Nautilus Chain

First Blockchain for RWA Streaming DeFi

White Paper

[Internal Review]

Steve Chen
steve@zebec.io
Internal - Confidential

V0.2

May 2023

Nautilus Blockchain: Streaming DeFi for Real World Assets

Abstract

Nautilus is a high performance modular blockchain built for real world DeFi and Web3 applications. Nautilus offers exceedingly high performance and superior security through its adoption of sovereign optimistic rollup and highly secure and distributed data availability layer. Nautilus will have an initial TPS of 2,000, with much higher rates soon to come. Nautilus achieves such a high rate through parallelizing transactions instead of processing them linearly. Nautilus blockchain is also optimized for real world asset based DeFi applications with a continuous stream of revenues or payments. The vast majority of all commercial transactions involve future payments or revenues. Nautilus offers a viable and efficient blockchain based solution for streaming DeFi applications that are based on real world assets and scenarios. With its core innovation in performance and security, Nautilus opens up trillions of real world assets to blockchain and smart contract based digital financial applications.

1 The Problem and Opportunity

Real World Assets (“RWAs”) are assets that exist off-chain, but are tokenized and brought on-chain to be used as a source of yield within DeFi. The potential impact that RWAs could have on DeFi seems transformative. RWAs can offer yields to DeFi which are sustainable, reliable, and backed by traditional asset classes.

RWAs can render DeFi to become more compatible with external markets, resulting in greater liquidity, capital efficiency, and investment opportunities. RWAs allow DeFi the ability to bridge the gap between decentralized financial systems and traditional financial systems. RWAs can represent tangible assets, such as gold and real estate, as well as intangible assets, such as government bonds or carbon credits.

The main driving force behind bringing real world assets onto the blockchain is the belief that, in the long-term, DeFi will offer unique opportunities and market efficiencies to asset holders, which cannot be found in traditional financial systems. The ability to easily fractionalize and disperse RWAs in DeFi renders previously unfractionalized, total sum, private credit investments to become accessible to a new set of investors. Fixed income is the predominant market in the RWA space.

Furthermore, RWAs allow DeFi the ability to bridge the gap between decentralized financial systems and traditional financial systems. This means DeFi can begin to address the sea of liquidity, opportunities, and value which exists outside the digital

asset space (~US$1T in digital assets vs. >US$600T in asset value in traditional financial systems).

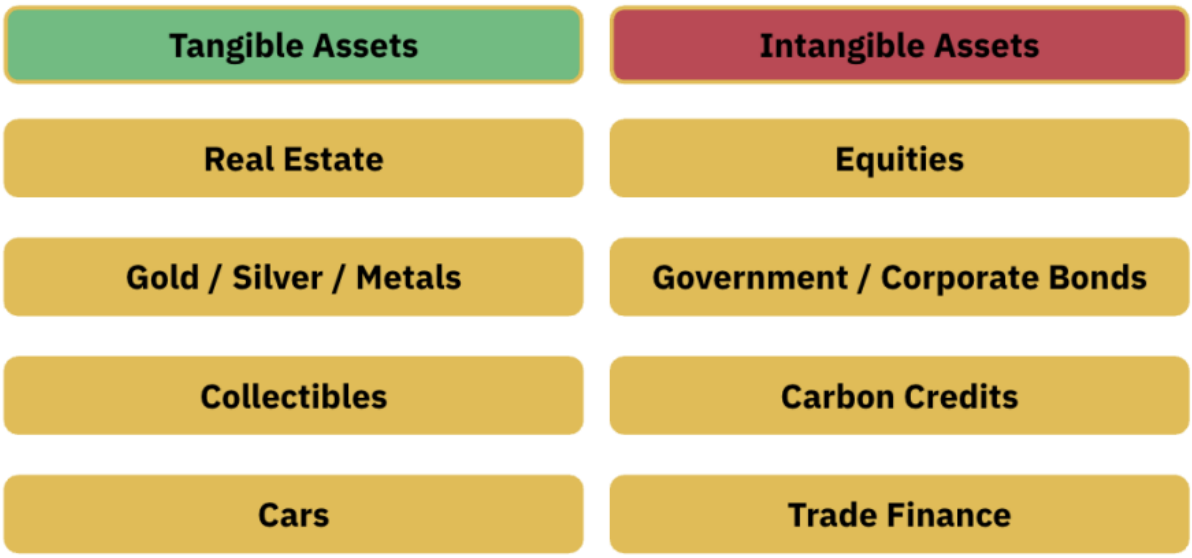


Figure 1: Categories of Real World Assets [Placeholder image, to be replaced]

Limitations. Over the course of history, traditional financial systems ("TradFi") have relied on intermediation systems, consisting of middle-men, background checks, and regulations. While these intermediation systems have helped instill a certain level of security and control, they have been at the expense of optimal market efficiency and opportunities for asset holders.

Market efficiency and opportunities dwindle when market participants are unwilling to pay fees to rent-seeking intermediaries, are denied access to the market by a centralized regulator, or are uncomfortable transacting in a system in which their assets are controlled by a third-party. The main driving force behind bringing real world assets onto the blockchain is the belief that, in the long-term, DeFi will offer unique opportunities and market efficiencies to asset holders, which cannot be found in traditional financial systems.

In particular, RWA based DeFi applications benefit from the following properties:

1. **Atomic settlement**: The combination of cryptography and decentralized consensus leads to strong finality guarantees of economic transactions—mitigating double-spend attacks and fraud in a tamper-resistant manner, thereby increasing capital efficiency and reducing systemic risks.

2. **Transparency**: Public block explorers and data dashboards provide granular and clear insight into the risk exposure and collateralization of DeFi as a whole. Furthermore, the source code of DeFi apps is open-source and can be reviewed by anyone.

3. **User control**: Non-custodial asset management is achieved through private keys, allowing DeFi apps to interface with assets in a trust-minimized manner. Decentralized autonomous organizations (DAOs) also allow for collective ownership of assets and applications.

4. **Reduced costs**: DeFi apps operate more efficiently and autonomously since the need for intermediaries is minimized. This facilitates low switching costs for moving capital across apps, creating an efficient market for app-level fees. Scaling technologies also make microtransactions feasible by reducing network-level fees.

5. **Composability**: Having a common settlement layer for running autonomous code allows for permissionless composability between new and existing DeFi apps. Developers don't have to worry about being deplatformed, further incentivizing collaboration.

Current layer-1 blockchains are insufficient in terms of providing the necessary performance to handle the execution and data availability requirements of real world DeFi applications, especially for streaming type of payment applications. Nautilus blockchain is purposely built for real world streaming DeFi type of applications. Nautilus can also be used for Web3 applications that involve a massively large number of users and large amounts of user interactions, such as GamiFi applications.

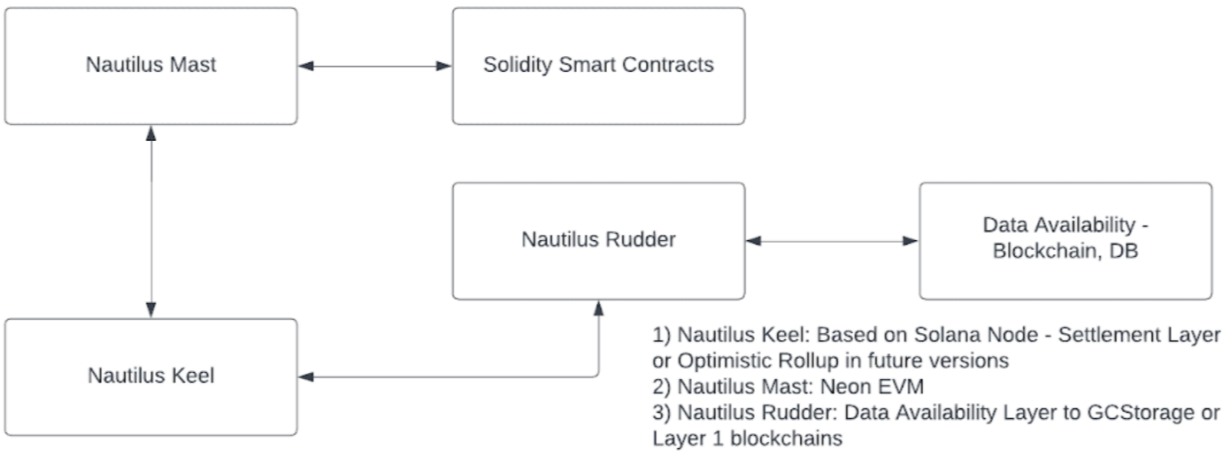
2 Nautilus: High Performance and Security

Nautilus blockchain is designed from the ground up to address the issues facing the wider adoption of Real World Asset in DeFi systems. The Nautilus blockchain is implemented in **two phases**: 1) modular blockchain architecture based on proven high performance layer 1 blockchain as settlement layer with EVM compatibility and data availability module to store transaction hashes on off-chain platforms 2) sovereign optimistic rollup blockchain with EVM compatibility and data availability module to store transaction hashes on off-chain platforms and other layer one blockchains.

Roadmap:
Stage 1: May 2023
Nautilus Blockchain v1.0:
- Settlement Layer: Nautilus Keel v1.0 based on Solana node
- Execution Layer: Nautilus Mast v1.0 based on Neon EVM, Solidity smart contracts, EVM compatible
- Data Availability Layer: Transaction hash stored on GCS

Stage 2
Nautilus Blockchain v2.0:
- Settlement Layer: Nautilus Keel v2.0 - Optimistic Rollup
- Execution Layer: Nautilus Mast v1.0 - EVM compatible Solidity smart contracts
- Data Availability Layer: Transaction hash stored on Layer 1 blockchains e.g. Bitcoin, Ethereum, BSC...

Phase I: High Performance EVM compatible modular blockchain



Phase I implementation of Nautilus Blockchain

In phase I implementation of the Nautilus blockchain, focus is put on modularity of the overall architecture and high performance. The phase I implementation of Nautilus blockchain is composed of Settlement Layer, Data Availability Module, and EVM Module. Settlement Layer is a modified and optimized version of the Solana node. The key differentiator for the Nautilus blockchain is EVM compatibility. Developers from the wider Ethereum community can deploy applications on the Nautilus blockchain without the need of extensive modifications. In phase I, Nautilus blockchain offers both the high performance of Solana and the wide adoption and ecosystem of EVM compatible smart contracts.

Phase I of the Nautilus blockchain is designed with modularity as a top consideration. Although the phase I implementation can be run with a data availability layer, the current implementation stores transaction hashes on cloud storage through the Data Availability Module to demonstrate and prove the benefits of modular architecture. In later implementations, data availability will be stored on other layer 1 blockchains.

Phase II: Sovereign Optimistic Rollup Blockchain

Nautilus blockchain implements two key innovations: **sovereign optimistic rollup** and **decentralized and highly secure data availability layer**:

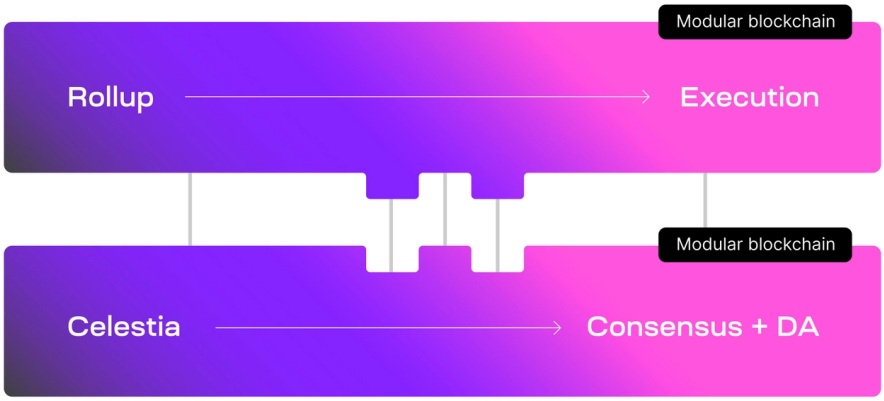


Figure 2: Nautilus Architecture [Placeholder image, to be replaced]

1. **Optimistic Rollup**

   An optimistic rollup is an approach to scaling Ethereum that involves moving computation and state storage off-chain. Optimistic rollups execute transactions outside of Ethereum, but post transaction data to Mainnet as calldata.

   Optimistic rollup operators bundle multiple off-chain transactions together in large batches before submitting to Ethereum. This approach enables spreading fixed

costs across multiple transactions in each batch, reducing fees for end-users. Optimistic rollups also use compression techniques to reduce the amount of data posted on Ethereum.

Optimistic rollups are considered “optimistic” because they assume off-chain transactions are valid and don't publish proofs of validity for transaction batches posted on-chain. This separates optimistic rollups from zero-knowledge rollups that publish cryptographic proofs of validity for off-chain transactions.

Optimistic rollups instead rely on a fraud-proving scheme to detect cases where transactions are not calculated correctly. After a rollup batch is submitted on Ethereum, there's a time window (called a challenge period) during which anyone can challenge the results of a rollup transaction by computing a fraud proof. If the fraud proof succeeds, the rollup protocol re-executes the transaction(s) and updates the rollup's state accordingly. The other effect of a successful fraud proof is that the sequencer responsible for including the incorrectly executed transaction in a block receives a penalty.

If the rollup batch remains unchallenged (i.e., all transactions are correctly executed) after the challenge period elapses, it is deemed valid and accepted on Ethereum. Others can continue to build on an unconfirmed rollup block, but with a caveat: transaction results will be reversed if based on an incorrectly executed transaction published previously.

2. **Data Availability Layer**

Data availability in blockchains refers to the ability of nodes to download the data contained within all blocks propagated through a peer-to-peer network. The idea is that your node can independently verify that the information it receives is correct by executing all the transactions in the blocks they receive from peers to ensure that the changes proposed precisely match those computed independently by the node. This means nodes do not have to trust that the senders of the block are honest. This is not possible if data is missing.

Data availability refers to the confidence a user can have that the data required to verify a block is really available to all network participants. For full nodes on Ethereum layer 1 this is relatively simple; the full node downloads a copy of all the data in each block - the data has to be available for the downloading to be possible.

A block with missing data would be discarded rather than being added to the blockchain. This is "on chain data availability" and it is a feature of monolithic blockchains. Full nodes cannot be tricked into accepting invalid transactions because they download and execute every transaction for themselves. However, for modular blockchains, layer 2 rollups and light clients, the data availability landscape is more complex, requiring some more sophisticated verification procedures.

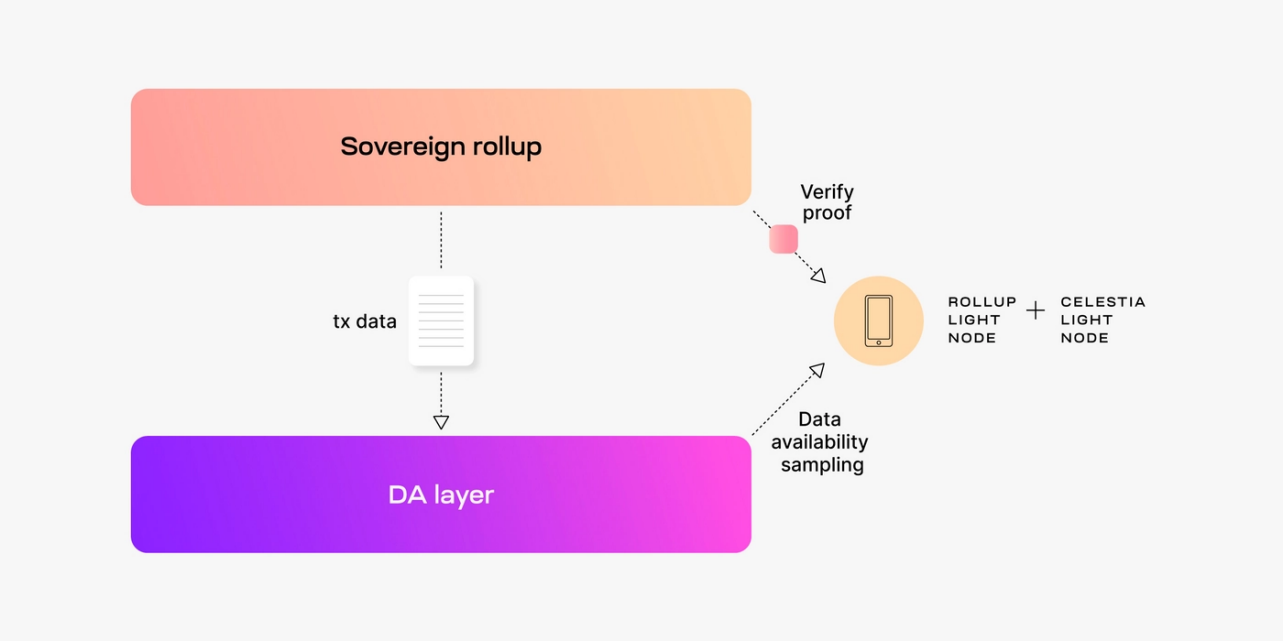


Figure 3: Nautilus - Sovereign Rollup [Placeholder image, to be replaced]

By combining these ideas, Nautilus achieves Superior speed, security, and customization for powerful real world streaming DeFi applications. Sovereign Rollup system offers the following features:

1. Wide selection of execution environment
2. Independent computational layer
3. Option to hard fork when necessary

2.1 Sovereign Optimistic Rollup

Optimistic rollups are off-chain scaling solutions. Each optimistic rollup is managed by a set of smart contracts deployed on the settlement layer. Optimistic rollups process transactions off the main underlying blockchain, but post off-chain transactions (in batches) to an on-chain rollup contract. Like layer-1 blockchains, this transaction record is immutable and forms the "optimistic rollup chain."

The architecture of an optimistic rollup comprises the following parts:

On-chain contracts: The optimistic rollups' operation is controlled by smart contracts running on the settlement blockchain. This includes contracts that store rollup blocks, monitor state updates on the rollup, and track user deposits. In this sense, the settlement blockchain serves as the base layer or "layer 1" for optimistic rollups.

Off-chain virtual machine (VM): Although contracts managing the optimistic rollup protocol run on the settlement blockchain, the rollup protocol performs computation and state storage off chain. The off-chain VM is where applications live and state changes are executed; it serves as the upper layer or "layer 2" for an optimistic rollup.

As optimistic rollups are designed to run programs either written or compiled for the EVM, the off-chain VM incorporates many EVM design specs. Additionally, fraud proofs computed on-chain allows the Ethereum network to enforce the validity of state changes computed in the off-chain VM.

Optimistic rollups are described as 'hybrid scaling solutions' because, while they exist as separate protocols, their security properties are derived from the data availability layer. Among other things, the data availability layer guarantees the correctness of a rollup's off-chain computation and the availability of data behind the computation. This makes optimistic rollups more secure than pure off-chain scaling protocols (e.g., sidechains) that do not rely on Ethereum for security.

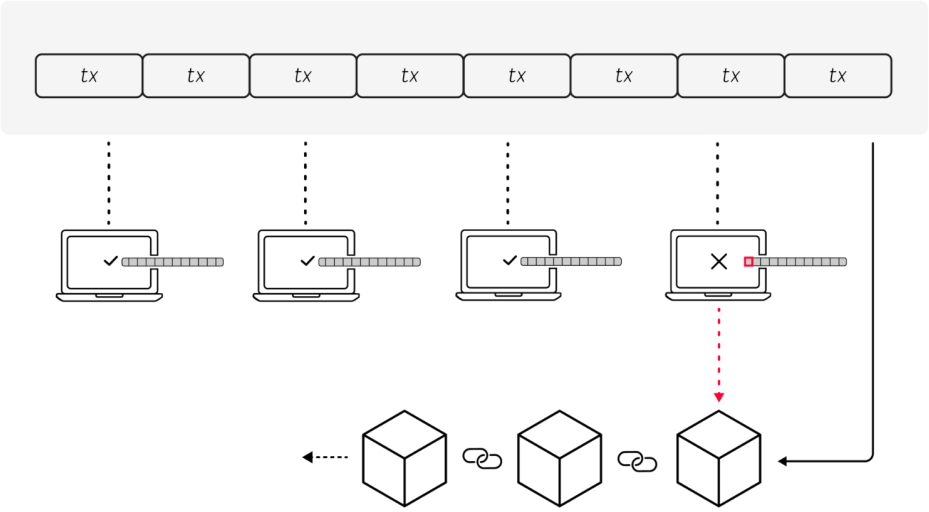


Figure 4: Optimistic Rollup Conceptualized [Placeholder image, to be replaced]

2.2 Secure and Distributed Data Availability Layer

A modular blockchain is a blockchain that handles a specific function, such as execution, consensus, or data availability and relies on other blockchains and off-chain systems to perform the remaining tasks. The modular blockchain stack comprises

different modular chains that work together in different ways to achieve set objectives.

The data availability layer in a modular blockchain stack is usually responsible for storing transaction data, although it may also provide consensus on the ordering of transactions. For example, modular blockchains that focus on execution (e.g. rollups and validiums) rely on off-chain data availability layers to store data behind state updates.

A data availability layer itself is a modular chain since it concentrates on storing data and outsources execution to other chains. Unlike regular blockchains, a pure data availability layer will not check the validity of data published by block producers. Nodes only have to come to consensus on the ordering of transactions and confirm that the right fees were paid.

Data availability sampling (DAS)

Data Availability Sampling (DAS) is a way for the network to check that data is available without putting too much strain on any individual node. Each node (including non-staking nodes) downloads some small, randomly selected subset of the total data. Successfully downloading the samples confirms with high confidence that all of the data is available. This relies upon data erasure coding, which expands a given dataset with redundant information (the way this is done is to fit a function known as a polynomial over the data and evaluating that polynomial at additional points). This allows the original data to be recovered from the redundant data when necessary. A consequence of this data creation is that if any of the original data is unavailable, half of the expanded data will be missing! The amount of data samples downloaded by each node can be tuned so that it is extremely likely that at least one of the data fragments sampled by each client will be missing if less than half the data is really available.

Data availability committees

Data Availability Committees (DACs) are trusted parties that provide, or attest to, data availability. DACs can be used instead of, or in combination with(opens in a new tab)↗ DAS. The security guarantees that come with committees depends on the specific set up. Ethereum uses randomly sampled subsets of validators to attest to data availability for light nodes, for example.

DACs are also used by some validiums. The DAC is a trusted set of nodes that stores copies of data offline. The DAC is required to make the data available in the event of a dispute. Members of the DAC also publish on-chain attestations to prove that the said data is indeed available. Some validiums replace DACs with a proof-of-stake (PoS) validator system.

Here, anyone can become a validator and store data off-chain. However, they must

provide a “bond”, which is deposited in a smart contract. In the event of malicious behavior, such as the validator withholding data, the bond can be slashed. Proof-of-stake data availability committees are considerably more secure than regular DACs because they directly incentivize honest behavior.

Data Availability for Nautilus

Two key features of Nautilus's DA layer are data availability sampling (DAS) and Namespaced Merkle trees (NMTs). Both features are novel blockchain scaling solutions: DAS enables light nodes to verify data availability without needing to download an entire block; NMTs enable execution and settlement layers on Nautilus to download transactions that are only relevant to them.

Data availability sampling (DAS)

In general, light nodes download only block headers that contain commitments (i.e., Merkle roots) of the block data (i.e., the list of transactions).

To make DAS possible, Nautilus uses a 2-dimensional Reed-Solomon encoding scheme to encode the block data: every block data is split into k × k chunks, arranged in a k × k matrix, and extended with parity data into a 2k × 2k extended matrix by applying multiple times Reed-Solomon encoding.

Then, 4k separate Merkle roots are computed for the rows and columns of the extended matrix; the Merkle root of these Merkle roots is used as the block data commitment in the block header.

Namespaced Merkle Trees (NMTs)

Nautilus partitions the block data into multiple namespaces, one for every application (e.g., rollup) using the DA layer. As a result, every application needs to download only its own data and can ignore the data of other applications.

For this to work, the DA layer must be able to prove that the provided data is complete, i.e., all the data for a given namespace is returned. To this end, Nautilus is using Namespaced Merkle Trees (NMTs).

An NMT is a Merkle tree with the leafs ordered by the namespace identifiers and the hash function modified so that every node in the tree includes the range of namespaces of all its descendants. The following figure shows an example of an NMT with height three (i.e., eight data chunks). The data is partitioned into three namespaces.

3 Nautilus: Real World DeFi Based on Payment and Revenue Streams

Nautilus is a general purpose modular high performance blockchain, and can be used for a variety of applications, ranging from DeFi, GameFi, to any Web3 application that runs on smart contracts. Nautilus is especially suitable for real world DeFi systems that taps into a continuous and predictable stream of payments or revenues. With predictable future asset streams, innovative, robust, and real world DeFi systems can be built on Nautilus.

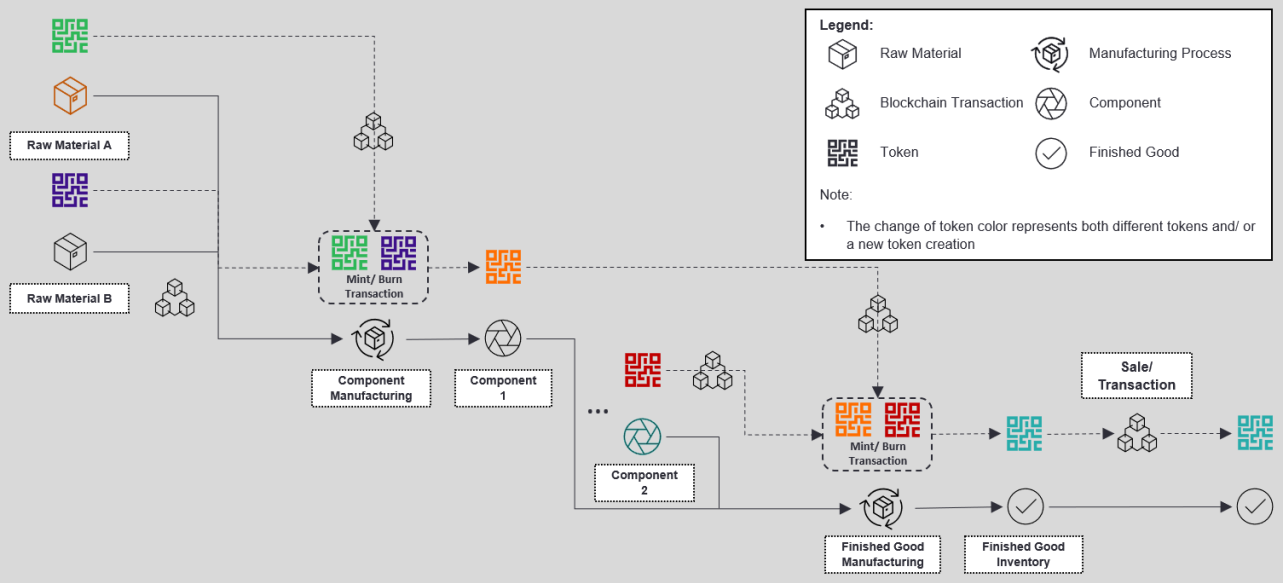
3.1 Payment and Revenue Streams

A guaranteed payment stream or future revenue stream can be the most trusted source of collateral for commercial lending and other financial activities. Nautilus is optimized for tracking and streaming payments through automated smart contracts. In order to efficiently and effectively stream financial payments, blockchain has to be able to handle large quantities of payment transactions at very low cost. [To be expanded with more technical details]

3.2 Real World Assets

Tokenizing real-world assets allows DeFi to tap into some of the largest financial markets. Global real estate was valued at $327 trillion in 2020 and non-financial corporate debt at over $87 trillion in 2022. These are colossal markets to which tokenization can bring enhanced liquidity and new investors. Bringing real world assets on chain requires the following key components:

1. Modules that implement financial services business logic. Modules are composed to realize complex financial services processes.
2. A smart contracting engine to develop and deploy contracts directly to the Provenance Blockchain.
3. Off-chain client-side agreements using the Contract Execution Environment



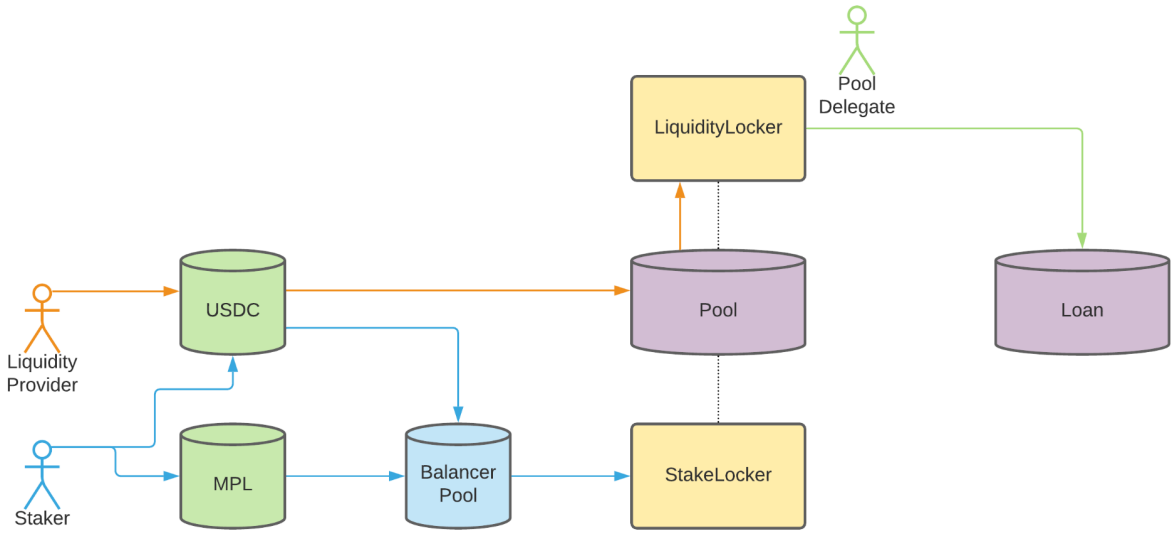
[To be expanded with more technical details]

4 A World with Nautilus

RWAs compose the majority of global financial value. For instance, the fixed income debt market is worth an estimated $127 trillion, the total value of global real estate is approximately $362 trillion, and gold has an $11 trillion market capitalization.

Real-world assets are tangible assets or financial primitives with the potential to serve as collateral in the DeFi industry. Here are the most popular examples of RWAs:

Cash
Precious Metals (gold, silver …)
Real estate
Corporate debt
Insurance
Salaries and invoices
Consumer goods
Credit notes
Royalties
Receivables
Salary Payments



RWAs already underpin lending and yield generation activities in the traditional finance world. The set of new DeFi applications enabled by Nautilus is quite broad and encompasses commercial paper based DeFi markets, real asset based DeFi markets, and fixed income based DeFi markets. We list here some of the possibilities, many of which are also exciting directions of ongoing and future research:

Stablecoins

Stablecoins are a perfect example of successful real-world asset use in DeFi, with three of the top seven crypto tokens by market capitalization being stablecoins (a combined $136 billion). Issuing companies such as Circle maintain an audited reserve of USD assets and mint USDC tokens for use across DeFi protocols.

Synthetic tokens and Derivatives

Synthetic tokens represent another use case involving the bridging of RWAs to DeFi. Synthetic tokens allow on-chain trading of derivatives linked to currencies, stocks and commodities. Leading synthetic token trading platform Synthetix had $3 billion worth of assets locked in its protocol at the peak of the 2021 bull run.

Lending protocols

Another exciting adoption of RWA in DeFi involves lending protocols. Unlike primitive lending protocols that rely on crypto-native borrowing, RWA-focused DeFi platforms service borrowers with real-world businesses. This model offers relatively stable returns insulated from crypto volatility. [To be expanded with more technical details]

5 Summary

Nautilus is a high performance modular blockchain built for real world DeFi and Web3 applications. Nautilus offers exceedingly high performance and superior security through its adoption of sovereign optimistic rollup and highly secure and distributed data availability layer.

1. High Throughput

   Nautilus will have an initial TPS of 2,000, with much higher rates soon to come. Nautilus achieves such a high rate through parallelizing transactions instead of processing them linearly.

2. Easy for Developers

   Solidity is the de facto Web3 language, and developers tend to prefer working with Solidity rather than a more specialized langue. Nautilus lets Solidity developers build high-throughput applications in an EVM environment.

3. Reliable
   Even though Nautilus can eventually surpass Solana in TPS, it will be far more reliable because it is a sovereign rollup. This means that even if the underlying L1 consensus layer suffers an outage, it will still be able to execute transactions.

Use Cases:

Nautilus is perfect for any high-throughput applications that want to operate with an EVM. This includes use cases like payments, DeFi, and Web3 gaming applications.

Payments

Nautilus can be used for high-throughput payment applications, like payroll, invoicing, and merchant services.

DeFi

DeFi has long needed a solution like Nautilus that will provide low gas fees and lightning-fast transactions. Nautilus is perfect for dApps like DEXs, payment streaming, and lending platforms.

Web3 Gaming

Great games have not yet been developed on-chain, because they require too much throughput unless they are running on their own execution chain. An L3 like Nautilus will be able to support Web3 games that require high scalability.

References

[1] Satoshi Nakamoto. Bitcoin: A peer-to-peer electronic cash system. *Decentralized business review*, page 21260, 2008.

[2] Vitalik Buterin et al. Ethereum white paper.

[3] Ethereum: A secure decentralized generalized transaction ledger. https://ethereum.github.io/yellowpaper/paper.pdf.

[4] Rollup-centric ethereum roadmap. https://ethereum-magicians.org/t/a-rollup-centric-ethereum-roadmap/4698.

[5] An incomplete guide to rollups. https://vitalik.ca/general/2021/01/05/rollup.html.

[6] Lido.fi. https://lido.fi/.

[7] Rocketpool. https://rocketpool.net/.

[8] Superfluid staking. https://docs.osmosis.zone/osmosis-core/modules/superfluid/.

[9] The cryptoeconomics of slashing. https://a16zcrypto.com/ the-cryptoeconomics-of-slashing/.

[10] Merged mining. https://developers.rsk.co/rsk/architecture/mining/.

[11] Blockspace: An introduction with chris dixon. https://www.generalist.com/briefing/ blockspace.

[12] Rainbow bridge. https://near.org/bridge/.

[13] Uni should become an oracle token. https://gov.uniswap.org/t/uni-should-become-an-oracle-token/11988.

[14] State of research: increasing censorship resistance of transactions under proposer/builder separation (pbs). https://notes.ethereum.org/@vbuterin/pbs_censorship_resistance.

[15] Committee-driven mev smoothing. https://ethresear.ch/t/ committee-driven-mev-smoothing/10408.

[16] Shutter - in-depth explanation of how we prevent front running. https://blog.shutter.network/shutter-in-depth-explanation-of-how-we-prevent-frontrunning/.

[17] Removing trusted relays in mev-boost using threshold encryption. https://ethresear.ch/t/removing-trusted-relays-in-mev-boost-using-threshold-encryption/13449.

[18] Paths toward single-slot finality. https://notes.ethereum.org/@vbuterin/single_slot_ finality.

[19] Soulbound. https://vitalik.ca/general/2022/01/26/soulbound.html.