The scalable Layer 1 runtime for dApps on highly distributed networks

Ting Yang
ting.yang@rhonix.io

February 3, 2023
v0.1

Abstract

Rhonix is a layer 1 blockchain platform, designed from the ground up to support decentralization, on chain data storage, security, economics and the scaling needs of Web3. Further, our smart contracts can be formally verified in our programming language Rholang. Our language model allows us to have concurrent smart contracts, which makes composition possible. Based on the latest research from the reflective higher order calculus (Rho-calculus), our programming language solves a series of problems preventing blockchain platforms from realizing mainstream adoption.

Its unique conflict detection algorithm will be accompanied by the newly developed proof-of-stake consensus algorithm currently named 'Weaver', that is being developed and finalised by the Rhonix core dev team.

We are aiming to make concurrent state transitions, which should enable execution scalability. Our consensus algorithm is currently being developed and finalized by our core devs to meet this goal. We are trying to distance ourselves from the paradigm of total ordering of blocks because that makes execution sequential. Instead, we first make execution concurrent and then match the consensus algorithm to enable concurrent execution.

Once implemented, it will allow all nodes to produce and verify blocks concurrently, and by doing so achieve single-shard scalability. Our aim is also to minimise the amount of messages that are disseminating in the network. To achieve this, all consensus decisions are only computed on the local copy of our directed acyclic graph (DAG).

Further, large data can be stored directly on chain, removing its dependency on other data storage solutions such as IPFS. In addition, the smart contracting language Rholang contains SQL-like querying capabilities, which makes data manipulation logic a first-class citizen.

1 Introduction

The last 15 years of development of online services have all been about digital asset management platforms from large corporations. They help billions of people upload, disseminate, and manage zettabytes of data. While many of the online digital asset management platforms began with more open models, they are becoming less open every day.

The emergence of a decentralized global data network represents a major shift, yet the potential of the blockchain to completely upend the market doesn't stop there. Having developed the consensus algorithm known as proof-of-work, the Bitcoin network chose to use it to store a ledger recording the balances at the Bitcoin holder's addresses. This choice has obvious but limited utility. A more sophisticated choice for what to store with a consensus algorithm, is the state of the virtual machine. This choice, originally conceived and developed by Ethereum, turns the global data store into a global computer.

More generally, the market has been exploring decentralized alternatives [1], like blockchains, in building the kinds of online services that have sprung up and insinuate themselves into modern life, over the last few decades.

At the core of every blockchain is an economically-secured, leaderless consensus algorithm. Essentially, algorithms of this type, be they proof-of-work, proof-of-stake, or some other kind, allow computer programs, that do not trust one another, to come to agreement on a value. Since they agree on the value, they can store a local copy for easy access and only run the algorithm if there is a change to that value. Such a capacity, if it were scalable enough, makes it possible to deploy a global decentralized data network.

2 Current runtimes for dApps

As we know, current proof-of-work runtimes for dApps (decentralized applications) are inherently wasteful, spending inordinate compute cycles. Basically, the protocol uses extensive energy resources to secure the network.

With that, current proof-of-work consensus algorithms, like Bitcoin only process less than 11 transactions per second. Further, even current proof -of -stake consensus algorithms have problems with realizing their advertized

[1] https://en.wikipedia.org/wiki/Decentralization

transaction rates. Chains which have higher transaction numbers like Solana [1], rely on linear block production. This means that their virtual machines, are all sequential in their computation, and all transactions must be processed *sequentially* through the validators.

3 Rhonix Protocol

Rhonix brings together four major technological components in delivering its RNode software:

- RSpace, a new kind of key-value store
- Rholang, a new kind of programming language
- The newly developed consensus algorithm currently named 'Weaver'
- Kademlia communications and node discovery protocols

Each of these components and the innovations they realize is designed to meet specific market needs. In general, Rhonix would much rather follow state-of-the-art practice and engineering where it makes sense to do so. At the same time, Rhonix recognizes that if we are going to build a platform sound enough to rebuild the world's data and financial networks on it, it needs to be of a completely different kind of quality than we find in most Internet software.

4 Technology

The Rhonix techonology stack is built with multiple layers. The RSpace layer is the persistence layer for storage and execution of smart contracts. On top of this is Rholang, the by-design fully concurrent language for decentralized networks. With Rholang there are smart-contracts executed as additional layers, that implements the consensus protocol.

4.1 RSpace - A new kind of store

In the last decade the technical communities, especially those involved in big data, have seen a rethinking of storage and retrieval. In particular, a dialectic around the no-SQL [2] alternative to relational data stores, has been developed. First, a wave of storage systems based on the key-value, together with the map-reduce paradigm emerged. A backlash followed this, levying critiques of the key-value store paradigm in terms of the semantics of both query and transactions. RSpace threads the needle, offering a no-SQL store, but with a clear query semantics and a clear transactional semantics. It goes beyond this by offering a critical feature necessary to support user-controlled concurrency in queries; the ability to store code and data. In fact, putting code and data on equal footing in the storage layer actually derives from a consistency constraint coming from one of the oldest rules of logic, the law of the excluded middle.

Meredith devised a version of this for Microsoft's BizTalk process orchestration [1], a business process automation platform, then updated the idea to use so-called delimited continuation in SpecialK, and finally proposed the RSpace design as a further refinement, that allow Rhonix to scale.

Given the need to organize and handle the world's data in a decentralized way, we begin with a store that builds and improves upon what we have learned from the last decade of big data. Essentially, a component like this is sitting as a private asset inside all the major digital asset management platforms, from Google to Facebook. One key difference, however, is that our store is open source, and fits into a decentralized, public infrastructure, and its features and functions are derived from a specific concurrency semantics, embodied in Rholang.

4.2 Rholang - A new concurrent programming language

For designing a new language one has to look at the requirements needed for writing smart-contracts for distributed systems.

A careful analysis of the various models of computation, from Turing machines [2] to lambda calculus [3], from Petri nets [4] to the π-calculus [5], shows that there are four properties we are interested in relative to this market's requirements.

- Completeness - can we say everything we need to say?
- Compositionality - can we build more complex programs out of simpler ones?
- Concurrency - can we build programs that have parts that run simultaneously?
- Complexity - can we measure the cost of computational resources

Table 1 shows that the π-calculus, and more generally, the family of models of computation known as the mobile process calculi, is the only one that has all four features necessary. Analyzing this table makes it possible to compare also different technologies, that other decentralized networks are using.

[1]https://en.wikipedia.org/wiki/Microsoft_BizTalk_Server
[2]https://en.wikipedia.org/wiki/Turing_machine
[3]https://en.wikipedia.org/wiki/Lambda_calculus
[4]https://en.wikipedia.org/wiki/Petri_net
[5]https://shorturl.at/douvW

	Completeness	Compositionality	Concurrency	Complexity
Turing machines	✔	X	X	✔
lambda calculus	✔	✔	X	X
Petri nets	✔	X	✔	✔
CCS	✔	✔	✔	X
π-calculus	✔	✔	✔	✔

Table 1: Comparison of different models of computation

It is also worth mentioning, that it's not just the blockchain, that demands concurrency as the model of computation. The programming model for Internet-scale programs has been under considerable pressure to move to a concurrent model of computation for quite some time. For the last two decades, two trends have been putting pressure on the programming model from below, as well as from above. From below, we see that Moore's law ended in the early 2000's [1]. In the previous era, a developer could write some C code, sit on their hands for a year, and their code's performance would double because of advances in processor speed. This trend ended when limits to sequential processing speeds were hit in the early 2000. The predominant way for computation to speed up, has been to put more cores per die, more chips per box, more boxes per rack, more racks per data center. Code that doesn't take advantage of this concurrency doesn't scale. Likewise, from above, the commercialization of the Internet has resulted in user demand for programs that are globally accessible by millions of concurrent users.

This means that the programming model used for programming web applications, whether they are on the blockchain or not, must evolve to be concurrent to have sufficient use. The mobile process calculi forms the basis for that evolution, not only because they represent significant advances in language design, but because a sound basis for static analysis of programs. The importance of this feature is hard to state. Concurrent programming is many times harder than sequential programming. Without significant support from static program analysis, the bugs in concurrent code will become overwhelming as the numbers of programmers writing concurrent code increases.

4.2.1 Rho-calculus vs π-calculus

As mentioned, the π-calculus is just one example of a family of models enjoying all the features necessary to address this market. Since Turing Award winner Robin Milner put forward the model, several models of computation sharing many of the π-calculus features have been identified and studied, including the join calculus, the blue calculus, and the ambient calculus. Each of these has interesting properties, but ultimately fails in one way or another to map as well to programming the Internet as the π-calculus. There is one model, however, that derives from the π-calculus, but fixes a small lacuna and at the same time adds some powerful features that are quite common in programming the Internet, namely Meredith and Radestock's Rho-calculus [3].

The Rho-calculus plugs a hole in the π-calculus by making names first class elements of the model. The π-calculus is parametric in a theory of names; that is, given a theory of names, the π-calculus will produce a theory of processes that get computation done in terms of communications that use those names as channels. The π-calculus is agnostic as to whether names are telephone numbers, email addresses, blockchain addresses, or all of the above. However, the pure π-calculus can only exchange names between processes. It's like people getting work done entirely by exchanging phone numbers. It turns out it is possible, in theory, to do just that, but it's a lot of work. What happens on the Internet, however, is that not only data, but code gets shipped around from process to process, and the rho-calculus supports this feature. The Rholang language specification paper describes in detail the grammar and semantics of this language.

4.2.2 Namespaces and sharding

The Rho-calculus achieves this ability to ship processes by making names be the codes of processes. Once names are code, it is possible to encode all the different kinds of common telecommunications notions of addresses into the Rho-calculus setting, everything from email to blockchain addresses embed nicely, and thus it is straightforward to embed the most common addressing scheme on the Internet: URI's and URL's. This latter is important not just because it is the way the entire World Wide Web is organized, but also because it identifies a very powerful feature that the Rho-calculus refines: namespaces. URI's organize the web into a tree of resources (a forest actually, but trees will suffice for our discussion). Each URI is a path from the root of the tree along the branches to the leaf, or endpoint that holds the resource. Because of this path structure, you can indicate entire groups or spaces of resources using only partial paths. This allows us to organize and search spaces of resources in terms of the tree and path structure. The Rho-calculus kicks this paradigm up a notch by identifying these spaces programmatically.

The reason this feature is so critical in this market is, because most transactions are isolated. We need a programmatic way to segment, organize, and reorganize transactions

[1] `https://interestingengineering.com/no-more-transistors-the-end-of-moores-law`

so that they are grouped in terms of the resources they have in common. This is commonly called sharding in the blockchain space. The Rho-calculus namespace capability provides an extremely powerful approach to sharding. Specifically, the same approach can be used to rejoin forks, interoperate with other networks, as well as give speed-ups on transactions that are isolated.

4.2.3 Operational semantics and correct-by-construction language design

This discussion wouldn't be complete without mentioning the interplay between the correct-by-construction methodology and the type system used by Rholang. The Rho-calculus provides a Turing complete operational semantics of Rholang's core features. Each of Rholang's additional features are defined in terms of a mapping back to the core calculus. This means that the language is correct-by-construction. Contrasting it with Solidity [1] [2], Solidity doesn't yet have a formal semantics. When it does eventually achieve one, there will be a proof obligation that any compiler in production compiles Solidity to EVM byte codes in a way that preserves the semantics.

Rholang cannot suffer such an attack because the language and the execution mechanism are both derived directly from the Rho-calculus.

It is also worth noting that a clean operational semantics is necessary to identify when one program is substitutable for another, in other words, when is it safe to put a different program in place of another in a wider execution context. This notion goes all the way back to Liskov's substitutability principle, and in the practical terms of the maintenance of a system long term, this is a critical feature. You need to know when you can swap out upgrades or fixes for old or failing components. In software development, this is predicated upon having a semantics of the programming language, and operational semantics are by far the most widely used form of programming language semantics in theory and practice.

This principle of substitutability is closely connected to static analysis and especially the form of static analysis, commonly called type checking. All programs inhabiting a given type are substitutable in any context requiring that type. This is the basis of correct plug-n-play software and is used in all serious production settings. However, the type systems for most popular modern languages are relatively weak, only ensuring that data structure supplied matches the data structure expected. They say nothing about *program* structure, let alone program *behavior*. Over the last two decades, a new class of type-systems (behavioural-types) has emerged, but have yet to find their way into a mainstream programming language.

4.2.4 Object capabilities

Further, Rholang provides a very elegant solution to enable OCAPs. Object capability security model (OCAP) is a superior model to ACL (access control list). ACL is now used everywhere and in all the blockchains as well. It is very prone to hacks. That's why linux has SELinux, AppArmor and so on, which is an enhancement on ACL, but not the solution. Since Rholang has unforgeable names as first-class citizens, users can generate an unforgeable name and use it as a capability (you can think about this as a key) - pass this capability to anyone, restrict further sharing of this capability. So for a decentralized computer with lots of users and no trust, having OCAPs is a huge benefit.

4.3 A new kind of consensus algorithm

Most blockchains out there have a notion of a leader. Networks with proof-of-work do not have a leader because all miners are equal. This is why PoW is so resilient. But in PoS networks, many blockchains use pBFT like consensus, that elects a leader. So all kinds of problems infer from that: censorship, failure of safeness and DDoS is easier, because you can only need to attack one proposing validator. We are aiming for a leaderless consensus where all validators are equal. With 8 CPU cores, each core is doing computation on its own, but in the end all cores have to agree on what has been computed by the whole CPU, under Byzantine conditions (some cores can cheat). This is an abstract description of what our consensus algorithm is doing.

With a further improvement of the consensus algorithm currently named 'Weaver', one can follow logic's lead. In the late 80's and early 90's Girard's linear logic was revolutionizing our notion of what both logic and proof are. Specifically, linear logic is resource sensitive. Unlike either classical or intuitionistic logic, where the proposition "A & A" is the same as the proposition A, linear logic takes into account the resources necessary to establish a proposition. One might think of it in terms of establishing properties of chemical compounds. Many assays to establish that a compound has a given property, require modifying or even destroying a quantity of the compound. More prosaically, in classical logic, saying "I have a dollar" and "I have a dollar" isn't sensitive to the possibility that this could mean "I have two dollars." In point of fact, valid linear proofs are balanced in the sense that all resources must be carefully accounted for, and linear proofs prevent double-spend. Needless to say, it seems natural to look to linear logic for clues about consensus, especially economically secured consensus.

Among the different semantics for linear logic, games semantics stands out as both intuitive and pluripotent, with many variations illuminating a wide variety of features of linear logic. Hyland and Ong's games semantics provides an extremely faithful interpretation of the logic. In Hyland-Ong's games semantics [4] each move of player

[1] `https://en.wikipedia.org/wiki/Solidity`
[2] `https://medium.com/mycrypto/the-ethereum-virtual-machine-how-does-it-work-9abac2b7c9e`

and opponent must be justified by previous moves and it is this justification structure that their semantics uses to establish strategies, which are single threaded. In 2009, Meredith proposed using this kind of structure to secure network protocols in the communication between instances of an earlier version of RSpace. Following these insights, our proof-of-stake consensus algorithm imposes and exploits a justification structure on blocks to detect equivocation, provide liveness constraints, and fairness constraints. These properties are typically ensured by imposing certain patterns of communication, and the justification structure gives a view into just enough of the history of communication that these properties, or their violations can be detected.

Armed with a means to detect violations reliably, proof-of-stake can be economically secured. Participants, that put up stake in the token used to prevent denial of service attacks can have their stake partially depleted or entirely lost, if they commit a violation. In the long run, only participants that play by the rules are able to stay in the game. This is considerably more efficient than wasting cycles guessing numbers.

4.4 Validators

Validators play an important part of the ecosystem, due to the importance of securing and validating the network and its transactions. As we have seen with the rush to become a Bitcoin miner, the market is already seeing a surge for putting together staking services for proof-of-stake nodes. To become economically attractive to validators, the network needs high transaction volume due to low transaction cost. Yet, to be economically attractive to developers of dApps, the network needs a critical mass of validators. Likewise, dApps incentive to come to the Rhonix network will be because it's one of the best networks offering this transaction throughput and low-cost transactions, with a trust model similar to Bitcoin's that also addresses user-level smart contract security.

Since blockchains are all about consensus, we're aiming to rely heavily on consensus sharding so local shards are free to agree on values without the necessity to involve the whole network to provide safety since we know it's not scalable. In contrast to Layer 2 roll-up centric designs, our approach (with a concurrency built-in mathematical model) for execution promises much easier orchestration of smart contracting on our network.

Our aim is to provide users with the lowest transaction fees, and this is only possible if you can manage computation in the most effective way since it's always a balance between decentralisation and transaction costs.

Also, we're trying to get away from the scarcity viewpoint where everything is just a fight for blockspace and order. The access to creating a block space should be made easier and maximally possible by those wishing to make transaction fees lower.

One might ask how can you make the token valuable in this context? The token value is the value of the compute infrastructure, decentralisation overheads and transaction fees. Transaction volume is therefore the key component to a vibrant and successful network.

We're aiming to build a hierarchical organisation of shards so that in such a sharded setting, all shards have the possibility to compete for a higher level in the hierarchy.

Rhonix's goal is to establish the shard that will be at the root of the tree structure (the root shard is in fact the highest level in the hierarchy of shards) and to maintain this position with our first-mover advantage.

The higher the shard in the tree structure of shards, the larger transactional area the shard facilitates, so these transactions are permitted to be more costly. That is, the higher your shard is in the hierarchy, the more compute space you control. By analogy, you can think of the higher-level shards as computing bank-to-bank transfers and local shards as computing cash transfers.

But no one can promise that this will be forever because this is the nature of decentralised permissionless computing infrastructure. It promotes competition. This is why Rhonix needs to create the best place for the developers to come so they feel supported and incentivised to grow with our network, and if we do so, the Rhonix root shard will organically grow to be the most valuable and the most secure shard, providing guidance for the establishment of local shards.

5 Governance

If we are talking about a global, economically secured, decentralized computer and data storage network, there are critical questions about how the network is governed that are inescapable. In Rhonix, the simplest thing to do is to treat forks as shards. Sharding, as mentioned above, is an elegant mechanism for establishing the economic bridge between two or more networks. This ensures, that even when disputes around governance happens, the network stays open and scalable as before.

5.1 Jurisdiction

One absolutely critical reason to explore these more scalable governance structures is because blockchain technology cuts across jurisdictions. In the same way that the commercialization of the Internet brought about global commerce and global markets, the blockchain goes further. It is perfectly possible to write a smart contract that cryptographically locks up a digital asset in a manner that all parties to the contract agree is unfair or undesirable.

Geopolitically based jurisdictional boundaries are of no practical use in these situations. Communities sharing common interests and common goals, despite being geographically, culturally, and even politically diverse are more aligned with the basic organizational components of the blockchain. Blockchain will provide fundamental challenges to global jurisprudence. Dubai, with its openness towards decentralized technologies, is a great location for Rhonix, to grow in a fast-changing dynamic environment. Leading decentralized/-centralized exchanges and layer-1 platforms are moving there due to regulatory support for these kinds of emerging technologies.

5.2 Tokenomics

The Rhonix token token is the native utility token for the Rhonix network and is needed for paying executing of smart contracts by the validators. Additionally, the Rhonix token token secures the network by rewarding the validators automatically by a smart contract (proof-of-stake smart contract) if they behave honestly, otherwise they are slashed. The reward is dynamic adjusted by the proof-of-stake consensus algorithm, based on network participation. The reward is not predefined and there is no fixed guarantee of a reward.

There will be 10.000.000.000 tokens minted at genesis. The tokenomics paper by Rhonix Labs FZE references the amount of tokens released to participants in the token distribution. All other remaining tokens will be in the staking pool at genesis. One of the core values of Rhonix is that inflation will be very low at the beginning. The inflation rate at which validators will get rewarded additional to transactions costs, will be regulated by a smart contract. The inflation rate is dependent on the numbers of transaction blocks over a longer timeframe. Only by significant user growth of the network, the circulation supply will grow substantially.

6 Ecosystem

The following dApps show, how smart-contracts are currently written for running on the Rhonix network. They show the vast possibility for new ecosystem participants.

6.1 RCAT - An On-chain asset tracker

The community involved with Rhonix has helped build a next layer tool, Rhonix Asset Tracker (RCAT). RCAT allows developers to package large data blobs with metadata assets, such as audio or video files with information about the creators and rights holders of the audio or video data. The metadata is vital to provide search capabilities for data blobs that do not enjoy semantic search functions, as one finds in relational data.

Further the community has also run experiments building a dApp, called RSong, factored into the RCAT backend. With the Rsong player front end web app it allows artists to offer songs to their audiences. The audio data is stored on-chain and delivered also from there. The payments for audience access to the audio data go directly to the artist on the basis of use and the rights management data in the associated data. The 3rd party dApp, Proof, provided a crowd sourced verification of news stories, again with the data stored on-chain and delivered from chain.

Stepping back to consider the wider view, Rhonix offers a more solid and stable way to reinvigorate the developer interest in blockchain by creating a dApp ecosystem. Decentralizing digital asset platforms was the impulse that had developers exploring technologies like blockchain in the first place. Rhonix reminds us of that objective and shows that its vast potentiality can be realized today.

6.2 RDrive - On-Chain file storage

RDrive is a utility to mount a Rhonix shard as a filesystem. This means that all NFTs, domain names, tokens, etc. are viewable as regular files. This allows even a casual non-technical user to interact with the blockchain and perform basic web3 tasks such as uploading and transferring ownership of assets with a simple drag and drop. The ultimate use case for RDrive has been demonstrated on the server side, turning most off-the-shelf legacy applications into web3 enabled hybrid dApps to create viable alternatives to existing Big Tech for purchasing and streaming movies and music from chain.

6.3 Godot Engine SDK - Editor plugin

Gaming has become an important part of our lives. It's what we grow up with, how we spend time with friends and how we discover new ones. This is why every time a large AAA publisher introduces blockchain into their games the community rejects it, creating a massive backlash. Players simply see it as a predatory monetary scheme rather than improving their experience.

Our community at Rhonix believes that the gaming community is more than just simple NFTs, and we want the players to feel cared for and not exploited. This is why making SDKs for game engines with interaction to the Rhonix chain is of high value for dApps that want to use it for players. The community is planning to organize several game jams to let game developers discover new ways of using our technology within games.

This is the first SDK, that connects Godot to the blockchain, by realizing a plugin for the Godot Engine, one of the most popular free and open-source game engines. On top of that, we're looking to bring RDrive to Godot, giving game developers a way to store game assets directly on chain, which would allow them to publish assets on a marketplace, push new content and updates, or give a community ability to continuously update the game through a DAO which would increase its longevity.

6.4 Dappy - The non-DNS provider

Another interesting project in our ecosystem is Dappy, a no-DNS web browser that together with the use of the Rhonix network, provides the most secure web3 browsing experience. It allows anyone to register, renew and co-own domain names and publish websites without relying on centralized and fragile infrastructure. Dappy browser comes with a built-in wallet, uses best security practices and provides end users with the best possible web3 experience. It is being developed by Fabcotech and its code is open source [1].

6.5 Self-sovereign identity, data privacy, and decentralization

This last point has been a concern of both the market and the public sector alike. The market has responded to the dominance by a few centralized companies of the digital asset management platform capabilities by pursuing key alternatives: self-sovereign identity, data privacy, and decentralization. These pursuits have all converged on the technology known as blockchain.

6.5.1 Data privacy

The market has grown increasingly anxious over the hoarding of personal information by the large online companies, especially in the wake of the number of security breaches these companies have experienced in the past several years. Many terms have entered the common parlance, describing the growing public anxiety. for personal data vaults have come from many sectors, including the World Wide Web's inventor, Sir Tim Berners-Lee. Simultaneously, the public sector has responded with fairly strong regulations, such as the European Union's GDPR. Rhonix will provide data privacy and self-sovereignty by allowing people to have control over their own data, by creating their own smart-contracts.

7 Summary

Rhonix offers a more grounded and unique combination of technical innovation and economic opportunity than current decentralized networks. The mobile process calculi have simply dominated the fields of protocol design and protocol analysis for decades. Rhonix rho-calculus based language, Rholang offers developers a chance to come into the modern world where programming language semantics meets protocol design. By putting math first but making it available for everybody to use, we can bring scalable smart-contracts to the users.

8 Roadmap

We are in the process of setting up a new TestNet for usage for developers. The past TestNet 1.0 and MainNet 1.0, which were using the RNode runtime, showed strong stability but was only a serial block producer (single-parent). The new Testnet 2.0 will implement our new consensus algorithm and include our newly developed leaderless-block-merge, where we can merge multiple blocks into one. This gives us the concurrency for transactions, which is needed to lead a fast-growing ecosystem of dApps.

We at Rhonix are continuously improving the technology with new features and performance improvements. You can join the weekly tech updates and community debriefs on Thursdays on Discord.

References

[1] Anatoly Yakovenk. Solana: A new architecture for a high performance blockchain. `https://solana.com/solana-whitepaper.pdf`, Accessed: 2023-01-14.

[2] NoSQL. `https://en.wikipedia.org/wiki/NoSQL`, Accessed: 2023-01-14.

[3] L.G. Meredith and Matthias Radestock. A reflective higher-order calculus. *Electronic Notes in Theoretical Computer Science*, 141(5):49–67, 2005. Proceedings of the Workshop on the Foundations of Interactive Computation (FInCo 2005).

[4] J.M.E. Hyland and C.-H.L. Ong. On full abstraction for pcf: I, ii, and iii. *Information and Computation*, 163(2):285–408, 2000.

[1] `https://dappy.tech/`

THIS WHITEPAPER PROVIDES AN INITIAL SUMMARY OF CERTAIN TECHNICAL AND BUSINESS ESSENTIALS UNDERLYING THE RHONIX LABS FZE PROTOCOL. THIS DOCUMENT IS EXPECTED TO EVOLVE OVER TIME, AS THE PROJECT PROCEEDS. THE RHONIX LABS FZE TEAM MAY POST MODIFICATIONS, REVISIONS AND/OR UPDATED DRAFTS FROM TIME TO TIME, INCLUDING BEFORE, DURING, AND AFTER THE CREATION OF ANY TOKENS, AND WHILST NETWORK(S) BASED ON THE RHONIX LABS FZE PROTOCOL ('RHONIX LABS FZE NETWORKS') ARE IN OPERATION.

THIS DOCUMENT SETS FORTH A DESCRIPTION OF THE RHONIX LABS FZE PROTOCOL, REFERENCE SOFTWARE IMPLEMENTATION, AND POTENTIAL RHONIX LABS FZE NETWORKS. THIS INCLUDES DESCRIPTIONS OF THE PROTOCOL ITSELF AND THE USE OF TOKENS SUCH AS THE PROPOSED RHONIX LABS FZE TOKEN, RHONIX TOKEN. THE POTENTIAL RHONIX LABS FZE NETWORK IS A DECENTRALIZED, PUBLIC BLOCKCHAIN UPON WHICH PEER TO PEER TRANSACTIONS CAN BE CARRIED OUT BY USERS AND DEVELOPERS. DEVELOPERS HOLD AND CONSUME UNITS OF NETWORK CAPACITY ON THE PROPOSED RHONIX NETWORK TO BUILD AND MAINTAIN DISTRIBUTED APPLICATIONS AND USERS STAKE, USE AND TRANSFER UNITS OF NETWORK CAPACITY ON THE PROPOSED RHONIX NETWORK. THESE UNITS OF NETWORK CAPACITY ARE EXCLUSIVELY REPRESENTED BY CRYPTOGRAPHIC UTILITY TOKENS.

THIS DOCUMENT IS PROVIDED FOR INFORMATION PURPOSES ONLY AND IS NOT A BINDING LEGAL AGREEMENT. ANY SALE OR OTHER OFFERING OF RHONIX LABS FZE TOKENS WOULD BE GOVERNED BY SEPARATE TERMS & CONDITIONS. IN THE EVENT OF CONFLICT BETWEEN APPLICABLE TERMS & CONDITIONS AND THIS DOCUMENT, THE TERMS & CONDITIONS GOVERN. THIS WHITEPAPER IS NOT AN OFFERING DOCUMENT OR PROSPECTUS, AND IS NOT INTENDED TO PROVIDE THE BASIS OF ANY INVESTMENT DECISION OR CONTRACT.

Legal disclaimer

As of the date of publication, the Rhonix Labs FZE team have no plans to launch any public Rhonix Labs FZE Networks, and Rhonix Labs FZE Tokens are a proposed token with no known potential uses outside of Rhonix Labs FZE Networks, and no such use is intended. This document does not constitute advice nor a recommendation by the Rhonix Labs FZE team, its officers, directors, managers, employees, agents, advisers or consultants, or any other person to any recipient of this document on the merits of purchasing, otherwise acquiring, or holding Rhonix Labs FZE Tokens or any other cryptocurrency or token. The purchase and holding of cryptocurrencies and tokens carries substantial risks and may involve special risks that could lead to a loss of all or a substantial portion of any money invested. Do not purchase tokens unless you are prepared to lose the entire amount allocated to the purchase. the purchase.

Rhonix Labs FZE Tokens, if and when they are created and made available, should not be acquired for speculative or investment purposes with the expectation of making a profit or immediate re-sale. They should be acquired only if you fully understand the intended functionality of the Rhonix Labs FZE Tokens, and you intend to use the Rhonix Labs FZE Tokens for those purposes only, and it is legal for you to do so. No promises of future utility or performance or value are or will be made with respect to Rhonix Labs FZE Tokens, including no promise any Rhonix Labs FZE Networks will be launched, no promise of inherent value, no promise of any payments, and no guarantee that Rhonix Labs FZE Tokens will hold any particular value.

Rhonix Labs FZE Tokens are not designed and will not be structured or sold as securities. Rhonix Labs FZE Tokens will hold no rights and confer no interests in the equity of the Rhonix Labs FZE business or any future Rhonix Labs FZE Networks. Rhonix Labs FZE Tokens are designed and intended for future use on public Rhonix Labs FZE Networks that may be created using the Rhonix Labs FZE protocol, for the purposes of trading and governance transactions, or for the operation of a node. Proceeds of any sale of Rhonix Labs FZE Tokens may be spent freely by Rhonix Labs FZE for any purpose, including but not limited to the development of its business and underlying technological infrastructure, absent any conditions set out in this document.

This whitepaper is not a prospectus or disclosure document and is not an offer to sell, nor the solicitation of any offer to buy any investment or financial instrument or other product in any jurisdiction and should not be treated or relied upon as one. Any distribution of this whitepaper must be of the complete document including the cover page and this disclaimer and the accompanying boilerplate in their entirety.

All information in this document that is forward looking is speculative in nature and may change in response to numerous outside forces, including technological innovations, regulatory factors, and/or currency fluctuations, including but not limited to the market value of cryptocurrencies.

This whitepaper is for information purposes only and will be subject to change. The Rhonix Labs FZE team cannot guarantee the accuracy of the statements made or conclusions reached in this whitepaper. The Rhonix Labs FZE team does not make and expressly disclaims all representations and warranties (whether express or implied by statute or otherwise) whatsoever, including but not limited to: any representations or warranties relating to merchantability, fitness for a particular purpose, suitability, wage, title or noninfringement; that the contents of this document are accurate and free from any errors; and that such contents do not infringe any third party rights. The Rhonix Labs FZE business, Rhonix Labs FZE team, and operators of any Rhonix Labs FZE Networks shall have no liability for damages of any kind arising out of the use, reference to or reliance on the contents of this whitepaper, even if advised of the possibility of such damages arising.

This whitepaper includes references to third party data and industry publications. The Rhonix Labs FZE team believes that the information reproduced in this whitepaper is accurate and that the estimates and assumptions contained herein are reasonable. However, there are no assurances as to the accuracy or completeness of this data. The information from third party sources contained herein has been obtained from sources believed to be reliable; however, there are no assurances as to the accuracy or completeness of any included information. Although the data is believed to be reliable, the Rhonix Labs FZE team has not independently verified any of the information or data from third party sources referred to in this whitepaper or ascertained the underlying assumptions relied upon by such sources.

Please note that Rhonix Labs FZE is in the process of undertaking a legal and regulatory analysis of the functionality of the protocol, proposed Rhonix Labs FZE Tokens, and the operation of its business. Following the conclusion of this analysis, the Rhonix Labs FZE team may decide to amend the intended functionality of Rhonix Labs FZE Tokens in order to ensure compliance with any legal or regulatory requirements to which it is subject, which may affect the utility, fungibility, or any other properties of Rhonix Labs FZE Tokens.

Any Rhonix Labs FZE Tokens could be impacted by regulatory action, including potential restrictions on the ownership, use, or possession of such tokens. Regulators or other competent authorities may demand that the mechanics of the Rhonix Labs FZE Tokens be altered, entirely or in part. Rhonix Labs FZE may revise the Rhonix Labs FZE protocol or Rhonix Labs FZE Token mechanics to comply with regulatory requirements or other governmental or business obligations. Nevertheless, Rhonix Labs FZE believes it has taken all commercially reasonable steps to ensure that the design of Rhonix Labs FZE Tokens is proper and in compliance with currently considered regulations as far as reasonably possible.

No regulatory authority has examined or approved any of the information set out in this whitepaper. The publication, distribution or dissemination of this whitepaper does not imply compliance with applicable laws or regulatory requirements.