MiCAR WHITE PAPER

“KITE TOKEN”

VERSION 1.0

OCTOBER 2025

White Paper in accordance with Regulation (EU) 2023/1114 of 31 May 2023 on markets in crypto-assets (MiCAR)

NOTE:
THIS CRYPTO-ASSET WHITE PAPER HAS NOT BEEN APPROVED BY ANY COMPETENT AUTHORITY IN ANY MEMBER STATE OF THE EUROPEAN UNION. THE PERSON SEEKING ADMISSION TO TRADING OF THE CRYPTO-ASSET IS SOLELY RESPONSIBLE FOR THE CONTENT OF THIS CRYPTO-ASSET WHITE PAPER.

00 Table of contents

COMPLIANCE STATEMENTS

01 Date of notification

2025-10-17

02 Statement in accordance with Article 6(3) of Regulation (EU) 2023/1114

This crypto-asset white paper has not been approved by any competent authority in any Member State of the European Union. The person seeking admission to trading of the crypto-asset is solely responsible for the content of this crypto-asset white paper.

03 Compliance statement in accordance with Article 6(6) of Regulation (EU) 2023/1114

This crypto-asset white paper complies with Title II of Regulation (EU) 2023/1114 and, to the best of the knowledge of the management body, the information presented in the crypto-asset white paper is fair, clear and not misleading and the crypto-asset white paper makes no omission likely to affect its import.

04 Statement in accordance with Article 6(5), points (a) (b) (c) of Regulation (EU) 2023/1114

The crypto-asset referred to in this white paper may lose its value in part or in full, may not always be transferable and may not be liquid.

05 Statement in accordance with Article 6(5), point (d), of Regulation (EU) 2023/1114

The utility token referred to in this white paper may not be exchangeable against the good or service promised in the crypto-asset white paper, especially in the case of a failure or discontinuation of the crypto-asset project.

06 Statement in accordance with Article 6(5), points (e) and (f), of Regulation (EU) 2023/1114

The crypto-asset referred to in this white paper is not covered by the investor compensation schemes under Directive 97/9/EC of the European Parliament and of the Council or the deposit guarantee schemes under Directive 2014/49/EU of the European Parliament and of the Council.

Summary

07 Warning

This summary should be read as an introduction to the crypto-asset white paper.

The prospective holder should base any decision to purchase this crypto-asset on the content of the crypto-asset white paper as a whole and not on the summary alone. The offer to the public of this crypto-asset does not constitute an offer or solicitation to purchase financial instruments and any such offer or solicitation can be made only by means of a prospectus or other offer documents pursuant to the applicable national law.

This crypto-asset white paper does not constitute a prospectus as referred to in Regulation (EU) 2017/1129 of the European Parliament and of the Council (36) or any other offer document pursuant to Union or national law.

08 Characteristics of the crypto-asset

The crypto-asset referred to in this white paper is named “KITE Token” (**KITE**). KITE is a fungible cryptographic token native to the KITE AI blockchain platform, a Layer-1 blockchain for agentic payments with verifiable identity and programmable governance (**KITE AI or Kite Network**). KITE is intended to serve as the native currency of KITE AI.

KITE is not pegged to any currency, is not redeemable, and is not intended as a medium of exchange outside the project ecosystem.

KITE functions as the network’s utility token and is used for staking purposes, reward distribution, and as a prerequisite for performing specific agent and service-related activities within the ecosystem. In particular, it has three core functions:

- It is used for protocol rewards. Initially, rewards are fully distributed in KITE, and over time they will gradually be paid in stablecoins instead.
- It is required for network participation. KITE users may perform services for the Kite network in exchange for rewards. There are three roles that a user can take: module owner, validator, and delegator. Any user who wishes to act as a module owner, validator or delegator must lock a specified amount of KITE as staked.
- It is a fungible and liquid asset that can be used as a means of exchange or store of value within the network. It enables smart contracts, interoperability between different modules, and coordination between network participants.

09 Utility token benefits and transferability

KITE grants access to network services by enabling the activation of specific operational roles within the protocol. Holders may stake KITE in order to activate network roles (validator, module owner or delegator) and may use KITE to enable participation in network coordination mechanisms, including interaction with modules, cross-module interoperability and smart contract transactions.

10 Key information about the offer to the public or admission to trading

The admission to trading is sought in order to enable KITE to be admitted to trading on platforms for crypto-assets in the EU.

The total supply is capped at 10 billion KITE. The circulating supply at launch will represent 27% of the total supply. Kite AI Ltd. expects the token to be admitted to trading on leading EU-based crypto-asset platforms that operate in full compliance with MiCAR.

Part A – Information about the offeror or the person seeking admission to trading

A.1 Name

Kite AI Ltd.

A.2 Legal Form

6EH6.

A.3 Registered Address

3rd Floor, Palm Grove House, Road Town,

Tortola, VG1110, British Virgin Islands.

A.4 Head Office

3rd Floor, Palm Grove House, Road Town,

Tortola, VG1110, British Virgin Islands.

A.5 Registration Date

2025-03-25.

A.6 Legal Entity Identifier

Not applicable.

A.7 Another Identifier Required Pursuant To Applicable National Law

Not applicable.

A.8 Contact Telephone Number

+1 (345) 324 3516

A.9 E-Mail Address

piano@horizonsglobal.io

A.10 Response Time (Days)

030.

A.11 Parent Company

KITE Foundation.

A.12 Members Of The Management Body

Full Name	Business Address	Function
Marc Piano	3rd Floor, Palm Grove House, Road Town, Tortola, VG1110, British Virgin Islands	Director

A.13 Business Activity

The core activity is the support of KITE and a Layer-1 blockchain for agentic payments with verifiable identity and programmable governance.

A.14 Parent Company Business Activity

The core activity is the support of KITE and a Layer-1 blockchain for agentic payments with verifiable identity and programmable governance.

A.15 Newly Established

True.

A.16 Financial Condition For The Past Three Years

Not applicable.

A.17 Financial Condition Since Registration

Kite Foundation and Kite AI Ltd. do not have a three-year financial reporting history. KITE AI is currently supported with financial backing from a software development company that has raised a total of USD$35,000,000.

PART B – INFORMATION ABOUT THE ISSUER, IF DIFFERENT FROM THE OFFEROR OR PERSON SEEKING ADMISSION TO TRADING

B.1 Issuer Different From Offeror Or Person Seeking Admission To Trading

False

B.2 Name

Not applicable.

B.3 Legal Form

Not applicable.

B.4 Registered Address

Not applicable.

B.5 Head Office

Not applicable.

B.6 Registration Date

Not applicable.

B.7 Legal Entity Identifier

Not applicable.

B.8 Another Identifier Required Pursuant To Applicable National Law

Not applicable.

B.9 Parent Company

Not applicable.

B.10 Members Of The Management Body

Not applicable.

B.11 Business Activity

Not applicable.

B.12 Parent Company Business Activity

Not applicable.

Part C – Information about the operator of the trading platform in cases where it draws up the crypto-asset white paper and information about other persons drawing the crypto-asset white paper pursuant to Article 6(1), second subparagraph, of Regulation (EU) 2023/1114

C.1 Name

Not applicable.

C.2 Legal Form

Not applicable.

C.3 Registered Address

Not applicable.

C.4 Head Office

Not applicable.

C.5 Registration Date

Not applicable.

C.6 Legal Entity Identifier

Not applicable.

C.7 Another Identifier Required Pursuant To Applicable National Law

Not applicable.

C.8 Parent Company

Not applicable.

C.9 Reason For Crypto-Asset White Paper Preparation

Not applicable.

C.10 Members Of The Management Body

Not applicable.

C.11 Operator Business Activity

Not applicable.

C.12 Parent Company Business Activity

Not applicable.

C.13 Other Persons Drawing Up The White Paper Under Article 6 (1) Second Subparagraph MiCAR

Not applicable.

C.14 Reason For Drawing Up The White Paper Under Article 6 (1) Second Subparagraph MiCAR

Not applicable.

PART D – INFORMATION ABOUT THE CRYPTO-ASSET PROJECT

D.1 Crypto-Asset Project Name

Kite AI

D.2 Crypto-Assets Name

Kite Token (KITE)

D.3 Abbreviation

KITE

D.4 Crypto-Asset Project Description

Kite AI is the first blockchain designed for agentic payments (i.e. payments initiated and executed autonomously by AI agents without human intervention).

The platform enables autonomous AI agents to operate with verifiable identity, programmable governance and seamless transactions.

Its purpose-built Layer-1 blockchain, together with the Agent Passport system, allows AI agents to act as first-class economic participants and to unlock new capabilities through composable interactions.

Kite AI was developed by AI and data infrastructure specialists from Databricks, Uber and UC Berkeley.

D.5 Details Of All Persons Involved In The Implementation Of The Crypto-Asset Project

Full Name	Business Address	Function
Marc Piano	3rd Floor, Palm Grove House, Road Town, Tortola, VG1110, British Virgin Islands	Director, Kite Foundation

D.6 Utility Token Classification

True.

D.7 Key Features Of Goods/Services For Utility Token Projects

KITE functions as the network's utility token and is used for staking purposes, reward distribution, and as a prerequisite for performing specific agent and service-related activities within the ecosystem (for further details see Sections F and G below).

D.8 Plans For The Token

The project is structured as a multi-layer infrastructure designed to enable autonomous AI agents to perform transactions, authorization flows and coordinated operations through blockchain-based mechanisms.

The architecture is composed of a base layer optimized for stablecoin payments and state channels, an application platform layer providing standardized interfaces for identity, authorization and micropayment execution, a programmable trust layer enabling cryptographic delegation and constraint enforcement, and a service and agent ecosystem layer designed to support discovery, interoperability and SLA-based interactions between agents and services.

The project has defined the core identity model based on a hierarchical delegation structure (user → agent → session), the constraint enforcement logic through pre-authorized spending rules and

session-scoped keys, and the micropayment framework using off-chain state channels with on-chain settlement. Development has also included compatibility with existing standards, enabling interaction of the agents across the ecosystems without adaptation.

The project architecture has been structured according to five core implementation pillars:

- a three-layer identity framework (user → agent → session), ensuring cryptographic delegation without direct key sharing;
- programmable governance and constraint enforcement, executed through Standing Intents and Delegation Tokens that mathematically bind agent spending and behavior;
- native compatibility with external agent protocols such as A2A, MCP and OAuth 2.1, enabling direct interoperability without middleware workarounds;
- a state-channel-based micropayment architecture, where agents exchange signed updates off-chain with a single settlement transaction on-chain, enabling sub-cent streaming payments for high-frequency interactions;
- a dedicated stablecoin payment lane, providing predictable settlement costs, guaranteed block space allocation for agent transactions and native support for batching, fee routing and commission logic.

Future technical extensions are planned in the areas of:

- verifiable credentials for agents (with cryptographic attestation about capabilities, certifications, and permissions);
- primitives for verifiable inference, proving not only what decision an agent produced but also linking it cryptographically to model parameters, prompts and data sources;
- portable reputation mechanisms across the agent network;
- trustless discovery layer where agents locate compatible services through verifiable attestations rather than centralized directories; and
- extended traceability layer so that every agent action pairs with attestations from relevant parties, creating complete audit trails for complex multi-party interactions.

D.9 Resource Allocation

Not applicable.

D.10 Planned Use Of Collected Funds Or Crypto-Assets

Not applicable.

PART E – INFORMATION ABOUT THE OFFER TO THE PUBLIC OF CRYPTO-ASSETS AND THEIR ADMISSION TO TRADING

E.1 Public Offering Or Admission To Trading

ATTR

E.2 Reasons For Public Offer Or Admission To Trading

The admission to trading is sought in order to enable KITE to be listed on leading EU-based crypto-asset platforms that operate in full compliance with MiCAR. The objective is to provide market participants with transparent and regulated access to the token, ensure liquidity within a regulated environment, and facilitate broader adoption and utility of the token across the network.

E.3 Fundraising Target

Not applicable.

E.4 Minimum Subscription Goals

Not applicable.

E.5 Maximum Subscription Goal

Not applicable.

E.6 Oversubscription Acceptance

False.

E.7 Oversubscription Allocation

Not applicable.

E.8 Issue Price

Not applicable.

E.9 Official Currency Or Any Other Crypto-Assets Determining The Issue Price

Not applicable.

E.10 Subscription Fee

Not applicable.

E.11 Offer Price Determination Method

Not applicable.

E.12 Total Number Of Offered/Traded Crypto-Assets

The total supply is capped at 10 billion KITE. The circulating supply at launch will represent 27% of the total supply. Kite AI Ltd. expects the token to be admitted to trading on leading EU-based crypto-asset platforms that operate in full compliance with MiCAR.

E.13 Targeted Holders

ALL.

E.14 Holder Restrictions

No. Token distribution will be carried out to holders in accordance with the laws and regulations applicable in each EU Member State.

E.15 Reimbursement Notice

KITE holders do not hold any right of withdrawal, reimbursement or refund.

E.16 Refund Mechanism

Not applicable.

E.17 Refund Timeline

Not applicable.

E.18 Offer Phases

Not applicable.

E.19 Early Purchase Discount

Not applicable.

E.20 Time-Limited Offer

False.

E.21 Subscription Period Beginning

Not applicable.

E.22 Subscription Period End

Not applicable.

E.23 Safeguarding Arrangements For Offered Funds/Crypto-Assets

Not applicable.

E.24 Payment Methods For Crypto-Asset Purchase

KITE can be purchased through the following methods:

- Centralized exchanges
- Decentralized exchanges

E.25 Value Transfer Methods For Reimbursement

Not applicable.

E.26 Right Of Withdrawal

Not applicable.

E.27 Transfer Of Purchased Crypto-Assets

KITE will be transferred to prospective holders' wallets via compatible blockchain networks.

E.28 Transfer Time Schedule

The transfer of KITE to holders will occur as follows:

- purchase via trading platform: KITE transfers processed upon purchase confirmation through exchanges. Delivery typically takes between 5 minutes and up to 15 minutes, depending on the particular exchange internal procedures;
- acquisition via KITE AI: KITE acquired through the KITE blockchain are transferred to the users' on-chain wallet immediately after the transaction is finalized.

E.29 Purchaser's Technical Requirements

Not applicable.

E.30 Crypto-Asset Service Provider (CASP) Name

Not applicable.

E.31 CASP Identifier

Not applicable.

E.32 Placement Form

NTAV.

E.33 Trading Platforms Name

Not applicable. Kite AI Ltd. intends for the token to be admitted to trading on leading EU-based crypto-asset platforms that operate in full compliance with MiCAR.

E.34 Trading Platforms Market Identifier Code (Mic)

Not applicable.

E.35 Trading Platforms Access

Prospective KITE holders must create a trading account on the trading platforms where KITE is listed, by completing the specific onboarding procedure requested by each platform.

E.36 Involved Costs

There are no costs involved in relation to the access of investors to the trading platforms.

E.37 Offer Expenses

Not applicable.

E.38 Conflicts Of Interest

There are no potential conflicts of interest of the persons involved in the admission to trading, arising in relation to the admission to trading.

E.39 Applicable Law

Not applicable.

E.40 Competent Court

Arbitration before Cayman International Mediation and Arbitration Centre (CI-MAC) seated in the Cayman-Islands.

PART F – INFORMATION ABOUT THE CRYPTO-ASSETS

F.1 Crypto-Asset Type

Crypto-asset other than asset-referenced tokens or e-money tokens (Utility Token).

F.2 Crypto-Asset Functionality

KITE functions as the network's utility token and is used for staking purposes, reward distribution, and as a prerequisite for performing specific agent and service-related activities within the ecosystem. In particular, it has three core functions:

- It is used for protocol rewards. Initially, rewards are fully distributed in KITE, and over time they will gradually be paid in stablecoins instead.
- It is required for network participation. KITE users may perform services for the Kite network in exchange for rewards. There are three roles that a user can take: module owner, validator, and delegator. Any user who wishes to act as a module owner, validator or delegator must lock a specified amount of KITE as stake.
- It is a fungible and liquid asset that can be used as a means of exchange or store of value within the network. It enables smart contracts, interoperability between different modules, and coordination between network participants.

As noted above, participants in the network may assume one of three roles - module owner, validator or delegator - each with distinct responsibilities and staking requirements. Their respective functions are outlined below.

MODULE OWNER (Stake Requirement: 30,000,000 tokens)

A module owner is a business entity which develops and operates a module on the KITE network. A module owner can be a single user who represents the module or a group of users who collectively authorize transactions via multi-signature wallet. The module owner is responsible for ensuring that the module expectations are met and that performance metrics are achieved.

VALIDATOR (Stake Requirement: 1,000,000 tokens)

KITE validators participate in a Proof-of-Stake consensus, locking up tokens to secure the network. Each validator must select a specific module to stake on, aligning their incentives with that module's performance. Validator responsibilities include network security, governance participation, and community engagement.

If validators fail to uphold their responsibilities or behave maliciously, they may be subject to slashing. Validators also risk slashing if their chosen module behaves maliciously or fails to meet network expectations. This system enforces accountability and ensures validators remain aligned with the network's long-term health and integrity.

DELEGATOR (Stake Requirement: 166,667 tokens)

KITE delegators lock up tokens to help secure the network and support their chosen module. Like validators, they must select a specific module to stake on, aligning their incentives with its success. While they do not perform any operational duties, delegators still share in rewards and may be slashed if the module they back behaves maliciously or fails to meet network expectations.

F.3 Planned Application Of Functionalities

The functionalities of KITE become effective from mainnet launch, as the token is required for staking to operate as a validator, module owner or delegator. During the initial phase, protocol rewards are distributed in KITE, with a gradual transition planned towards stablecoin-based rewards, while KITE will continue to serve as the mandatory staking and coordination asset for participation in the network.

F.4 Type Of White Paper

OTHR.

F.5 The Type Of Submission

NEWT.

F.6 Crypto-Asset Characteristics

KITE is the native utility token of the KITE network and functions as a staking and coordination asset within a Layer-1 Proof-of-Stake consensus infrastructure composed of the base chain and service-specific AI modules.

The total supply of KITE is capped at 10,000,000,000 (10 billion) units. KITE distribution follows a sub-linear emission schedule with four-year linear unlocks, and all initial allocations are subject to locking mechanisms to ensure controlled circulation.

KITE is required to access operational roles within the network, including module owner (30,000,000 KITE required), validator (1,000,000 KITE required) and delegator (166,667 KITE required). Staking is necessary to activate these roles, and the protocol enforces alignment by associating each validator and delegator with a specific module, with slashing mechanisms applied in cases of malicious behavior or performance failure.

KITE is fungible and transferable and may be used as a settlement asset for on-chain coordination, smart contract execution and cross-module interoperability, while transaction fees (gas) within the network are denominated and paid in stablecoins to ensure fee predictability and avoid volatility exposure.

Staking yields are targeted at approximately 4% annualized on staked value. In the initial phase, protocol rewards are distributed in KITE, with a progressive transition to stablecoin-denominated rewards over time, while the requirement to stake KITE to obtain operational permissions remains unchanged.

The token model integrates reward-based incentives and slashing-based penalties: module owners, validators and delegators receive rewards in proportion to their performance and contribution to the network, while those failing to meet protocol criteria for availability or integrity forfeit pending unlocks and/or future rewards.

F.7 Commercial Name Or Trading Name

Kite AI.

F.8 Website Of The Issuer

www.gokite.ai.

F.9 Starting Date Of Offer To The Public Or Admission To Trading

2025-11-17.

F.10 Publication Date

2025-11-14.

F.11 Any Other Services Provided By The Issuer

Agent Passport and agentic-payments infrastructure on Kite Layer 1.

F.12 Language Or Languages Of The White Paper

English.

F.13 Digital Token Identifier Code Used To Uniquely Identify The Crypto-Asset Or Each Of The Several Crypto Assets To Which The White Paper Relates, Where Available

Not applicable.

F.14 Functionally Fungible Group Digital Token Identifier, Where Available

Not applicable.

F.15 Voluntary Data Flag

False.

F.16 Personal Data Flag

True.

F.17 Lei Eligibility

True.

F.18 Home Member State

Italy.

F.19 Host Member States

Austria; Belgium; Bulgaria; Croatia; Cyprus; Czechia; Denmark; Estonia; Finland; France; Germany; Greece; Hungary; Ireland; Italy; Latvia; Lithuania; Luxembourg; Malta; Netherlands; Poland; Portugal; Romania; Slovakia; Slovenia; Spain; Sweden.

Part G – Information on the rights and obligations attached to the crypto-assets

G.1 Purchaser Rights And Obligations

The rights attached to the token are limited to the ability to stake and participate in network operations. The token does not grant ownership rights, voting rights over any legal entity, nor any redemption or repayment rights.

G.2 Exercise Of Rights And Obligation

The exercise of rights attached to KITE is limited to on-chain functional interactions permitted by the blockchain network. Holders may stake KITE in order to activate network roles (validator, module owner or delegator) and may use KITE to enable participation in network coordination mechanisms, including interaction with modules, smart contract activation and cross-module interoperability. All such rights are exercised exclusively through blockchain network.

G.3 Conditions For Modifications Of Rights And Obligations

Not applicable.

G.4 Future Public Offers

Not applicable.

G.5 Issuer Retained Crypto-Assets

Not applicable.

G.6 Utility Token Classification

True.

G.7 Key Features Of Goods/Services Of Utility Tokens

KITE grants access to network services by enabling the activation of specific operational roles within the protocol. Holders may stake KITE in order to activate network roles (validator, module owner or delegator) and may use KITE to enable participation in network coordination mechanisms, including interaction with modules, cross-module interoperability and smart contract transactions.

G.8 Utility Tokens Redemption

KITE does not include any redemption mechanism and cannot be redeemed for fiat currency or goods. KITE grants digital access to network services, without involving any redemption or reimbursement claim.

G.9 Non-Trading Request

True.

G.10 Crypto-Assets Purchase Or Sale Modalities

Not applicable.

G.11 Crypto-Assets Transfer Restrictions

KITE is subject to an unlock schedule which varies by holder groups. The team and investors are subject to a 1-year cliff with a 4 year unlock. 30% of the ecosystem and community allocation unlocks at launch with the remainder subject to a 4-year linear vesting period. 60% of the module allocation unlocks at launch with the remainder subject to a 4-year linear vesting period.

G.12 Supply Adjustment Protocols

False.

G.13 Supply Adjustment Mechanisms

Not applicable.

G.14 Token Value Protection Schemes

False.

G.15 Token Value Protection Schemes Description

Not applicable.

G.16 Compensation Schemes

False.

G.17 Compensation Schemes Description

Not applicable.

G.18 Applicable Law

British Virgin Islands law.

G.19 Competent Court

Arbitration before Cayman International Mediation and Arbitration Centre (CI-MAC) seated in the Cayman Islands.

Part H – Information on the underlying technology

H.1 Distributed Ledger Technology

The network operates on a custom Layer-1 Proof-of-Stake blockchain built using Avalanche Layer-1 technology, integrating AI service modules and a stablecoin-based gas model. The network introduces a dedicated payment primitive with a separate fast-lane mempool and fee market, where only whitelisted stablecoins are used both for value transfer and fee payment. The blockchain operates as the base layer for modules that offer AI services to KITE AI users, with state channels enabling high-frequency off-chain interactions anchored by on-chain settlement. The architecture maintains compatibility with existing standards such as A2A, MCP and OAuth 2.1, enabling agents to transact and coordinate natively across ecosystems.

H.2 Protocols And Technical Standards

The network follows an EVM-compatible architecture and uses the EIP-712 structured signing standard for transaction authorization. Protocol-level interoperability is implemented through native compatibility with A2A, MCP and OAuth 2.1 specifications. The payment lane supports memo fields including merchant category codes (MCC) and purpose codes. Authorization tokens follow a JWT structure bound to session keys through cryptographic signatures.

H.3 Technology Used

KITE AI's architecture represents a purposeful departure from traditional blockchain design. Instead of building another general-purpose chain that happens to support payments, every architectural decision optimizes for one goal: enabling autonomous agents to operate with mathematical safety guarantees.

The architecture stacks purpose-built layers from blockchain to applications, each solving specific challenges in the agent autonomy puzzle:

- Base Layer: An EVM-compatible L1 optimized for stablecoin payments, state channels, and settlement. Unlike general-purpose chains, every optimization targets agent transaction patterns.
- Platform Layer: Abstracts blockchain complexity through agent-ready APIs for identity, authorization, payments, and SLA enforcement. Developers interact with familiar patterns while the platform handles cryptographic proofs and on-chain settlement.
- Programmable Trust Layer: Introduces novel primitives including Kite Passport (cryptographic agent IDs), Agent SLAs (smart contract interaction templates), and compatibility bridges to A2A, MCP, and OAuth 2.1. This layer ensures agents can interact with both Web3 and traditional services seamlessly.
- Ecosystem Layer: Two interconnected marketplaces - an Application Marketplace for AI services and an Agents Ecosystem connected through standard protocols. Services register once and become discoverable by millions of agents.

Together, these layers let agents transact with predictable fees, enforce SLAs, and accumulate reputation across accounts and interactions. KITE AI's architecture achieves transformative results: dramatic reduction in payment costs through state channels, significant latency improvement via dedicated agent lanes, and complete elimination of credential management overhead through cryptographic identity. More critically, true agent autonomy is realized with programmable guarantee of constraint compliance.

H.4 Consensus Mechanism

The network operates as a Layer-1 Proof-of-Stake (PoS) blockchain, using the same PoS mechanism adopted by Avalanche Layer-1 architecture. Validators are required to stake KITE and participate in consensus to secure the network. Slashing applies in case of validator misbehavior or failure to meet network performance requirements, ensuring economic accountability and alignment with protocol objectives.

H.5 Incentive Mechanisms And Applicable Fees

The network applies a Proof-of-Stake incentive model. Validators, delegators and module operators receive protocol rewards, initially distributed in KITE and transitioning progressively to stablecoins, targeting approximately 4% annual reward on staked value. Validator operations receive a minimal base reward to cover L1 operating costs. Transaction fees (gas) are paid exclusively in whitelisted stablecoins rather than in KITE, ensuring fee stability and predictable cost exposure.

H.6 Use Of Distributed Ledger Technology

True.

H.7 DLT Functionality Description

The distributed ledger operates as follows:

Three-Layer Identity Architecture

KITE AI introduces the first hierarchical identity model that separates user (root authority), agent (delegated authority), and session (ephemeral authority) identities. Each agent receives its own deterministic address derived from the user's wallet using BIP-32, while session keys are completely random and expire after use. This defense-in-depth architecture ensures graduated security: compromising a session affects only one transaction; compromising an agent remains bounded by user-imposed constraints; and user keys secured in local enclaves inaccessible to any external third parties, which are thus highly unlikely to be compromised, represent the only point of potential unbounded loss. While funds remain compartmentalized for security, reputation flows globally across the system. Every transaction and interaction contribute to a unified reputation score, establishing a cryptographic root of trust that spans users, agents, and services throughout the KITE AI platform.

Programmable Governance Beyond Smart Contracts

While smart contracts enable programmable money, agents require compositional rules that span multiple services. KITE AI implements a unified smart contract account model where users own a single on-chain account holding shared funds. Multiple verified agents operate through session keys with cryptographically enforced spending rules: "ChatGPT limit $10,000/month, Cursor limit $2,000/month, other agents limit $500/month." Rules can be temporal (e.g., increase limits over time), conditional (e.g., reduce limits if volatility spikes), and hierarchical (cascade through delegation levels). These are not policies but programmatically enforced boundaries.

Agent-Native Payment Rails with State Channels

Beyond stablecoins and payments-first blockchains, the real revolution goes deeper. KITE AI creates agent-first transaction types. Beyond simple transfers, KITE AI implements programmable micropayment channels optimized for agent patterns. Instead of the traditional over-complicated multi-party card rails of authenticate, request, pay, wait, and verify, payments are instantly settled during agent interaction within the same channel. Two on-chain transactions (open and close) enable thousands of off-chain signed updates, achieving sub-hundred-millisecond latency at $1 per million requests. This architectural inversion, treating per-request and streaming micropayments as first-class behaviors with sub-cent precision and instant finality, unlocks agent-native economics which were previously impossible.

Chain of Trust with Cryptographic Proof

Every action in the KITE AI system creates a cryptographically verifiable audit trail, establishing an immutable chain of custody from user to agent to service to outcome. This transforms agent operations from black box mysteries into transparent, provable sequences where every decision can be verified and every constraint can be enforced mathematically.

The chain of trust operates through three critical principles:

No Direct Key Access: The revolution starts with a simple insight: agents should never touch private keys directly. Instead, each operation receives a one time, task scoped session key with surgical precision permissions. An agent authorized to purchase data feeds receives a key valid only for specific providers, exact amounts, and narrow time windows. When the task completes or the window

expires, the key becomes cryptographically useless. Even total compromise affects only that single operation.

Fine-Grained Task Authorization: Permissions are scoped at the task level, not the agent level. This granular authorization extends beyond simple spending limits. Permissions scope down to individual API endpoints, specific data types, and conditional triggers. An agent authorized to “purchase data feeds” receives a session key valid only for strictly enforced specific data providers, amounts, and time windows. This granularity makes broad compromises mathematically impossible.

Reputation Without Identity Leakage: The relationship between users and agents creates a fascinating paradox: shared reputation with independent identity. Each user and corresponding agent accumulate their own track record of successful operations, building trust through performance. For each user-agent binding, the reputation of the duo is inherited as a combination of the user’s reputation and the agent’s reputation, and the operation outcome affects both the reputation of the user and the agent, where the cryptographic binding to the controlling user ensures accountability flows upward. Services know that behind every agent stands a real user with real stakes, even when that user's identity remains private. Users can selectively disclose ownership when beneficial, but privacy remains the default.

This architecture enables something unprecedented: complete transparency without sacrificing privacy. Regulators can verify that agents operated within legal boundaries. Services can confirm payment authorization. Users can audit every action their agents took. Yet sensitive business logic, trading strategies, and personal information remain encrypted. The system proves compliance without revealing secrets.

Sovereignty through Separation

KITE AI rigorously separates decentralized asset management from developer services, creating an architecture where security and usability reinforce rather than compromise each other.

Decentralized Asset Model: The decentralized asset model ensures complete user sovereignty. All funds remain in self-custodial wallets controlled by smart contracts that even KITE AI cannot access. Users maintain exclusive control over their digital wealth while agents operate within mathematically enforced boundaries. Users maintain independent access to their funds through standard blockchain interfaces, regardless of KITE AI’s availability. This is not a promise or a policy. It is a cryptographic guarantee.

KITE AI Platform Services: Meanwhile, the KITE AI platform provides the abstraction layer that makes agent operations practical. APIs handle complex cryptographic operations, protocol translations, and cross chain interactions. Developers work with familiar patterns while the platform manages key derivation, session management, and constraint compilation. The platform never touches assets directly. It simply provides the tools to make asset operations seamless.

This separation solves a fundamental tension in blockchain systems. Pure decentralization often means terrible user experience. Pure centralization means unacceptable risk of single point failure and user assets compromise. KITE AI achieves both security and usability by separating concerns architecturally. Assets remain decentralized and sovereign. Services remain centralized and optimized. Users get the best of both worlds without compromise.

The implications extend beyond technical architecture. Regulators can audit the platform without accessing user funds. Developers can build sophisticated applications without managing private keys. Users can delegate complex operations without surrendering control. Each stakeholder operates in their domain of expertise while the system maintains security guarantees across all layers.

Zero Migration Friction to Existing Standards

Rather than creating another isolated protocol that fragments the ecosystem further, KITE AI embraces existing standards as first principles. This is not grudging compatibility or minimal compliance. It is not architectural integration that makes KITE AI agents native citizens of multiple protocols simultaneously.

Google's A2A protocol enables direct agent coordination across platforms. KITE AI agents speak A2A fluently, coordinating with agents from any ecosystem using established communication patterns.

When a KITE AI agent needs to coordinate with a Google agent to complete a complex workflow, they communicate directly without translation layers or compatibility bridges.

Anthropic's MCP ensures model interoperability across the entire LLM ecosystem. Claude, GPT, and other emerging models interact through the same protocol layer, making model choice a business decision rather than a technical constraint. A KITE AI agent can leverage Claude for reasoning, GPT for generation, and specialized models for domain tasks, all within a single workflow.

OAuth 2.1 compatibility means compatibility with human-centric services and existing login/payment flows, thus existing services don't need to rebuild their authentication systems. A service supporting OAuth today can accept KITE AI agents tomorrow with minimal changes. This backward compatibility creates a gradual migration path for developers where they can adopt KITE AI infrastructure without abandoning the broader internet and enterprise ecosystems.

X402 standard compatibility enables agent-native payments, while Agent Payment Protocol integration specifically optimizes for stablecoin payments on KITE AI, ensuring KITE AI remains compatible with future developments in the agent economy. As new standards emerge, KITE AI's architecture allows rapid integration without breaking existing functionality.

These standards embrace transforming adoption dynamics. Developers don't choose between KITE AI and their existing stack. They add Kite to enhance what already works. Services don't rebuild for agents. They extend existing APIs with agent awareness.

Terminology and Core Concepts

Understanding KITE AI requires grasping the fundamental building blocks that enable autonomous agent operations. These are not merely technical definitions but the architectural choices that transform AI agents from sophisticated chatbots into economic actors.

Core Entities

The KITE AI ecosystem operates through four fundamental entity types, each playing a critical role in the agent economy.

User: The human principal who owns and controls a fleet of AI agents. More than account holders, users represent the bridge between traditional legal frameworks and autonomous systems. They delegate specific capabilities to agents while maintaining ultimate authority, manage master wallets that serve as the root of cryptographic trust, and set global policies that cascade through all their agents. Users remain the legally responsible entities, ensuring that agent autonomy never means absence of accountability.

Agent: A personal AI assistant or autonomous program acting on behalf of a user. These are not simple scripts or API wrappers but sophisticated entities that execute complex tasks, interact with multiple services simultaneously, and handle real money within cryptographically enforced boundaries. Each agent maintains its own wallet, accumulates its own reputation, and operates under its own governance policies. Yet they remain mathematically bound to their controlling user through BIP-32 derivation, creating provable ownership without key exposure.

Service: Any external offering that agents interact with to accomplish tasks. Services span from data APIs providing real-time market feeds to GPU providers offering inference compute, from SaaS applications exposing business logic to other agents offering specialized capabilities. Services can integrate through multiple protocols: MCP for model interactions, A2A for agent coordination, or traditional OAuth for legacy compatibility. Each service maintains sovereignty over its access policies while participating in the global agent marketplace.

Merchant/Provider: The businesses, developers, or infrastructure operators who make services discoverable and consumable by millions of agents. Providers do not just list APIs; they define SLAs with automatic penalties, establish reputation through verifiable performance, and participate in programmable commerce where every interaction becomes a micropayment. The merchant role transforms B2B services from manual integration nightmares into plug-and-play agent resources.

Identity and Trust Infrastructure

Identity in the agent economy transcends usernames and passwords. It requires cryptographic proof, verifiable delegation, and portable reputation.

KITE AI *Passport*: The cryptographic identity card that makes agent operations possible. Unlike traditional credentials that merely authenticate, the passport creates a complete trust chain from user to agent to action. It binds to existing identities like Gmail or Twitter through cryptographic proofs, enabling users to leverage their existing digital presence. The passport contains not just identity but capabilities: what an agent can do, how much it can spend, which services it can access. Most critically, it enables selective disclosure. An agent can prove it belongs to a verified human without revealing which human, preserving privacy while maintaining accountability.

Decentralized Identifier (DID): A globally unique, cryptographically verifiable identifier that establishes immutable binding between agents and users. DIDs are not random strings but structured identifiers that encode relationships. A user might have did:kite:alice.eth while her trading agent has did:kite:alice.eth/chatgpt/portfolio-manager-v1. This hierarchy makes authority chains instantly verifiable. Any service can cryptographically confirm that a session belongs to an agent, that agent belongs to a user, and that user authorized the current operation. No central authority needed, no API calls required, just mathematical proof.

Verifiable Credentials (VCs): Cryptographic attestations that prove specific capabilities or authorizations. A VC might certify that an agent passed compliance training, holds a trading license, or maintains a reputation above a threshold. These are not self-declared claims but cryptographically signed attestations from trusted authorities. VCs enable fine-grained access control where services can require specific proofs before granting access. An exchange might require KYC verification, a data provider might require payment history, a compute cluster might require reputation scores. The beauty lies in composability: agents accumulate credentials over time, building portable proof of capabilities.

Proof Chains/Audit Trails: Every agent action creates an immutable, tamper-evident log anchored to the blockchain. These are not just logs but cryptographic proof of what occurred. They establish complete lineage from user authorization through agent decision to final outcome. When disputes arise, proof chains provide indisputable evidence. When regulators investigate, they see complete transparency. When users audit their agents, they verify every action. Traditional systems rely on logs that can be altered or deleted. Proof chains provide mathematical certainty about historical events.

Wallets and Payment Architecture

The agent economy demands sophisticated wallet structures that enable programmable control without sacrificing security.

EOA Wallet (Externally Owned Account): The traditional blockchain wallet controlled by a private key. In KITE AI's architecture, users maintain master EOA wallets that serve as the root of authority. These wallets live in secure enclaves, hardware security modules, or user devices, never exposed to agents or services. The EOA signs the initial authorizations that delegate specific powers to agent operations, creating the foundation of the trust chain. While simple in concept, EOAs become powerful when combined with smart contract delegation.

AA Wallet (Smart Contract Account): The revolutionary advance that makes agent payments possible. Account Abstraction wallets are not just addresses but programmable accounts with built-in logic. AA enables developers to write smart contract code that governs spending, bundles transaction execution, and allows third parties to pay gas fees. These smart contracts interact with multiple accounts, perform cross-program communications, and integrate customized logic. In KITE AI's model, the user owns a single on-chain AA account holding shared funds in stablecoins. Multiple agents operate this account via session keys, but only within their authorized limits. One treasury, multiple operators, perfect isolation.

Embedded Wallets: Self-custodial wallets integrated directly into applications, abstracting complexity while maintaining sovereignty. Users don't manage seed phrases or private keys, yet they maintain complete control over funds. Embedded wallets enable one-click agent authorization, automatic session management, and transparent fund flows. They make blockchain invisible to users who think in dollars, not tokens, while preserving the cryptographic guarantees that make agent operations safe.

Agent Payment Protocol: A comprehensive system that enables payment flows impossible with traditional infrastructure. This protocol supports micropayments down to fractions of cents, streaming payments that flow continuously based on usage, pay-per-inference models where every API call carries value, and conditional payments that release based on performance. Credit cards charge

$0.30 minimum, making micropayments absurd. ACH takes days to settle, making real-time payments impossible. The Agent Payment Protocol enables instant, global, programmable value transfer at machine speed.

On/Off-Ramp API: The bridge between traditional finance and the agent economy. Through integration with providers like PayPal and banking partners, users fund agent wallets with credit cards while merchants withdraw earnings to bank accounts. The ramp handles compliance, fraud prevention, and currency conversion invisibly. Users never need to understand blockchain; they just see dollars in and dollars out. This abstraction layer makes agent payments accessible to billions who will never own crypto but need agent services.

Governance and Safety Mechanisms

Autonomous agents require sophisticated governance that ensures safety without sacrificing capability.

SLA (Service-Level Agreement) Contracts: Smart contracts that transform vague service promises into mathematically enforced guarantees. Unlike traditional SLAs that rely on legal enforcement, these contracts automatically execute penalties and rewards. An SLA might specify 99.9% uptime with automatic pro-rata refunds for downtime, response times under 100ms with tiered pricing based on performance, or data accuracy requirements with slashing for errors. These programmable agreements create trust through code rather than courts.

Programmable Trust/Intent-Based Authorization: Users express their intentions through mathematical constraints that compile to blockchain enforcement. Instead of hoping agents respect policies, the system ensures they cannot violate them. Intents can specify spending caps that cannot be exceeded even if the agent tries, temporal windows outside which operations automatically fail, whitelisted merchants or blacklisted categories enforced at protocol level, and complex conditional logic like "if volatility exceeds 20%, reduce limits by half." These intents automatically expire, preventing forgotten authorizations from becoming vulnerabilities. The user's intent becomes immutable law.

Session Keys/Ephemeral Keys: Temporary cryptographic keys that implement zero-trust session management. Generated for each agent task, these keys are completely random, never derived from permanent keys, ensuring perfect forward secrecy. A session key might authorize "transfer maximum $10 to providers A, B, or C for data feeds between 2:00 PM and 2:05 PM today." The key executes its authorized operation then becomes cryptographically void forever. Even total session compromise affects only one operation for minutes with bounded value. This architecture prevents the cascading failures that plague traditional API key systems where one breach means total compromise.

Reputation System: Trust scores that accumulate based on verifiable behavior rather than self-reported ratings. Every successful payment increases reputation. Every failed delivery decreases it. Every policy violation triggers penalties. But unlike traditional ratings that can be gamed, KITE AI's reputation derives from cryptographic proofs of actual behavior. High reputation agents access better rates, higher limits, and premium services. Low reputation agents face restrictions and additional verification. Reputation becomes portable across services, solving the cold-start problem where new relationships begin from zero trust. An agent with proven history on one platform can present verifiable credentials to new services, bootstrapping trust through cryptographic proof rather than promises.

Detailed Design

The Three Layer Identity Imperative

Human payment systems recognize only two entities: the payer and the payee. This binary model suffices when humans initiate every transaction, evaluate every risk, and maintain control over every key. Agent systems shatter this assumption. They require a third layer that enables delegation without the catastrophic vulnerability of key sharing.

User Identity (Root Authority)

The user maintains ultimate control as the cryptographic root of trust. Their private keys live in secure enclaves, hardware security modules, or protected device storage, never exposed to agents, services, or even the KITE AI platform itself. Users can instantly revoke all delegated permissions with a single transaction, set global constraints that cascade through all agents, and monitor every operation

through immutable proof chains. This is not theoretical control through terms of service; it's mathematical control through cryptographic enforcement.

Agent Identity (Delegated Authority)

Each AI agent receives its own deterministic address mathematically derived from the user's wallet using BIP-32 hierarchical key derivation. When you create a ChatGPT agent for portfolio management, it gets address 0x891h42Kk9634C0532925a3b844Bc9e7595f0eB8C, provably linked to your wallet yet cryptographically isolated. This address serves as the agent's on chain identity, enabling it to authorize sessions for spending operations.

The mathematical derivation creates powerful properties. Anyone can verify the agent belongs to you through cryptographic proof, yet the agent cannot reverse the derivation to access your private key. The agent maintains its own reputation score, coordinates with other agents autonomously, and operates within user defined boundaries that the blockchain enforces absolutely. Even total agent compromise remains bounded by smart contract constraints.

Session Identity (Ephemeral Authority)

For each task execution, the system generates a completely random session key with address 0x333n88Pq5544D0643036b4c955Cc8f8706g1dD9E. These keys are never derived from wallet or agent keys, ensuring perfect forward secrecy. They're single-use authorization tokens that execute specific actions without exposing permanent credentials.

The session key generation happens entirely locally, without server communication or private key transmission. Once generated, a DID session registers in the agent network, self-containing the proof of authority, chain of trust, and validity period. The session validates through its time window then becomes permanently invalid. Even quantum computers cannot resurrect expired sessions.

This three-tier model provides defense in depth that payments-first chains lack. Compromising a session affects only one transaction. Compromising an agent is limited by user-imposed constraints. Only user key compromise enables unbounded loss, and secure enclave protection makes this nearly impossible.

Identity Resolution Through Standards

KITE AI extends ENS standards to claim agent ownership with human readable identifiers:

- User ID: did:kite:alice.eth
- Agent ID: did:kite:alice.eth/chatgpt/portfolio-manager-v1

Public resolvers enable instant verification:

- GetAgent (AgentID) → AgentID, AgentDomain, AgentAddress
- ResolveAgentByDomain (AgentDomain) → AgentID, AgentDomain, AgentAddress
- ResolveAgentByAddress (AgentAddress) → AgentID, AgentDomain, AgentAddress
- GetAgentBySession (SessionID) → AgentID, AgentDomain, AgentAddress, SessionInfo

Any service can verify the complete authority chain without contacting KITE AI or the user, enabling permissionless interoperability.

Programmable Governance and Money

Smart contracts revolutionized programmable money, but agents require programmable governance that spans services, evolves over time, and responds to changing conditions. These are not simple spending limits; they are sophisticated control systems that make agent autonomy safe.

Compositional Rules Across Services

Constraints combine through boolean logic to create sophisticated policies. "Total spending across all platforms < $1000/day AND no single transaction > $100 AND only verified providers". These rules compile to smart contract code that evaluates atomically. Agents cannot circumvent limits by splitting transactions or distributing operations across services. The blockchain becomes the ultimate arbiter, enforcing rules with mathematical certainty.

Temporal Evolution of Trust

Static limits ignore the reality that relationships build over time. KITE AI implements progressive trust: “Start with $10/day limit, increase by $10 weekly up to $100/day after trust is established.” The blockchain automatically adjusts limits based on time and behaviour. No manual intervention, no forgotten updates, just programmatic trust evolution that reflects actual agent performance.

Conditional Response to External Signals

Markets change, threats emerge, opportunities appear. Agent constraints must adapt: "If volatility > 20%, reduce trading limit by 50%." Oracle networks feed real time signals into smart contracts, triggering automatic adjustments. When markets panic, agents automatically become more conservative. When security breaches occur, new authorizations freeze. The system responds to threats faster than humans can react.

Hierarchical Cascade Through Organizations

Enterprise deployments require nested governance: “ChatGPT agent limit $10,000/month, Cursor limit $2,000/month, other agents limit $500/month.” Constraints cascade through delegation levels, with child limits automatically bounded by parent limits. A department cannot exceed its division's budget. An agent cannot exceed its tier's allocation. Organizational policies propagate cryptographically.

The Unified Account Model

KITE AI designs a unified smart contract account model that elegantly balances simplicity with control. The user (EOA 0xUser) owns a single on-chain account holding shared funds in USDC or pyUSD. Multiple verified agents—Claude, ChatGPT, Cursor—each operate this account via their own session keys (0xSession01 through 03), but only through sessions that enforce rules and quotas.

When an agent executes a task, spending comes from the shared pool within its session allowance, and the account pays merchants programmatically. Result: one treasury, per session risk isolation, and fine grained, auditable control across all the user's agents. No fund fragmentation, no complex reconciliation, just elegant unified management.

Spending Rules versus Policy

The system distinguishes between on chain rules and off chain policies based on their evaluation requirements:

Dimension	Spending Rules	Policy
Scope	Asset or stablecoin related, leveraging programmable money	Full control and flexibility for complex logic
Evaluation	Entirely on chain through smart contracts, ensuring transparency	Securely off chain in user's local or KITE AI’s TEE
Interoperability	Integrates with any on chain account, fully decentralized	Platform specific but can integrate third party systems
Use Cases	Spending limits, rolling windows	Session TTL, categories, recipient lists

Session Key Implementation with Smart Contracts

The AA wallet smart contract implements session keys through a plugin architecture.

The user signs authorization transactions with their EOA wallet, representing cryptographic intent. The transaction packages as a UserOperation and sends to a bundler for on chain execution. Sessions expire automatically when validUntil timestamps pass, or users can actively revoke by calling removeSessionKeyPermission().

Programmable Escrow Contracts

Beyond direct payments, KITE AI implements programmable escrow that places smart contracts between agents and merchants. Buyers sign payment intents, funds authorize into escrow with expiries, and later capture partially or fully based on outcomes. The protocolized contract exposes the complete commerce lifecycle, including authorize, capture, charge, void, reclaim, refund, etc., while remaining non-custodial and permissionless.

Additional "Operators" can submit transactions and cover gas but remain cryptographically constrained by the payer's signed hash. Standards like ERC-3009 enable signature based, gasless pre-approval. Because its code, escrow extends with business rules like revenue splits, token swaps, or privacy modules, composing payments with other smart contracts. Every transaction becomes programmable, auditable, and reversible.

Compatible with Agent Protocols

Modern agents don't operate in isolation. They integrate with Google's A2A protocol, Anthropic's MCP, and enterprise OAuth systems. Payment infrastructure must provide native bridges to these ecosystems, not awkward adapters.

Protocol Translation

Seamless mapping between agent communication standards enables universal interoperability. A KITE AI agent speaks A2A to Google agents, MCP to Claude, OAuth to enterprise services—all through the same identity and payment rails. This is not middleware translation; it's native multilingual capability.

Identity Federation

Unified authentication eliminates the N×M credential problem. Users authenticate once with existing providers; agents inherit permissions across all connected services. No duplicate credentials, no synchronization nightmares, just elegant identity flow.

Compliance Frameworks

Built in support for regulatory requirements across jurisdictions ensures global operability. GDPR in Europe, CCPA in California, data residency requirements in China—all handled automatically through configurable compliance modules.

The integration happens through extended protocol definitions, combined with the A2A protocol.

This architecture ensures KITE AI agents participate as first class citizens in every major agent protocol while maintaining their unique capabilities.

Programmable Micropayment Channels

Direct money transfers work for occasional payments but fail catastrophically for agent patterns. Agents make thousands of tiny, frequent calls—messages, API requests, inference queries. Traditional payments would cost more in fees than value transferred. State channels transform this impossible economy into a profitable reality.

A state channel enables two parties to transact off chain with just two on chain transactions: open to lock funds and close to settle. Between these anchors, parties exchange thousands of signed updates instantly. This yields high throughput, low latency, and fees amortized across millions of interactions. AI inference requiring sub hundred millisecond latency at $1 per million requests becomes economically viable only through programmable micropayment channels.

Channel Variants for Every Pattern

KITE AI implements multiple channel types optimized for different interaction patterns:

Unidirectional channels flow value from user to merchant for simple metering. Perfect for API consumption, data feeds, and inference requests where value flows one direction.

Bidirectional channels enable refunds, credits, and two-way value exchange. Services can pay agents for data, agents can receive rebates, errors trigger automatic refunds.

Programmable escrow channels embed custom logic in state transitions. EVM developers write arbitrary rules: conditional releases, multi-party splits, time locked vesting. The channel becomes a mini smart contract.

Virtual channels route value through intermediaries without new on chain contracts. Agent A pays Agent C through hub B, enabling network effects without setup overhead.

Privacy preserving channels keep interactions confidential. Only channel open and close appear on chain. Thousands of micropayments remain private between participants, protecting competitive intelligence and usage patterns.

Net effect: every message becomes a payment with sub-cent precision, instant finality, and minimal on chain footprint. Perfect for agent pay per use and streaming economics.

How State Channel Limitations Become Advantages in Agent Economy

State channels carry known limitations from their Ethereum origins. But agent use patterns transform these bugs into features.

Open/Close Channel Overhead

Traditional users rarely open and close channels, making setup costs prohibitive. But agents send hundreds of inferences to the same service over minutes. Setup costs amortize perfectly across concentrated bursts of activity. The "overhead" becomes negligible.

Liveness Assumption

Channels assume participants stay online to contest disputes. Human users go offline unpredictably. But professional services with reputation at stake won't risk fraud for micropayments. The game theory strongly favors honest behavior.

Griefing Attacks

Malicious participants could force expensive dispute responses. But agents and services maintain reputation scores. Griefing destroys reputation for minimal gain. The attack becomes economically irrational.

Predefined Participant Sets

Channel membership cannot change without closing. But agent interactions naturally have fixed participants. User, agent, service—the relationships are determined at task initiation.

Parallel Transaction Processing

Channels process updates sequentially, working best for turn based applications. Agent interactions are inherently turn based: request, response, request. The limitation perfectly matches the use case.

Dedicated Stablecoin Payment Lane

General purpose blockchains treat payments as just another transaction type, competing for block space with NFT mints, DeFi swaps, and smart contract deployments. This creates an economic absurdity: critical payment operations wait behind speculative trades while fees swing wildly based on jpeg popularity. A simple $10 payment might cost $50 in gas during congestion, priced in volatile tokens that change value between initiation and settlement. Bank style metadata gets bolted on through fragile off chain systems. Privacy remains an afterthought. The result is a payment system that fails at being a payment system.

KITE AI introduces a fundamental architectural innovation: a dedicated payment primitive with its own fast lane mempool and fee market. This is not optimization of existing systems; it's recognition that payments deserve first class infrastructure. Only whitelisted stablecoins can serve as both transfer

assets and fee payment, guaranteeing predictable costs denominated in real money. The system cleanly distinguishes payments (which can include commissions) from transfers, implements bank grade memo fields natively, enforces compliance policies on chain, and supports batching and state channel micropayments as core primitives.

Architectural Principles

The payment lane operates on five foundational principles that transform blockchain payments from afterthoughts into primary citizens:

Independent Fast Lane: A separate mempool and fee market ensures payments never compete with general transactions for block space. Each block reserves guaranteed quota for payment operations.

Stablecoin Exclusivity: Both value and fees must use whitelisted stablecoins. This eliminates the absurdity of paying transaction fees in volatile tokens that might double in price during confirmation.

Payment Semantics: The protocol distinguishes payments (supporting commissions, reconciliation data, and business logic) from simple transfers (peer to peer value movement).

Banking Alignment: Native support for invoice numbers, merchant category codes, purchase order references, and purpose codes brings blockchain payments to banking standards.

Agent Native Capabilities: Intent references, mandates, carts, and micropayment channels layer seamlessly on the payment primitive, enabling sophisticated agent operations.

Core Payment Operations

The payment lane implements five primary operations that cover the complete spectrum of payment needs:

sendPayment

The primary payment operation routes through the dedicated Payment Mempool with rich semantic support:

sendTransfer

Plain transfers without commission support use the Payment Mempool but with simplified semantics. Same fee and policy rules apply, but no commission or complex reconciliation fields.

estimatePaymentFee

Returns base_fee, priority_fee_range, commission_estimate, and expected_slo_ms for fast lane latency SLO. Enables cost prediction before submission.

sendPaymentBatch

Batch payments compress multiple transfers into single transactions. On chain compact encoding uses recipient sorting and amount compression. BLS aggregate signatures dramatically reduce gas costs.

Key features:

- Per item sub memos or shared batch memo;
- Optional atomicity (all succeed or all revert) for payroll use cases;
- Oversized batches automatically shard with Merkle proofs for parallel verification.

Payment Channels

State channels integrate natively with the payment lane through three operations:

- openPaymentChannel: Lock funds with guaranteed SLO;
- commitPaymentChannel: Update channel state off chain;
- closePaymentChannel: Final settlement through payment lane.

Enables "each message equals a payment" economics. Account for approximately $0.000001 per interaction inside channels, amortizing fees across millions of micropayments.

Node and Consensus Architecture

The payment lane requires specialized node infrastructure to maintain performance and security guarantees.

Payment Mempool

The Payment Mempool operates as an independent queue with sophisticated prioritization:

Structure and Quotas: Each block reserves X% exclusively for payment transactions. This guaranteed quota ensures payments never starve during general network congestion.

Queue Management: Four sub queues handle different operation types:

- Payment queue for standard payments with commission;
- Transfer queue for simple value movements;
- Batch queue for multi recipient operations;
- Channel queue for state channel settlements.

Priority Algorithm: Transactions order by earliest deadline first, then fee per byte, then arrival time (FIFO). This ensures time sensitive payments process reliably.

Admission Filtering: Pre mempool checks reject invalid transactions immediately:

- Non-whitelisted stablecoins blocked;
- Denylisted recipients rejected;
- Invalid signatures discarded;
- Expired deadlines filtered;
- Insufficient fees denied.

Fee Market

The payment lane implements an independent fee market optimized for stable, predictable costs:

Separate EIP-1559 Implementation: Base fee for payments adjusts independently from general transactions. Congestion in DeFi doesn't affect payment costs. The base_fee_payment variable tracks payment specific congestion.

Stablecoin Settlement Options:

- Option 1: Validators receive stablecoins directly. Simplest approach requiring minimal protocol changes;
- Option 2: Protocol enshrined AMM automatically swaps stablecoins to native staking assets. Maintains existing validator economics.

Commission Processing: Protocol governance caps commission basis points. Commissions extract from fee addon rather than principal amount, keeping reconciliation clean.

Compliance Infrastructure

Compliance operates through the PolicyRegistry system contract maintaining three registries:

StablecoinWhitelist: Maps token addresses to configuration.

RecipientAllowlist/Denylist: Address sets with verifiable proof support. Entries can be individual addresses or Merkle roots for efficient batch updates.

RuleSet Engine: Composes multiple compliance rules:

- Geographic restrictions by IP or declared location;
- Time windows for permitted operations;
- Transaction limits by amount or frequency;
- Category restrictions for merchant types.

Travel Rule Compliance: The protocol aligns with FATF Travel Rule and IVMS101 standards. Stores only reference hashes on chain. Plaintext PII lives off chain with verifiable assertions linking for audit.

Privacy Mechanisms

Three privacy modes serve different use case requirements:

None Mode: Full transparency for public payments. All details visible on chain. Suitable for public goods funding or transparent operations.

Stealth Mode: Recipients generate one time addresses for each payment. View keys enable selective disclosure. Balances and transaction graphs remain private while maintaining auditability.

Shielded Mode: Zero knowledge proofs hide amounts and parties. Stablecoin shielded pool processes private transfers. Compliance maintained through selective disclosure mechanisms.

Memo Tiering: Two tier memo structure separates public and private data:

- Public tier: Merchant visible data for reconciliation;
- Private tier: Encrypted data visible only to view key holders.

Semantic Payment Types

The protocol maintains clear distinction between payment types for accurate economic measurement:

Payments include:

- Commission splits for platforms and aggregators;
- Reconciliation metadata for accounting systems;
- Intent and mandate references for agent operations;
- Order and cart bindings for commerce.

Transfers provide:

- Simple peer to peer value movement;
- No commission overhead;
- Minimal metadata requirements;
- Maximum throughput optimization.

Nodes, explorers, and indexers differentiate these types to calculate accurate metrics like Gross Merchandise Value (GMV) and platform take rates.

Optimization Strategies

Batch Processing

Large scale operations benefit from sophisticated batching:

Deduplication and Sorting: Remove duplicate recipients and sort for optimal compression. Amounts compress using delta encoding from previous values.

Sharding Strategy: Large batches split into shards with Merkle proof linkage. Nodes verify shards in parallel, maintaining throughput.

Failure Handling Options:

- Partial Tolerance: Skip failed recipients, process successful ones;
- Atomic Mode: All succeed or all revert for critical operations.

Export Formats: Indexers generate CSV and Parquet files for one click reconciliation. Standard accounting software imports directly.

Developer Integration

EIP-712 Signing Standard: Separate type definitions for Payment, Transfer, and BatchItem prevent cross domain replay attacks.

SDK Methods:

- signPayment(): Generate payment signatures;
- signBatch(): Create batch payment signatures;
- simulatePayment(): Test execution without submission;
- watchPaymentReceipt(): Monitor confirmation status;
- exportReconciliation(): Generate accounting reports.

Agent Native Features

The payment lane provides primitives that enable sophisticated agent behaviors:

Intent and Mandate System: Every payment references its authorizing intent. Smart contracts verify payments fall within authorized parameters. Limits, categories, and time windows enforce automatically.

Cart Integration: Payments reference cart snapshots through memo fields. Receipts include content fingerprints. Disputes resolve through cryptographic proof of cart contents at payment time.

Micropayment Streaming: Agent to agent interactions carry payment vouchers in every message. Settlements trigger every N messages or T time interval. Automatic fallback to on chain settlement handles channel failures.

Programmable Settlement: Templates define complex fund distribution:

- Platform fees extract automatically;
- Merchant shares distribute proportionally;
- Creator royalties process recursively;
- Tax withholdings reserve properly.

Security and Risk Controls

The payment lane implements comprehensive protection against attacks and operational risks:

Denial of Service Prevention:

- Pre admission validation stops invalid transactions at the edge;
- Rate limiting throttles addresses generating excessive load;
- Economic spam becomes impossible due to stablecoin fee requirements.

MEV Resistance:

- First come first served ordering within priority tiers;
- Deadline enforcement prevents holding transactions;
- Private mempools available for sensitive operations.

Stablecoin Freeze Handling:

- Protocol detects issuer freeze events;
- Affected funds marked as “undistributable balance”;
- Receipts annotate freeze status;
- Reconciliation remains accurate despite disruption.

Emergency Controls:

- Governance can pause specific operations;
- Circuit breakers trigger on anomalous volumes;
- Rollback mechanisms handle critical failures.

This dedicated payment infrastructure transforms blockchain payments from expensive afterthoughts into efficient, compliant, privacy preserving operations that match or exceed traditional payment systems while enabling capabilities impossible in legacy finance.

Security with Programmable Trust

The promise of autonomous agents collapses without cryptographic security guarantees. Users cannot delegate real authority if agent compromise means unbounded loss. Services cannot accept agent requests without verifiable authorization. Regulators cannot approve agent operations without auditable compliance. KITE AI's security architecture provides mathematical certainty where traditional systems offer only promises.

Core Cryptographic Components

The KITE AI protocol achieves unbreakable security through three interlocking cryptographic primitives that create a complete chain of authorization from user intent to agent action.

Standing Intent: The Root of Authority

The Standing Intent (SI) represents the user's cryptographically signed declaration of what an agent may do. This is not a policy document or configuration file; it's a mathematical proof of authorization that cannot be forged or exceeded.

The Standing Intent signed with the user's private key becomes the immutable root of trust. Every subsequent operation must trace back to a valid SI. The capabilities define mathematical boundaries that cannot be exceeded regardless of agent behavior, model hallucination, or service compromise. The expiration ensures forgotten authorizations cannot persist indefinitely.

Delegation Token: Agent Authorization

The Delegation Token (DT) enables agents to authorize specific sessions for particular operations without exposing their permanent credentials.

The delegation token cryptographically proves the agent authorized this session for this operation within the Standing Intent's boundaries. The hash linkage ensures the agent cannot exceed user defined limits. The short expiration minimizes exposure from compromised sessions. The operation scope prevents session reuse for unauthorized actions.

Session Signature: Execution Proof

The Session Signature (SS) provides the final cryptographic proof for transaction execution.

Services verify all three signatures before accepting operations. The Standing Intent proves user authorization. The Delegation Token proves agent delegation. The Session Signature proves current execution. This triple verification makes unauthorized actions cryptographically impossible rather than merely prohibited.

Provable Security Properties

The protocol provides mathematical guarantees about system behavior under adversarial conditions, transforming security from trust based to proof based.

Theorem 1: Bounded Loss

Statement: Given Standing Intent SI with capabilities C and duration D, the maximum extractable value MEV under complete agent compromise is:

$MEV \leq C.max\_daily \times D$

Proof: By construction of the protocol, each transaction requires a valid Standing Intent signature. The cryptographic signature verification ensures only transactions with valid SI can execute. The provider enforced capabilities guarantee no single transaction exceeds C.max_tx and daily aggregates cannot exceed C.max_daily. Since the Standing Intent expires after duration D, no transactions can execute beyond this time. Therefore, the maximum extractable value is bounded by the product of daily cap and duration, regardless of agent compromise.

This theorem provides users with precise risk quantification. A user authorizing an agent with $100 daily limit for 30 days knows their maximum exposure is exactly $3,000, not a penny more. This mathematical certainty enables confident delegation of real financial authority.

Theorem 2: Unforgeability

Statement: Without access to the user's private key, an adversary cannot create a valid Standing Intent for unauthorized agents.

Proof: Standing Intent validity requires signature verification against the user's public key using ECDSA or EdDSA algorithms. Under the assumption that these signature schemes are existentially unforgeable under chosen message attack (EUF-CMA), an adversary without access to the user's private key cannot produce a valid signature for a new Standing Intent with non-negligible probability. The cryptographic hardness of the discrete logarithm problem ensures computational infeasibility of key recovery from public information.

This theorem guarantees that attackers cannot create fake authorizations. Even with complete knowledge of existing Standing Intents, observation of all transactions, and control over compromised agents, adversaries cannot forge new authorizations for unauthorized agents. The user’s private key remains the sole source of delegation authority.

Additional Security Properties

Beyond the formal theorems, the protocol provides additional security guarantees:

Forward Secrecy: Compromising a session key reveals only that session's operations. Past and future sessions remain secure due to independent key generation.

Principle of Least Privilege: Each layer of delegation reduces authority. Sessions have less power than agents, agents less than users. Authority only flows downward.

Automatic Expiration: All authorizations include expiration timestamps. Forgotten delegations cannot persist indefinitely. Time becomes an automatic revocation mechanism.

Non-Repudiation: Cryptographic signatures provide undeniable proof of authorization. Users cannot deny authorizing agents, agents cannot deny authorizing sessions, and sessions cannot deny executing transactions.

Revocation Mechanism

Authorization without revocation is incomplete security. The protocol implements a comprehensive revocation system with multiple enforcement layers ensuring compromised or misbehaving agents cannot continue operating.

Immediate Local Revocation

Users broadcast revocation messages to KITE Hub, which instantly propagates to all registered providers. This peer-to-peer propagation ensures near instantaneous revocation without waiting for blockchain confirmation. Services receiving revocation notices immediately reject all requests from revoked agents.

The revocation message contains:

- Agent identifier being revoked;
- Revocation timestamp;
- User signature proving authority;
- Optional reason code for analytics.

Network effects amplify revocation speed. Popular services with many connections propagate faster. Critical services prioritize revocation messages. The entire network typically learns of revocations within seconds.

Cryptographic Revocation

Users sign revocation certificates that providers cache and verify against.

Even if network propagation fails, cryptographic proof of revocation prevents continued operation. Services check revocation certificates before accepting requests. Cached certificates persist across restarts. Permanent revocations cannot be reversed, providing strong security guarantees.

Economic Revocation

Slashing conditions in agent bonds create economic incentives against continued operation post revocation:

Agent Bonds: Agents stake tokens as collateral for good behavior. Bonds are proportional to authorization limits, creating aligned incentives.

Slashing Triggers: Operating after revocation triggers automatic slashing. The protocol detects post revocation transactions and burns staked bonds.

Reputation Impact: Slashed agents suffer permanent reputation damage. Future authorizations require higher bonds. Services may refuse slashed agents entirely.

Distribution: Slashed funds are distributed to affected parties. Users recover losses from misbehaving agents. Services receive compensation for processing invalid requests.

This economic layer ensures that even if cryptographic and network revocations fail, agents have strong financial incentives to respect revocations. The cost of ignoring revocation exceeds any potential gain from continued operation.

Graceful Degradation in Adversarial Conditions

The revocation system degrades gracefully under various failure modes:

Network Partition: Local revocation continues within network segments. Cryptographic certificates provide eventual consistency.

Hub Failure: Peer to peer propagation continues without central coordination. Services share revocation information directly.

Blockchain Congestion: Off chain revocation mechanisms operate independently. On chain slashing provides eventual enforcement.

Service Offline: Cached revocations persist. Services enforce revocations immediately upon restart.

This multi-layer approach ensures revocation remains effective even under adversarial conditions or system failures. Users maintain ultimate control over their agents regardless of infrastructure state.

Agent Flows

The lifecycle of agent interactions encompasses three critical phases: authorization establishment, continuous communication, and value exchange through payments. Each phase builds on cryptographic foundations while maintaining intuitive developer experiences and user control.

Authorization transforms human identity into agent capability through a carefully orchestrated flow that bridges traditional web authentication with blockchain settlement.

The Authorization Challenge

The flow begins when an agent attempts to access a service without valid credentials. The service returns a 401 Unauthorized response, indicating required authentication methods. This triggers the authorization sequence that converts one-time human authentication into persistent agent capability.

Key insight: Gmail/OAuth proves the user's web identity (represented as did:kite_chain_id:claude:scott for example, bundling Claude App with Scott's Gmail ID). The KITE AI session token transforms this one-time proof into a time bounded, policy guarded capability the agent can use safely without repeatedly exposing the user's primary credentials.

Authorization Actors and Sequence

Four actors participate in the authorization flow:

- Agent (e.g., Claude): The AI system requesting service access;
- Service (MCP server or service agent): The resource being accessed;

- Web Credential Provider (Gmail): The identity verification source;
- KITE AI Platform and Chain: The authorization and settlement layer.

The authorization sequence proceeds through six critical steps:

Step 1: Initial Request Failure

The agent calls a service without a valid token or with an expired one. The service replies with 401 Unauthorized, initiating the authorization process.

Step 2: Discovery Phase

The service indicates it requires web credentials and points the agent to Gmail or other supported providers. The agent retrieves the authorization server metadata through standard discovery mechanisms.

Step 3: OAuth Authentication

Standard OAuth 2.1 flow executes with Gmail. The user signs in and provides consent. The agent performs the token request and receives an access token bound to the agent application, user's Gmail identity, and redirect details. This creates cryptographic proof that a real human authorized this specific agent.

Step 4: Session Token Registration

After obtaining web credentials, the agent generates a local session key and registers a session token with KITE AI Platform and Chain. This registration binds:

- Agent and application identity;
- Scopes and allowed operations;
- Quotas and time to live (TTL);
- Proof chain back to user authorization.

The session private key never leaves the local environment. A DID session registration call ensures the session is registered, linked, and recognizable by other services across the network.

Step 5: Service Retry

The agent retries the original service call, now presenting the session token signed by the ephemeral session key.

Step 6: Verification and Execution

The service verifies the token and checks it against KITE AI's registry and policies. If the token is valid, policies and quotas pass, and the token falls within its time window, the service executes the request and returns a successful response.

JWT Token Structure

When a session key is authorized and created, a corresponding JWT token encapsulates all necessary authorization information.

Once created, subsequent calls to any service in the network contain both the JWT token and the session public key used for encryption. This dual credential system ensures both authorization and authenticity.

Agent Reputation and Trust Accumulation

Existing blockchains treat all accounts as equals. A newly created address possesses identical capabilities to one with years of legitimate history. This egalitarian approach fails catastrophically for agent systems where behavioral history should determine authorization scope.

Trust Dynamics for Agent Systems

Agent systems require sophisticated trust dynamics that traditional blockchains cannot provide:

Progressive Authorization: New agents must start with minimal permissions, perhaps $10 daily limits and restricted service access. Each successful operation contributes to reputation scores, automatically expanding capabilities.

Behavioral Adjustment: Successful operations should increase authorization over time. A reliable agent might see limits increase from $10 to $100 to $1,000 as trust accumulates. Conversely, violations should trigger automatic constraint tightening, reducing limits or requiring additional verification.

Trust Portability: Trust must transfer across services. An agent with established reputation on one platform should bootstrap trust on new platforms without starting from zero.

Verification Economics: Without native reputation, every agent interaction starts from zero trust, forcing expensive verification for routine operations. This overhead makes micropayments economically impossible.

Proof Chain Architecture

KITE AI provides a complete proof chain from session to agent to user to reputation, all verified by credited authorities. This chain enables instant trust verification without repeated authentication:

Service providers and users leverage this proof chain to make authorization decisions based on verifiable history rather than blind trust. For example, a user might grant:

- Read access to agents with reputation > 100;
- Write access to agents with reputation > 500;
- Payment authority to agents with reputation > 750;
- Unlimited access to agents with reputation > 900.

This graduated trust model enables safe progressive autonomy while maintaining security.

Agent Communication Flow

Human transactions occur in isolation. Agent operations require continuous communication and coordination. This fundamental difference demands native support for persistent connections, multi-party coordination, and verifiable message exchange.

Agent to Agent Messaging (A2A)

Encrypted channels enable negotiation, discovery, and coordination without exposing strategies or private information. The A2A protocol provides structured communication through Agent Cards that serve as the source of truth for capabilities and endpoints.

Agent Card Structure:

Peers fetch Agent Cards to learn capabilities and supported security schemes. The card declares endpoint URLs, authentication methods, and method availability. Most critically, it includes the Agent DID and session security scheme so any A2A compatible peer understands how to validate session scoped credentials.

Service Level Agreements

Programmable contracts enforce response times, availability guarantees, and performance metrics with automatic penalties. Unlike traditional SLAs based on legal enforcement, these execute automatically through smart contracts:

- Response Time: Service must respond within 100ms or face automatic penalties;
- Availability: 99.9% uptime with pro rata refunds for downtime;
- Accuracy: Error rates below 0.1% or reputation slashing;
- Throughput: Guaranteed 1,000 requests per second minimum.

These programmable SLAs transform vague promises into cryptographically enforced guarantees.

Reputation Networks

Persistent identity and performance tracking enables trust building across multiple interactions and services. Every interaction contributes to global reputation:

- Successful payments increase scores;
- Fast responses boost reputation;
- Failed deliveries decrease trust;
- SLA violations trigger penalties.

Reputation becomes portable across the entire network, solving the cold start problem where new relationships begin from zero trust.

Agent Payment Flow

Traditional payment rails built for large, infrequent charges with human approval loops fail catastrophically for agent micropayments. Monthly subscriptions and one-time checkouts assume high fixed fees, settlement delays, chargeback windows, and per transaction minimums that make per message pricing impossible.

KITE AI inverts this model. Agents open state channels to services, every interaction carries a signed micro voucher, and a single close nets all balances on chain. Fees amortize across millions of calls. This enables true micropayments at approximately $0.000001 per message, streaming payments at rates per second or token, automatic quotas, and instant revocation, all enforced by smart accounts and session keys. The result: usage-based pricing that feels real time like network packets, not billing periods.

Channel Opening

The smart account deposits a limit (e.g., $50) into the channel contract. This deposit acts as a bond for honest behavior. Both parties collectively sign a state update to initialize the channel's state. After initialization, they can transact quickly and freely off chain without blockchain overhead.

Pay Per Interaction (Off Chain)

Each message or API call carries a voucher signed by the session key.

The channel tracks old state and new state, requiring all participants to agree on state changes. The merchant accepts work if the voucher validates and falls within quota and TTL. Streaming uses many small increments, potentially $0.000001 per chunk. Policies including spend caps, allowlists, and rolling windows are enforced locally by the KITE AI sidecar before voucher production.

Channel Settlement

Channels close through either cooperative or disputed paths:

Cooperative Close (Happy Path):

Parties co-sign the final state and submit once. Unused funds immediately return to the user. Gas costs are minimal. Settlement is instant.

Disputed Close:

Either party posts their latest signed state to the blockchain. The counterparty can challenge within the window by presenting a newer state, ensuring only the most recent update settles. If consensus breaking situations occur, either party can trigger the on-chain contract to close the channel and distribute funds.

Common dispute scenarios include:

- Participants going offline and failing to propose state transitions;
- Participants refusing to co-sign valid state updates;
- Participants attempting finalization with outdated states;
- Participants proposing invalid state transitions.

This dispute mechanism provides trustlessness, ensuring honest parties can exit deposits at any point regardless of counterparty behavior. The blockchain serves as the ultimate arbiter, guaranteeing that honest behavior is always rewarded and malicious behavior always punished.

Economic Transformation

This payment architecture enables an economic model impossible with traditional infrastructure:

Granular Pricing: Services price at the level of individual API calls, inference tokens, or compute milliseconds. A conversation might cost $0.02 in language model inference, $0.001 in embedding generation, and $0.0001 in storage.

Real Time Economics: Prices adjust dynamically based on load, priority, and resource availability. Peak hours might cost more. Premium models command higher rates. Urgent requests pay priority fees.

Automatic Budgets: Agents operate within cryptographically enforced spending limits. When budgets approach limits, agents automatically throttle or halt operations.

Instant Settlement: No waiting for monthly bills or reconciliation. Every interaction settles immediately. Services receive payment instantly. Users see real time spending.

The transformation from billing periods to packet economics enables business models that are impossible with traditional payments. Every message becomes a payment. Every payment becomes programmable. Every program becomes verifiable. This is the economic foundation for truly autonomous agents.

H.8 Audit

True.

H.9 Audit Outcome

An audit of the technology used was conducted by Halborn. The assessment reported no critical or high-severity issues.

PART I – INFORMATION ON RISKS

I.1 Offer-Related Risks

The admission to trading of KITE entails risks related to market volatility, liquidity availability and operational execution. Price fluctuations may occur due to secondary market dynamics, and liquidity may be limited in early trading phases.

I.2 Issuer-Related Risks

Not applicable.

I.3 Crypto-Assets-Related Risks

Market Risk: Crypto-assets are notoriously volatile, with prices subject to significant fluctuations due to market sentiment, regulatory news, technological advancements, and macroeconomic factors.

Liquidity Risk: Crypto-assets may suffer from low liquidity, making it difficult to buy or sell large amounts without affecting the market price, which could lead to significant losses, especially in fast-moving market conditions.

Custodial Risk: Risks associated with the theft of crypto-assets from exchanges or wallets, loss of private keys, or failure of custodial services, which can result in the irreversible loss of crypto-assets.

Smart Contract Risk: Smart contracts are code running on a blockchain, executing the programmed functions automatically if the defined conditions are fulfilled. Bugs or vulnerabilities in smart contract code can expose blockchain users to potential hacks and exploits. Any flaw in the code can lead to unintended consequences, such as the loss of crypto-assets or unauthorized access to sensitive data.

Regulatory and Tax Risk: Changes in the regulatory environment for crypto-assets (such as consumer protection, taxation, and anti-money laundering requirements) could affect the use, value, or legality of crypto-assets in a given jurisdiction.

Counterparty Risk: In cases where crypto-assets are used in contractual agreements or held on exchanges, there is a risk that the counterparty may fail to fulfill their obligations due to insolvency, compliance issues, or fraud, resulting in loss of crypto-assets.

Reputational Risk: Association with illicit activities, high-profile thefts, or technological failures can damage the reputation of certain crypto-assets, impacting user trust and market value.]

I.4 Project Implementation-Related Risks

As the network relies on validator and module participation, there is an operational risk of underperformance or service disruption. To mitigate malicious use or misaligned incentives, the protocol applies a staking-and-slashing mechanism that economically penalises misconduct.

I.5 Technology-Related Risks

Private Key Management Risk and Loss of Access to Crypto-Assets: The security of crypto-assets heavily relies on the management of private keys, which are used to access and control the crypto-assets (e.g. initiate transactions). Poor management practices, loss, or theft of private keys, or respective credentials, can lead to irreversible loss of access to crypto-assets.

Settlement and Transaction Finality: By design, a blockchain's settlement is probabilistic, meaning there is no absolute guaranteed finality for a transaction. There remains a theoretical risk that a transaction could be reversed or concurring versions of the ledger could persist due to exceptional circumstances such as forks or consensus errors. The risk diminishes as more blocks are added, making it increasingly secure over time. Under normal circumstance, however, once a transaction is confirmed, it cannot be reversed or cancelled. Crypto-assets sent to a wrong address cannot be retrieved, resulting in the loss of the sent crypto assets.

Scaling Limitations and Transaction Fees: As the number of users and transactions grows, a blockchain network may face scaling challenges. This could lead to increased transaction fees and slower transaction processing times, affecting usability and costs.

Economic Self-sufficiency and Operational Parameters: A blockchain network might not reach the critical mass in transaction volume necessary to sustain self-sufficiency and remain economically viable to incentivize block production. In failing to achieve such inflection point, a network might lose its relevance, become insecure, or result in changes to the protocol's operational parameters, such as the monetary policy, fee structure and consensus rewards, governance model, or technical specifications such as block size or intervals.

Network Attacks and Cyber Security Risks: Blockchain networks can be vulnerable to a variety of cyber-attacks, including 51% attacks, where an attacker gains control of the majority of the network's consensus, Sybil attacks, or DDoS attacks. These can disrupt the network's operations and compromise data integrity, affecting its security and reliability.

Consensus Failures or Forks: Faults in the consensus mechanism can lead to forks, where multiple versions of the ledger coexist, or network halts, potentially destabilizing the network and reducing trust among participants.

Bugs in the Blockchain's Core Code: Even with thorough testing, there is always a risk that unknown bugs may exist in a blockchain protocol, which could be exploited to disrupt network operations or manipulate account balances. Continuous code review, audit trails, and having a bug bounty program are essential to identify and rectify such vulnerabilities promptly.

Smart Contract Security Risk: Smart contracts are code running on a blockchain, executing the programmed functions automatically if the defined conditions are fulfilled. Bugs or vulnerabilities in smart contract code can expose blockchain networks to potential hacks and exploits. Any flaw in the code can lead to unintended consequences, such as the loss of crypto-assets or unauthorized access to sensitive data.

Dependency on Underlying Technology: Blockchain technology relies on underlying infrastructures, such as specific hardware or network connectivity, which may themselves be vulnerable to attacks, outages, or other interferences.

Risk of Technological Disruption: Technological advancements or the emergence of new technology could impact blockchain systems, or components used in it, by making them insecure or obsolete (e.g. quantum computing breaking encryption paradigms). This could lead to theft or loss of crypto-assets or compromise data integrity on the network.

Governance Risk: Governance in blockchain technology encompasses the mechanisms for making decisions about network changes and protocol upgrades. Faulty governance models can lead to ineffective decision-making, slow responses to issues, and potential network forks, undermining stability and integrity. Moreover, there is a risk of disproportionate influence by a group of stakeholders, leading to centralized power and decisions that may not align with the broader public's interests.

Anonymity and Privacy Risk: The inherent transparency and immutability of blockchain technology can pose risks to user anonymity and privacy. Since all transactions are recorded on a public ledger, there is potential for sensitive data to be exposed. The possibility for the public to link certain transactions to a specific address might expose it to phishing attacks, fraud, or other malicious activities.

Data Corruption: Corruption of blockchain data, whether through software bugs, human error, or malicious tampering, can undermine the reliability and accuracy of the system.

Third-Party Risks: Crypto-assets rely on third-party services such as exchanges and wallet providers for trading and storage. These platforms can be susceptible to security breaches, operational failures, and regulatory non-compliance, which can lead to the loss or theft of crypto-assets.

I.6 Mitigation Measures

The use of stablecoin-based gas fees contributes to cost predictability, reducing exposure to fee volatility typically associated with crypto-assets.

PART J – INFORMATION ON THE SUSTAINABILITY INDICATORS IN RELATION TO ADVERSE IMPACT ON THE CLIMATE AND OTHER ENVI-RONMENT-RELATED ADVERSE IMPACTS

J.1 ADVERSE IMPACTS ON CLIMATE AND OTHER ENVIRONMENT-RELATED ADVERSE IMPACTS

(A) MANDATORY INFORMATION ON PRINCIPAL ADVERSE IMPACTS ON THE CLIMATE AND OTHER ENVIRONMENT-RELATED ADVERSE IMPACTS OF THE CONSENSUS MECHANISM

General information	
S.1. Name Name reported in field A.1.	Kite AI Ltd.
S.2. Relevant legal entity identifier Identifier referred to in field A.2.	6EH6.
S.3. Name of the crypto-asset Name of the crypto-asset, as reported in field D.2.	Kite Token (KITE)
S.4. Consensus Mechanism	The network operates as a Layer-1 Proof-of-Stake (PoS) blockchain, using the same PoS mechanism adopted by Avalanche Layer-1 architecture. Validators are required to stake KITE and participate in consensus to secure the network. Slashing applies in case of validator misbehavior or failure to meet network performance requirements, ensuring economic accountability and alignment with protocol objectives.
S.5. Incentive Mechanisms and Applicable Fees	The network applies a Proof-of-Stake incentive model. Validators, delegators and module operators receive protocol rewards, initially distributed in KITE and transitioning progressively to stablecoins, targeting approximately 4% annual reward on staked value. Validator operations receive a minimal base reward to cover L1 operating costs. Transaction fees (gas) are paid exclusively in whitelisted stablecoins rather than in KITE, ensuring fee stability and predictable cost exposure.
S.6. Beginning of the period to which the disclosure relates	The date when the token launches which is expected to be November 2025.
S.7. End of the period to which the disclosure relates	Until modified by governance vote.
Mandatory key indicator on energy consumption	

S.8. Energy consumption Total amount of energy used for the validation of transactions and the maintenance of the integrity of the distributed ledger of transactions, expressed per calendar year.	Proof-of-Stake model minimizes energy use. While quantitative energy data is not yet available, the total amount of energy does not exceed 500,000 kilowatt-hours.
Sources and methodologies	
S.9. Energy consumption sources and Methodologies Sources and methodologies used in relation to the information reported in field S.8	Energy consumption is measured at each validator node and a transaction based methodology is used to determine energy consumption.