mechanisms that govern resource allocation, incentivize leaderless participation, and support long-run system sustainability. These mechanisms include rules governing native-token issuance and supply adjustment, transaction-fee design, and compensation structures for transaction validators. Together, these features determine how scarce resources—such as block space or computational capacity—are allocated and managed over time through protocol-level rules and validator incentives. While these mechanisms do not, by themselves, guarantee desirable outcomes, they shape the incentives faced by participants and influence collective network behavior over time. As blockchain systems have evolved, experimentation with these economic design choices has become a central aspect of protocol design, reflecting ongoing efforts to balance security, usability, and operational viability.

Incentive alignment in blockchain systems also shapes the ongoing development and maintenance of the underlying protocol itself. In the absence of a formal organizational hierarchy or centralized authority, developers and validators who hold native tokens are economically exposed to network performance and credibility. This exposure creates incentives

for stakeholders to propose and coordinate technical updates that preserve or enhance long-term network value (and, by extension, native-token value), with adoption ultimately depending on decentralized coordination among participants rather than formal, centralized command.²³ Protocol evolution therefore emerges endogenously from the interaction of technical design choices and economic incentives, rather than from hierarchical control.

These protocol-level design choices have implications that extend beyond technical operation, influencing how blockchain systems are adopted, governed, and integrated into real-world economic activity. In the following section, I examine applications across sectors, illustrating how blockchain architectures translate from technical design into practical deployment.

4. Applications Across Sectors

Blockchain-based applications differ substantially in their economic rationale and practical viability across domains. Although early narratives often focused on broad applicability,

²³ In practice, notwithstanding the absence of formalized hierarchical authority, coordinated updates emerge through informal social consensus in which a participant's relative influence is shaped by economic stake, technical capacity, and network position. For example, proposed updates to the Bitcoin protocol are articulated through Bitcoin Improvement Proposals (BIPs) (see Taki, Amir, 2011, "BIP-1: BIP Purpose and Guidelines," Bitcoin Improvement Proposals, accessed at <https://github.com/bitcoin/bips/blob/master/bip-0001.mediawiki>). Adoption is signaled through validator (i.e., PoW miner) behavior—such as version signaling through feature flags included in blocks—and ultimately realized only if a sufficient share of economically relevant nodes and validators elect to run the updated software.

By contrast, updates to the Ethereum protocol are coordinated primarily through an established group of developers commonly referred to as Ethereum Core Developers. Although any participant may submit an Ethereum Improvement Proposal (EIP) (see Becze, Martin, and Hudson Jameson, 2015, "EIP-1: EIP Purpose and Guidelines," Ethereum Improvement Proposals, accessed at <https://eips.ethereum.org/EIPS/eip-1>), decisions regarding inclusion and implementation are reached through off-chain deliberation and client development rather than through direct validator voting. More broadly, many newer blockchain systems retain significant protocol-development authority within a founding team or core development group. While participation as a node or validator in such networks is typically permissionless and leaderless, control over the evolution of the protocol rules themselves is often significantly more centralized.

experience has shown that blockchain systems tend to be most effective in settings characterized by fragmented trust, limited institutional enforcement, or coordination problems that are difficult to address through centralized intermediaries. In this section, I highlight several prominent application areas, emphasizing both their promise and limitations.

a. Decentralized Financial Intermediation

Decentralized financial intermediation represents one of the most developed application domains of blockchain technology. Through smart contracts and distributed execution, blockchain-based systems reproduce selected core functions traditionally performed by trusted financial intermediaries, including asset exchange, liquidity provision, and credit formation. While decentralized exchanges and lending protocols differ in their specific mechanisms, they share a common architectural logic: market rules are encoded directly into on-chain smart contracts, with economic incentives and open-source code²⁴ substituting for institutional discretion and centralized control.

In conventional market microstructure, centralized exchanges (CEXs) rely on off-chain matching engines and custodial ownership records to support faster execution,²⁵ deeper liquidity, and more sophisticated order types. By contrast, decentralized exchanges (DEXs) execute trades and manage liquidity directly on-chain through smart contracts. This design

²⁴ In standard practice, smart contracts deployed on public blockchains are accompanied by publicly available source code that can be independently verified against the compiled bytecode executed on-chain. This transparency allows market participants to confirm that the deployed contracts implement their purported functions without hidden logic or discretionary backdoors, reinforcing the verifiability and contestability of protocol-enforced market mechanisms.

²⁵ That is, traditional financial intermediaries typically hold assets in street name or omnibus accounts to facilitate efficient exchange through centralized coordination, while tracking beneficial ownership separately.

reduces counterparty risk and enables permissionless participation by both traders and liquidity providers. These features are particularly valuable in environments where access to trusted financial intermediaries is limited or where censorship resistance is a priority, as participation and asset listing are governed by protocol rules rather than discretionary approval by a centralized operator.

At the same time, on-chain execution introduces other tradeoffs. Because transaction processing competes with other on-chain activity for limited block space, DEXs expose users to higher transaction costs, latency, and congestion, as well as to liquidity fragmentation across protocols and blockchains.²⁶ While on-chain order books have long been technically feasible,²⁷ CEXs have generally been more effective at supporting high-frequency trading with deep order books, tight spreads, and complex order types at scale. Moreover, DEXs are inherently constrained to assets native to, or represented on, a given blockchain environment, and therefore cannot natively support cross-chain asset exchange or direct conversion into fiat currency. These limitations persist even as DEXs offer advantages in

²⁶ In addition to execution-related constraints, some decentralized exchange designs—particularly automated market-making mechanisms—expose liquidity providers and traders to risks such as impermanent loss and price slippage. These risks further affect market quality and participant incentives and reflect specific implementation choices rather than inherent properties of decentralized exchange architectures. See also Lehar and Parlour (2025), who compare Uniswap’s AMM model with a traditional limit-order-book model and identify conditions under which one market design may dominate the other.

²⁷ On-chain order books have been technically feasible since early blockchain implementations. However, because adding, modifying, and removing liquidity from the order book requires on-chain transactions, maintaining and updating an on-chain order book has historically been costly and slow, particularly on shared execution environments. These constraints have limited the economic viability of fully on-chain order books relative to off-chain matching engines, especially for high-frequency liquidity providers and traders.

For a non-technical overview of decentralized exchange designs, including automated market makers and order-book-based models, see Kim, Seoyoung (2023), “Chapter 6: Decentralized Exchanges and Marketplaces,” *DeFi For Dummies*.

transparency, composability, and censorship resistance. Similar tradeoffs arise in decentralized versus centralized lending. In traditional centralized finance (CeFi), lenders rely on credit assessment, balance-sheet intermediation, and legal enforcement to extend undercollateralized, trust-based credit to a broad range of borrowers. By contrast, decentralized lending protocols enable users to borrow and lend tokenized assets through on-chain smart contracts that require overcollateralization and enable forced liquidation under predefined conditions, reflecting the absence of identity-based credit assessment and enforceable legal claims. These design choices enable permissionless participation and efficient settlement, but at the cost of capital efficiency and scope, confining most activity to volatile digital assets with observable on-chain prices.

Because decentralized lending protocols cannot rely on borrower identity or legal recourse, credit risk is managed primarily through collateral seizure rather than ex-ante screening or other relationship-based enforcement. Substantial overcollateralization, combined

with liquidation mechanisms triggered by price oracles²⁸ and third-party liquidators,²⁹ substitutes for credit evaluation. While this structure reduces discretionary counterparty risk, it can amplify volatility during market stress by inducing liquidation cascades and limits the feasibility of long-term, undercollateralized lending relationships that characterize most consumer and commercial credit markets today.

At the extreme, some decentralized protocols support flash loans, which provide short-duration liquidity that must be repaid within the same block of transactions in which it is borrowed. Repayment is enforced atomically³⁰ at block finalization: if the transaction sequence does not result in successful repayment by the end of the block, the entire sequence is reverted and the loan is never issued. By eliminating intertemporal default risk, flash loans substitute

²⁸ Because blockchains are closed systems that cannot directly access off-chain information, so-called price “oracles” are required to transmit external market price information to on-chain lending protocols. Oracles therefore enable on-chain smart contracts to assess collateral values and determine whether liquidation thresholds have been breached, but also introduce a reliance on external data providers and associated risks related to data accuracy, latency, and manipulation.

In response to these limitations, decentralized oracle networks were developed to reduce reliance on any single data provider by using open-source smart contracts to transparently aggregate inputs from multiple off-chain sources into a determinative on-chain price feed. While such designs can mitigate certain failure modes associated with centralized data feeds, they do not eliminate the fundamental dependence on off-chain information and associated verification challenges. See Ellis, Steve, Ari Juels, and Sergey Nazarov (2017), “*ChainLink—A Decentralized Oracle Network*,” accessed at <https://research.chain.link/whitepaper-v1.pdf>. See also Harper, Colin (2018), “*What Is ChainLink? A Beginner’s Guide to Decentralized Oracles*,” CoinCentral, accessed at <https://coincentral.com/what-is-chainlink-a-beginners-guide-to-decentralized-oracles/>.

²⁹ Other participants in the decentralized lending ecosystem, known as liquidators, are economically incentivized to monitor collateral positions and initiate liquidation when predefined conditions are met. Rather than being executed automatically by the lending protocol itself, liquidations are carried out by these third-party actors who compete to seize and liquidate the collateral backing undercollateralized loans in exchange for a liquidation fee. In this way, solvency constraints are enforced through open participation and economic incentives, without discretionary intervention or centralized control. See, for example, the Compound lending protocol’s documentation on “*Liquidation*,” accessed at <https://docs.compound.finance/liquidation/>.

³⁰ Atomic enforcement, or atomicity, refers to the property that a sequence of instructions is treated as an indivisible unit: either all steps execute successfully, or none are executed and the system state does not reflect partial execution.

execution guarantees for creditworthiness. While such mechanisms expand execution flexibility and support arbitrage or liquidation strategies, they do not constitute intertemporal credit and therefore do not relax the structural constraints that limit decentralized lending markets. Instead, they illustrate the boundary conditions of trust-minimized financial intermediation: as reliance on legal enforcement and borrower identity is removed, viable extensions of undercollateralized credit converge toward arrangements that fully settle within a single execution cycle.

Transaction-ordering discretion introduces additional tradeoffs for decentralized financial applications. On Ethereum and similar blockchains, the block-proposal process permits validators to determine not only which transactions are included in a given block but also their execution order. This discretionary feature has given rise to the strategic pursuit of maximal extractable value (MEV), formerly known as *miner* extractable value, whereby proposed blocks are constructed or selected to maximize the block proposer's compensation.

The pursuit of MEV has produced an industry of specialized block-building intermediaries that aggregate public and privately submitted transaction requests and construct blocks with particular execution sequences.³¹ Builders submit these blocks with competing bids to the validator selected to propose the next block. Thus, the validator's

³¹ Public transaction requests are broadcast to the network and remain in a public memory pool (mempool) until selected for execution. In contrast, private submissions avoid public disclosure before execution and may allow a submitter's own transactions to be strategically sequenced around targeted transactions pending in the public mempool.

compensation may include both publicly specified priority fees attached to individually routed transaction requests and private fees paid for inclusion in a bundled sequence.³²

Private-transaction routing can support and de-risk economically useful activities, such as the liquidation of decentralized loans that have breached a pre-set loan-to-value (LTV) ratio.³³ It can also facilitate harmful strategies, including frontrunning and sandwich attacks,³⁴ that can disadvantage other market participants, particularly those who are less sophisticated or unaware of private transaction routing. Concentration among builders, relays, or validators may create additional censorship and access concerns. These risks have particular implications for on-chain implementations of regulated trading venues, where nontransparent or selectively applied transaction ordering may be incompatible with regulatory requirements.

³² For privately submitted transaction bundles, the submitter's own transactions generally specify a *max priority fee* of zero, with compensation paid through the private bundle-level arrangement. Targeted constituent transactions drawn from the public mempool retain the publicly specified priority fees paid by their respective originators. This arrangement reduces the risk that the submitter's transactions will execute—and incur priority fees—outside the requested sequence.

³³ Private transaction requests can reduce the risk that an unsuccessful liquidation attempt will nevertheless be executed and incur transaction fees by providing a cheaper and more reliable means of cancellation before inclusion if another liquidator acts first or the position no longer satisfies the liquidation conditions. By contrast, attempting to cancel a transaction already broadcast to the public mempool generally requires submitting a replacement transaction with a higher priority fee (and the same transaction nonce), without assurance that the replacement will reach the relevant block proposer first. Transaction ordering can also be used more opportunistically: a liquidator may seek priority over a borrower's pending transaction to add collateral or repay debt, thereby liquidating the position and earning the associated liquidation fee before the borrower restores the loan-to-value ratio to an acceptable level.

³⁴ In a *frontrunning attack*, a sophisticated trader monitors the public mempool to identify pending transactions expected to move an asset's price, such as transaction requests to swap USDT for ETH (i.e., to buy ETH) on a decentralized exchange, and submits a bundled request to place the trader's own USDT-for-ETH swap immediately before the targeted transactions. Frontrunning is often combined with *backrunning*, whereby the trader would also include a request to place a final ETH-for-USDT swap (i.e., to sell ETH) immediately after the targeted transactions, allowing the trader to exit their accumulated position after the intervening purchases have increased the price of ETH. This combination of frontrunning and backrunning the targeted transactions is known as a *sandwich attack*.

Susceptibility to ordering-based value extraction, however, differs across blockchains. Stellar limits deterministic priority by shuffling transactions within a proposed block prior to execution, though validators still retain discretion over which transactions are included.³⁵ Solana's high-throughput, parallel-execution architecture reduces reliance on the longer block times associated with other blockchains but does not eliminate strategic ordering: the scheduled block proposer retains selection and ordering discretion, and MEV extraction is supported by block-engine services that integrate off-chain auctions with the on-chain execution of ordered bundles.³⁶ Accordingly, differences across blockchain architectures alter the form and incidence of MEV extraction rather than necessarily eliminating it. Ultimately, the suitability of a blockchain for regulated financial applications depends not only on application-layer functionality but also on the protocol rules at the consensus and execution layers governing transaction inclusion and execution order. Taken together, these features highlight how decentralized financial intermediation substitutes relationship-based enforcement for open-source, protocol-based execution, constraining capital efficiency in some contexts while expanding access in others. In particular, rule-based, collateral-driven designs may be preferred in settings where discretionary, identity-based credit assessment is viewed as opaque, exclusionary, or difficult to contest. This tradeoff reinforces the importance of institutional context in determining where blockchain-based financial services can meaningfully substitute for existing arrangements.

³⁵ See the section on "Transaction Lifecycle" in Stellar's developer documentation, accessed at <https://developers.stellar.org/docs/learn/fundamentals/transactions/transaction-lifecycle>.

³⁶ See the section on "MEV Protection with Jito DontFront" in Solana's developer documentation, accessed at <https://solana.com/docs/defi/mev-protection>.

b. Digital Assets and Tokenization

With the introduction of natively programmable platforms, blockchains also support richer forms of on-chain recordkeeping and conditional execution. By enabling application-specific logic to be embedded directly into ledger updates, these systems allow ownership records to incorporate constraints, permissions, and automated actions that extend beyond simple value transfer, expanding the range of institutional arrangements that can be implemented on-chain. Conceptually, this expanded functionality allows blockchains to serve as programmable recordkeeping systems capable of enforcing more complex ownership and transfer rules.

Tokenization refers to the use of blockchains to create, update, and maintain authoritative records of transaction and ownership associated with a particular asset, claim, or right.³⁷ In such systems, tokens function as abstract units of account: transferring a token from one account to another updates the shared ledger to reflect a change in ownership or control as defined by the relevant legal or contractual framework. In this respect, tokens function analogously to title-based instruments—such as vehicle titles—where the record of ownership is distinct from the underlying physical asset, but governs rights of transfer and control.

Overall, tokenization has been applied across a wide range of domains, from native digital assets and financial instruments to claims tied to off-chain assets and activities, with varying degrees of institutional reliance and effectiveness. In practice, the adoption of tokenized assets has depended not only on technical design, but also on legal classification,

³⁷ Some digital assets, such as bitcoin or ether, are natively digital and self-contained on-chain such that the asset itself and its ownership record coincide. By contrast, many tokenized assets rely on off-chain legal, custodial, or institutional arrangements, and tokenization alone does not eliminate these dependencies.

governance structures, and institutional readiness—issues examined in greater detail in Section 5.

c. Limits of Applicability

While blockchain-based systems have enabled meaningful innovation across a range of application domains, their effectiveness depends critically on institutional context. Blockchain architectures impose distinctive tradeoffs, and their advantages tend to be realized only where conventional trust mechanisms are costly, unreliable, or contested. In environments with strong legal enforcement, well-functioning intermediaries, and low coordination costs, centralized databases and organizational hierarchies often remain more efficient.

As decentralized applications have evolved, they have demonstrated the capacity to support increasingly sophisticated and composable functionality to enhance user experience. However, this functional expansion has been accompanied by growing implementation complexity, including layered dependencies, upgradeable components, and persistent permissioning mechanisms.³⁸ Although such systems may remain formally open-source and verifiable, the practical ability of most market participants to meaningfully understand or

³⁸ The growth in implementation complexity is not merely theoretical. Early decentralized exchange protocols—such as Uniswap v1 and v2—relied on a relatively small number of core contracts and dependencies, making their effective functional behavior easier to assess through direct inspection. By contrast, more recent protocol iterations—such as Uniswap v4—introduce substantially more modular and composable architectures, including numerous dependencies, hooks, upgradeable components, and persistent permissioning structures. While these designs expand functionality and flexibility, they significantly raise the cost of independent verification for most users and introduce reliance on default permissions and execution assumptions that are rarely examined in practice, even when the codebase remains formally open-source.

See, for instance, codebases for Uniswap v1 and v2, accessed at <https://github.com/Uniswap/v1-contracts/tree/master/contracts> and <https://github.com/Uniswap/v2-core/tree/master/contracts>. In comparison, see core and supporting periphery codebases for Uniswap v4, accessed at <https://github.com/Uniswap/v4-core/tree/main/src> and <https://github.com/Uniswap/v4-periphery/tree/main/src>.

independently audit effective system behavior has diminished. In practice, enhanced functionality often comes at the cost of reduced comprehensibility, shifting reliance toward specialized auditors, tooling, or reputation-based trust rather than direct inspection of protocol logic.

Related limitations arise when blockchain systems are used to coordinate or record activity that depends on off-chain events or representations. While blockchains can enforce consistency and integrity of on-chain records, they cannot independently verify the truthfulness of real-world inputs. In applications such as supply-chain tracking or provenance systems, for example, the transfer of a token may be designed to correspond to the completion of a physical checkpoint. If an actor misrepresents an off-chain action—such as falsely claiming that an inspection has occurred—that misrepresentation is faithfully preserved and propagated on-chain. In such cases, validity is not eliminated but carried over from off-chain institutional verification. Blockchain systems faithfully preserve and propagate recorded information, but where the accuracy of that information depends on real-world actions, reliance on trusted data providers, auditors, certifiers, and enforcement institutions remains necessary both off-chain and on-chain.

In addition to technical and institutional constraints, early adoption of blockchain-based applications was shaped by usability and onboarding frictions inherent in permissionless, rule-based systems. Such systems typically place greater responsibility on end users, requiring them to manage private keys, navigate unfamiliar interfaces, and bear the full consequences of operational errors without recourse to centralized support or discretionary remediation. These design features are not incidental, but reflect the same architectural choice to replace

institutional mediation with rule-based execution. As a result, barriers to adoption have often included not only technical complexity, but also rational user aversion to systems in which errors are irreversible and accountability is diffuse.

These limits underscore that the performance of blockchain-based applications cannot be evaluated in isolation from their institutional surroundings. In many cases, the feasibility and adoption of blockchain-based systems have depended not only on technical design and economic incentives, but also on legal interpretation, regulatory classification, governance structures, and institutional readiness.

5. Legal, Governance, and Institutional Considerations

In practice, technical feasibility and incentive alignment have often proven insufficient to ensure institutional adoption where legal classification remains uncertain, governance authority is diffuse or opaque, and accountability mechanisms are ill-defined. In this section, I examine how these considerations have influenced the trajectory of blockchain adoption, highlighting the interaction between protocol design and existing legal and institutional frameworks.

a. Legal Classification and Regulatory Uncertainty

A central challenge for blockchain-based applications has been the difficulty of mapping decentralized recordkeeping architectures onto existing legal categories. Regulatory frameworks governing securities, commodities, payments, and custodial arrangements were developed around identifiable intermediaries and centralized recordkeepers. Importantly, however, the legal classification of an asset has never depended on the particular technology used to record ownership or transfers. Recording ownership on a spreadsheet rather than a

physical ledger—or on a blockchain rather than a spreadsheet—does not, by itself, alter the legal rights, obligations, enforcement mechanisms, or determinative authority associated with that asset. An asset does not become subject to a different regulatory regime merely because its ownership records are maintained on a blockchain rather than through pre-existing conventional systems.³⁹

Early efforts at tokenization—particularly tokenized securities—illustrate this tension. In practice, decentralized architectures complicated early regulatory engagement not by altering the legal or economic characteristics of assets, but by introducing new recordkeeping infrastructures without a shared conceptual model distinguishing changes in technology from changes in underlying legal substance. Although blockchain-based systems could technically support issuance, transfer, and settlement, uncertainty surrounding regulatory classification and compliance requirements limited institutional participation.

In many cases, existing legal categories were applied to new infrastructures without clear guidance as to how traditional enforcement and compliance requirements should map

³⁹ Recent legislative and regulatory efforts have recognized the distinction between the legal and economic characteristics of an asset and the technology used to record or transfer ownership of that asset. These efforts seek to reduce uncertainty surrounding blockchain-based recordkeeping by clarifying that regulatory classification should turn on the nature of the underlying claim, rather than the medium through which ownership is documented, while preserving existing jurisdictional boundaries.

For example, the Financial Innovation and Technology for the 21st Century Act (FIT21) delineates regulatory oversight of digital assets between the Commodity Futures Trading Commission (CFTC) and the Security and Exchange Commission (SEC), specifying that a “security,” as defined under the Securities Act of 1933, “does not include a digital commodity or permitted payment stablecoin.” See §401 (“*Treatment of Digital Commodities and Other Digital Assets*”), H.R. 4763, 118th Congress (2023-2024), “*Financial Innovation and Technology for the 21st Century Act*,” accessed at <https://www.congress.gov/bill/118th-congress/house-bill/4763/text>.

Similarly, the Digital Asset Market Clarity Act of 2025 (CLARITY Act) further defines digital commodities and clarifies the allocation of regulatory authority over the issuance and exchange of native digital assets and certain tokenized assets. See H.R. 3633, 119th Congress (2025-2026), “*Digital Asset Market Clarity Act of 2025*,” accessed at <https://www.congress.gov/bill/119th-congress/house-bill/3633/text>.

onto decentralized execution environments. As a result, adoption was often delayed not by technical constraints, but by ambiguity regarding permissible design choices and the allocation of enforcement responsibility. In several early attempts, this uncertainty was resolved only by treating the blockchain as a duplicative or auxiliary recordkeeping system, with conventional transfer agents and off-chain books retaining ultimate legal authority in the event of discrepancy.⁴⁰

By contrast, applications such as stablecoins achieved faster adoption primarily because they solved a pressing market-structure problem: the need for a stable on-chain unit of account to support trading, settlement, and liquidity provision within crypto-native markets. This

⁴⁰ For example, the first tokenized registered fund approved in the United States—the Arca U.S. Treasury Fund (ArCoin)—required numerous iterations and amended filings before receiving a statement of effectiveness. See Arca U.S. Treasury Fund, Notice of Effectiveness, July 6, 2020, accessed at https://www.sec.gov/Archives/edgar/data/1758583/999999999520001647/xsIEFFECTX01/primary_doc.xml. See also Bovaird II, Charles (2020), “Arca Uses Ethereum In First SEC-Registered Fund For Digital Shares,” Forbes, July 6, 2020, accessed at <https://www.forbes.com/sites/cbovaird/2020/07/06/arca-launches-first-sec-registered-fund-to-offer-digital-shares/>, and Nelson, Danny (2020), “605 Days Later: How ArCoins Got the SEC Go-Ahead As An Ethereum-Traded Treasurys Fund,” CoinDesk, accessed at <https://www.coindesk.com/markets/2020/07/29/605-days-later-how-arcoins-got-the-sec-go-ahead-as-an-ethereum-traded-treasurys-fund/>.

To satisfy existing securities law and AML obligations, the issuer implemented transfer restrictions through a permissioned extension of the ERC-20 standard, limiting transfers to whitelisted addresses. Ultimately, regulatory approval was conditioned on the Ethereum blockchain serving as a supplementary recordkeeping system, with the traditional transfer agent maintaining the authoritative off-chain books and retaining determinative authority in the event of any discrepancy between on-chain and off-chain records. See Arca U.S. Treasury Fund, Amendment No. 8 to Form N-2, June 24, 2020, accessed at <https://www.sec.gov/Archives/edgar/data/1758583/000121465920005869/s624200n2a2.htm>, which states:

“Although records of peer-to-peer transactions are viewable on Ethereum, record and beneficial ownership of the Fund’s shares is reflected on the records of DTAC, LLC, the Fund’s transfer agent (the “Transfer Agent”). The Transfer Agent is regulated by the Securities and Exchange Commission (“SEC”). The Transfer Agent’s records constitute the official shareholder records of the Fund and govern the record ownership of ArCoins in all circumstances.

Potential shareholders must be “whitelisted” by the Transfer Agent prior to purchasing ArCoins, whether directly or in a peer-to-peer transaction. In order to be whitelisted, each potential shareholder must have completed the Fund’s Anti-Money Laundering/Know Your Customer (“AML/KYC”) process including documentation required for a shareholder to establish a Fund account with the Transfer Agent (see “Anti-Money Laundering/Know Your Customer” and “Plan of Distribution – Purchasing Shares”). The Transfer Agent is also responsible for ensuring that each potential transferee has established a Fund account on the books and records of the Fund.”

adoption preceded—and in some cases outpaced—regulatory clarity or robust governance arrangements.

Over time, however, regulatory engagement followed precisely because privately issued, fiat-backed stablecoins rely on identifiable issuers, centralized reserve management, and custodial structures that can be reconciled with existing legal frameworks.⁴¹ While these designs depart from the ideal of decentralized governance, they provide clear points of accountability that facilitate necessary oversight. The stablecoin case thus reinforces that legal clarity and enforceable accountability—rather than decentralized recordkeeping alone—have been central to institutional acceptance.⁴²

b. Governance and Accountability in Decentralized Systems

Public blockchains introduce distinctive governance and accountability challenges because operational roles are voluntarily assumed by developers, validators, and users rather than assigned within formal hierarchies or centralized ownership structures. While this architectural choice does not alter the underlying legal nature of the assets recorded and transferred on-

⁴¹ Federal legislative efforts—initially reflected in the proposed STABLE Act and culminating in the enactment of the GENIUS Act—have sought to clarify regulatory oversight of stablecoin issuers, including issuance authority, disclosure obligations, redemption rights, reserve management, and supervisory accountability. These measures reflect a broader recognition that stablecoin risks and regulatory treatment depend primarily on issuer structure and asset backing, rather than on the medium used for recordkeeping and settlement.

See H.R. 2392, 119th Congress (2025-2026), “*STABLE Act of 2025*,” accessed at <https://www.congress.gov/bill/119th-congress/house-bill/2392/text>, and GENIUS Act, Pub. L. No. 119-27 (2025), accessed at <https://www.govinfo.gov/app/details/PLAW-119publ27>.

⁴² See also Parlour and Rajan (2025), who compare traditional payment rails with emerging alternatives, including blockchain-based stablecoin transfers, and explore the factors that may facilitate or impede adoption.

chain, it complicates the assignment of responsibility for system design, protocol evolution, and compliance-related functions.

Traditional institutions rely on clearly defined authority structures to make decisions, enforce rules, and resolve disputes. By contrast, protocol evolution in public blockchains typically requires coordination among core developers who propose changes, validators who choose whether to adopt updated software, and end-market participants whose adoption and use of the system ultimately determines network value. Contentious upgrades may result in delays, stalled development, or network fragmentation, as competing groups attempt to reconcile disagreements. In such cases, governance outcomes emerge from collective behavior rather than authoritative command, and disputes are resolved through iterative coordination mechanisms—such as protocol improvement proposal processes—or through participant exit

and migration to alternative networks, including via protocol forking,⁴³ rather than through centralized adjudication.⁴⁴

Related accountability challenges arise at the application layer. Due to the permissionless and append-only nature of blockchain systems, deployed smart contracts may continue to operate and remain accessible indefinitely, even after their original developers have ceased involvement or no longer exercise effective control. Once deployed, application logic can be invoked by any participant interacting with the network, complicating traditional notions of control, recall, and responsibility. As a result, accountability in decentralized systems must primarily rely on ex-ante design choices—such as code constraints, permissioning mechanisms, or upgrade controls—rather than ex-post intervention.

This dynamic further complicates regulatory and institutional engagement, particularly where deployed applications generate ongoing economic effects without a clearly identifiable

⁴³ In blockchain systems, governance disputes may result in a hard fork, whereby a shared codebase diverges into separate, incompatible versions that recognize different rule sets and transaction histories going forward. Such forks create distinct blockchains with separate native tokens, fragmenting the developer, validator, and user community. Forking thus functions as a distinctive governance mechanism, substituting decentralized coordination and voluntary alignment for authoritative dispute resolution. See, for instance, Wilcke, Jeffrey (2016), “*To Fork or Not To Fork*,” accessed at <https://blog.ethereum.org/2016/07/15/to-fork-or-not-to-fork>, and Buterin, Vitalik (2016), “*Onward from the Hard Fork*,” accessed at <https://blog.ethereum.org/2016/07/26/onward-from-the-hard-fork>.

⁴⁴ A prominent example occurred in the Bitcoin network, where heterogeneous preferences regarding block size limits culminated in a hard fork on August 1, 2017. Many participants advocated on-chain scaling by allowing substantially larger block sizes, most notably proposed in BIP-101, which failed to achieve broad consensus. The eventual adoption of BIP-91—which increased effective capacity only indirectly and in a manner designed to support layered scaling solutions such as the Lightning Network—prompted the exit of participants favoring substantial direct increases in block limits and the subsequent emergence of Bitcoin Cash, which adopted a block-size limit of eight megabytes. See Andresen, Gavin, 2015, “*BIP-101: Increase Maximum Block Size*,” accessed at <https://github.com/bitcoin/bips/blob/master/bip-0101.mediawiki>, and Hilliard, James, 2017, “*BIP-91: Reduced Threshold Segwit MASF*,” Bitcoin Improvement Proposals, accessed at <https://github.com/bitcoin/bips/blob/master/bip-0091.mediawiki>. See also Graham, Luke, 2017, “*A new digital currency is about to be created as the bitcoin blockchain is forced to split in two*,” CNBC, accessed at <https://www.cnbc.com/2017/07/31/blockchain-fork-will-create-new-digital-crypto-currency-bitcoin-cash.html>.

operator capable of modifying or withdrawing the system. The accountability challenges posed by immutable, permissionless applications are illustrated by regulatory actions involving Tornado Cash, a set of smart contracts deployed on the Ethereum blockchain. In 2022, the U.S. Department of the Treasury's Office of Foreign Assets Control (OFAC) designated Tornado Cash in connection with illicit financial activity, sanctioning "all property and interests in property of the entity above, Tornado Cash,"⁴⁵ notwithstanding that the core smart contracts were immutable and continued to function without an identifiable operator capable of modifying or disabling them.

Later in 2024, the U.S. Court of Appeals for the Fifth Circuit held that immutable smart contracts do not constitute "property" subject to sanctions under the International Emergency Economic Powers Act (IEEPA),⁴⁶ leading the Treasury Department to lift the designation and remove Tornado Cash from its Specially Designated Nationals (SDN) list.⁴⁷ The episode highlights the tension between traditional enforcement tools and decentralized applications that persist beyond the control of their original developers.

Accountability challenges may arise not only from the absence of an identifiable operator but also from the concentration of de facto control in a nominally leaderless system, as summarized in Table 2. Validator concentration can create opportunities for censorship or consensus

⁴⁵ See U.S. Department of Treasury (2022), "U.S. Treasury Sanctions Notorious Virtual Currency Mixer Tornado Cash," accessed at <https://home.treasury.gov/news/press-releases/jy0916>.

⁴⁶ See Fifth Circuit U.S. Court of Appeals (2024), Case 23-50669, "Van Loon v. Department of the Treasury," accessed at <https://law.justia.com/cases/federal/appellate-courts/ca5/23-50669/23-50669-2024-11-26.html>.

⁴⁷ See also U.S. Department of Treasury (2025), "Tornado Cash Delisting," accessed at <https://home.treasury.gov/news/press-releases/sb0057>.

disruption. Even where transaction execution is decentralized, protocol governance may be captured through concentrated token ownership, administrative privileges, or the disproportionate agenda-setting influence of core developers, foundations, and other technically or economically prominent participants. Concentration can also arise through infrastructure and access layers that are not formally part of protocol governance. Widespread reliance on a dominant consensus client or execution client can expose the network to correlated software failures, while dominant wallet applications and other interfaces can shape transaction routing, staking, permissions, and application access through preconfigured defaults. Thus, control over the software and interfaces through which participants interact with the system may confer substantial de facto influence without corresponding formal accountability.--- Insert Table 2 here ---

- c. For institutional participants, these forms of concentration—and the resulting uncertainty regarding who exercises effective influence and how conflicts are resolved—can pose a significant barrier to adoption. This distinction between leaderless execution and leaderless control—two concepts that are often conflated in popular discourse—raises important questions about transparency, legitimacy, and responsibility, particularly where protocol changes can materially affect asset functionality, compliance obligations, or economic outcomes. This governance ambiguity helps explain why, in current institutional practice, certain applications—such as tokenized securities—have been implemented with blockchains functioning as duplicative or shadow recordkeeping systems rather than as determinative legal ledgers.
- ### Institutional Readiness and Operational Integration

Finally, the feasibility of blockchain-based applications depends on the readiness of existing institutions to integrate decentralized systems into operational, legal, and compliance frameworks. Issues such as custody, liability, reversibility, and auditability remain central to institutional adoption, particularly in regulated sectors where accountability and remediation are essential.

In many contexts, blockchain systems have exposed gaps rather than resolved them. While on-chain settlement can reduce certain operational risks, it does not replace the need for off-chain enforcement, dispute resolution, or accountability in cases of loss, error, or fraud. Tokenization may instead add complexity through layered custodial arrangements, including

custody of the underlying asset and custody or control of the tokens representing interests in that asset. Each layer introduces distinct risks of loss, insolvency, unauthorized transfer, or inconsistency between on-chain and off-chain ownership records. Related accounting treatment—including recognition, measurement, and derecognition—may likewise depend on off-chain determinations of legal ownership, control, and custodial responsibility rather than solely on the blockchain record.

Thus, operational integration requires alignment between on-chain transaction finality and legal settlement finality. A token transfer may be validated and finalized under a blockchain’s consensus rules without conclusively transferring legal ownership of the underlying asset or claim—particularly where an off-chain record remains authoritative or where insolvency, fraud, error, or competing claims permit legal intervention. For instance, guidance published by the Committee on Payments and Market Infrastructures and the International Organization of Securities Commissions states that a systemically important stablecoin arrangement should “clearly define the point at which a transfer of the stablecoin through the operational method used becomes irrevocable and unconditional.”⁴⁸

These gaps are not incidental, but reflect deliberate architectural choices to replace institutional discretion with rule-based execution. As a result, hybrid or permissioned architectures have often emerged as pragmatic compromises, allowing institutions to capture

⁴⁸ See Committee on Payments and Market Infrastructures and International Organization of Securities Commissions (2022), “Application of the Principles for Financial Market Infrastructures to Stablecoin Arrangements,” Section 3.4, accessed at <https://www.bis.org/cpmi/publ/d206.pdf>. The guidance distinguishes legal settlement finality from technical settlement and notes that distributed-ledger arrangements may create misalignment between the two.

some benefits of blockchain-based recordkeeping while retaining oversight and control mechanisms consistent with regulatory expectations, such as off-chain compliance processes mapped to on-chain transfer restrictions.⁴⁹ Taken together, these considerations illustrate that blockchain adoption is shaped as much by institutional compatibility as by technical capability. Public blockchains are neither inherently incompatible with regulation nor universally suitable across contexts. Instead, their viability reflects tradeoffs among decentralization, governance, and institutional integration—tradeoffs that must be evaluated relative to the legal and organizational environments in which they are deployed.

6. Concluding Remarks

Blockchain technology represents a distinctive approach to digital recordkeeping and coordination, enabling shared state without assigning recordkeeping authority to a centralized institution. However, the potential benefits of decentralization must be evaluated relative to the coordination and enforcement costs that motivated the historical emergence of trusted intermediaries, including scalability constraints, governance frictions, operational complexity, and the need for standardized integration with prevailing legal and regulatory frameworks.

Thus, blockchain adoption is best understood as an institutional-design choice rather than a technological inevitability. Different blockchain systems reflect distinct priorities, trading off flexibility, standardization, and institutional alignment in response to specific coordination

⁴⁹ See, for instance, ERC-1404, which was proposed as an extension to the ERC-20 token standard that enforces transfer restrictions through an on-chain whitelist of approved account addresses, with eligibility determined through off-chain KYC/AML processes (Gierlach, Ron, James Poole, Mason Borda, and Lawson Baker, 2018, “ERC-1404: Simple Restricted Token Standard,” accessed at <https://github.com/ethereum/EIPs/issues/1404>).

and agency problems. Accordingly, blockchain-based solutions are most likely to generate value in environments where conventional trust mechanisms are costly, unreliable, or contested, rather than in settings where established legal institutions and authorized intermediaries already perform these functions efficiently. Moreover, decentralization at the consensus or execution layer does not eliminate reliance on off-chain verification, legal enforcement, or concentrated sources of de facto control elsewhere in the system.

Looking forward, ongoing developments in protocol design, governance mechanisms, and regulatory engagement will continue to shape the trajectory of blockchain systems and their applications. Rather than converging on a single dominant architecture, the ecosystem is likely to remain heterogeneous, reflecting variation in economic environments, legal frameworks, and organizational needs. Understanding these institutional boundary conditions is therefore essential for evaluating when—and under what circumstances—blockchain technology can serve as a complementary addition to existing financial and informational infrastructure.

References

118th Congress (2023-2024), H.R. 4763, “*Financial Innovation and Technology for the 21st Century Act*,” accessed at <https://www.congress.gov/bill/118th-congress/house-bill/4763/text>

119th Congress (2025-2026), H.R. 2392, “*STABLE Act of 2025*,” accessed at <https://www.congress.gov/bill/119th-congress/house-bill/2392/text>

119th Congress (2025-2026), H.R. 3633, “*Digital Asset Market Clarity Act of 2025*,” accessed at <https://www.congress.gov/bill/119th-congress/house-bill/3633/text>

Andresen, Gavin, 2015, “*BIP-101: Increase Maximum Block Size*,” Bitcoin Improvement Proposals, accessed at <https://github.com/bitcoin/bips/blob/master/bip-0101.mediawiki>

Arca U.S. Treasury Fund, Amendment No. 8 to Form N-2, June 24, 2020, accessed at <https://www.sec.gov/Archives/edgar/data/1758583/000121465920005869/s624200n2a2.htm>

Arca U.S. Treasury Fund, Notice of Effectiveness, July 6, 2020, accessed at https://www.sec.gov/Archives/edgar/data/1758583/999999999520001647/xsIEFFECTX01/primary_doc.xml

Becze, Martin, and Hudson Jameson (2015), “*EIP-1: EIP Purpose and Guidelines*,” Ethereum Improvement Proposals, accessed at <https://eips.ethereum.org/EIPS/eip-1>

Blatchford, Jack & Matt Allen (2025), “*The Bitcoin Renaissance Series 2.0: A Technical Deep Dive*,” accessed at https://fcatalyst.com/bin-public/060_www_fidelity_com/external_fcat/documents/Bitcoin-Renaissance-2.0-A-Technical-Deep-Dive.pdf

Bovaird II, Charles (2020), “*Arca Uses Ethereum In First SEC-Registered Fund For Digital Shares*,” Forbes, July 6, 2020, accessed at <https://www.forbes.com/sites/cbovaird/2020/07/06/arca-launches-first-sec-registered-fund-to-offer-digital-shares/>

Buterin, Vitalik (2014), "*Ethereum: Now Going Public*," Ethereum Foundation Blog, accessed at <https://blog.ethereum.org/2014/01/23/ethereum-now-going-public/>

Buterin, Vitalik, and Fabian Vogelsteller (2015), "ERC-20: Token Standard," Ethereum Improvement Proposals, accessed at <https://eips.ethereum.org/EIPS/eip-20/>

Buterin, Vitalik (2016), "*Onward from the Hard Fork*," accessed at <https://blog.ethereum.org/2016/07/26/onward-from-the-hard-fork>

Buterin, Vitalik (2017), "*Sharding FAQ*," accessed at https://vitalik.eth.limo/general/2017/12/31/sharding_faq.html

Committee on Payments and Market Infrastructures and International Organization of Securities Commissions (2022), "Application of the Principles for Financial Market Infrastructures to Stablecoin Arrangements," accessed at <https://www.bis.org/cpmi/publ/d206.pdf>

Compound (2022), "Liquidation," *Compound III Documentation*, accessed at <https://github.com/compound-finance/compound-finance.github.io/blob/0ba466bbeaf3b8fd39cef4d6ab77074ee897839d/docs/pages/v3/liquidation.md>

Dai, Wei (1998), "*b-money*," accessed at <https://www.weidai.com/bmoney.txt>

Demsetz, Harold (1967), "*Toward a Theory of Property Rights*," American Economic Review, Vol. 57, No. 2

Ellis, Steve, Ari Juels, and Sergey Nazarov (2017), *“ChainLink—A Decentralized Oracle Network,”* accessed at <https://research.chain.link/whitepaper-v1.pdf>

Entriken, William, Jacob Evans, Nastassia Sachs, and Dieter Shirley (2018), *“ERC-721: Non-Fungible Token Standard,”* Ethereum Improvement Proposals, accessed at <https://eips.ethereum.org/EIPS/eip-721>

[GENIUS Act, Public Law 119-27 \(2025\), “Guiding and Establishing National Innovation for U.S. Stablecoins Act,”](#) accessed at <https://www.govinfo.gov/app/details/PLAW-119publ27>

Gierlach, Ron, James Poole, Mason Borda, and Lawson Baker, 2018, *“ERC-1404: Simple Restricted Token Standard,”* accessed at <https://github.com/ethereum/EIPs/issues/1404>

Graham, Luke, 2017, *“A new digital currency is about to be created as the bitcoin blockchain is forced to split in two,”* CNBC, accessed at <https://www.cnbc.com/2017/07/31/blockchain-fork-will-create-new-digital-crypto-currency-bitcoin-cash.html>

Harper, Colin (2018), *“What Is ChainLink? A Beginner’s Guide to Decentralized Oracles,”* CoinCentral, accessed at <https://coincentral.com/what-is-chainlink-a-beginners-guide-to-decentralized-oracles/>

Higgins, Stan (2014), *“Stellar Network Fork Prompts Concerns Over Ripple Consensus Protocol,”* CoinDesk, accessed at <https://www.coindesk.com/markets/2014/12/09/stellar-network-fork-prompts-concerns-over-ripple-consensus-protocol/>

Hilliard, James, 2017, *“BIP-91: Reduced Threshold Segwit MASF,”* Bitcoin Improvement Proposals, accessed at <https://github.com/bitcoin/bips/blob/master/bip-0091.mediawiki>

International Organization for Standardization (2019), “ISO/IEC 7810:2019(en) Identification Cards — Physical Characteristics,” accessed at <https://www.iso.org/standard/70483.html/> and <https://www.iso.org/obp/ui/en/#iso:std:iso-iec:7810:ed-4:v1:en>

Kim, Seoyoung (2023), “DeFi For Dummies”, John Wiley & Sons, Inc.

Lehar, Alfred, and Christine A. Parlour (2025), “Decentralized Exchange: The Uniswap Automated Market Maker,” *Journal of Finance*, Vol. 80, No. 1, pp. 321–374.

Lightspark Team (2015), “Bitcoin Script: The Programming Language of Money,” accessed at <https://www.lightspark.com/glossary/bitcoin-script>

Nakamoto, Satoshi (2008), “Bitcoin: A Peer-to-Peer Electronic Cash System,” accessed at <https://bitcoin.org/bitcoin.pdf>

Nelson, Danny (2020), “605 Days Later: How ArCoins Got the SEC Go-Ahead As An Ethereum-Traded Treasurys Fund,” CoinDesk, accessed at <https://www.coindesk.com/markets/2020/07/29/605-days-later-how-arcoins-got-the-sec-go-ahead-as-an-ethereum-traded-treasurys-fund/>

Parlour, Christine A., and Uday Rajan (2025), “New Payment Rails: Implications for Banking,” *Annual Review of Financial Economics*, Vol. 17, pp. 207–223.

Rosenfeld, Meni (2012), “Overview of Colored Coins,” accessed at <https://bitcoil.co.il/BitcoinX.pdf>

Solana Foundation (2026), “MEV Protection with Jito DontFront,” *Solana Documentation*, accessed at <https://github.com/solana-foundation/solana->

[com/blob/ab8988e2fda4e2c4ce76f6660d0aecbdf80d906b/apps/docs/content/docs/en/defi/mev-protection.mdx](https://stellar.org/docs/blob/ab8988e2fda4e2c4ce76f6660d0aecbdf80d906b/apps/docs/content/docs/en/defi/mev-protection.mdx)

Stellar Development Foundation (2017), “*Using Stellar for icos*,” Stellar Blog, accessed at <https://stellar.org/blog/foundation-news/using-stellar-for-ico>

Stellar Development Foundation (2025), “Transaction Lifecycle,” *Stellar Documentation*, accessed at <https://github.com/stellar/stellar-docs/blob/2b092bf85c2fa38d3cce4d861628de244e0c89a2/docs/learn/fundamentals/transactions/transaction-lifecycle.mdx>

Szabo, Nick (2005), “*Bit gold*,” accessed at <https://unenumerated.blogspot.com/2005/12/bit-gold.html>

Taki, Amir, 2011, “BIP-1: BIP Purpose and Guidelines,” Bitcoin Improvement Proposals, accessed at <https://github.com/bitcoin/bips/blob/master/bip-0001.mediawiki>

U.S. Court of Appeals, Fifth Circuit (2024), Case 23-50669, “*Van Loon v. Department of the Treasury*,” accessed at <https://law.justia.com/cases/federal/appellate-courts/ca5/23-50669/23-50669-2024-11-26.html>

U.S. Department of Treasury (2022), “*U.S. Treasury Sanctions Notorious Virtual Currency Mixer Tornado Cash*,” accessed at <https://home.treasury.gov/news/press-releases/jy0916>

U.S. Department of Treasury (2025), “*Tornado Cash Delisting*,” accessed at <https://home.treasury.gov/news/press-releases/sb0057>

Wilcke, Jeffrey (2016), *"To Fork or Not To Fork,"* accessed at

<https://blog.ethereum.org/2016/07/15/to-fork-or-not-to-fork>

Table 1. Institutional Boundary Conditions Favoring Centralized versus Public-Blockchain Systems

Institutional Feature	Conditions Favoring Centralized Systems	Conditions Favoring Public Blockchain Systems
Trust and accountability	There is a trusted operator subject to effective oversight	Fragmented trust or concern about discretionary abuse
Coordination	Participants can rely on an established intermediary	Participants lack an accepted coordinating authority
Rule specification and modification	Discretion, reversibility, and remediation are dominant concerns; appropriate actions require interpretation or judgment	Required actions and contingencies can be encoded as precise, formulaic rules ex ante; predictable execution is a dominant concern
Performance	Speed, scale, and privacy are dominant concerns	The need for public verifiability or censorship resistance justifies the operational inefficiencies
Off-chain dependence	Underlying claims require substantial institutional verification or enforcement	Relevant assets and actions can be represented and enforced substantially on-chain
Governance	Clear hierarchical authority is desirable and effective	Open participation and exit provide meaningful benefits

Table 2. Sources and Implications of Concentration in Nominally Leaderless Systems

Source of Concentration	Mechanism	Principal Implications
Validators	Concentrated stake or hash power	Censorship, consensus disruption
Governance participants	Concentrated token ownership, smart-contract privileges, or agenda-setting authority	Capture of blockchain protocol or application-level decisions
Consensus and execution clients	Widespread reliance on a dominant software implementation	Correlated software failures and disproportionate influence by client developers
Wallets and other interfaces	Preconfigured routing, staking, permission, and access defaults	De facto control over the choices users encounter and exercise

Appendix A. Glossary

This Appendix provides an alphabetized glossary of selected technical terms used in this paper.

Append-only Ledger

A recordkeeping structure in which historical entries cannot be altered or deleted, but only extended through the addition of new records, ensuring tamper-evident persistence of past state.

Application Layer

The layer comprising smart contracts and on-chain applications that define conditional logic and economic interactions executed on the underlying protocol layer, subject to its security guarantees.

Application-Layer Risk

Sources of vulnerability arising from smart-contract logic and implementation choices, distinct from failures of the underlying blockchain protocol.

Composability

The ability of smart contracts and applications to interact, build upon, and reuse select components through standardized interfaces, enabling modular system design.

Consensus Mechanism

A protocol that enables distributed participants to agree on the valid state and ordering of the ledger in the presence of potential adversaries, balancing safety and liveness without centralized oversight.

Cryptographic Primitives

Foundational cryptographic tools—such as hash functions and digital signatures—that enable verifiable commitments, data integrity, and authorization without reliance on trusted intermediaries.

Cryptographic Verification

The process by which transaction validity, authorization, and data integrity are confirmed using cryptographic proofs rather than institutional attestation or identity-based trust.

Execution Environment

The computational framework provided by a blockchain that determines how smart-contract logic is processed, including expressiveness, determinism, and execution costs.

Incentive Alignment

The design of economic mechanisms—such as token issuance, fees, and rewards—that motivate decentralized participants to act in ways that sustain system security and functionality over time.

Interface Standards

Agreed-upon specifications for smart-contract behavior and data structures that promote interoperability, reduce duplicative implementation, and enable composability across applications.

Permissioned System

A recordkeeping or transaction-processing system in which participation, validation rights, or access are restricted to authorized entities, typically under institutional governance and legal accountability.

Permissionless System

A system that allows any participant to read from and write to the ledger (subject to protocol rules) without requiring approval, identity verification, or authorization from a centralized intermediary.

Protocol-Defined Coordination

The set of rules embedded in the blockchain protocol that govern transaction validation, state transitions, execution semantics, and consensus, enabling collective enforcement without institutional authority.

Protocol Layer

The base layer of a blockchain system responsible for consensus, ledger maintenance, and enforcement of core validity rules, independent of application-specific logic.

Public Blockchain

A digital recordkeeping system in which the ledger is publicly accessible and participation in transaction validation is open to unvetted participants, governed by protocol rules rather than centralized institutional control.

Smart Contract

A programmable on-chain application that automatically executes predefined logic when specified conditions are met, enabling conditional transactions and state updates without discretionary intervention.

Trustless

The ability of participants to establish the validity of records and rule compliance through cryptographic proofs and protocol-enforced mechanisms, without reliance on a trusted intermediary or identity-based verification.

Validator

A participant responsible for proposing, verifying, or finalizing transactions and blocks under the rules of the blockchain's consensus mechanism, typically compensated through protocol-defined incentives.

Appendix B. Digital Signatures and Public-Private Key Authorization (Stylized Example)

This appendix provides a stylized example showing how digital signatures enable validators to confirm that a transaction request was authorized by the account owner (i.e., the holder of the corresponding private key), without revealing that private key and without relying on a trusted intermediary for identity verification. The example also demonstrates how transaction requests signed without the correct private key are rejected by the network.

Suppose an account is associated with the following public–private key pair:

$$Key_{pub} = \{3, 33\}, \quad Key_{priv} = \{7\}.^{50}$$

To submit a transaction request authorizing the transfer of 4 ETH to another account, a simplified digital signature accompanying the request is formed as follows:

1. Signing (authorization).

The account owner raises the requested transaction value, 4, to the power of the private-key exponent, 7, and divides the result by 33. The resulting remainder,

$$4^7 \bmod 33 = 16,$$

⁵⁰ The conforming public–private key pair generated for this example—i.e., one that satisfies the structural requirements of the signature scheme—does not reflect the cryptographic parameters or key-generation procedures employed in operational blockchain systems such as Bitcoin or Ethereum. This stylized example is intentionally simplified to demonstrate the logic of public-private key authorization using digital signatures. In practice, blockchain systems sign cryptographic hashes of structured transaction data rather than raw transaction values, employ substantially larger key sizes, and use more complex signature schemes. The core intuition, however, is the same: transaction requests are validated if they can be publicly verified as authorized by the relevant private key, without revealing that key or relying on institutional identity checks.

serves as the digital signature authorizing the transaction.

2. Verification (public validation).

Validators apply the publicly known key, $Key_{pub} = \{3, 33\}$, by raising the submitted digital signature, 16, to the power of the public-key exponent, 3, and again dividing the result by 33:

$$16^3 \bmod 33 = 4.$$

Because the result matches the requested transaction value, validators confirm that the transaction was authorized by the holder of the corresponding private key—without learning the private key or the identity of the account owner.

Now consider a malicious actor attempting to submit a fraudulent transaction request transferring 4 ETH from this same account. Because this unauthorized user does not know the private key, suppose he incorrectly guesses a value of 8.

1. Invalid signing attempt.

Using the incorrect private-key exponent, the malicious actor computes:

$$4^8 \bmod 33 = 31,$$

and submits 31 as the purported digital signature.

2. Verification failure.

Validators again apply the public key:

$$31^3 \bmod 33 = 25 \neq 4.$$

Because the verification output does not match the requested transaction value, the transaction request is rejected as invalid.

Thus, this example highlights how digital signatures and control of an account's private key serve as the basis for authorization in blockchain systems, enabling transaction validity to be confirmed or denied without disclosing sensitive information or relying on trusted intermediaries.