thirdlyr – A Layer-3 architecture for atomic transactions and decentralized data availability for maritime and cross-border finance

thirdlyr

Feb 2025, v0.1

Abstract

We introduce thirdlyr, a Layer-3 blockchain architecture purpose-built for atomic transactions that integrate payments and decentralized data availability, specifically for maritime trade and cross-border finance. thirdlyr is designed to provide high throughput, cost-efficient data anchoring, and decentralized storage, ensuring a seamless and verifiable settlement process. We present the core architectural innovations behind thirdlyr, detail its transaction lifecycle, and outline the protocol's roadmap for scalability and adoption in maritime and cross-border financial ecosystems.

Contents

1 Introduction

Even though the maritime industry moves 80% of the world's goods, maritime trade and cross-border finance still rely on inefficient and costly legacy infrastructure. Payments and trade documents remain fragmented across banking systems, regulatory databases, and supply chain intermediaries. The result is a high-friction settlement process, prone to delays, manual reconciliation, and disputes. A common reason for slow payments is when there is a request for information (RFI) by an intermediary bank, where there is not enough information about a particular payment during a telegraphic transfer. The cascading of documents from one bank to another is slow, which leads to the delay. The lack of transparency on the fund flow also leads to a frustrating experience for the sender.

While emerging fintechs and blockchain-based solutions have emerged to address friction and pain points in consumer payments, these solutions do not integrate verifiable trade documents directly into transaction execution.

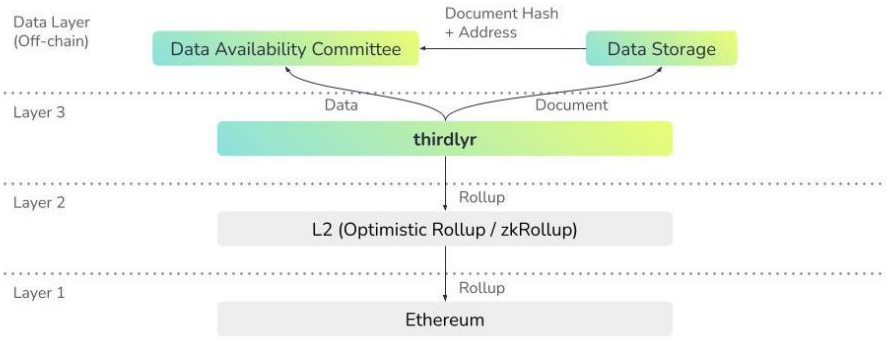
To streamline and modernize the payments process, we introduce thirdlyr – a unified atomic transaction framework, linking financial settlements with provable data storage, ensuring end-to-end visibility and compliance in global trade. In this paper, we introduce key concepts and outline the overall architecture of thirdlyr. We aim to align pain points experienced by compliance-focused and global industries such as maritime and trade with our purpose-built blockchain. We will also share the possibilities of linking financial settlements with provable data storage.

2 Layer-3 Blockchain

thirdlyr is the first data-integrated Layer-3 blockchain built on Arbitrum Orbit, designed to synchronize payments with off-chain document verification and storage in a single atomic transaction. Layer-3 blockchains represent the next evolution of rollup technology, providing application-specific optimizations that extend beyond the scalability benefits of traditional Layer-2 solutions. Transactions on thirdlyr are optimized for high efficiency and cost-effectiveness, leveraging an optimistic rollup model to minimize fees while ensuring compliant-ready data storage.

As a Layer-3 built on Arbitrum Orbit, thirdlyr also benefits from custom execution logic, allowing for domain-specific optimizations in maritime finance, trade documentation, and institutional payments. The modular structure ensures that financial entities can interact with the blockchain without requiring full exposure to complex smart contract management or blockchain-based data storage mechanisms.

3 Architecture



thirdlyr is a Layer-3 blockchain architecture purpose-built for atomic transactions that integrate financial settlements with decentralized data availability. The architecture is composed of three core components that work together to ensure efficient execution, secure data anchoring, and scalable storage:

- Layer-3 Execution Environment: A smart contract execution framework optimized for atomic transaction finality and cross-chain interoperability.
- Data Availability Committee (DAC): A cryptographic data anchoring system ensuring off-chain data verifiability and redundancy.
- Decentralized Data Storage (DDS): A distributed storage layer optimized for maritime trade documents, compliance records, and payment receipts.

thirdlyr operates as an Arbitrum Orbit-based Layer-3 blockchain, leveraging optimistic rollup technology to optimize transaction costs while ensuring the security guarantees of Ethereum. Unlike traditional Layer-2 solutions that primarily focus on scalability, thirdlyr is designed to integrate financial execution with verifiable data storage. Transactions on thirdlyr do not solely involve token transfers but also include commitments to external off-chain data, which are subsequently referenced in smart contract execution.

The execution layer supports atomic transactions, meaning that payment transfers and associated document attestations occur within a single verifiable operation. If a condition tied to document verification fails (e.g., missing bill of lading confirmation), the transaction does not execute. This ensures that maritime trade payments, regulatory filings, and escrow conditions are handled in a trustless and automated manner.

Smart contracts within thirdlyr implement programmable financial logic that interacts with oracles, credit scoring mechanisms, and AI-generated synthetic data models. The contract architecture is modular, allowing users to compose multi-step transactions that involve multiple counterparties, spanning trade finance, freight settlements, and international payments.

To maintain efficiency without overloading on-chain storage, thirdlyr employs a Data Availability Committee (DAC) that ensures off-chain data integrity while providing cryptographic proofs for verification. The DAC is an enhanced AnyTrust model, where selected validators manage availability attestations for documents referenced in on-chain transactions. When a transaction involves a trade document, the document's cryptographic Merkle root and storage address are published on-chain, while the actual document is stored in decentralized storage nodes. The DAC ensures that the data remains accessible and verifiable, even though it is not permanently stored on the blockchain itself.

thirdlyr extends the standard Arbitrum AnyTrust model by introducing distributed storage coordination, allowing DAC nodes to act as both availability validators and optional storage providers. This enhances redundancy while keeping storage costs minimal. The DAC operates under a staking-based security model, where misbehavior, e.g., withholding data, results in economic slashing. The DAC follows the commit-reveal scheme for ensuring verifiable off-chain storage. The process involves:

- Commit Phase: DAC nodes commit to storing a document by signing and submitting a Merkle root on-chain.
- Reveal Phase: Upon request, DAC nodes must provide the original document or a corresponding proof of inclusion within the committed Merkle tree.
- Slashing Mechanism: If a DAC node fails to reveal data when challenged, it is penalized, and its stake is redistributed to honest participants.

This model ensures low-cost, high-throughput data availability guarantees for maritime trade documents, payment receipts, and compliance reports. While the DAC ensures data availability, the actual long-term storage and retrieval of trade documents and transaction records occurs in thirdlyr's Decentralized Data Storage (DDS) layer. DDS is a sharded, fault-tolerant storage system optimized for maritime and financial records, enabling businesses to store large amounts of data without the prohibitive costs of on-chain storage. thirdlyr's DDS employs RaptorQ erasure coding, a next-generation rateless erasure code algorithm that allows data to be split into fragments while ensuring reconstructability. Compared to traditional storage redundancy techniques, RaptorQ enables efficient retrieval with minimal replication overhead. More details on the algorithm will be furnished in Section 5.

4 THIRD Token

THIRD is the native utility and incentive token of the thirdlyr ecosystem, designed to facilitate network security, storage incentives, and computational rewards across its Layer-3 blockchain. Unlike conventional ERC-20 utility tokens that serve only as a medium of exchange, THIRD is deeply integrated into thirdlyr's data availability, decentralized storage, and payment automation frameworks.

THIRD serves as the primary transaction fee token within thirdlyr, ensuring low-cost execution of atomic transactions. As a Layer-3 blockchain operating under an optimistic rollup model, thirdlyr inherits Ethereum's security while maintaining its own execution and settlement logic. THIRD is used to pay for gas fees within the ecosystem, optimizing the economic model for maritime payments, trade settlements, and decentralized AI-based credit evaluation.

The Data Availability Committee (DAC) and Decentralized Data Storage (DDS) nodes are rewarded in THIRD tokens for maintaining data integrity and uptime. Instead of storing raw documents on-chain, thirdlyr ensures that data references remain cryptographically linked while keeping costs manageable. DAC participants must stake THIRD tokens to act as availability providers, and their continued participation is secured through slashing conditions that penalize misbehavior, such as data withholding or failure to respond to retrieval requests. The decentralized storage layer utilizes THIRD to reward nodes based on retrieval frequency, storage duration, and geographic distribution, ensuring high redundancy and fault tolerance.

In later phases of thirdlyr's development, autonomous AI-driven payment agents will execute cross-border settlements and enforce on-chain compliance checks. These AI agents require computational resources to operate, and THIRD tokens are used to compensate validators and execution nodes that process agent-driven transactions. Furthermore, thirdlyr's generative AI models for synthetic financial data generation require decentralized compute resources. Nodes that contribute processing power for on-chain AI model training and federated learning are rewarded in THIRD, creating an incentive-driven ecosystem where AI, blockchain, and decentralized storage converge.

THIRD also functions as a governance token, allowing stakeholders to propose and vote on key network upgrades, storage policies, and economic parameters. Holders can stake THIRD tokens to gain voting power proportional to their holdings, ensuring that long-term participants drive decision-making rather than short-term speculators. The staking model extends beyond governance, allowing long-term storage providers and validators to earn additional yield through secured staking pools. These pools dynamically adjust staking rewards based on network demand, ensuring that economic incentives align with network activity.

THIRD follows a deflationary issuance model, where a portion of transaction fees and slashing penalties are permanently removed from circulation. Unlike traditional gas tokens, THIRD features a dynamic fee adjustment mechanism that

scales transaction costs based on network congestion, ensuring that high-volume trade settlements and AI-driven transactions remain economically viable. As thirdlyr grows, the demand for THIRD tokens will be driven by increasing data storage needs, AI computation requirements, and cross-border transaction volumes. The token's utility extends beyond mere speculation, embedding itself directly into the payment, compliance, and automation frameworks of the thirdlyr ecosystem.

5 Roadmap

5.1 Phase One: Decentralized Data Storage (DSS)

The first phase for thirdlyr involves establishing a distributed and decentralized data storage system for scalable data storage with high throughput and fidelity. In general, there are two extremes in which data can be stored in a decentralized and distributed data storage.

In a system of N nodes, we can give each node a copy of a file of arbitrary size or break the file up into N equal chunks and distribute each chunk to a node. In the former, any server has a full copy of the file and in the case of node failure, we'd still have *N-1* nodes available serving the full file. This gives maximum redundancy but requires more space. In the latter, chunking the file into N chunks leads to the lowest cost and space requirement. However, any single node failure results in failing to reconstitute the file.

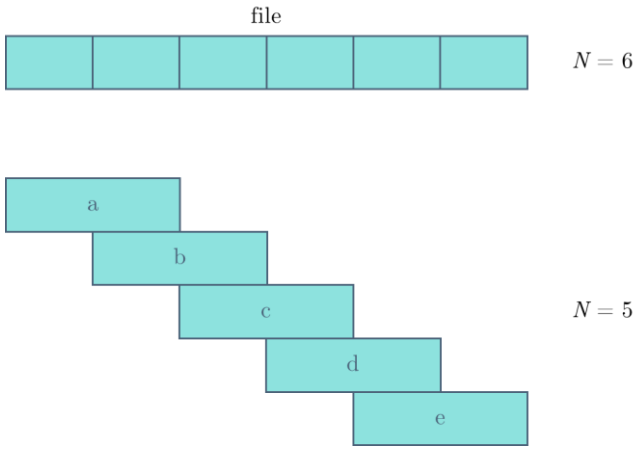
We can express the probability of data reconstruction P for either scenario as:

$$P_{replication} = 1 - f^N \quad (1)$$

$$P_{chunk} = (1-f)^N \quad (2)$$

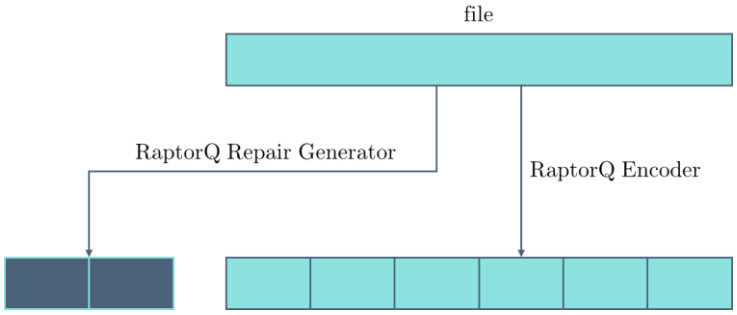
where f is the probability of node failure, N is the number of chunks. In (1), the probability of file construction in naïve full node replication is high when N is high. On the other hand, in (2), the probability of replication in naïve chunking is low at any non-zero value f.

Erasure coding is the middle ground, balancing redundancy with storage requirements. In essence, erasure coding results in a case where a file is divided into N chunks but still can be reconstituted with M chunks destroyed. This is more useful and applicable in real-life situations where nodes may not always be online and available.



To illustrate this, we have a file that is divided into multiple chunks for storage in nodes. In the first chunking method, you divide the chunks equally into N pieces and intuitively we can see that losing any part of the chunk will lead to an incomplete file. In the second chunking method, there are fewer chunks, but each chunk is larger compared to the earlier chunk and each chunk also overlaps with the preceding and next chunks. In theory, you minimally need only chunks a, c, and e to reconstitute the entire file because of the redundancy offered by chunks b and d. While this is a simplification of erasure coding, it is sufficient to convey how it helps maintain data fidelity and redundancy in distributed systems.

In thirdlyr DSS, we adopt a variant of the rapid tornado (raptor) algorithm named RaptorQ forward error correction to enhance fault tolerance, efficient retrieval, and minimal redundancy overhead. RaptorQ is a form of rateless erasure code, in which a potentially limitless sequence of encoding symbols can be generated from a given set of source symbols (file chunks). The original source symbols can ideally be recovered from any subset of the encoding symbols of size equal to or only slightly larger than the number of source symbols, which limits network overhead. This is because the RaptorQ encoder also generates redundant parity chunks through the repair generator, and these chunks can be used to fully reconstitute the file even when there are chunk losses.



Here is the general sequence of events in a RaptorQ encoding event:

- A file is divided into k source symbols (chunks).
- r additional parity symbols are generated.
- The encoded data is distributed across storage nodes.
- Any subset of k' symbols is sufficient to reconstruct the full file, where $k' = k + \epsilon$, with ϵ being a small overhead factor.

Thus, we can express the probability of a successful data reconstruction P using RaptorQ as:

$$P_{RaptorQ} = 1 - P_{fail} \tag{3}$$

where:

$$P_{fail} = \sum_{i=0}^{k\prime-1} \binom{N}{i} (1 - f)^i f^{(N-i)} \quad (4)$$

where:

- P_{fail} is the probability that fewer than k′ encoded symbols are retrievable, making file reconstruction impossible.
- N is the total number of encoded symbols stored across nodes ($N = k + r$).
- f is the failure probability of a single storage node.

This results in a robust decentralized data storage system that independent nodes or DAC members can participate in. Even in a situation where ~20% of the nodes fail, full data reconstitution is still achievable and available.

5.2 Phase Two: Maritime and Cross-Border Payment Oracle

The maritime industry operates in an environment where real-time and historical data are critical for financial settlements, regulatory compliance, and operational efficiency. Traditional trade finance relies on fragmented data sources, making transactions slow, opaque, and prone to disputes. Thus, as part of the second phase, thirdlyr introduces a Maritime and Cross-Border Payment Oracle (Maritime Oracle), a decentralized oracle network tailored for maritime and cross-border payments, enabling secure, verifiable, and automated data retrieval from external trade, logistics, and regulatory systems. The Maritime Oracle will aggregate and verify data from multiple external sources to provide accurate and tamper-proof inputs for smart contracts. This ensures that payments and trade document verification are executed as atomic blockchain transactions, reducing delays and manual reconciliation.

The Maritime Oracle aggregates and verifies data from AIS tracking (geospatial data), SGTradeX (port data flow), port authorities, and banking APIs. This enables real-time vessel tracking, automated customs clearance, and accurate foreign exchange validation. By connecting to port call information systems and logistics networks, thirdlyr ensures that transactions are executed only when predefined trade conditions are met, reducing fraud and improving compliance.

In the first phase, we anticipate the payments to be triggered manually when conditions are met. With the oracles in the second phase, thirdlyr smart contracts can automate trade finance and maritime payments using verified data. Payments for port fees, cargo handling, and freight insurance can be triggered upon real-time vessel arrival confirmation. Cross-border trade settlements can be automated by validating customs records and AML compliance before transaction execution. By integrating with global trade registries, thirdlyr ensures that invoices, bills of lading, and logistics updates are securely linked to financial transactions.

Oracle operators earn rewards based on data accuracy and network demand, while dishonest actors face slashing penalties. Fees are dynamically adjusted based on usage, ensuring a sustainable incentive model. The Maritime Oracle is designed as a modular product, serving both thirdlyr's Layer-3 and external blockchain ecosystems. Initially deployed on thirdlyr's Layer-3 (Arbitrum Orbit), it will expand to other Ethereum L2s via Chainlink Cross-Chain Interoperability Protocol (CCIP).

By providing trustless, automated trade finance execution, thirdlyr's Maritime Oracle transforms how cross-border payments and shipping transactions are conducted, reducing fraud, settlement delays, and compliance risks.

5.3 Phase Two: On-chain Credit Scoring

thirdlyr introduces an on-chain credit scoring system leveraging payments data and decentralized data storage to assess borrower risk in a transparent and efficient manner. Traditional credit scoring models rely on opaque financial histories and centralized data providers, making access to credit restrictive, particularly in cross-border trade.

The current landscape of credit assessment in trade finance is highly fragmented, with businesses and individuals relying on disparate systems that do not communicate seamlessly. Many enterprises in emerging markets lack access to fair credit evaluation due to missing financial records or limited banking history. This results in inefficiencies where lenders either extend credit blindly or impose excessively high collateral requirements to offset the perceived risk.

thirdlyr's on-chain credit scoring system is built to address these inefficiencies by leveraging real-time payments data, verified trade transactions, and stored financial records in a decentralized format. The system captures historical invoice settlements, payment behavior, trade counterparties, and transactional consistency, ensuring a more holistic view of borrower reliability. Instead of relying on credit bureaus that operate in silos, thirdlyr's scoring mechanism aggregates data directly from decentralized payments and document attestations.

The underlying architecture consists of three key components: on-chain payment history, off-chain document validation, and a metascoring algorithm that integrates multiple creditworthiness indicators into a single verifiable credit score. The metascoring algorithm assigns weights to various financial behaviors, including repayment timeliness, trade frequency, transaction volume, and risk exposure derived from counterparties' financial health. By processing these factors in a transparent and auditable framework, lenders gain access to real-time, fraud-resistant credit assessments.

The metascoring function is a probability expressed as:

$$P(C_t) = \sigma(\alpha T + \beta H + \gamma R + \delta D + \epsilon S) \tag{5}$$

$$\sigma(x) = \frac{1}{1 + e^{-x}} \tag{6}$$

where:

- $P(C_t)$ is probability of repayment at time t
- T is transaction volume and frequency.
- H is payment consistency and history.
- R is risk exposure derived from counterparties' creditworthiness.
- D represents off-chain document-backed verification scores.
- S is settlement efficiency, representing the time taken to clear obligations.
- α, β, γ, δ, ε are tunable weights calibrated based on risk models.
- is the sigmoid function, ensuring the output is a probability between 0 and 1.

- $\sigma(x)$ is the sigmoid function, ensuring that the output is a probability between 0 and 1

The scores in the metascore are calculated internally and is out of scope in this whitepaper. Apart from the score, we consider temporal effects in credit rating as well, i.e. our credit scoring model incorporates a recursive Bayesian updating process, which dynamically adjusts the probability of creditworthiness based on new payments data and trade documentation verifications. This ensures that the score is always up-to-date and reflective of actual risk. The Bayesian update rule is given by:

$$P(C_t|D_t) = \frac{P(D_t|C_t)P(C_{t-1})}{P(D_t)} \tag{7}$$

where:

- $P(C_t|D_t)$ is the updated probability of creditworthiness given new data.
- $P(D_t|C_t)$ is the likelihood of observing new payment and trade data given past credit behavior.
- $P(C_{t-1})$ is the prior probability of the borrower's creditworthiness before the update.
- $P(D_t)$ is the normalizing factor ensuring a valid probability distribution.

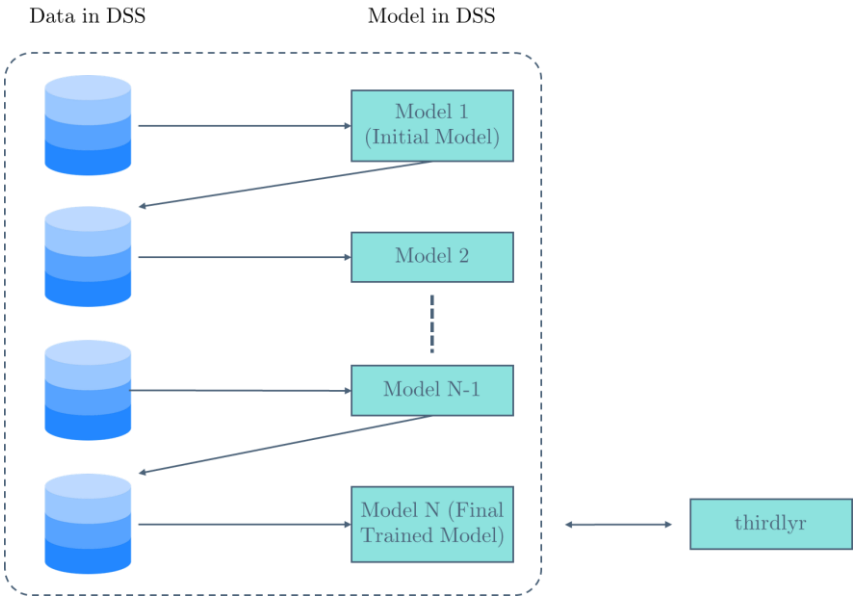
Borrowers with a strong history of on-time settlements and verified trade flows receive higher probabilities of creditworthiness and lower financing costs, while risky behavior leads to probability degradation over time. This ensures that businesses can build creditworthiness organically through consistent trade performance, rather than relying solely on historical data.

Smart contracts enforce risk-based loan approvals and collateralization levels by using the generated probabilities as deterministic inputs. Lenders can configure custom risk tolerances, automatically adjusting loan interest rates and credit limits based on the borrower's up-to-date financial track record. The integration of on-chain verifiable creditworthiness probabilities with smart contract lending ensures a permissionless, trust-minimized credit environment, removing the inefficiencies of traditional underwriting processes.

To the best of our knowledge, thirdlyr's approach to redefine credit evaluation by utilizing transaction history, on-chain payment behaviors, and stored trade documentation to generate a decentralized, verifiable credit score will be a world first.

5.4 Phase Three: Synthetic Data Generation

Apart from on-chain credit scoring models, thirdlyr also leverages its blockchain and decentralized storage system to run generative AI models for synthetic payment data. Traditional machine learning models rely on centralized servers, introducing privacy concerns and single points of failure. By distributing model training, inference, and data synthesis across decentralized compute nodes and storage layers, thirdlyr ensures privacy, fault tolerance, and regulatory compliance.



The generative AI framework operates on a federated learning architecture where multiple nodes train local models using differentially private gradients instead of sharing raw data. These updates are aggregated via secure multi-party computation (MPC) and homomorphic encryption, allowing collaborative learning while preserving input confidentiality.

thirdlyr's Decentralized Storage System (DSS) acts as the backbone for storing trained AI models and synthetic data. Instead of a centralized repository, DSS distributes model checkpoints and inference results across storage nodes, similar to how IPFS stores AI-generated content in traditional decentralized architectures. When an AI model completes training, its cryptographic hash is stored in the DSS, allowing for secure and efficient retrieval while preventing unauthorized modifications.

Unlike centralized AI architectures, where model inference lacks transparency, thirdlyr implements on-chain verifiability for generative AI models. The thirdlyr blockchain records the hashes of trained models and validation proofs, ensuring immutability and integrity. Each model operates within a trusted execution

environment (TEE) to maintain confidentiality of weights and training data while enforcing predefined validation constraints to prevent adversarial manipulation.

Privacy-preserving synthetic data generation relies on homomorphic encryption and differential privacy techniques that allow computations to be performed on encrypted datasets without exposing underlying transaction details. A transformation function applies controlled noise to real payment data, generating synthetic records that retain statistical fidelity while ensuring individual transactions cannot be reverse-engineered.

To align with thirdlyr's optimistic rollup framework, AI-generated synthetic data is assumed valid upon submission but remains subject to fraud-proof mechanisms. Validators can challenge faulty synthetic outputs, triggering re-execution or dispute resolution to maintain the integrity of synthetic datasets used for financial modeling and compliance verification.

Smart contracts govern AI model validation, storage, and incentive distribution. Compute nodes participating in model training and inference receive incentives via THIRD tokens, ensuring decentralized collaboration and preventing reliance on centralized AI providers. Fraudulent or low-quality contributions are penalized through a challenge-response dispute mechanism, reinforcing data integrity. This approach ensures that synthetic datasets can be generated in a trust-minimized, scalable, and privacy-preserving manner.

By integrating decentralized AI training, encrypted synthetic data storage, and on-chain validation, thirdlyr establishes a novel paradigm for privacy-preserving synthetic data generation in financial applications. This system supports use cases such as risk modeling, credit scoring, fraud detection, and liquidity simulations without reliance on centralized AI providers. Through a trust-minimized, scalable, and cryptographically secure architecture, thirdlyr redefines how synthetic financial data can be generated and utilized within the global trade and payments ecosystem.

5.5 Phase Three: Payment Agents

In the final phase, thirdlyr will introduce payment agents. Building on top of what we will build in prior phases, these agents are fully autonomous AI-driven entities designed to execute financial transactions, optimize liquidity flows, and facilitate cross-border settlements with minimal human intervention. These agents operate within the ecosystem, leveraging the decentralized storage system (DSS), generative AI infrastructure, on-chain credit scoring mechanisms, and real-time oracles to enhance decision-making and transaction efficiency.

The agents function as intelligent financial intermediaries, capable of managing accounts, processing payments, and ensuring compliance with regulatory requirements. By integrating generative AI, these agents can dynamically assess risk profiles, predict optimal transaction routes, and execute trades with precision. Through continuous learning, they adapt to market conditions, identifying trends and anomalies that influence payment flows and counterparty behaviors.

Payment agents utilize the decentralized oracles within thirdlyr to access and verify external financial data, such as real-time exchange rates, liquidity pools, and trade documentation. This ensures that transactions are executed at the most favorable rates while maintaining compliance with international regulations. The integration of on-chain credit scoring allows these agents to assess counterparties' creditworthiness and determine appropriate settlement terms, enabling seamless peer-to-peer and institutional transactions without requiring third-party intermediaries.

To ensure security and reliability, thirdlyr payment agents execute transactions through smart contract-driven workflows that are resistant to fraud and manipulation. The optimistic rollup infrastructure ensures that transactions are validated efficiently while preserving decentralization and minimizing costs. Additionally, trusted execution environments (TEEs) enable the agents to perform confidential computations, ensuring that sensitive financial data remains private while still benefiting from the verifiability and transparency of blockchain technology.

By embedding payment agents into thirdlyr infrastructure, the ecosystem moves toward a model where AI-driven financial automation is seamlessly integrated with decentralized finance. These agents act as the core for next-generation payment networks, enabling intelligent cross-border settlements, decentralized treasury management, and autonomous financial services. As thirdlyr continues to evolve, payment agents will play a pivotal role in redefining how businesses and individuals interact with financial systems in a fully automated, trustless environment.