ACCUMULATE Protocol Litepaper v2.0

A Universal Interconnect Protocol for DApps and DeFi

December 17, 2021

Abstract

The Accumulate Protocol ("Accumulate") is an identity-based, Delegated Proof-of-Stake blockchain designed to power the digital economy through interoperability with Layer 1 blockchains, integration with enterprise tech stacks, and interfacing with the World Wide Web. Accumulate bypasses the trilemma of security, scalability, and decentralization by constructing its blockchain entirely around Accumulate Digital Identifiers (ADIs) and adding validation to every layer. Identities are defined not by tokens but by hierarchies of keys, which support more complex operations than are possible with the simple and constrained smart contract-based frameworks of other blockchains. The use of digital identities as the core basis of blockchain also provides greater flexibility over key management and enables the creation of independent chains that are processed and validated in parallel over Accumulate's network. Combined with innovative features such as Scratch Space for consensus building, and Anchoring for added security, Accumulate has the potential to become a fast, secure, and linearly scalable blockchain purpose-built for powering decentralized finance applications.

1. Introduction

Decentralized finance (DeFi) is an emerging market whose rapid growth is a testament to the mainstream adoption of blockchain technology by financial institutions. A variety of blockchain networks have joined the DeFi revolution and added functionality to enable borrowing, lending, yield farming, and trading on their platforms. However, most protocols are focused on how to build on existing blockchains rather than how to manage interactions with institutions and validate ownership of financial instruments. This approach works for trading tokens, but once a blockchain interacts with regulators and banks, or processes transactions in the real world (e.g., rental property) it is forced to depend on a set of validators that enter that data into the blockchain. Accumulate takes a pragmatic approach to DeFi by creating an entirely new framework centered around identity that allows individuals and financial institutions to validate authorship, manage their keys over time, build a multi-party consensus off the blockchain, and provide an audit trail for private market assets in its quest to become "the bridge to the digital economy."

1.1 Choosing Identity as the Atomic Unit of the Blockchain

A typical blockchain is organized around addresses, as pioneered by the first blockchain, Bitcoin. Most addresses are a hash of a script that applies a public key and are controlled by a public/private key pair. People are familiar with addresses as random, pseudonymous strings of alphanumeric characters that provide both privacy and transparency for token transactions on the blockchain. However, addresses are neither user friendly nor built for storing ordered data sets. Relying on the first and last characters of an address leaves one open to a Man-in-the-Middle attack. Ordering data sets with addresses is impractical and assigning keys to ordered data sets with different levels of authority is all but impossible.

Accumulate Digital Identifiers (ADIs) are an innovation originally introduced by Factom, which replaces addresses as the atomic unit of the blockchain in Accumulate. Unlike hashes, ADIs are human-readable identities, identifiers, and domains that are chosen by the user or assigned by an organization. ADIs are based off URLs and operate similarly to websites in that a website consists of a domain and a key (e.g. EPP key). Just like an owner of a website can upgrade or downgrade security, modify content, and change visibility to the public without having to create an entirely new website, so too can the owner of an ADI manage their identity. As we explain in the following section, ADIs power DeFi applications by managing identities, tokens, keys, and data with flexibility, security, and order.

1.2 Managing Your Keys Over Time

The ability to update keys and upgrade technology over time is critical to building a blockchain to manage data and value for businesses. However, businesses that depend on smart contracts must often commit themselves to the constant flow of new contracts replacing old contracts, or to preserving certain keys forever. Since an address is essentially the hash of a public key in most blockchains, changes to a public key necessarily requires a change of address. Moving private keys to new wallets and signing transactions to move assets exposes the asset holder to security risks, while audits of the blockchain may flag such movements as tax events.

Accumulate breaks this paradigm by introducing a set of hierarchical keys with different security features that separate keys from the assets they protect. With ADIs, it is possible to manage the keys to data and value over time without issuing new contracts or moving digital assets, similar to how changing the lock on your home is possible without moving to a new property. On Accumulate, the keys to an ADI can be managed like signers on an account so new ADIs don't have to be issued when a company shifts responsibilities over time. The departure or promotion of an employee does not require a new ADI, just new security.

1.3 Minimizing Vulnerability to Attacks

All blockchains are susceptible to 51% attacks where a majority of miners control the hashing power of a Proof-of-Work (PoW) blockchain, or a majority of validators hold a majority of cryptocurrency in a Proof-of-Stake (PoS) blockchain. While 51% attacks are less likely to occur on PoS blockchains due to decreased incentives for bad actors, new projects and smaller blockchain networks are still vulnerable to attacks. To protect against this threat, Accumulate applies a solution that was originally developed by Factom. Accumulate "anchors" the transactions of one blockchain to another so that a successful attack requires both blockchains be compromised. Factom anchored to Bitcoin and Ethereum, effectively buying their security. Accumulate is even more versatile and can anchor transactions to any Layer 1 blockchain, including Solana and Cosmos. See Section 2.4 for a detailed description of anchoring.

1.4 Writing Consensus into the Blockchain

Immutability is one of the defining features of blockchain technology. Everything written to the blockchain is permanent and unalterable. Immutability provides the platform with the means to audit transactions, guarantee data integrity, and resolve disputes. However, immutability tends to come at a price. Developers are faced with a tradeoff between the expense and difficulty of showing their work on the blockchain on one hand, and the lack of accountability when the work of arriving at a consensus is done off the blockchain on the other. A blockchain platform only realizes the benefits of immutability when it posts conclusions. When the process is too cumbersome and costly, the platform loses its appeal as a practical means of verification. Conversely, when the process compromises on measures that ensure immutability, users are less likely to rely on it.

Accumulate provides users with an innovative solution to this dilemma, which challenges every blockchain platform. Accumulate incorporates "Scratch Space", a mechanism for intermediate data that records "discussions" to a transient blockchain called the Pending Chain and submits "conclusions" to the permanent blockchain once a consensus is reached. Data has limited availability, but cryptographic proof of an event written in Scratch Space is immutable. See Section 2.5 for a detailed description of Scratch Space.

1.5 Automating Security and Promoting Best Practices

Security and key management are often left to the user, and following best practices may take considerable time and effort. Users may also be forced to choose between warm and cold key

storage, which can make large transactions risky and day-to-day transactions needlessly complex. For example, a person may want to secure the bulk of their Bitcoin in cold storage with high priority keys and maintain a smaller balance in their wallet with more accessible low priority keys to make everyday purchases. In Accumulate, users can upgrade or downgrade security (by cycling keys), enable or disable Multisig, backup lower priority keys, and reassign key priority all without moving their tokens or exposing sensitive information. By depending on Multisig, eliminating scripting signatures, and having fixed address transaction security, Accumulate limits the possibility of hackers exploiting a vulnerability in the signatures.

2. Components and Features

The Accumulate core team has created a new paradigm for blockchain based on identity management and introduced a variety of innovative features that will power DeFi applications. In this section, we will explore the core components of Accumulate that underlie its security, scalability, interoperability, and flexibility.

2.1 Accumulate Digital Identifiers

Accumulate Digital Identifiers (ADIs) are UTF-8 strings of human readable text that enable smart contracts, consensus building, validator networks, and enterprise level management of digital assets on the Accumulate blockchain. UTF-8 is an encoding system for Unicode that accounts for 97% of all webpages and seamlessly integrates ADIs with servers and web-based apps. Applying Internet standards for addressing and routing to the blockchain allows Accumulate to scale with the Internet and become the first URL indexable blockchain.

Building identifiers directly into the architecture also enables key management over time, adding flexibility for enterprises and consistent transaction speed as the network scales. Specific advantages of building ADIs into the protocol include:

- Security (e.g. Multisig) can be upgraded or downgraded as necessary
- Parties can use ADIs as a sender or receiver of tokens instead of addresses
- ADIs can be bought, sold, or managed by multiple parties
- Parties can update their ADIs to use new keys, enabling key management over time

In addition, the construction of ADIs allows the creation of an unlimited number of accounts and sub-identities. Each sub-identity is processed only by a portion of the network, and since Accumulate is sharded on identity, the network becomes infinitely scalable. Identities and sub-identities are defined by hierarchies of keys rather than tokens, which enables the support of complex identity operations and efficient key management. For example, a business may assign an ADI to each department, and each department may create a hierarchy of keys with different permissions or levels of security. The permission and security of each key may be managed by the employees within a department based on their role or security privilege. The following comparison between Bitcoin and Accumulate illustrates the simplicity of ADIs and the logic of key hierarchies:

Bitcoin Address	Accumulate Address
1AALw9rFoX1DKBv57NtQo6Xsi6KVGPr2g	acc://Corp
3Mi2L2uCRWNY96cbcYdNidZLn2XGSh6jXd	acc://Corp/Eng
bc1q69r6gfrzp6yl9ju2ulqwp78qsmpqefal7s55qq	acc://Corp/Eng/Payments

2.2 ADI Accounts

Accumulate is best described as a network of chains. Each ADI is its own independent set of chains that can be validated in parallel with other ADIs, and every feature that is managed by an ADI can be considered an account of that ADI. As a result, there is not a single chain that can be considered the Accumulate blockchain. ADIs have sub-chains that are defined by their accounts. The accounts for ADIs include:

- Token Accounts, which implement a token account, and track both transactions against the account and deposits to the account
- Data Accounts, which track and organize data that is validated or approved by an identity
- Staking Accounts, which allow you to stake ACME tokens to secure the network in exchange for rewards
- Scratch Accounts, which collect data for consensus building across the Accumulate network and allow the blockchain to coordinate Multisig validation

Every account is also its own, ever growing Merkle Tree, which is a hierarchical data structure most notably used by the Bitcoin blockchain to successively merge hashed data (e.g. transactions) until a single hash, called the Merkle root, is obtained. Merkle Trees separate data validation from the data itself and significantly reduce the volume of data that a validator must maintain to authenticate a transaction. Applied to Accumulate, Merkle Trees make it possible to prove that the contents contained within a chain are complete and immutable without requiring proofs from other chains in the network. This allows the network to scale.

Keys are managed by Key Books that specify a set of Key Pages of different priorities. High priority keys can be kept in cold storage to back up keys that are lost or compromised. Low priority keys can be used in wallets and applications. Different accounts and sub-identities can specify different Key Pages, meaning that the Key Pages managing administrative tasks (high priority) and low value token transactions (low priority) can be different. To see how key hierarchies may be useful, consider a user with a large balance of BTC. This user may manage the majority of their BTC in cold storage using a set of high priority keys with Multisig enabled, but use a set of low priority keys for buying groceries. Accumulate also allows an identity to make a variety of security updates (e.g. enabling Multisig) without moving tokens, which encourages best practices and improves security.

2.3 Validator Accumulator Architecture

A major difference between Accumulate and Factom, which is the basis of many features on which Accumulate incorporates and improves, is the validator accumulator (ValAcc) architecture. This innovation allows a user to organize transactions so they can be sorted into their own chains and their own state, specifying a different validator for each chain type. Validators collect data in the real world and validate records that need to be written in the blockchain. Ordered hashes of validated transaction are fed to accumulators that add the hashes to the Merkle Tree for a particular ADI or account. We refer to accumulators as Block Validator Networks (BVNs) because each accumulator can communicate with another. The output of a BVN is effectively a hash that can be fed into other accumulators, ending with a summary hash of the entire network. A visual description of this architecture is provided in Section 3.

A Directory Network (DN) ties all the BVNs together by collecting the summary hash for all the BVNs and anchors them into the Directory Network Block for every block period. The DN acts as a directory service for Accumulate and resolves questions about the state of Accumulate at every block. This may include collecting signatures from the block validators to prove that their state at the end of a block is actually validated across the protocol.

The ValAcc architecture also makes use of Patricia Tries. These data structures are very much like Merkle Trees, except that they are only used to track the state of a blockchain. Each key has a defined position in a Patricia Trie, and its value can be updated. The root hash of a Patricia Trie represents the state of the entire system. In other words, Patricia Tries are designed to be able to create small cryptographic proofs about the particular state of values at a particular point of time. On a blockchain, a Patricia Trie can be used to prove the balance of an address at a particular block height. A Patricia Trie inside of a BVN has a root that contains the state of all the different chains. When you update a Patricia Trie, you only have to fix the nodes that are impacted by changing a leaf (i.e. data), so it ends up being very efficient to update. Because validation is very simple, it means that transaction cost across entire blockchain should be analogous in cost to making a post on a website.

2.4 Anchoring

Accumulate periodically inserts the Merkle root proofs of its data into Layer 1 chains (e.g. Bitcoin, Ethereum, Solana) in a process called "Anchoring". In other words, a hash in one system is placed in another, which effectively allows you to buy the security of a larger and more secure blockchain for the cost of a single transaction. Factom pioneered the use of Anchoring as a security feature, but realized early on that submitting an Anchor for every transaction was not economically viable and was affected by price instability. Consider the cost of a single transaction on the Bitcoin network in a bull market, or the gas fees on Ethereum during a craze like Cryptokitties.

The solution developed by Factom and applied by Accumulate involves the collection of many entries submitted by many applications and reducing the state of the blockchain to a single hash. This hash is aggregated into a single anchor and written to Bitcoin. In other words, a large group of DN blocks are aggregated into a single anchor similar to how an entire block's worth of entries are aggregated into a Merkle Tree. In the 5 years that Factom was anchoring director blocks to

the Bitcoin network, a total of 255,243 transactions were recorded for a cumulative fee of $47,127, for which we can calculate an average cost of just $0.18 per anchor.

2.5 Scratch Space

The Merkle Tree of a chain defines the membership and order of transactions in that chain, while the Patricia Trie defines the chain's current state. Order, membership, and state are needed to validate authorship and provide an audit trail of an event on the blockchain. If that event is deleted from the Patricia Trie, its associated chain is effectively deleted from the protocol. However, the existence of the event still exists in the hashes that are present in the Merkle Tree. This data structure allows Accumulate to provide a token chain called Scratch Space that deletes transaction history but maintains a cryptographically provable balance for the account similar to how a bank maintains proof of your transactions but only provides transaction data for a period of 3-4 years.

Scratch Space is essentially a mechanism for intermediate data that allows users to record the process of reaching a consensus to the blockchain without changing its state. The Scratch Chain facilitates blockchain-based communication between users for the purpose of managing multi-signature transactions, voting on governance issues, and creating multi-party attestations. Consensus is powered by Tendermint, a PoS consensus engine with a proven history of security. Accumulate will run more than 30 Tendermint networks that we call BVNs, building more parallelism into the blockchain. While data is not guaranteed after 20,000 blocks, or approximately 2.3 weeks, proof of an event created in scratch space will be available forever.

2.6 Sharding

Sharding is a method of splitting and storing a blockchain network into smaller partitions, known as shards. For most blockchains, sharding usually depends on the inefficient serial execution of transactions. Checking the validity of a transaction almost always requires an enumeration of the entire blockchain. Accumulate shards on identity by handling each ADI as a logical shardable unit that can be processed independently from the rest of the blockchain. Each identity is its own chain, and because each chain is its own ever growing Merkle Tree, proving its contents are complete and immutable does not require proofs from other structures in Accumulate.

The Accumulate Network supports a wide range of sharding features:

- Network Sharding which allows nodes to be attached to the Accumulate network without having to process all the messages on the network.
- Route Sharding where messages can be routed more efficiently within the protocol (assuming better infrastructure than on the general network).
- Execution Sharding allows the work of processing and validating transactions to be sharded differently for actual execution as opposed to message routing.

ADIs manage all operations on the Accumulate network, including smart contracts, consensus building, validator networks, and enterprise level management of digital assets. Sharding on

identity therefore ensures that all of these essential operations are processed in parallel, and that the network scales as capacity is added.

2.7 Managed Chains

When chains are created, a manager Key Book can be specified. The manager Key Book reviews validated transactions before transactions are placed on the main chain. The review process allows the manager Key Book to enforce additional rules and requirements, and possibly restructure the transactions once transactions have been validated by the manager Key Book. This provides a powerful smart contracting capability to Accumulate and allows unlimited computational power to be added to the Accumulate Network, all coordinated and managed by the Accumulate Network. In the case of a token chain, for example, the managing Key Book has the authority to apply additional rules to the transaction before validating it, allowing the transaction to be entered onto the token chain and processed. For a security token, these rules may be determined by enforcement agencies like the Securities and Exchange Commission. In the case of a token chain, for example, the managing ADI has the authority to apply additional rules to the transaction before validating it, allowing the transaction to be entered onto the token chain and processed. For a security token, these rules may be determined by enforcement agencies like the Securities and Exchange Commission.

To see how managed chains could be applied in the real-world, consider a security token issued on the Accumulate network by Party A (an ADI) on a managed chain that Party A controls. Tokens issued by Party A could be distributed to Party B (another ADI) to token chains that are controlled by Party B but also managed by Party A. Since Party A is the manager of all chains that hold this particular security token, they have the authority to enforce the restriction that any chains holding the token must be accredited. Any attempt to send the security tokens to chains not managed by A or to ADIs without accreditation would be rejected by Party A. Note that accreditation is a real-world status, and the protocol allows the validator(s) of Party A to use real-world platforms to establish accreditation status of ADIs.

3. System Overview

Accumulate is constructed from a set of independently operated subchains. At the top level of the Accumulate network, Root Identities are used to route transactions to a particular Block Validator Network (BVN), which build blocks for the Accumulate blockchain. Root identities are URL based addresses (e.g. acc://Corp/Eng/Payments) that are hashed in order to create a chain ID that is then fed into the BVN. The amount of state that these block validators need to have online for themselves in order to process transactions is very small, which should allow Accumulate to scale with adoption.

The Directory Network (DN) and BVN are interlocked but independent chains. In other words, they can be viewed as side chains of one another. As illustrated in Figure 1, BVNs are fed into a DN, which ties BVNs together by collecting the summary hash for all the BVNs and putting them in a Directory Network Block for every block period.

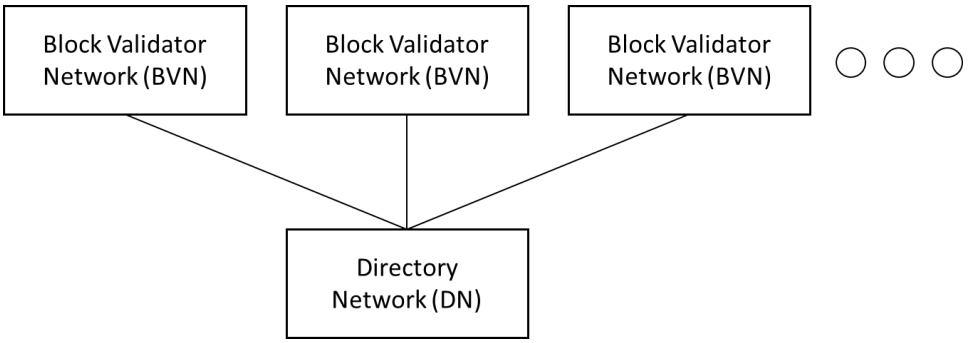

Figure 1: General architecture of Block Validator Network and Directory Network

A closer look at BVNs and DNs reveals an alternating network of accumulators and validators that operate in parallel, across every chain, and at every level of the network. Validation occurs on a chain-by-chain basis, with interactions between chains communicated with synthetic transactions (i.e. transactions that are generated by the protocol itself). If a transaction is validated, it is stored as a hash in a key/value database (i.e. Merkle Tree) and fed into an accumulator as illustrated in Figure 2.

Streams of hashes collected by accumulators are used to build Merkle DAG Trees, where DAG refers to a directed acyclic graph. These data structures are similar to standard Merkle Trees, except that a DAG does not need to be balanced and its non-leaf nodes are permitted to contain data. Merkle DAG Trees are used by the protocol because computing their root requires half the hashes as balancing a Merkle Tree.

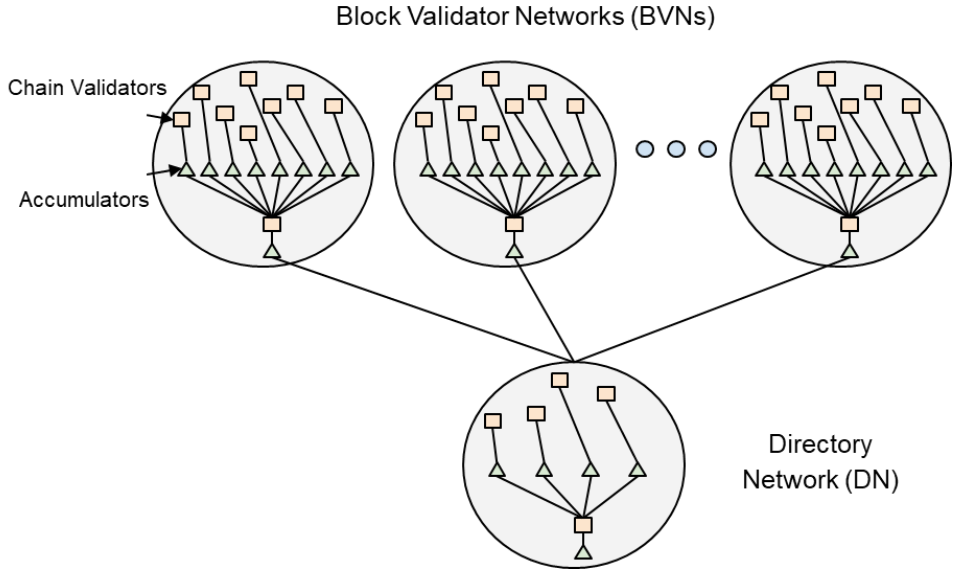

Figure 2: The components of Block Validator and Directory Block Networks

Putting this all together, Accumulate is a network of chains with identity as the core basis of blockchain. Identity is represented by ADIs, which are URL based addresses that enable smart contracts, consensus building, validator networks, and the management of digital assets. ADIs are independent chains that contain subchains for tokens, data, Scratch Space, and keys. Key hierarchies with customizable security and key management specified by Key Books build flexibility, security, and best practices into the blockchain. Managed chains add smart contract functionality and facilitate compliance with regulatory agencies. Transactions that are controlled by ADIs are validated and accumulated into BVNs and the DN, and periodically anchored to Layer 1 blockchains like Bitcoin, Ethereum, or Solana for added security.

4. Conclusion

Accumulate is a multi-chain platform that allows the distribution and validation of its contents across an unbounded number of cooperating servers. Its architecture, based on identity and dynamic key management, enables the efficient and scalable implementation of integrations that provide Financial services, Exchange gateways, Oracles, DeFi services, IOT security, and more. Chains are independently verifiable, allowing applications to finely tune what parts of Accumulate they need. Payments are supported through powerful token gateways, unbounded transaction rates, and low transaction fees. Paying in tokenized real-world currencies as well as a range of cryptocurrencies is also supported on the network. Sophisticated key management in Accumulate ensures high security around tokens, and the ability to manage that security over time. Powerful multi-signature transactions for tokens and data provide the means of creating managed protocols that allow the participants to be onboarded and off boarded over time without disrupting validation processes and procedures. Most importantly, Accumulate is built to evolve over time to integrate the best-in-class mechanisms to distribute transactions, validation proofs, information between applications and users.