UNILAYER: CROSS-CHAIN INFRASTRUCTURAL NETWORK

Document release: 17.02.2023

UniLayer version: alfa 0.7.3

Table of Contents

Brief overview

UniLayer is a one-of-a-kind interoperability platform and L1 blockchain that enables transfers of data and assets across all blockchains. While many aspects of blockchain technology have undergone significant growth and development, there still remains an essential need for secure, direct, cross-chain operability. UniLayer is building the technology to fill that gap and accelerate universal operability.

What Sets Unilayer Apart?

UniLayer is one of the few L1 networks that has been built from the ground up, developing novel, fundamental technologies to realize an uncompromising vision of decentralized interoperability.

UniLayer runs “Universal Nodes”, embedded directly within the blockchain networks that UniLayer connects, validating transactions natively on those networks, while also facilitating cross-chain transactions.

UniLayer’s cross-chain logic smart contracts (“Logix”) allow a single smart contract to connect separate blockchains simply and securely, all through UniLayer’s decentralized L1 network.

UniLayer can natively and seamlessly integrate with any kind of chain: L0, L1, L2.

We are on a mission to connect every blockchain.

Abstract

The idea behind crypto currencies was a peer-to-peer cryptography-based system, in which both parties for p2p transactions would use chains of records signed with hashes. The idea saw the first practical implementation with Bitcoin released by Satoshi Nakamoto in 2009. More than a decade later, we enjoy a huge market of dozens of cryptocurrencies of various relative value, which use various mechanisms. There are, however, new challenges introduced by this situation.

First and foremost, as of now there is no automated way to convert these currencies into each other. Customers have to rely on centralized and decentralized exchangers (CEX, DEX) instead, which defeats the very purpose of eliminating the need for trust-based mechanisms in a transaction.

The second issue is the limited ability of a customer to spend their coins on actual goods and services. Instead of a payment method, cryptocurrencies are seen as an investment and financial tool, which was not their intended purpose back in 2009.

The UniLayer project proposes a solution which would solve both issues, seeking to help finally implement the vision proclaimed by Nakamoto.

General Info

The classic Blockchain Trilemma describes the three main limitations of blockchain technologies: scalability, decentralization, and security. In short, it is seen as very hard to improve on one of these components without damaging the other two (improving security will come at the cost of decentralization and scalability, more scalability will mean more centralization and less security and so on). In practice, this means that emerging chains, DApps and projects grow slower than they would have liked, some struggle with security issues, and the community as a whole is fragmented into individual ecosystems and separate networks, centralized beyond what has been the original idea.

The excessive centralization issue persists due to a lack of mechanisms that would allow for a seamless transition from one ecosystem into the other. This is further worsened by the fact that competing systems often have conflicting, even mutually exclusive implementations of their one basic philosophy, which hinders the development of the decentralized finances (DeFi) market as a whole.

Since their emergence, cryptocurrencies continue to expand over the traditional markets. This expansion is fueled by the interest of private and corporate customers alike, seeking to adopt the idea while it is still relatively young to make the most of it. This, in turn, encourages more and more developers to present their own vision of a cryptocurrency. While a very good thing in itself, this has the side effect of dozens of individual chains existing in their own parallel universes, competing bitterly for investment and users.

Each chain developer has their own idea of a perfect balance between the three “pillars” of the Trilemma. This leads to every chain having its own strong and weak sides. Ethereum, the current DeFi market leader in 2022, suffers from slow transactions and high transaction fees. Others lack sufficient liquidity and scalability or struggle with security issues. The result for the end user is the need to keep several wallets for different tasks. And when they need to transfer their own funds between wallets, they require help from a trusted third party.

The issue is anything, but new, and large ecosystems create their own solutions which allow exchange between blockchains within the ecosystem. The overall centralization burden, however, persists in the form of extensive server infrastructure (as opposed to a fully P2P landscape envisioned by Nakamoto) and oracles, since these ecosystems don't have the tools to interact with each other directly.

Another, less obvious problem with existing interchain solutions within ecosystems is that they tend to use not-blockchain solutions, which may (and often do) pose security risks.

Motivation

The existing blockchain market struggles with numerous issues, as expected from something in its earliest stages of development. The most important are inconvenience for the end user, the total liquidity being fragmented between individual ecosystems and separate blockchains, incompatible solutions requiring additional "duct tape" measures to combine, high and multiple transaction fees, slow transactions and the lack of overall standardization.

Safe scaling may also be achieved by using Layer 2 protocols, often called sidechains, which rely on the zero knowledge cryptography principle. They have less issues with security and centralization compared to the previous group. There are numerous other solutions (like Optimism or Arbitrum), some Ethereum-compatible, some independent, which offer different approaches to enhance the user experience.

What they have in common, however, are difficulties in building a community. Despite all the attention the DeFi opportunities are attracting, enrolling new customers and developers, providing them with the necessary training and accumulating liquidity is still hindered. This happens because every ecosystem has to do it on its own.

Our goal is to provide a means of communication and interaction which will allow developers to guide the evolution of their products in a way which would benefit the industry as a whole, at the same time enhancing the user experience.

UniLayer provides a native way for different blockchains to communicate, share liquidity and help each other improve, facilitating mutual compatibility and progress. For the end user, the implementation of UniLayer will mean full anonymity and transaction safety expected from a block chain, faster transactions and less transaction fees.

Transactions between ecosystems and blockchains are currently bothersome for more than a single reason. One such reason is the possible lack of integration of one of the chains into an existing wider DeFi protocol, which requires running and monitoring several software tools at the same time.

Another problem is that the multiple steps necessary to make such a transaction each require a fee, and these fees pile up, eventually making DeFi activities like trading, farming and staking notably less profitable (this also hinders the progress of blockchains which rely on the Proof of Stake principle to build liquidity).

From the user standpoint, a single tool to automatically transfer their funds between different blockchains in a single transaction will save time, increase cost efficiency and safety. From the point of view of the developers, such a tool would help create a large universal landscape which would allow for interaction between different architectures without the need to either interfere with their own logic or use cumbersome and costly intermediates, freeing their time and brainpower for working on the blockchains themselves.

PART I I Research of Interoperability Solutions

Looking back at the rapid evolution of the blockchain industry towards true interoperability over the past few years, it is clear that we can expect to see even greater opportunities for cross-chain data transfer, as well as increased crypto adoption. Interoperability offers many incredible advantages and unique solutions across a range of industries, allowing users to move their assets more freely within broader cross-chain systems with higher speed, lower fees and greater transparency. This helps build user trust in the data they are presented with due to enhanced trustless data provenance.

Enhanced interoperability will enable various types of digital assets such as cryptocurrencies or utility tokens to be transferred between different blockchains quickly via atomic swaps or sidechains while maintaining high levels of security through cryptographic methods like zero knowledge proofs (ZKPs) and multi-signature protocols (MSAs). Additionally, this would enable users to securely move their data across any chains needed by utilizing trusted execution environments (TEEs) which ensure that all operations occur within secure hardware enclaves. Furthermore, end-user trust in the data is likely to improve due to progress being made regarding making sure there is trustless data provenance when assets are moved freely within broader cross-chain systems using technologies such as Merkle trees for verifying asset transfers on distributed ledgers.

Moreover, true interoperability offers a new level of complexity over blockchain projects that can be attainable thus combining functionality from different chains in the same moment via novel interconnectivity protocols which enables transactions between multiple independent blockchains without requiring them all use identical consensus mechanisms or token standards.

Blockchain bridges

Blockchain Bridges are a type of technology that allow two different networks to communicate with each other, providing users on both chains access to unique features and enabling secure cross-blockchain transactions. These bridges have become increasingly important as the crypto industry evolves into an ecosystem with multiple blockchains, each with its own diverse set of scalability solutions.

In order to understand the concept of blockchain bridges further, it is important to recognize that these types of bridges typically utilize a centralized approach which is managed by operators or organizations. This means that users can expect high levels of speed, efficiency and connectivity when using these trusted bridge protocols; however, there is also a reliance on reputation and security when using them.

Currently there are three main types of blockchain bridges available: Blockchain Bridges, Trusted Bridges and Trustless Bridges. Blockchain Bridges are protocols that allow two or more distinct blockchains to communicate directly without requiring any third party services or custodial accounts - allowing users on both chains access each other's features while keeping funds secured. Examples include Cosmos Hub's Interblockchain Communication (IBC) protocol which enables communication between different blockchains as well as Polkadot's Substrate framework for building interoperable decentralized applications (dApps). Trusted Bridge Protocols provide higher levels of trust than traditional blockchain bridging mechanisms since they rely on external entities such as central banks or governments in order to ensure secure data exchange between two separate networks. Trustless Bridge Protocols offer even greater security measures than their trusted counterparts since they do not require any third-party involvement at all – instead relying solely upon cryptographic algorithms for authentication purposes during network interaction processes such as atomic swaps between currencies built on different blockchains like Bitcoin and Ethereum using technologies like Lightning Network & Atomic Swap DEXes respectively.

Trusted bridges

Trusted, or custodial bridge protocols are managed by an operator or organization in a centralized manner and offer high levels of speed and efficiency for users who must trust the reputation of the bridge provider for security purposes. The main advantage that these types of bridges have over trustless bridges is their ability to quickly process transactions due to their reliance on external sources such as "oracles" which provide real-time data feeds from multiple sources via APIs, webhooks & other communication methods .

The use of trusted bridge protocols allows organizations to reduce infrastructure costs associated with managing blockchain networks while also increasing scalability, however there are some drawbacks associated with this approach. By relying on third party services, users must place their trust in the operator or organization managing the system; making them vulnerable points for attack or data manipulation due to centralization issues. Furthermore, should any changes be made within the system itself, it would require manual intervention from a trusted third party which can lead to delays and potential security breaches if not properly monitored and handled efficiently . To mitigate these risks , organizations using robust security measures such as role-based access control (RBAC) systems , intrusion detection/prevention systems (IDS/IPS), encryption technologies , secure authentication methods like multi-factor authentication (MFA) , public key infrastructure (PKI) solutions , cryptographic hashing algorithms & digital signature schemes etc., all along with regular monitoring processes and periodic threat assessments.

Should any changes be made within the system itself, it would require manual intervention from a trusted third party which can lead to delays and potential security breaches if not properly monitored and handled efficiently.

In contrast with trusted bridges, backend-based bridges rely on their own infrastructure rather than relying on external sources such as oracles. This type of system offers higher levels of flexibility and scalability along with easier integration but comes at the cost of increased vulnerability since it becomes an even more attractive target for attackers due to its centralization point within the system which must be trusted by users.

Oracle problem

The Oracle Problem is a common issue that arises when relying on Oracles to verify transactions. An Oracle is an entity that provides data from the blockchain to external sources such as validity of the exact transaction and its parameters, while also providing access to other critical information needed for various use cases within distributed ledger technologies (DLT). This trust in Oracles makes them vulnerable points where attackers could manipulate data being transmitted through them, creating potential risks for users depending on those Oracles accuracy or integrity when providing outside-world data inside the blockchain environment.

In order to address this problem, numerous solutions have been proposed which aim at mitigating these vulnerabilities. Chainlink's decentralized network of oracles based off consensus format seeks to provide a secure and reliable way of connecting smart contracts with external resources by using multiple nodes running their own versions of the same codebase in order to come up with an agreement about what should be considered valid input. While this method offers increased security compared traditional centralized solutions, it has its own drawbacks in terms of cost and speed due to having multiple copies running simultaneously.

Physical level integration attempts like Cosmos Hub offer another solution - exchanging information between different chains without needing any additional software layer but rather revising ways of representing data natively on each chain so they can understand each other's messages. Such methods make it easier for two blockchains connected via Hub protocol to communicate directly while still allowing user control over how much interaction they want between chains and how many layers are used during the communication process which helps reduce risk associated with using centralized Oracles since there no single point failure system subject abuse attacks etc.

Finally, local convenient solutions provided by Cosmos SDK powered Tendermint BFT module enable custom blockchains natively interoperate with other blockchains plus Stargate update facilitating interblockchain communication protocol solving minimal set problem functions yet still requiring certain degree trust in backend point failure system subject abuse attacks etc.

Backend problem

The backend of a blockchain system can present itself as a weak point due to its centralization and the necessity for trust in order for successful bridging between networks to occur. While having a backend does provide flexibility and scalability benefits, it also serves as a central point of failure which could become susceptible to attack if not properly secured against malicious actors attempting access into either network being connected together via blockchain bridging technology.

In order to reduce the impact of such weaknesses, various protocols have been developed over time that use cryptographic mechanisms such as zero-knowledge proofs (ZKPs) and multi-party computation (MPC). Examples include Atomic Cross Chain Swaps (ACCS), XCLAIM protocol, RenVM from Ren Protocol project, Chainlink providing decentralized network oracles consensus based format, etc., all aiming at providing secure asset transfers across different chains without having any third party involved in them. Additionally projects like Polkadot have proposed their own decentralized parachain bridging system using validators participating in consensus process based on Proof-of-Stake mechanism thus allowing assets transfer from one chain onto another securely while keeping privacy intact at all times during this process.

Another solution with potential is sidechains technologies like Blockstream's Liquid Network which allow transactions between two blockchains using atomic swaps thus allowing assets exchange across multiple blockchains while still keeping security guarantees similar ones offered by PoW algorithms eliminating centralization risk associated with backends solutions currently available in market today. This approach has already been implemented successfully within some exchanges around the world and involves establishing two separate chains where each chain contains only one half of an overall transaction; these two halves are then combined through atomic swap technology making it possible for users on both sides of the transaction to be able verify completion without relying on any third parties or intermediaries.

It has to be admitted that solutions with the necessity to rely on something centralized are not pure Web3 solutions; however there are certain measures which can be taken in order reduce this weak point including implementing stronger cryptography algorithms such as ZKPs and MPC into existing systems along with utilizing sidechain technologies like Blockstream's Liquid Network which enable more secure asset transfers across different networks while still preserving user privacy throughout the entire process . Furthermore , further development on how to provide even greater security guarantees than what current methods offer is required.

Step Towards Trustless Interoperability Solutions

Finally, trustless solutions are striving to create a truly secure environment where non-custodial platforms relying purely on smart contracts algorithms underlying blockchains provide high levels of decentralization by locking blockchain assets for added security. However, these emerging technologies typically sacrifice speed and incur higher fees in return for enhanced security compared to traditional custodian models. Nevertheless, this approach proven to be a step in the right direction.

To ensure the security and reliability of trusted bridge systems, it is important to have robust protocols in place that can detect any malicious activities or data manipulation attempts. Additionally, it is equally important to choose reliable sources for external data validation (oracles) so as not to create potential vulnerabilities within the system.

To ensure the security and reliability of trusted bridge systems, it is important to have robust protocols in place that can detect any malicious activities or data manipulation attempts. Additionally, reliable sources must be chosen for external data validation (oracles) so as not to create potential vulnerabilities within the system; this could include both centralized oracles such as Chainlink's network of Oracles which offers slow speeds but more secure solutions than other options available; however also decentralized networks such as Cosmos' native Cosmos Hub which enables inter-blockchain communication through consensus protocols rather than relying solely upon centralized Oracles alone. Another example towards achieving convenient cross-chain communications without having to sacrifice decentralization is Polkadot XCM's unified data definition standards enabling data exchange across multiple blockchains - although this requires revising how we currently represent our data in order for this solution to work effectively. Furthermore, with Cosmos' Stargate update developers are now able to create custom chains natively capable of interoperating with other blockchains within its own ecosystem while requiring minimal specific functions from each chain involved in bridging process - all powered by Tendermint BFT module that allows users access highly secure and more efficient transactions between different DLT ecosystems taking part into interconnecting bridge channels established amongst them respectively .

Physical Level Integration

Physical level integration is an important step in achieving interoperability between different blockchain networks and Polkadot XCM has made significant progress in this area. The Polkadot XCM allows for the exchange of data between various networks by revising and changing the way that data is represented. This process can be quite uncomfortable, but it's necessary to achieve interoperability. In order to do so, certain protocols need to be followed such as Hyperledger Fabric's Chaincode Protocol which defines how chaincodes are written, stored and executed on a distributed ledger system; Ethereum Virtual Machine (EVM) which is a runtime environment for smart contracts based on Ethereum; Interledger Protocol (ILP), an open protocol suite for sending payments across different ledgers using a routing algorithm called pathfinding; and ISO 20022 standard for messaging formats used in financial transactions.

Additionally, when implementing new protocols like Hyperledger Fabric's Chaincode Protocol one must carefully consider all aspects involved including writing chaincodes that are secure enough while still being efficient enough not to overburden the system itself - something especially challenging when dealing with large scale enterprises who require high throughput rates without sacrificing security levels. Similarly, EVM needs careful consideration since it provides the base layer upon which smart contracts are built - if mistakes are made at this stage then they will likely propagate throughout future iterations once deployed in production environments causing serious issues down the line that could cost valuable time & resources trying to resolve them correctly afterwards. Lastly ILP should also be taken into account since its pathfinding algorithms require extensive testing & optimization prior deployment so they work reliably under real life scenarios while avoiding potential pitfalls along their execution paths due lack of proper preparation beforehand.

Local convenient interoperability solution

Despite the ever-growing demand for a more efficient and user-friendly crosschain communication system, it was only recently that Cosmos delivered on this with its Stargate update which included the Inter-Blockchain Communication Protocol (IBC). IBC provides a solution to the cross-chain communication problem while requiring minimal functions such as consensus synchronization mechanisms like Tendermint Core; cryptographic primitives including Merkle trees, SHA256 hashes and Schnorr signatures; network layer protocols such as libp2p or gRPC; messaging queues like Kafka, RabbitMQ or NATS Streaming Server; databases like MongoDB, PostgresQL or LevelDB ; programming languages such as JavaScript , Go , Rust etc. This enables developers to create custom blockchains from scratch with native interoperation capabilities between different blockchains by leveraging existing industry standard technologies.

In addition to providing a much needed boost in terms of convenience and scalability, creating blockchain solutions with interoperation capability has become significantly simpler thanks to Cosmos SDK's Tendermint BFT module. The SDK allows developers to quickly customize their projects without having to worry about compatibility issues between different networks and data formats. Furthermore, several documents have been released by Cosmos outlining how these innovations can be best utilized in practice - including Whitepapers on Governance models for chain interoperability using Byzantine Fault Tolerance consensus algorithms, Security considerations when developing multi-blockchain applications using distributed ledger technology (DLT) systems - further enabling researchers and developers alike to apply their knowledge in real world scenarios.

Global Convenient Interoperable Solutions

So nowadays with the interoperability evolution we can see projects which aim to provide a global and convenient trustless and interoperable solution. This type of solution relies on algorithms, math, smart contracts running over underlying blockchain networks where assets remain locked until transaction completes successfully eliminating risk associated with single points failure common among traditional trusted/custodial bridge solutions. Whilst offering higher levels of security than custodial ones they come at increased fees and slower speeds making them less attractive compared to others available today.

Trustless (noncustodial) Bridges

The concept of trustless (noncustodial) bridges has been gaining traction in recent years due to the need for secure and decentralized solutions that enable users to transfer assets across multiple blockchain networks. Such platforms are built on sophisticated algorithms, cryptographic mechanisms, distributed systems engineering and smart contracts which together form a completely trustless environment.

Examples of such technologies include zero-knowledge proofs for user privacy, multi-signature authentication for asset custody services, sharding protocols like Polkadot Substrate Sharding protocol for scalability purposes or atomic swaps protocols such as Lightning Network Atomic Swaps Protocol which facilitate token exchange without counterparty risk. Additionally dual token models provide an efficient way to transfer assets across multiple chains simultaneously.

However, despite all these advantages there are certain drawbacks associated with trustless bridges including increased fees related to gas cost payments used by miners within the chain verifying transactions or slow speeds caused by high level of decentralization needed in order to achieve maximum security against malicious actors potentially attacking the system using weak points in its architecture.

To tackle this issue several projects have been developed lately aiming at providing global trustless solutions suitable for seamless interoperation among different chains while maintaining high levels of security and performance such as Zcash Zerocash proofing scheme or Thor Chain which is being developed as an interoperable decentralized liquidity network designed specifically for cross-chain assets swaps leveraging concepts like Byzantine Fault Tolerance consensus algorithms and advanced cryptography techniques.

Conclusion

Moving forward, blockchain projects must prioritize security over performance and ensure that fundamental security best practices are followed. This includes using decentralized networks with least privilege access, in order to minimize reliance on centralized oracles and backends. Additionally, monitoring systems should be put in place to detect any suspicious activity on their platforms.

Moreover, trustless bridges have become increasingly popular due to their ability to provide a more secure solution when compared with traditional custodial solutions while still allowing users the benefits of interoperability such as faster speed and lower fees. Finally, global convenient trustless solutions strive towards creating an environment where fully decentralized peer-to-peer networks can be used by anyone around the world for practical use cases.

The complexity of this problem is further compounded by the question of whether it is possible to receive all these important qualities without sacrificing performance levels;

however there are certain strategies which can help bridge this gap between security and performance requirements.

As followed there should be implemented a scalable L1 network utilizing Delegated Proof-of Stake (DPoS) consensus protocol which would increase overall network throughput whilst simultaneously ensuring transaction integrity through digital signatures from multiple validators within a given node set providing physical level integration capability across different blockchains - thus enabling reliable real-time communication between nodes located anywhere in the world with low latency rates and minimal risk associated with data breaches or other malicious activities targeting user accounts or transactions stored within those chains' underlying databases structures.

References

1. "What Are Blockchain Bridges And How Can We Classify Them?", by Arjun Chand, https://blog.li.fi/what-are-blockchain-bridges-and-how-can-we-classify-them-560dc6ec05fa

2. "What is blockchain interoperability: A beginner's guide", https://cointelegraph.com/blockchain-for-beginners/what-is-blockchain-interoperability-a-beginners-guide-to-cross-chain-technology

3. "Blockchain Interoperability – How Does It Work?", by Alex Hickey, https://lisk.com/blog/posts/blockchain-interoperability-how-does-it-work

4. Zero Knowledge Proofs (ZKP) https://en.wikipedia.org/wiki/Zero-knowledge_proof

5. Multi Party Computation (MPC) https://en.wikipedia.org/wiki/Secure_multi-party_computation

6. Michael Abd-El-Malek, Gregory R. Ganger, Garth R. Goodson, Michael K. Reiter, and Jay J. Wylie. Fault-Scalable Byzantine FaultTolerant Services. In SOSP, 2005

7. Admin Accounts and Multisigs, https://blog.openzeppelin.com/admin-accounts-and-multisigs/

8. Blockchain bridges, https://ethereum.org/en/bridges/

9. The Interoperability Trilemma, by Arjun Bhuptani

https://blog.connext.network/the-interoperability-trilemma-657c2cf69f17

10. Why Do Blockchains Need Oracles?

- Forbes https://www.forbes.com/sites/matthewmastando/article/what-are-blockchain-oracles/

11. Blockchain Bridges: How Do They Even Work?

. Available at: https://hackernoon.com/blockchain-bridges-how-do-they-even-work12. Y. Gurevich and I. Neeman. DKAL: Distributed-Knowledge Authorization Language . In CSF '08.

13. How Will Blockchain Bridges Be Beneficial To Centralized Business Models?

https://www.linkedin.com/pulse/how-blockchain-bridges-beneficial-centralized-models-caroline

14. Secure The Bridge: Cross-Chain Communication Done Right, https://medium.com/dragonfly-research/secure-the-bridge-cross-chain-communication-done-right-part-i-993f76ffed5d

15. ChainLink Research Whitepaper, https://research.chain.link/whitepaper-v2.pdf?_ga=2.94936159.2133617801.1674819366-1726614350.1674819366

16. Byzantine Fault Tolerance Explained, https://academy.binance.com/en/articles/byzantine-fault-tolerance-explained

17. Tendermint Explained — Bringing BFT-based PoS to the Public Blockchain Domain, https://medium.com/hackernoon/tendermint-explained-bringing-bft-based-pos-to-the-public-blockchain-domain-8e5b0d360fa3

18. Cosmos (2022), 'Cosmos Stargate Upgrade Overview, https://figment.network/resources/cosmos-stargate-upgrade-overview

19. Polkadot Wiki (2020), 'XCM', https://wiki.polkadot.network/docs/learn-xcm

20.Inter-Blockchain Communication Protocol (IBC): The TCP/IP Protocol of Blockchains, Bridges & Trends

https://johnniecosmos.medium.com/inter-blockchain-communication-protocol-ibc-the-tcp-ip-protocol-of-blockchains-bridges-trends-b38ed700c681

21. The Polkadot Parachain Host Implementers' Guide https://paritytech.github.io/polkadot/book/protocol-overview.html

22. Polkadot's Substrate 2.0 integrates oracles at a protocol level

https://cointelegraph.com/news/polkadot-s-substrate-2-0-integrates-oracles-at-a-protocol-level

23. XCLAIM protocol, https://eprint.iacr.org/2018/643.pdf

24. Atomic Cross Chain Swaps (ACCS), https://arxiv.org/pdf/1801.09515.pdf24. RenVM from Ren Protocol project https://github.com/renproject/ren/wiki/#renvm

25. Eli Ben-Sasson, Alessandro Chiesa, Daniel Genkin, Eran Tromer, and Madars Virza. SNARKs for C: Verifying Program Executions Succinctly and in Zero Knowledge, pages 90–108. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013.

26. Zerocash protocol, Whitepaper http://zerocash-project.org/paper

27. Connecting Blockchains: Instant Cross-Chain Transactions On Lightning, https://blog.lightning.engineering/announcement/2017/11/16/ln-swap.html

28. Dual-Token Economy/Model (Two-Token Economy), https://coinmarketcap.com/alexandria/glossary/dual-token-economy-model-two-token-economy

29. Thor Chain Project Overview https://thorchain.org/

PART II I Cross-chain Market Research

Cross-Chain Market Overview

The concept of blockchain interoperability has been gaining traction in the cryptosphere over recent years, with multiple approaches being proposed to achieve it. The two main visions are the multichain and cross-chain approach.

Multichain approach

In this vision, multiple blockchains exist simultaneously, isolated from each other but interacting via centralized hubs such as Polkadot. Depending on their requirements developers choose a blockchain that they believe will offer them the best flexibility and security for their projects. These blockchains can operate at different levels of decentralization and have different transaction speeds.

Cross-Chain approach

In the cross-chain world it doesn't matter for developers which chain to build a project on. Any Dapp can easily be used on other chains as well. In other words, different blockchains speak to each other, sharing data and information. Cross-chain bridges and such projects as Cosmos or LayerZero attempt to build fast and secure connections across different blockchains.

The challenge with cross-chain is that while it allows tokens to be bridged across chains, it still doesn't allow true data sharing at a blockchain level. Interconnected chains are a problem that a lot of extremely intelligent people within space are working toward solving. The Cosmos IBC is an example of an attempt to achieve this, but for now, it's restricted to the Cosmos ecosystem itself.

Potential threats

Two of the main issues related to are centralization and security. The Wormhole bridge hack (Jan 2022) is a glaring example of a centralized entity losing funds, and of how even a small slip in security can have huge repercussions. Another threat is the emergence of the custodial liquidity phenomenon. Some projects provide distributed custodial liquidity storage that is operated by a small group of individuals, leading to unwanted centralization and uneven distribution of control over the blockchain.

UniLayer obviously sticks to the cross-chain approach with the goal of reducing or even eliminating the aforementioned blockchain threats.

Cross-chain Product Category

By the middle of 2022 the opinion that the future of blockchain lies in cross-chain technology is increasingly dominant. Not surprisingly since dozens of projects have emerged recently. They offer unique cross-chain products or infrastructure technologies.

Let's divide cross-chain projects into the following categories:

- Bridge
- L0 network
- L1 network
- L2 network
- Composite Solutions

Bridges

A cross-chain bridge is a protocol that enables users to transfer tokens, utilize smart contracts, exchange data, and employ other features of interoperable chains. For example, the user could use a cross-chain bridge to transfer BTC to the Ethereum network and utilize smart contracts unavailable on Bitcoin. Depending on the needs, bridges use Oracles, Backend servers or Swarm of Nodes to improve transaction speed and security. Bridges do not store transaction data. They trust data stored in blockchains, though.

Among them: Wormhole, Hashport, Multichain.Xyz, Router protocol, DeBridge, Anyswap, etc.

L0 networks

A Layer 0 cross-chain network runs beneath the blockchain and connects seamlessly with all other interconnected chains, offering a more robust and advanced alternative for smart contracts. Layer-0 blockchain protocols provide the necessary tools for developers to create blockchains that allow for high cross-chain interoperability and facilitate faster communication between various chains (e.g., IBC Cosmos).

To mention some: LayerZero, Cosmos, Polkadot, Quant, Chainlink, AnalogOne, etc.

L1 networks

Layer 1 cross-chain projects refer to some base networks, such as Bitcoin or Ethereum, that can validate, finalize, and store transactions without the need for another network. Cross-chain data transfer takes place at the level of the interoperability protocol, and transactions are validated by the network of nodes based on the preferred mechanism of consensus.

Let's name some: Kardiachain, Fincor, Partisia, Nervos, Flare, etc.

L2 Networks

A Layer 2 cross-chain network operates on top of the blockchain and serves as a scaling solution for the underlying layer 1 network. These networks use off-chain methods to handle a large number of transactions, while still maintaining the security and immutability of the underlying blockchain. This allows for faster and cheaper transactions, without sacrificing the integrity of the network. Some examples of Layer 2 networks include: Lightning Network, Plasma, State Channels, and Matic Network.

Composite Solutions

These solutions are made up by combining multiple approaches together in order to address liquidity issues caused by lack of interoperability between blockchains. This often requires complex structures of composed pushers using BLS signature aggregation relays or Tendermint BFT adapted modules along with modified clients adapted according to particular requirements at hand.

In addition there have been hypotheses suggesting that it is possible to utilize custom designed zkSNARK circuits in order to improve interoperability even further as well as increase privacy when transacting across different ledgers/networks. Some examples could be Internet Computer infrastructure for Dapps (Canisters), PolyNetwork & many others offering similar services tailored specifically according customer needs & preferences via their own proprietary stack(s) comprised out variety technologies mentioned above among other innovations created specifically solve problems related interoperability within blockchain space & beyond it if applicable/necessary.

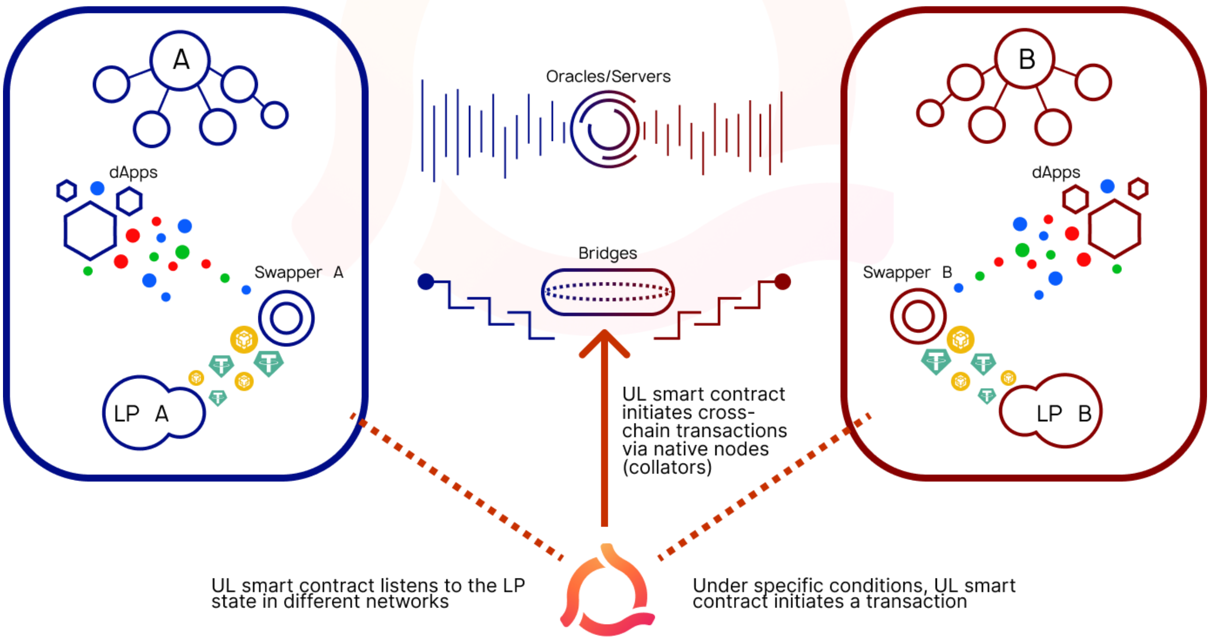
Conclusion

Cross-chain technology has become a driving force in the future of blockchain. The emergence of various cross-chain projects in different categories, such as bridges, L0 networks, L1 networks, and L2 networks, has opened up new possibilities for greater interoperability and communication between different blockchain networks. Each category offers its own unique benefits, from the ability to transfer tokens and utilize smart contracts on different networks, to faster and cheaper transactions. As the demand for cross-chain solutions continues to grow, we can expect to see further advancements and innovation in this field in the coming years.

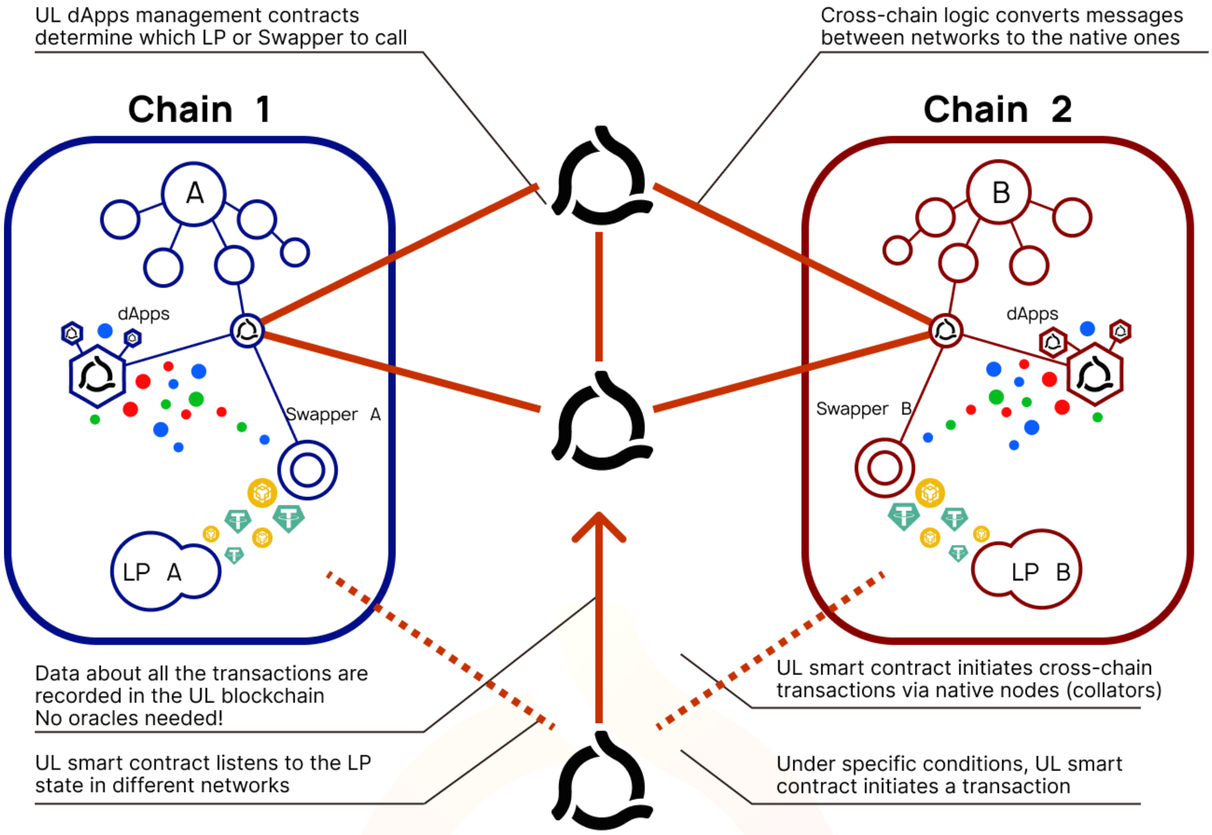
UniLayer offers a unique combination of the benefits of different cross-chain categories. With its advanced cross-chain technology and the ability to facilitate seamless communication and data transfer between different blockchain networks, UniLayer has the potential to revolutionize the way we interact with decentralized systems and create new opportunities for collaboration and innovation. The future of UniLayer is a promising one and it holds the potential to create a more decentralized, secure, and autonomous world.

UniLayer Approach

UniLayer belongs to the L1 network category and offers its product not only for applied tasks, such as an interop Dapp platform or a development SDK for native bridge on top of a specific blockchain ecosystem, but also creates a universal infrastructure for all participants in the entire blockchain market.



Example 1. UniLayer works alongside bridges or L0 solutions.



Example 2. UniLayer works as a single solution.

Competitors Report

This section is devoted to the comparison between the UniLayer network and some other projects that offer similar products and technologies on the market.

The Competitor List Criteria

Let's walk through the criteria that determine the shortlist of competitors for comparison with the UniLayer network product.

1. L1 network or a similar solution
2. Both EVM and non-EVM support
3. Interoperable Dapps development platform
4. Non-custodial liquidity
5. Related marketing offers and product vision

Shortlist of competitors

- Nervos
- Axelar
- Flare
- Fincor
- EYWA
- Nervos

Benefits of L1 and L2 in one solution

Nervos is a layered network solution designed for universal interoperable Dapps with the advantages of L1 and L2 networks. The layer 1 blockchain focuses on being secure, neutral, decentralized and open public infrastructure, while smaller, layer 2 networks can be purpose-built to best suit the context of their usage.

The layer 1 protocol is inspired by Bitcoin and is an open public proof of work-based (POW) blockchain. It serves as a decentralized custodian of value and crypto assets. Layer 2 protocols leverage the security of the layer 1 blockchain to provide unbounded scalability and minimal transaction fees.

Axelar

Blockchain platform for cross-chain Dapps development

Axelar is a developer-friendly cross-chain platform for interoperable application builders. It provides a solution to cross-chain communication needs and the API to access global liquidity and communicate with the entire ecosystem. Developers can go cross-chain without resorting to extra programming languages or added complexity.

The Axelar network consists of a decentralized L0 network which bridges blockchain ecosystems that speak different languages and a protocol suite with APIs on top, making it easy for applications to perform cross-chain requests. Network consensus comes from a Delegated Proof-of-Stake (DPoS) model.

Flare

Ecosystem of products for EVM Dapps scalability

Flare is a blockchain platform for building dApps that can access the value, information and liquidity of multiple chains with a single deployment on Flare. Flare can be referred to as a tricky combination of L1 blockchain, oracle, bridge and token wrapper technology.

This L1 network runs the Avalanche consensus protocol and can be used as a scaling method for leveraging the EVM capabilities. The oracle technology provides decentralized time series data to Dapps. Native cross-chain bridge allows smart-contracts to talk to different L1 chains. Wrapping technology can be used to get access to non-smart contract networks (e.g., Bitcoin).

Fincor

AI-based multi-chain platform for inter-chain transfers

Fincor is an L1 network with an AI-powered consensus mechanism that provides interoperability between different blockchains. Proof-of-unity (POU) consensus allows to connect any type of network to conduct both crypto and non-crypto activities on one platform. Cross-chain communications are direct on-chain account-to-account transfers with no need for any centralized authority to carry out the transactions.

An AI\machine learning network is used at multiple levels in the consensus to improve overall security of the entire blockchain. AI-based algorithms can also help developers to classify, predict and manage the blockchain state.

EYWA

The middle-layer system for cross-chain transactions

EYWA is an end-to-end system that allows different chains to interact with each other.

EYWA's cross-chain data and liquidity protocol are mainly designed for asset management in different blockchains. EYWA provides a basic solution for building DeFi cross-chain Dapps and protocols to get direct access to interchanged liquidity.

EYWA uses a combination of oracle networks based on a Proof-of-Authority (POA) consensus algorithm on top of IBC core technology. A gasless subsystem allows users to pay transaction fees in multiple assets.

	UniLayer	Fincor	Nervos	Flare	EYWA	Axelar
About	Cross-chain control Logix on L1 for any smart contract anywhere	AI driven Proof of Usage cross-chain consensus for all networks	Benefits of L1 and L2 in one solution	L1 cross-chain with EVM and Ripple consensus	The Middle Layer for cross-chain transactions	1-click building cross-chain Dapps
Consensus	uDPOS	AI POU	POW	mPOW	POA	dPOS
Layer 1	+	+	+	+	-	-
Liquidity Transfer	+	+	-	+	+	-
Cross-chain Control Logic	+	-	-	-	-	-
NFT transfer	+	+	-	-	+	-
Gasless	+	-	-	-	+	-
Cross-Platform	+	-	-	-	-	+
EVM and non-EVM	+	+	-	-	+	+

UniLayer Market Vision

The mission we undertake is to help create a unified decentralized landscape of blockchain ecosystems, while avoiding using forced solutions based on technologies foreign to the blockchain paradigm.

Previously we spoke briefly of the "three pillars of the Blockchain Trilemma", decentralization, scalability and security. The existing bridging solutions tend to target decentralization and, to an extent, scalability. Their safety, however, varies. Bitcoin, a blockchain-only decentralized system, has not been hacked a single time since its emergence back in 2009. Ethereum, which uses smart contracts, relies heavily on the competence of people writing these contracts, and there have been several notable cases of vulnerabilities created due to human error being exploited, sometimes with an enormous monetary loss to the blockchain's operator and their investors.

UniLayer strives to make its mission a reality by introducing a solution which does not rely on external technologies. Instead, native nodes of interacting blockchains are used as agents of the UniLayer system. This way, the liquidity can be moved seamlessly between blockchains, with no custom input required from the user, thus eliminating the possibility of human error or malicious actions.

The idea is to follow the original Satoshi Nakamoto's philosophy, utilizing lockhains only. Our implementation of said philosophy will help blockchains decentralize and improve their scalability without the impact on security unfortunately inherent to the existing solutions. The perceived result is a road playing field where blockchain developers can implement their ideas, while sharing the necessary common resources without the need to spend time and effort on avoidable supporting tasks.

The potential use cases, however, do not stop at helping in blockchain development. A tool such as UniLayer will allow customers to implement intricate investment and finance strategies without the risk of relying on third party transaction methods (which, unlike a developer, a user should not be expected to fully understand, introducing a possible additional security risk), while having a seamless transaction user experience.

Finally, UniLayer will help to use cryptocurrencies not only as an investment tool, but also as a convenient payment method. An introduction of an automated safe transaction protocol will allow vendors and service providers to accept virtually any coin, which would be converted into the cryptocurrency of their choice without the need to do anything manually once the system has been set up to use UniLayer.

PART III I UniLayer Structure & Ecosystem

Blockchain design and capabilities

UniLayer is an L1 protocol, and it doesn't use additional (overlying or underlying) networks. Instead of developing and being a 'classic bridge', UniLayer is creating a single data-transparent ecosystem for partner projects, and projects based on the UniLayer network. This unified system of data collection, storage and evaluation uses distributed ledger technology (DLT), such as blockchain technology to allow projects in the UniLayer ecosystem to maintain the highest transparency and objectivity with their data.

Unified integrated ecosystem based on UniLayer also allows us to greatly increase the security level for our partners within the ecosystem, because in fact, in order to hack a member of our ecosystem, one needs to hack the whole ecosystem, which is quite a different thing, you should admit.

The Unified integrated ecosystem based on UniLayer offers numerous advantages due to its decentralized nature which makes it exponentially more difficult than any centralized system available today for someone to hack into one project within this network since they would need to access all of them simultaneously.

Furthermore, this level of unified data that is transparently understandable by every project in the ecosystem opens up previously impossible opportunities for projects created on any kind of DLT platform or blockchain network - true interoperability between different systems that can exchange information incredibly smoothly and seamlessly without needing external bridges or other complex methods.

Most importantly, this approach allows us not just EVM based chains but non-EVM chains as well including those who using specific technologies like Zero Knowledge Proofs (ZKPs), Patricia Trees structures used for storing transactions securely inside blockchains; consensus algorithms like Practical Byzantine Fault Tolerance (PBFT) & Casper; sharding techniques commonly used by Ethereum 2nd layer solutions such as Plasma & ZK rollups along many other important aspects related directly or indirectly with ensuring high levels of trustlessness while maintaining scalability at all times when dealing with large amounts of transaction throughputs across multiple networks at once.

This level of unified data provides an improved replacement compared to classic bridges - allowing information to be exchanged smoothly and seamlessly without additional complexity - unlocking what was formerly thought impossible.

Furthermore, this level of unified data, transparent, understandable and accessible to every project in the ecosystem, is essentially an improved replacement of the classic bridges (although bridges can also be created on the basis of the UniLayer). It allows data to be exchanged incredibly smoothly and seamlessly. This is true interoperability, which opens up previously impossible opportunities for projects of all kinds created on any! networks.

In summary, UniLayer's integration capabilities enable developers from various backgrounds working on different DLT platforms or blockchain networks to collaborate together efficiently without sacrificing security or scalability requirements needed for real world applications – something never before seen in our industry.

uDPOS General Concept

The uDPoS (UniLayer Delegated Proof of Stake) algorithm is a consensus mechanism used to validate transactions in a L1 blockchain network. It is an improvement over traditional PoS (Proof of Stake) algorithms, as it combines the advantages of PoS and DPoS (Delegated Proof of Stake) mechanisms.

One of the key advantages of uDPoS is its ability to increase the speed and efficiency of the validation process. Unlike traditional PoS algorithms, where validation is determined by the number of coins a user holds, uDPoS uses a delegate system where users can vote for a select group of validators, known as "delegates," to validate transactions on their behalf. This reduces the number of validators needed to reach consensus, resulting in faster transaction processing times.

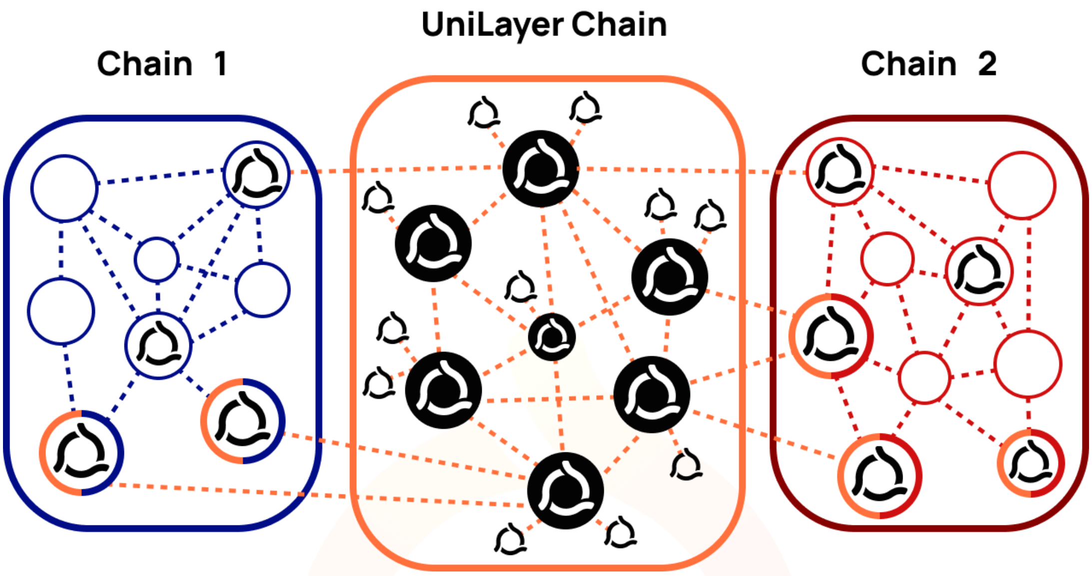
uDPoS also addresses some of the issues present in traditional DPoS mechanisms, such as the concentration of power among a small group of validators. In uDPoS, delegates are elected through a transparent and democratic process, ensuring a more decentralized and fair distribution of power.

Furthermore, uDPoS includes a unique feature called "delegated voting" in which each delegate has a certain number of votes proportional to the number of tokens held by their voters. This allows for a more accurate representation of the community's will and reduces the risk of centralization.

In summary, uDPoS is a consensus mechanism that offers a number of benefits over traditional PoS and DPoS algorithms. It increases the speed and efficiency of the validation process, promotes decentralization and fairness, and provides a more accurate representation of the community's will.

UniLayer Network types of nodes

UniLayer is a L1 blockchain using masternodes and a DPOS consensus protocol. A masternode validates the transactions and participates in administering the network. A native token, ULR, is used to grant masternodes rewards for each transaction.



On the picture above we can explore the architecture of UniLayer Cross-Chain Network.

Chain 1 and Cain 2 have their own nodes of the same color.

UniLayer Universal Nodes (embedded nodes) can be deployed on top of any existing blockchain infrastructure. They play both sides: original chain's nodes and UniLayer nodes. This type of nodes can play a role of collator, validator or MTN (see info below).

Solid black nodes here stand for UniLayer nodes. Though there could be a big difference between them.

The simplest Basic nodes are Wallets, Smart Contracts, L1 blockchain addresses that may contain both a wallet and a cross-chain smart contract

Basic Node: L1 blockchains contain addresses which may include wallets along with cross-chain smart contracts written using Solidity programming language which allow developers to create decentralized applications (dApps). In addition, existing address books stored on Ethereum Virtual Machine (EVM) are also able access this information provided by L1 Blockchains allowing users full control over funds without any third party involvement thus increasing privacy levels significantly during transfers between different blockchain networks .

Type of masternodes

Three types of masternodes exist in UniLayer:

1. Validator
2. Collator
3. MTN (Mixed Type Node)

Collators

Collators serve an important role as native nodes within partner chains that collect relevant information from these blockchains which can then be used by validator nodes when generating draft blocks.

Also collators create messages which can be utilized by the previously mentioned validator nodes. As such, they form an integral part of the overall system architecture, allowing for secure and efficient transactions between different blockchain networks.

1. They collect information in partner chains, generate draft blocks using said information and create proof that validators use.

2. Collators act as native nodes in partner networks. The data is therefore not transferred directly between chains but is coded in a format UniLayer understands and then decoded in a format native to the receiving network.

Collators monitor the status of these messages in their respective network to assure synchronization. Collators cannot modify these messages, however, for reasons of additional validation and compatibility across blockchains.

Validators

Validators verify collators messages; they also build consensus among themselves. The validators provide cross-chain transaction security. Validators add new blocks to the UniLayer blockchain, which make the data available for all nodes connected to the network.

Validators provide cross-chain transaction security. Deploying embedded UniLayer nodes on top of an existing infrastructure will allow providers to get rewards for validation.

These validators help generating draft blocks through signed messages which are then used by other validators for confirmation purposes.

Validators are specialized nodes within the UniLayer network responsible for creating UniLayer's blocks through signed messages. This is done by utilizing advanced cryptographic algorithms such as ECDSA, Schnorr signatures and Merkle trees in order to ensure the security of all data being transferred across different blockchain networks. Furthermore, validators also receive rewards from providers for their part in helping to secure the system while they remain active on it.

MTNs (Mixed Type Nodes)

Mixed Type Nodes combine both Validator and Collator functions into one entity - providing an additional layer of security due to their use of Delegated Proof of Stake (uDPoS) consensus algorithm on top of their transfer level activities. This helps ensure that all transaction processing remains secure throughout its lifespan while still being highly performant in terms of speed and throughput capacity when compared to traditional methods such as PoW or PoS algorithms.

PART IV I Core Technology

Physical-level chain intersection

UniLayer's Collateral nodes have the ability to inject a hot node from another blockchain into itself using Physical Level Chain Communication Protocol (PLCCP). PLCCP enables data flow from UniLayer directly onto other chains without involving RPC between nodes connected within different chains, thus decreasing computational load time and increasing transfer speeds significantly. This technology permits vanilla nodes from target chains to be used as program modules within UniLayer Node codebase; in order to resolve any dependencies between different codebases along with possible TCP port numbers conflicts that could arise due to multiple nodes running at once, additional measures are taken.

Specifically, by definition PLCCP there are some general alterations implied on target chain node codebase in order for it integrate properly: generic interface implementations must be provided for search operations within target chain databases; the construction of signed transactions is also necessary; reception mechanisms must be implemented in order receive new events from other blockchains - these represent only part of what this technology provides reference implementations for various components such as transaction signing schemes, state transition models and general data interfaces which further enhances its security profile against malicious actors trying exploit any vulnerabilities present within system architecture.

Moreover, PLCCP allows injection of an unlimited number of nodes from other chains but it is mandatory in order to utilize it - at least one node has to be injected.

Addressing

A UniLayer node acts as a full node (not a light node) of the partner chain therefore having access to data on all transactions in the chain.

Interaction with a specific node (or, more often) is done by passing its address in the form of a UniLayer smart contract.

A UniLayer node runs a service which deals with the connection specifics of partner chains. A synchronized node monitors the network for vault addresses. When an inbound connection is detected, its data is converted into a UniLayer transaction.

Thus, a transaction has an ID and addresses which are essentially the same for every chain.

UniLayer proceeds to process the transaction and waits for consensus. Once the majority of Validators verify the transaction, it is permanently recorded.

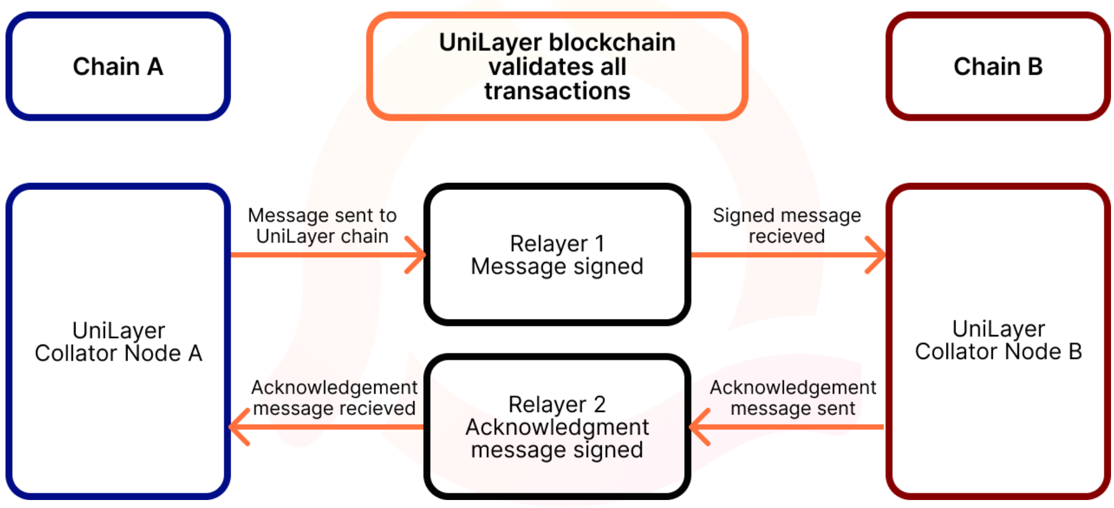
Onion-like transactions

Cross-chain transactions are stored in an onion-like construct, in which real addresses and their statuses are concealed in a meta-data hash.

The hashed data is then sequentially validated via the UniLayer DPOS consensus and stored in the UniLayer blockchain.

UniLayer smart contracts pick up data from the resulting blocks and send them to collator nodes in receiving networks, which decode them in the native format of these networks.

Cross-Chain Transport Protocol



The UniLayer CTCP acts as a universal bus, allowing to transfer data between partner chains

One of today's major challenges is that blockchains are inherently isolated systems. The UniLayer CTCP-technology allows us to safely overcome this limitation. This opens new alluring possibilities:

1. Seamless liquidity flows across blockchains.
2. Moving digital property objects (NFTs) across blockchains.
3. New options for cross-chain apps development.

The UniLayer CTCP uses an L1 blockchain, validating cross-chain messages via decentralized networks and storing all blocks in a DLT, to transfer data across blockchains. The UniLayer CTCP consists of two parts:

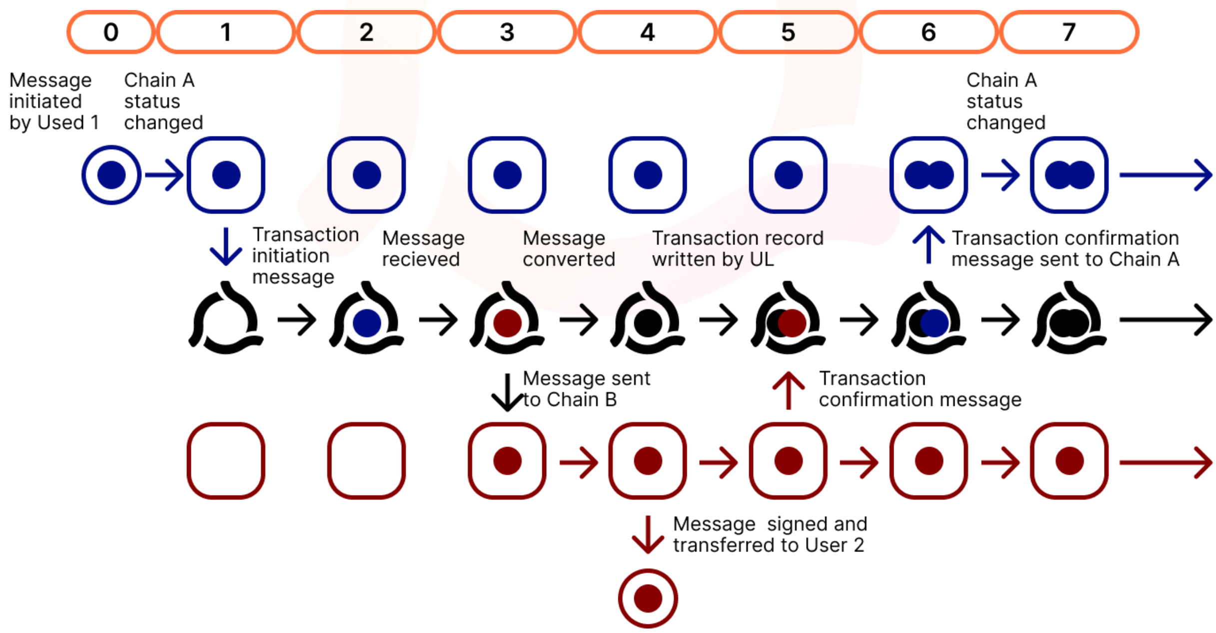
- L1 blockchain.
- Administrative cross-chain logic smart contracts.

The distributed ledger L1 blockchain processing cross-chain calls consists of Collators and Validators. Each cross-chain message is signed by a unique collective signature of validator nodes. The cross-chain call is recorded in the UniLayer L1 blockchain.

Each node has a ULR deposit in a specialized smart contract, which serves to prevent malicious activity. Beneficial actions are rewarded with ULR tokens. Nodes which undertake malicious actions are fined.

UniLayer cross-chain message validation is accessible by everybody due to UniLayer's infinite scalability.

Step-by-step CTCP transaction



Step 0. Data is composed into a message by the user or a smart contract and sent to a special smart contract in Blockchain A.

Step 1. Blockchain A (source) validates the transaction.

Step 2. A Collator node smart contract compresses the data and informs UniLayer Validator nodes of the new cross-chain call.

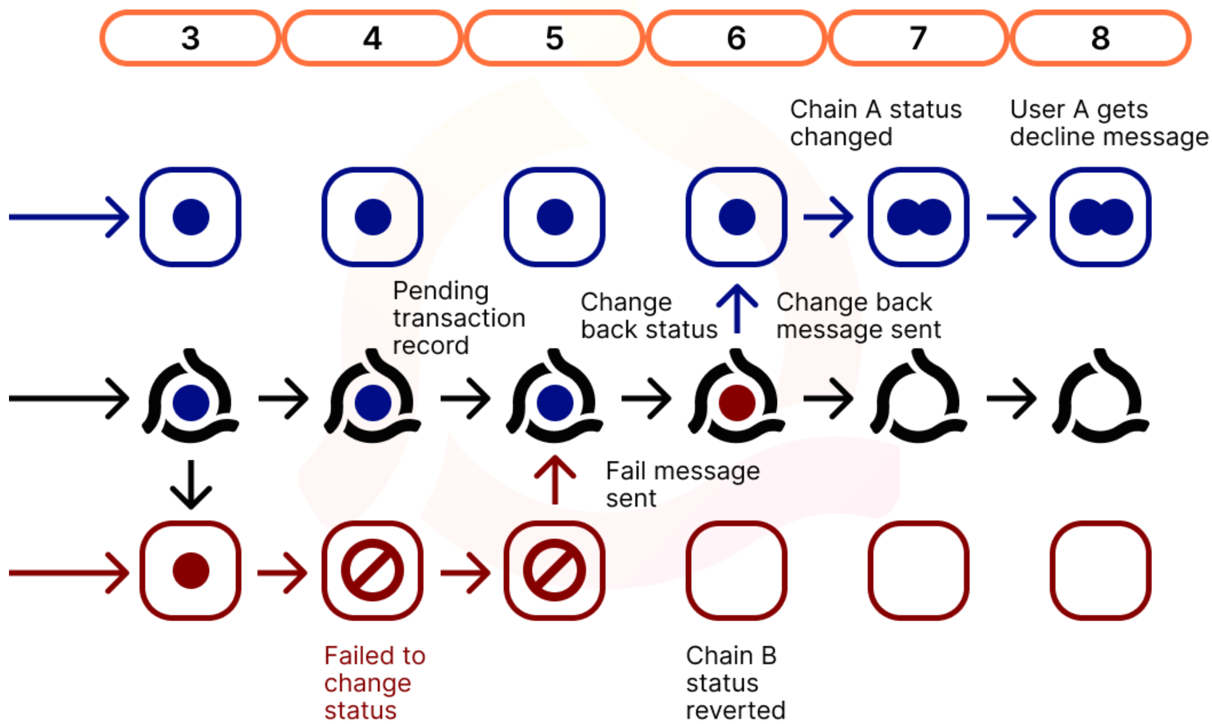
Step 3. UniLayer Validators sign the received data and store it in a new UniLayer internal block.

Step 4. Administrative smart contracts forward the signed message to a Collator in the destination network.

Step 5. A destination network Collator node smart contract verifies the signature of Validator nodes and converts the message into the native chain B format.

Step 6. If validation is successful, the Collator forwards the data to a destination smart contract in chain B.

Revert transaction



Blockchain interaction encounters stuck transactions quite frequently. This is especially true for cross-chain transactions. In case of a faulty transaction, every following transaction needs to be reverted to prevent loss of liquidity.

To make this easier, transactions are kept in the pending status until the problematic transaction is confirmed as failed by the destination chain. Reasons for a failure include unavailability or malfunctions in the destination node, gas expiration timeout, non-compliance with smart contract conditions etc.

Users can also create custom transaction reversion rules via a smart contract.

CTCP gas consumption

Obviously, there is a loss of liquidity for the gas payments. So, for the transfer from Address in Chain A to Address in Chain B would be

$$Res_2 = X_1 - (Gas_1 + 2 * GasUni + Gas_2)$$

Where:

- Res_2 - is a result liquidity in Chain 2 (receiver),
- X_1 - is a liquidity in Chain 1 (sender), which should be transferred
- Gas_1 - is a gas spent in Chain 1 to provide transaction
- GasUni - is a gas for the transfer inside UniLayer Chain
- Gas_2 - is a gas spent for the transaction in Chain

Example 1	**Example 2**	**Example 3**
Chain 1 sending the liquidity to a Pool.	In order for a transaction not to stop there should be x2 UniLayer Gas consumption.	Liquidity will transfer to Chain 2 from the Pool
$X_1 \rightarrow Pool_{12} + Gas_1$	$Pool_{12} \rightarrow Pool_{12} + GasUni*2$	$Pool_{12} \rightarrow Res_2 + Gas_2$

PART V I UniLayer Subsystems

Cross-chain Control Logic (LogiX)

Cross-chain control logic (LogiX) is one of the core subsystems in UniLayer. It allows smart-contracts in different blockchains to work in coherence under control of smart-contract deployed on UniLayer blockchain.

Most smart contracts are limited to a single instance ecosystem. Some cross-chain solutions can be combined with other networks at transport logic. For example, a smart contract on the Ethereum network can call and receive data from a smart contract on the nearby network. This could be done using Oracles or backend solutions. This approach assumes that the control logic of applications can be distributed across smart contracts in different networks.

There are three essential issues in this approach:
1. Use of oracles or other server solutions.
2. Full interaction with smart contracts in other networks is impossible without extending add-ons on top of called smart contracts.
3. The cross-chain history is discovered after the submission or is discovered on a focused application.

In our case any smart contract on UL becomes a central hub of information within the links of other blockchain. The LogiX smart contract keeps an overview of all logistics flows and information from this central point. Think of NFT, data, liquidity, or transaction state.

The aim of LogiX is to integrate all activities, systems and processes in a logical continuous flow based on the customer journey map. Through this integration, Dapp owners can set up collaborations and integrate with any certain chain, to ultimately provide customers with better user experience.

For example, a smart-contract with logiX can connect multiple LPs in different blockchains with each other, allowing carriers such as bridges, DeFi projects, NFT markets, and even other cross-chain platforms to develop a logistics strategy together. This logic depends on the joint objective that is set. When the goal has been set and the strategic process has been completed, the general coordination is transferred to a LogiX smart contract on the UniLayer blockchain. Then this smart contract takes care of the execution of the logistics activities. This approach bundles all available information to provide insight into multiple smart contracts on different chains.

Non-Fungible Vouchers

Since the emergence of Bitcoin, blockchain technologies have extended their reach beyond cryptocurrencies proper. One such development is the Non-Fungible Token or NFT.

Unlike a cryptocurrency token, an NFT holds no intrinsic value of its own. Instead, it signifies the ownership of an external valuable object, be it virtual (like in-game commodities) or physical.

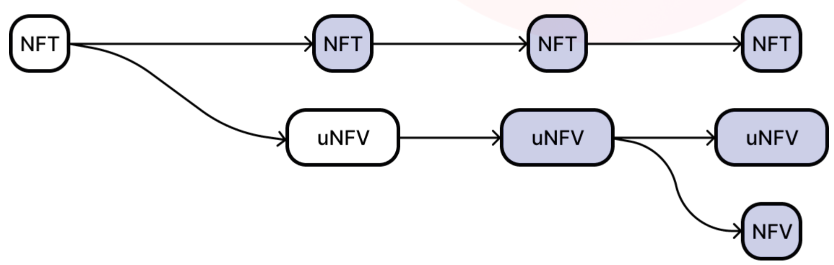
When it comes to cross-chain transactions, NFTs are associated with challenges unknown to cryptocurrency token owners, namely the issue of moving something non-fungible across virtual environments while preserving its uniqueness and non-fungibility.

UniLayer has developed an approach for transferring NFTs across blockchains. It is based on a new entity known as NFV (stands for Non-Fungible Voucher).

The UniLayer NFV functionality is comprised of its ability to:
1. Send an NFT to another chain as an NFV
2. Send the resulting NFV to a third chain as a replica of a replica
3. Send an NFT to another chain as an NFT, burning the original in the process (but retaining its full history).

Each variant suggests that only 1 instance of the NFT is accessible to anyone in any given moment.

NFT → NFV

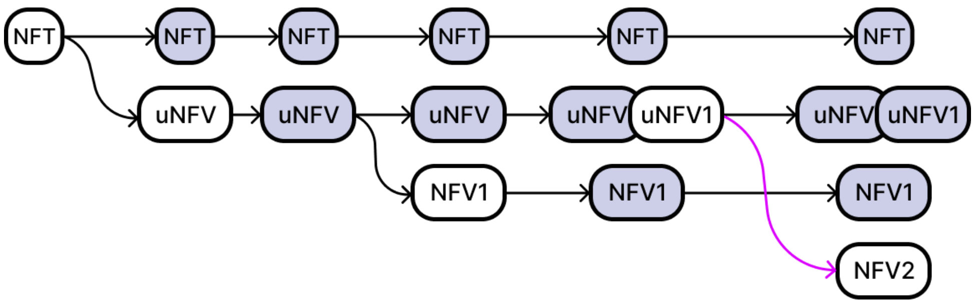


On the customer’s request, UniLayer creates an instance of a specified NFT, called uNFV. The original NFT is frozen in its chain of origin (chain 1), and can’t be deleted, altered of made another copy of! The only way to access the original NFT is to undo the transfer in its entirety step by step.

UniLayer then sends a request to the target chain (chain 2) to create its own instance. Once the request is fulfilled, the uNFV is frozen as well.

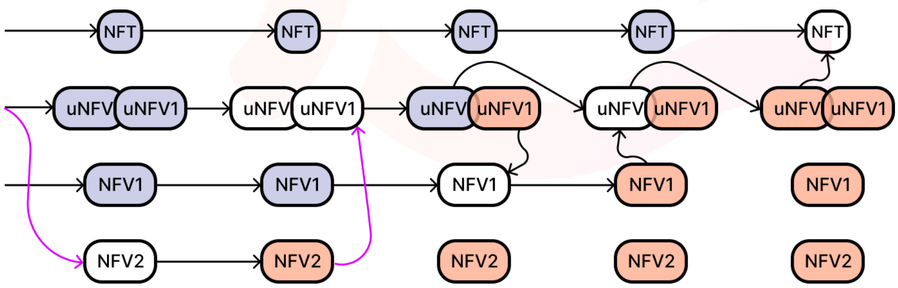
uNFVs store information on the original NFT transactions history, its content and a link to the frozen original.

NFV1 → NFV2

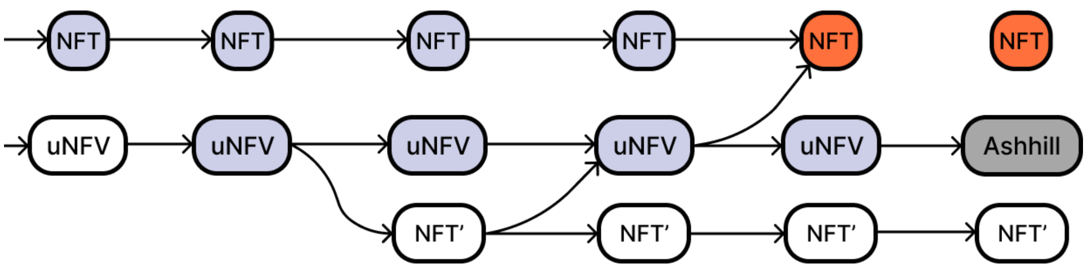


The mechanism remains the same. UniLayer labels the NFV to be transferred as NFV1, then treats it as if it were the original NFT, freezing it in its native chain (chain 2), creating an uNFV of NFIV (uNFV1), initiating a replica creation in the third chain (chain 3) and freezing uNFV1 (so that UniLayer now has two frozen replicas of the initial NFT, but neither can be used in any way other than to revert the process exactly backwards, undoing every step in sequence).

Return to NFT with burndown of NFVs



NFT → NFT'



The methods leave the initial NFT in its chain of origin. Should something happen with the frozen NFT or the entire chain, this may prove problematic.

UniLayer allows moving NFTs to another chain completely. The same multi-step approach is utilized, with the difference that the initial NFT is not frozen, but destroyed, and the token created in the target network is not NFV, but a fully independent NFT. The uNFV created in the process is retained, but frozen irreversibly, and serves as a proof of the transaction having taken place, and its details.

Gasless transactions

Gasless mode allows the ecosystem users to perform cross-chain operations in comfort, paying the fee in its entirety within the initial process. This eliminates the need to pay fees for some commodities and allows one to choose the payment source network.

A user can currently pay a cross-network transaction fee with the following means:

- Using an own source network commodity.
- Using a token from the transaction body.
- Using native network tokens.

Why gasless?

A cross-chain transaction causes sequential code execution in several block chains. Since any interaction with a blockchain is paid for in gas, performing a cross-chain transaction requires paying fees in each of the networks involved.

Here is a cross-network transaction example:

You exchange Ethereum commodities for BSC chain commodities using a cross-chain pool located in Harmony One.

This exchange transaction involves three blockchains: Ethereum → UNI → BSC. Gas needs to be paid for sequentially in ETH, UNI and BNB for the transaction to happen. This means the user needs to sign three transactions in different blockchains. The Gasless infrastructure automates payments for cross-network transactions in different blockchains. Using Gasless, the user needs to pay for the transaction once in the source network, using one of the means available. The cross-chain transaction then proceeds without the need for further user input.

Deal payment means: the user may choose one of the following ways to pay for a cross-network transaction:

- Paying for a cross-chain transaction with a source chain native token.

The user will be offered the opportunity to pay for a cross-chain transaction with a source chain native commodity.

- Paying with a token from the transaction body.

The user will be offered the opportunity to pay for a cross-chain transaction with the token being sent. The fee will be included in the transaction body to be signed by the user.

- Paying for a cross-chain transaction with UNI tokens.

The user will be offered the chance to pay for a cross-chain transaction with UNI tokens. The fee will be included in the transaction body to be signed by the user.

Paying in UNI tokens is the cheapest transaction fees payment option which saves 10 to 20% on fees due to the pool supremacy.

Gasless is best described this way: you pay for using the Gasless infrastructure in the way most convenient for you, while Gasless guarantees the successful cross-chain transaction on your behalf.

Paying for a cross-chain transaction with a source chain native token.

The deal payment is organized in the following fashion:

- Sending a meta-transaction.

The user sends a signed meta-transaction. The meta-transaction describes a cross-chain transaction the user wishes to process. The meta-transaction does not require gas since it's a simple request not needing to be processed in any blockchain.

- Cross-chain transaction cost evaluation.

The system estimates the amount of gas required for the network to finalize the cross-chain transaction. The cost of the deal in the native network commodity and said cost validity time is estimated based on the information provided.

- Signing of the cross-chain transaction by the user.

The user is offered the chance to sign the cross-chain transaction, including the fee estimated in Step 2.

- Processing of the cross-network transaction by the Gasless infrastructure.

Once the user has signed the cross-network transaction, the fee is sent to a smart contract within the Gasless infrastructure. The Gasless infrastructure then processes the cross-network transaction on the user's behalf.

If the user fails to sign the transaction within the designated time, the Gasless infrastructure will not process the cross-chain transaction for the estimated price. In this case, the user will have to send the meta-transaction again for the current cost to be estimated.

PART VI I Opportunities & Benefits

Cross-chain dApps

DApps (Decentralized Applications) will be able to run across all blockchain networks, unlocking cross-chain liquidity and allowing for effortless multi-chain scaling. This means that developers can easily develop applications on multiple chains simultaneously, taking advantage of the unique features each chain offers while also having access to a larger user base. Cross-chain dApps are made possible by UniLayer's Omnichain Universal Infrastructure, which provides secure interoperability between different blockchains including Ethereum, EOS, Tron and more. The infrastructure can be used in combination with Polkadot's Substrate framework as well as other technologies like Cosmos' IBC protocol and Interledger Protocols (ILP).

NFTs

NFTs (Non Fungible Tokens) will be freely transferable across all blockchains using UniLayer's platform. NFTs are digital assets with unique characteristics that make them highly desirable in gaming, art and collectibles markets due to their scarcity or uniqueness. With cross-chain compatibility enabled through UniLayer's platform, users will now be able to trade these tokens on any blockchain they choose without having to worry about incompatibilities between platforms or networks. Furthermore, this opens up the possibility of creating "cross-chain NFTs" which take advantage of the individual strengths of different networks in order to create new types of digital assets with enhanced functionality or even entirely novel use cases for existing ones.

DeFi

DeFi (decentralized finance) projects such as decentralized exchanges (DEXes), automated market makers (AMMs), lending protocols etc., will be able to access cross-chain liquidity via UniLayer's universal infrastructure making it easier for them to reach out a wider user base as well offering new options for users who may not have had access before due its lack of availability on certain chains or networks. Furthermore implementation costs associated with connecting these services across multiple chains becomes significantly lower with UniLayers technology thus providing potential cost savings over traditional methods such as atomic swaps etc.. Additionally since prices tend differ slightly between different blockchains because supply/demand dynamics vary depending on network size it creates an opportunity for arbitrage traders who can exploit price discrepancies in order capitalize upon profits from trading the same asset at differing prices across different platforms - something we expect will become increasingly popular amongst traders once more bridges are established between major chains via Unilayer's omnichain infrastructure

Metaverses & Gaming

Metaverses & gaming platforms benefit immensely from combining features separate blockchains to accelerate development of a truly multi - chain multiverse . Imagine an MMO game where players seamlessly move around the virtual world engaging various activities along the way while benefiting improvements offered by distinct characteristics native respective current inhabit eg increased speed scalability storage capacity transparency privacy protection provided particular blockchain utilized while playing game instance... Such possibilities would only be achievable through utilization technology like Unilayler ' s Omnichain Universal Infrastructure serve basis interconnecting metaverse comprising many disparate elements into a cohesive whole.

Bridges

Developers have option build own trustless bridge solution leveraging high level security and transparency provided Unilayler' s Omnichain Universal Infrastructure further supplemented additional components ensure highest degree safety standards maintained throughout process bridging multiple chains together order enable seamless frictionless interaction end users wishing interact participate within environment created crossing boundaries existing ecosystems.

PART VII I Consensus mechanism

uDPOS Algorithm

The UniLayer philosophy implies implementing as many features, as possible, using blockchain-based solutions, for they are the most secure and reliable. Doing so requires generating new blocks for every few transactions, which is traditionally done via a consensus (the majority of the nodes agreeing on a new version of the chain after a change being valid, this version being recorded by all connected nodes and used as a base for subsequent operations). A well-known theoretical issue is the risk of a malicious entity gaining control over 51% or more of the consensus authority (whatever this is, depending on the consensus method, see below). Such an entity then would be able to replace the actual chain with whatever version they see fit. This is known as a 50%+1 attack.

The first consensus mechanism to be used was Proof-of-Work (PoW). Basically, nodes would solve a complex mathematical problem, and the first one to find the answer will be granted the right to issue a new block, after which a new problem is presented and the cycle repeats (the process is called mining). Widespread application of this method has led to tremendous overuse of energy and specific hardware optimal for this task.

While being the most hack-proof (overpowering the existing consensus would require computing power exceeding that spent on generating the chain throughout the time of its entire existence), PoW is also probably the least convenient, the main bottleneck being the time it takes to generate a single block (plus it increases with each new block progressively). Aside from this scalability issue, PoW is also vulnerable to excessive centralization by means of large entities being more capable financially of accumulating large stocks of mining hardware than the average user. PoW is therefore mostly used in applications where operation speed is less of an issue, like baseline coins generation via mining.

The most common alternative to PoW is Proof-of-Stake (PoS). Instead of mining, nodes stake the chain's tokens (locking them up in a purpose-written smart contract), and the chances of a node to issue the next block are calculated proportionally to the amount of coins staked, while blocks themselves are generated on schedule. This solves the scalability issue inherent to PoW, but aggravates potential centralization even further (if tokens are available for purchase, a single entity can easily buy out an entire chain). It is also marginally less secure compared to PoW, although an attack is still going to be very expensive, with a loss being prohibitively costly.

On the other hand, scheduled issue of blocks allows for a much higher (and predictable) operations frequency. PoS systems protect themselves from malicious intent by "burning" the staked coins belonging to nodes which express undesirable behavior (like validating blocks containing data conflicting with the chain history).

In 2014 Daniel Larimer proposed an alternative to both PoW and PoS mechanisms, called Delegated Proof-of-Stake (DPoS). In DPoS, consensus is based on nodes called delegates or witnesses. Delegates are elected by other nodes, which provide the coins for staking, the voting power being dependent on the amount of coins staked. Unlike PoS, which can only punish stakers for obvious technical violations, in DPoS a delegate can be voted out of blocks validation by other users. Ideally, the more individual nodes there is in the network, the harder it is to centralize a DPoS system. DPoS also offers the highest operations rate among the three mechanisms.

While each of the mechanisms is best at certain applications, UniLayer uses uDPoS as a base. Its fast transactions rate is in line with UniLayer's seamless transaction philosophy, while the delegated staking principle serves as a means to protect the network and, by extension, all connected networks from an unwanted bulk buyout by an external player.

Proof of cross-chain consensus

We will prove the following property of the algorithm:

According to [1] and [4] :

If consensus exists in each chain `Pi`,

where `i, j` ∈ `{1, N} => {Pi} (1:N)`

for events `dist(i, -) = Ci` and `dtab(i, j, -) = Sb`,

then initialization:

Send all the values `dist(b, -)` to each of its neighbors.

`loop`:

Whenever a new value of `dist(p, c)` arrives from computer `Pb`:

(1) Set `dtab (b, p, c)` to this value;

(2) Recompute `dist(b, c)`;

(3) If the value of `dist(b, c)` has changed, then send the new value to every neighbor `Pb`, `there b=b+1`.

If communication lines stop failing and being repaired, then the computers eventually obtain and maintain the correct values of "`dist`" and "`dtab`" for all the missing subarrays.

The values of `dist` and `dtab` are correct if:

- `dist satisfies:`

```
dist(b,c)=
  {if b=c
      then 0
         else
         if ndist(b, c) < NN − 1
          then 1+ ndist(b, c)
       else ∞
 where: ndist(b, c) = min{dist(p, c) : c a neighbor of b
}
```

- and for all b and p:

```
if there is a working communication
      then dtab (b, p, -) = dist (p, -)
            else dtab (b, p, -) = 0.
```

[2], [3]

Wherever we meet line joining b and p the loop is:

- Send all its values

```
dist(b, -) to pi+1,
except for dist(b, b),
which pi+1 knows to be zero;
```

- Set `nbrs (b, p) to true`;
- Set all elements of

```
dtab(b, p, −) to ∞
except for dtab(b, p, p),
```

```
which is set to 0;
```

- Set `dist(b,p) to 1`

There is one last trick that we will use. Instead of setting the elements of

`dtab (b, p, −) to` ∞ when the line joining b and p is repaired, let them all, including `dtab (b, p, p)`, be set to 0 when the line fails. When the line is repaired, `dtab (b, p, p)` is set to 0 and the other elements are left unchanged.

This means that:

```
nbrs(b,p) equals true
      when dtab (b, p, p) = 0, and
it equals false
      when dtab (b, p, p) = ∞
```

[5]

To prove this property, we prove the following two properties, where the system is said to be stable if there are no unprocessed failure or repair notifications and no "`dist(b, c)`" messages in transit.

- C1. If the system is stable, then the values of dist and dtab are correct.
- C2. In the absence of failures and repairs, the system will eventually become stable.

These two properties are proved separately. [9]

Proof of property C1

Property C1 is a safety property - one which asserts that some predicate R is always true. (A predicate is a *boolean-valued function* of the system state.) Such a safety property is proved by finding a predicate Q with the following properties:

- `S1. Q is true initially.`
- `S2. Each action of the system leaves Q true.`
- `S3. R implies Q.`

A predicate satisfying this condition is called an invariant of the system. For a general discussion of this method, we refer the reader to [1] or [4].

In the proof of `C1,` R is the assertion that if the system is stable then the values of `dist` and `dtub` are correct. Recalling (1) and (2) of previous Section, it is easy to see that this is equivalent to:

```
R =
if the system is stable
then
      LPI: dist and dtab satisfy (2) and
      LP2: for all b and p :
      if there is a working communication
      then dtab (b, p, -) = dist (p, -)
      else dtub (b, p, −) = ∞.
```

Expression for each `Pi` is convenient because of Lamport's evidence of consensus in distributed systems. The expression for `Pi (i=i+1)` is convenient because the algorithm always recomputes the elements of DLT to keep condition `Pi true`.

To satisfy condition **S3** - the only condition that mentions R - the invariant Q; must imply that `Pi` and `Pi (i=i+1)` are satisfied when the system is stable. [6]

Proof of property C2

In order to formulate the predicate R2, we must state more precisely our assumptions about the order in which a computer receives messages and notifications of failure and repair. For each pair of computers b and p that are joined by a communication line, we assume that there are three lines of queues:

- `OQi (p, b)`: A queue of "`dist(p, -)`" messages in p's output buffer waiting to be sent over the line;
- `TQi (p, b)`: A queue of "`dist(p, -)`" messages currently being transmitted over the line;
- `IQi (p, b)`: A queue of "`dist(p, -)`" messages and failure and repair notifications for the line that b has received but not yet processed.

Note that there are two sets of unidirectional lines of queues for the bidirectional communication line joining `b and p`:

- `OQ(p, b), TQ(p, 6)` and `IQ(p, b)` for messages `from p to b`;
- `OQ(b, p), TQ(b, p)` and `IQ(b, p)` for messages `from b to p`. [9]

When the communication is working, messages move `from OQi(p, 6) to TQi(p, b) to IQi(p, 6)`. When the communication fails, a failure notification is placed at the `end of TQi(p, b)` and the messages in `TQi(p, b)` are thrown away.

Thereafter, messages from `OQi(p, b)` are thrown away instead of being moved into `TQi(p, b)`.

When the line is repaired, a repair notification is placed at the `end of IQi(p, b)` and normal message transmission is resumed. We let `TQi(p, b)` denote the `concatenation` of the queues `IQ{i= 1:N}(p, b)`, contains the entire line of queues of unprocessed messages that p has sent to b, together with b's unprocessed notifications of the failure and repair of the line joining b and p.

We now derive a predicate which expresses the assertion that `dtab (b, -, -)` will have the correct value when all outstanding messages and notifications have been processed.

We first define a predicate `Ui(b, p)` which states that `dtab(b, p,p)` will have the correct value when b has finished processing all repair and failure notifications in all the `IQi(b, p)`.

We next define a predicate `Wi(b, p, c)` for c # b, p, which essentially states that when b and p have finished processing their outstanding messages and notifications, then `dtab (b, p, c)` will wind up with the correct value - i.e.,

- with the current value of `dist(p, c)` if the communication line joining them is working,
- and with ∞ if it isn't.

We now define `UWij` to be the conjunction of the predicates `Ui (b, p)` and `Wi(b, p, c)` for all `b, p` and `c`

- with `b # p`
- and `c # b, p`.

The `UWij` is equal to invariant R then $i \in \{1, N\}$ and $j \in \{1, N\}$. To prove property **C2**, we must show that `r` satisfies conditions S1-3 from previous part. We can do it the same way as in previous part. [7]

Constants

Now we know `A(b, p)` – part of computers in superposition of distributed networks being at the same state on the same step.

In order to determine necessary and sufficient `A(b, p)` to reach consensus in strong eventual consistency we need to return to three lines of queues from previous part:

- `OQi (p, b)`: A queue of "`dist(p, -)`" messages in p's output buffer waiting to be sent over the line;
- `TQi (p, b)`: A queue of "`dist(p, -)`" messages currently being transmitted over the line;
- `IQi (p, b)`: A queue of "`dist(p, -)`" messages and failure and repair notifications for the line that b has received but not yet processed.

Now we can define predicate:

$$Y(b,p) = \{dtab\ (OQ_{min},\ TQ_{max},\ IQ_{max}) = S_i\ (A(b,p),\ A(b,\ p))\}$$

According to Lamport's evidence and because of two sets of unidirectional lines of queues for the bidirectional communication line joining b and p: OQ(p, b), TQ(p, 6) and IQ(p, b) for messages from p to b; and OQ(b, p), TQ(b, p) and IQ(b, p) for messages from b to p.

- `OQmin = 1 – 1/sqrt(2)` [8]
- `TQmax >=  1/sqrt(2)` [8]
- `IQmax >=  1/sqrt(2)` [8]

```
A(b,p) >=

= 1 - |(OQmax, TQmax, IQmax|*2 =

= 1 - (TQmax*IQmax/OQmin)*2 =

= 1 – ((sqrt(2)-1)/sqrt(2)) = 0,707213
```

Conclusion

We need **70%** of nodes on cross-chain network to be in the same state to reach cross-chain consensus.

References

[1] E.W. Dijkstra, A Discipline ofProgramming (Prentice-Hall, Englewood Cliffs, NJ, 1976).

[2] N. Francez, Distributed termination, ACM Trans. Prog. Lang. and Systems 2(l) (1980).

[3] C.A.R. Hoare, Communicating sequential processes, Comm. ACM tl(8) (1978).

[4] L. Lamport, Time, clocks and the ordering of events in a distributed system, Comm. ACM 21(7) (1978).

[5] L. Lamport, The 'Hoare Logic' of concurrent programs, Acta Znformat. 14(l) (1980).

[6] L. Lamport, Sometime' is sometimes 'not never': A tutorial on the temporal logic of programs, Proc. 7th Annual Symposium on Principles of Programming Languages, ACM SIGACT-SIGPLAN (1980).

[7] L. Lamport, Specifying concurrent program modules, ACM Trans. Prog. Lang. and Systems S(2) (1983).

[8] L. Lamport, Reasoning about nonatomic operations, Proc. 10th Annual Symposium on Principles of Programming Languages, ACM SIGACT-SIGPLAN (1983).

[9] S. Owicki and D. Gries, An axiomatic proof technique for parallel programs, Acta Informal. 6(4) (1976).

PART VIII I Governance

Holders, Leasers, and Stackers

Intro

The idea of profiting by staking coins has gained popularity in the blockchain community. Delegated PoS not only implements standard staking capabilities but also enables delegated staking (Leasing). Thus not only can users make a profit by freezing, removing a sizable volume of coins from the trading turnover for the benefit of the exchange rate, but it is also possible to further decentralize the network by determining the level of user confidence in Masternodes and Validators due to the ability to choose a target of coin-transfer for leasing purposes. This is very well visualized with the main nodes rating in the network.

Holders

Staking

Coins can be staked either by Masternode or by Validator. This is one of the main ways to prove the network's intentions. Since UniLayer is a two-tier network, both sides of the staking tier have an interest in keeping the network healthy. Masternodes contributing to the network are rewarded either for staking in-wallet or for storing their ULR as collateral. Both ways return a profit but in different ways and numbers.

Leasing

The UniLayer uses its leasing mechanism to issue new ULR coins and to increase the circulating supply of coins. Each user can lease their coins to Validators with the "Leasing" transaction type.

Interest on leasing is collected after 30 blocks. It is collected to the main balance and can be used immediately without the need to close the lease. Using the main wallet to collect interest allows multi-staking.

Masternode Rating System

The level of confidence in a network node depends on its rating. The rating is based on leasing volume, the number of leasers and Masternode network stability. The node's history, showing its growth and average confidence over time, is also taken into account.

Formulas

Rank of main node leasing value:

$$V_m = K_1 \times (\frac{CV}{TV})^2 \times \sqrt[4]{CV}$$

where:

Vm – main node leasing value

CV – current volume of leasing

TV – total volume of leasing

$K_1 = 0.25$ – rank of leasing users quantity coefficient

Rank of leasing users quantity:

$$U_m = K_2 \times (\frac{CQ}{TQ})^2 \times \sqrt[4]{CQ}$$

where:

CQ – current quantity of users in leasing

TQ – total quantity of users in leasing

$K_2 = 0.65$ – rank of leasing users quantity coefficient

Rank of successful blocks in the network value:

$$B_m = K_3 \times (\frac{SB}{TB})^2 \times \sqrt[4]{SB}$$

where:

SB – successful blocks of main node

TB – Total blocks of node in the network

$K_3 = 0.1$ – rank of successful blocks in the network coefficient

Total rank of main nodes:

$$R_m = V_m \times U_m \times B_m$$

where:

R_m – final indicator of main nodes ranks in the network

Token Flow (Reward System)

$Reward_i = K * (GasUni * 2) - C_R$

where:

1. $C_R = K_i * (GasUni * 2) * 0,2$
   - (20% of the value after the coefficient implementation)
2. $K_i \rightarrow min$

**Token flow formulas in the alfa-state. Will be updated soon*

General Conclusion

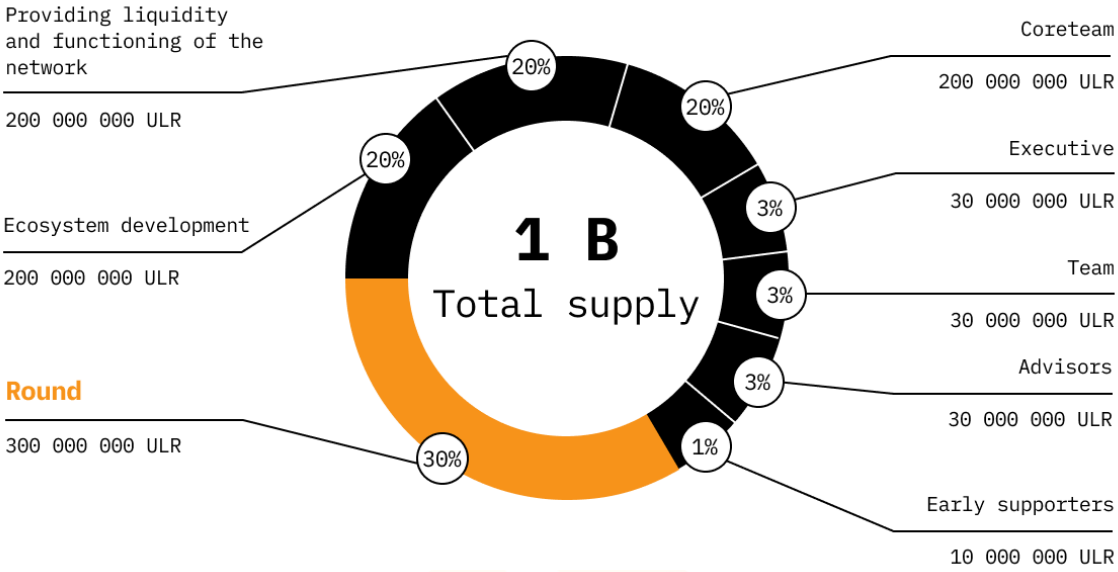
The future of trustless decentralized applications (DAPs) is upon us, and UniLayer technology is leading the charge. UniLayer's unique blockchain technology allows for true interoperability between different networks, breaking down barriers and enabling seamless communication and data transfer. This holds the potential to revolutionize the way we interact with decentralized systems and open up new opportunities for collaboration and innovation.

One particularly exciting application of UniLayer technology is in the realm of decentralized finance (DeFi). Trustless DAPs enable the creation of transparent, secure, and accessible financial services, disrupting traditional financial systems and empowering individuals to take control of their own financial futures.

But the potential of UniLayer is not limited to DeFi. It also has the power to transform industries such as supply chain management, voting systems, and gaming. With growing interest in blockchain technology, we can expect to see an increasing number of practical use cases for trustless DAPs in the near future. The future is here, and it's trustless, interoperable, and powered by UniLayer technology.

PART IX I Metadata

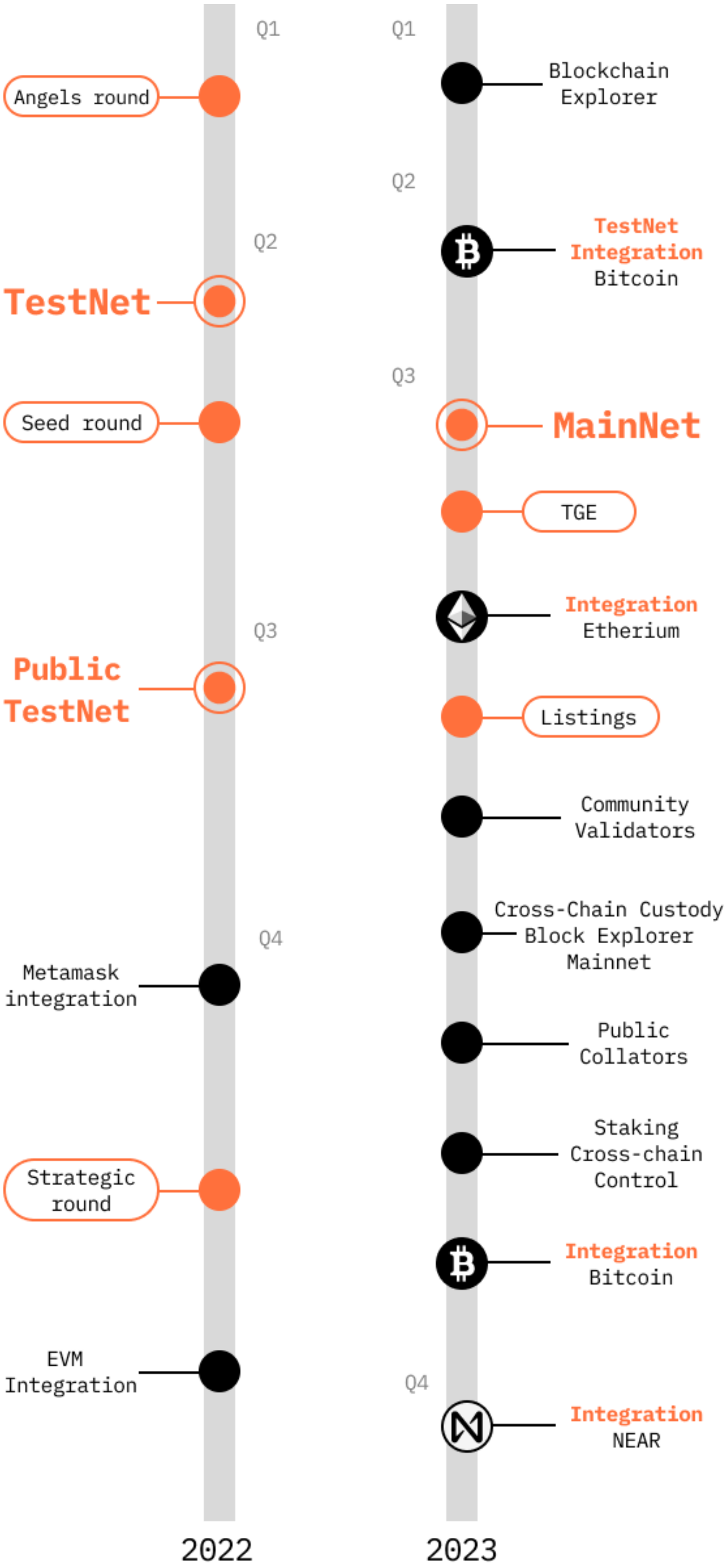
Tokenomics



Rounds and Vesting

Details for sale	Percent %	Token AMT	Price	TGE %	Vesting period	Goal	Evaluation
Angel round	3,3%	33 000 000	$0,015	3%	2 years	$0,5M	$15M
Seed round	4,7%	47 000 000	$0,015	3%	2 years	$0,7M	$15M
Strategic round	8%	80 000 000	$0,035	3%	2 years	$2,8M	$35M
Private round	12%	120 000 000	$0,050	3%	2 years	$6M	$50M
Private round for Influencers, Early Adopters, Node Owners	1%	10 000 000	$0,055	3%	2 years	$0,55M	$55M
Public round	1%	less than 10 000 000	$0,070	100%	-	$0,7M	$70M

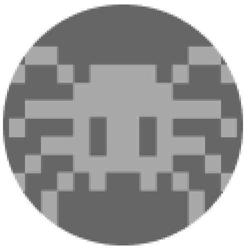
Roadmap


Q1
Angels round
Q2
TestNet
Seed round
Q3
Public TestNet
Q4
Metamask integration
Strategic round
EVM Integration
2022
Q1
Blockchain Explorer
Q2
TestNet Integration Bitcoin
Q3
MainNet
TGE
Integration Etherium
Listings
Community Validators
Cross-Chain Custody Block Explorer Mainnet
Public Collators
Staking Cross-chain Control
Integration Bitcoin
Q4
Integration NEAR
2023

Team

ALEX BELETS
CEO, Founder

Blockchain,ecosystem, cyber security.
Ex-SmartState founder, developer, and smart contracts engineer. Business developer. QA and tokenomics specialist. White hat hacker.



CORE ARCHITECT
CTO, Founder

Architecture, development, cyber security.
Enterprise cyber security expert with 10+ years experience. Blockchain lead architect. White-hat hacker.

SERGEY BELETS
Founder

Strategy, operations, ecosystem.
Tech entrepreneur with 20+ years experience. Ex-coo of 1 bln+ valuation hi-tech company Uniwagon. Hedera hash graph MVP developer.

ANDREY KURKIN
CBDO, Founder

Business development, cybersecurity, investor relationships.
Vc and IR expert in blockchain, GCC and Europe council at crypto market since 2014.

NIKITA ORLOV
Lead analyst

3 years in analysis for security and analytical intellectual systems. Experience in high load systems support for QA processes cross-database interactions.

SERGEY SAPEGIN
Core dev

Blockchain & cryptography.
High qualified AI development engineer and mathematician.
An inventor of original algorithms, data analysis methods, and architectural solutions.
Specialized in extracting, processing, and modeling data structures.

OLEG AZIZOV
Core dev

Network layers.
15 years as a programming and development mathematician.
Building on c and c++ since 80's QT engineer

MAX KOLTASHEV
Core dev

UX/UI.
9 years of backend development experience,
UI\UX for several fintech project, .net expert

MARK FRIEDLER
Advisor, Partnerships and BD

Silicon Valley based strategic sales and business development leader.
Bridgepointe technologies, senior enterprise technology consultant.
Dundon advisers LLC., senior adviser, blockchain/crypto SME
Crypto Mondays, San Francisco, founder.

ART MALKOV
Marketing advisor, Strategy growth

10+ years in digital marketing in emerging tech and blockchain space.
Zilliqa, ex-head of marketing, ex-Iotex head of growth.

NICK DINNERSTEIN
Senior Marketing Writer

Writer, editor, and content creator, with experience in traditional finance (LexION Capital), financial education (Money Masters), and numerous crypto and NFT projects across a number of blockchains. Additional duties as social media manager and social media advisor.