

X Layer Exchange OS

Builder-First Permissionless Financial Infrastructure

X Layer Exchange OS — a builder-first permissionless financial infrastructure where markets are created, connected, and composed, enabling the next chapter of on-chain finance.

Contents

Abstract	5
-----------------	----------

Disclaimers	5
--------------------	----------

1 Vision	6
1.1 Unleashed, not limited	6
1.2 Unified, not fragmented	7
1.3 Liquid, not islanded	7

2 Why Exchange OS	8
2.1 Next-Generation Markets to Match	8
2.2 The Trading Infrastructure to Build	8
2.3 The Onchain Financial Landscape to Reshape	9

3 Introducing Exchange OS	11
3.1 Overview: How Exchange OS Works	11
3.2 XIP-Exchange OS	11
3.3 Design Philosophy	11
3.4 Architecture	12

4 Governance Model	18
4.1 Governance Boundary	18
4.2 How Governance Operates	18

5	Ecosystem	19
5.1	Ecosystem Landscape	19
5.2	Deployer — People Who Identify Market Needs and Turn Them Into Realities	19
5.3	Issuer — Enables Assets	20
5.4	Interface / Distribution — Wherever the Users Are, That's Where the Market Is	20
5.5	Supporter — Service Providers	21
5.6	Market Maker — The Lifeline of Financial Markets	21
5.7	Open System of Exchange OS	22

6	Conclusion	23
----------	-------------------	-----------

Abstract

As a major upgrade to X Layer, Exchange OS is a builder-first, permissionless financial infrastructure on X Layer. Anyone can use this high-performance open protocol to launch and operate a market venue, from spot, perps and outcome markets to virtual game markets.

Within protocol-defined boundaries, each venue can set its own risk preferences, access rules, and service scope. A deployer can offer institutional-grade, KYC-verified services in one venue, and a fully permissionless Web3 experience in another. This allows different market operators to serve different users, regions, and risk preferences while relying on the same shared infrastructure.

Exchange OS stays decentralized where credible neutrality matters most: matching fairness, asset self-custody, risk enforcement, and final settlement. Every trading venue created by deployers shares one unified high-performance stack for matching, margining, liquidation, settlement, and risk. Accounts, balances, and margins are unified at the protocol layer. The same capital moves across every market, making cross-market capital efficiency and strategy composability native by design.

Built on X Layer, Exchange OS allows fragmented venues to access one open financial market network. Through deployers, issuers, interfaces, supporters, and market makers, every participant contributes to what they do best.

SCOPE OF THIS PAPER

This document presents the vision and design model of X Layer Exchange OS: the permissionless protocol that lets deployers deploy diverse market venues, the architecture, the unified account and margin model, the listing and oracle mechanisms, the liquidation framework, the governance structure, and the open ecosystem landscape. Contract interfaces and schemas, precise risk parameters (margin ratios, oracle deviation thresholds, leverage caps, slashing percentages), slot auction mechanics, liquidation logic, reference implementations, and other details are out of scope here and are maintained in the Exchange OS Protocol Specification. Subsequent revisions will be made to the specification rather than in this document.

Disclaimers

Not investment, financial, or legal advice

This paper is provided for information only. Nothing in this paper constitutes investment advice, a financial promotion, a recommendation to buy or sell any digital asset, or legal advice. Readers should do their own research and consult their own advisers before making any decision relating to the use of X Layer, Exchange OS, or any market deployed on Exchange OS.

1 Vision

Every generation of open networks has shifted power once held by centralized institutions back to any market participants. The internet unlocked the right to publish information. Blockchain unlocked the right to issue assets. Yet in financial markets, one core power remains centralized: whether a market can be created permissionlessly and at sufficiently low friction.

AMMs enabled tokenized spot trading, making it possible to create basic spot markets without permission. But in more complex market structures such as perps and outcome markets, existing solutions impose significant operational overhead, technical dependencies, or capital requirements that prevent deployers from freely and cost-effectively creating diverse markets at scale.

Also, trading demand is surging, while market creation still depends on platform assessment and deployment. **This is the first fundamental tension in today's trading infrastructure: market demand now moves faster than infrastructure can respond.**

The second contradiction comes from the user side. As types of different markets increase, users don't trade with a single directional view — they hold layered, interconnected positions across markets. Take this example: "I believe AI will have its breakout year — the odds will shift in its favor, AI tokens will get revalued, and I want to hold AI-linked blue chips for the long run." To the user, this is one coherent market view. Price action, outcome probability, and asset allocation are all expressions of the same thesis. But in practice, that view gets fragmented across spot, perps, and outcome markets — forcing the user to manage separate accounts, capital, margin, and risk. The intent is unified; the execution is fragmented.

The third contradiction sits at the ecosystem layer. Today's trading landscape is split by platform boundaries: liquidity is fragmented, order flow is locked inside separate interfaces, and strategies are difficult to compose across markets. Markets are not growing together within one ecosystem; they are being built repeatedly as walled gardens.

Therefore, the next generation of trading infrastructure must accomplish three things at once.

1.1 Unleashed, not limited

Decentralized financial markets should no longer depend on centralized platforms to decide what can be traded. Any deployer who identifies demand early should be able to permissionlessly deploy spot, perps, and outcome markets on the same open infrastructure. Market creation should no longer require permission from any central process: initiated by deployers, chosen by users, secured by governance, and validated by market liquidity.

1.2 Unified, not fragmented

As financial markets become more diverse, users are forced to navigate fragmented trading environments: separate accounts, separate balances, separate margin systems, and separate risk systems. Users' trading intent is scattered across different markets. A unified platform is needed to carry the users' complete trading intent, with a unified account that consolidates user assets and exposures to maximize capital efficiency and experience.

1.3 Liquid, not islanded

Today's trading ecosystem is divided by platform boundaries. Liquidity is fragmented, strategies cannot be composed, and order flow is trapped within individual venues. Trading markets should not remain isolated from one another. Deployers, asset issuers, users, interfaces, supporters, and market makers will participate together, enabling liquidity and order flow to be more broadly shared and distributed. Trading strategies should be able to compose, arbitrage, and stack across markets, raising the capability of the entire ecosystem and forming a continuously self-evolving trading network.

This is why we propose **Exchange OS: a major upgrade to X Layer — a shared market operating environment and high-performance trading infrastructure for all markets**. At the protocol layer, it is open to any deployer who wants to deploy their own market permissionlessly. It uses a decentralized risk framework to secure the system, enables users to move across spot, perps, outcome markets through one account, and makes cross-market liquidity natively composable.

Together, these form the core of next-generation trading infrastructure.

2 Why Exchange OS

2.1 Next-Generation Markets to Match

The evolution of blockchain-based financial markets is being driven by both demand and supply.

On the demand side, the scope of what users want to trade continues to expand. Markets are moving beyond token prices into event probabilities, real-world assets, private-market valuations, or any exposure. As users expand their expectations of what markets can support, markets need to respond with greater efficiency, broader coverage, and stronger real-time availability.

On the supply side, new market structures continue to emerge. Outcome markets, RWA markets, pre-market trading, and perps are not merely built for existing demand; they also create new demand through novel market mechanisms. Outcome market expands the category of tradable instruments to events. RWA market extends them from crypto-native assets to tokenized real-world assets. Pre-market trading extends them from listed assets to private-market valuations. Perps, through margin, funding rates, and liquidation mechanisms, allow users to trade without physical spot settlement.

Together, demand-side expansion and supply-side innovation create a broad network of independent markets. But while this network is vibrant, it remains highly fragmented: accounts, liquidity, order flow, risk controls, and settlement systems are separated from one another. Every new market has to bootstrap itself from scratch. What the next generation of markets needs is not simply another trading venue, but a shared infrastructure layer that connects them.

2.2 The Trading Infrastructure to Build

On-chain trading infrastructure has gone through multiple generations of technical evolution. AMMs made spot trading possible without relying on centralized market makers. Off-chain order books with on-chain settlement further improved trading performance. Appchains brought trading logic into more transparent, higher-performance dedicated execution environments.

But the next generation of financial infrastructure must solve more than just "how to trade." It must respond to how markets can be deployed permissionlessly, how users can access markets through a unified account, and how liquidity and order flow can move continuously in the ecosystem.

Exchange OS is designed to make market creation permissionless, unify market access for users, and reshape liquidity flows in the entire ecosystem.

This moment is here. Dedicated execution environments have been validated. The oracle and data-source ecosystem has matured. Market demand for on-chain perps and outcome markets has also been proven. These three prerequisites have matured at the same time, giving Exchange OS the

conditions to become the next generation of on-chain trading infrastructure.

PREREQUISITES	STATE
Dedicated Execution Environment	The architecture of "asset and governance + matching and execution" has been proven to be a viable engineering paradigm.
Oracle and Data Source Ecosystem	When creating a market, deployers require data sources for asset prices, and outcome market settlement. Today's ecosystem of on-chain and off-chain data sources is sufficiently diverse — from mainstream oracle networks and specialized market data aggregation services to self-hosted nodes and even manual adjudication, with viable solutions available in every form. This makes the design of "the protocol layer not prescribing data sources, but allowing deployers to choose their own" a realistic and feasible approach.
Demand for On-chain Spot, Perps and Outcome Markets	Over the past two years, dedicated perps and outcome markets have generated cumulative trading volumes of trillions of dollars, proving that market demand is no longer a hypothesis. This has transformed the question of "why should anyone be able to launch their own perps or outcome market?" from a speculative argument into a data-backed conclusion.

Table 1. Three prerequisites that enable X Layer Exchange OS today.

X Layer Exchange OS enables permissionless market deployment at the protocol layer, allowing spot, perps, and outcome markets to share the same infrastructure. These markets can be launched directly on Exchange OS with the broader ecosystem.

The next generation of on-chain trading networks will no longer be defined by a single trading venue. It will be powered by X Layer Exchange OS.

2.3 The Onchain Financial Landscape to Reshape

Exchange OS fundamentally changes how markets are created, accessed, capitalized, and connected: market creation expands from permissioned to permissionless; user access expands from a single frontend to any third-party interface; capital structure moves from fragmented balances to unified accounts; and the overall form evolves from a single trading product into an open financial market network.

Together, these shifts bring more than just additional markets. They create structural improvements in market bootstrapping efficiency, capital efficiency, and ecosystem collaboration. Deployers no longer need to rebuild trading backends from scratch. Users can access all markets with a single account. Ecosystem participants can build together on shared infrastructure. X Layer can support more substantive, high-frequency, and sustainable financial activity.

X Layer Exchange OS upgrades market creation from a centralized platform decision into an open protocol-level mechanism.

3 Introducing Exchange OS

3.1 Overview: How Exchange OS Works

1. A deployer stakes OKB and launches a market venue.
2. Each market venue can run in its risk-isolated markets. A deployer can offer institutional-grade, KYC-verified services through one market venue and a fully permissionless Web3 experience through another, and isolate the user groups and liquidity within each venue. Deployers can also launch simulated markets that deal only in assets with no real-world value.
3. To interact with the deployer's market venue, a user deposits assets into the X Layer EVM contract, signing with their own key.
4. The user trades on the deployer's market venue. The matching engine runs as protocol code; the deployer cannot front-run the trade or change how it is matched.
5. The user can sync assets onto X Layer at any time by signing a withdrawal transaction. If the X Layer TradeZone sequencer is offline, the user can still withdraw their assets directly onto the X Layer EVM using the escape hatch.
6. If a deployer misbehaves, a decentralized model can slash the deployer's stake under pre-published objective rules.

3.2 XIP-Exchange OS

Today, we are introducing XIP-Exchange OS — a major protocol upgrade to X Layer delivered through the X Layer Improvement Proposal (XIP) process. The proposal establishes an open deployment protocol for on-chain markets, bringing the deployer ecosystem natively onto X Layer, with Exchange OS as its implementation.

Historically, launching a market meant building an exchange from the ground up: every component had to be designed, integrated, and operated independently. Exchange OS fundamentally changes this model. Once a market is deployed, the protocol provides the complete trading stack out of the box, freeing deployers to focus on the markets they create rather than the infrastructure beneath them.

3.3 Design Philosophy

The design philosophy of Exchange OS can be distilled into three principles:

1. Assets belong to the users.
2. Markets belong to the deployers.

3. Infrastructure belongs to everyone.

The infrastructure ensures that capabilities that must remain neutral — order matching, asset self-custody, risk enforcement, and final settlement — are handled end-to-end at the protocol layer. User funds stay secure and order matching stays fair, no matter which deployer operates the market.

Everything else is left to the deployer's discretion: which assets to list, which data sources to trust, and which users to serve.

3.4 Architecture

Exchange OS is built on X Layer across two infrastructure environments — a separation designed to place trust and performance in the execution domains best suited to each.

- The **X Layer EVM** anchors assets and enforces governance.
- The **X Layer TradeZone** handles high-frequency order matching and execution.

The two environments stay in sync through the protocol's native state-synchronization mechanism.

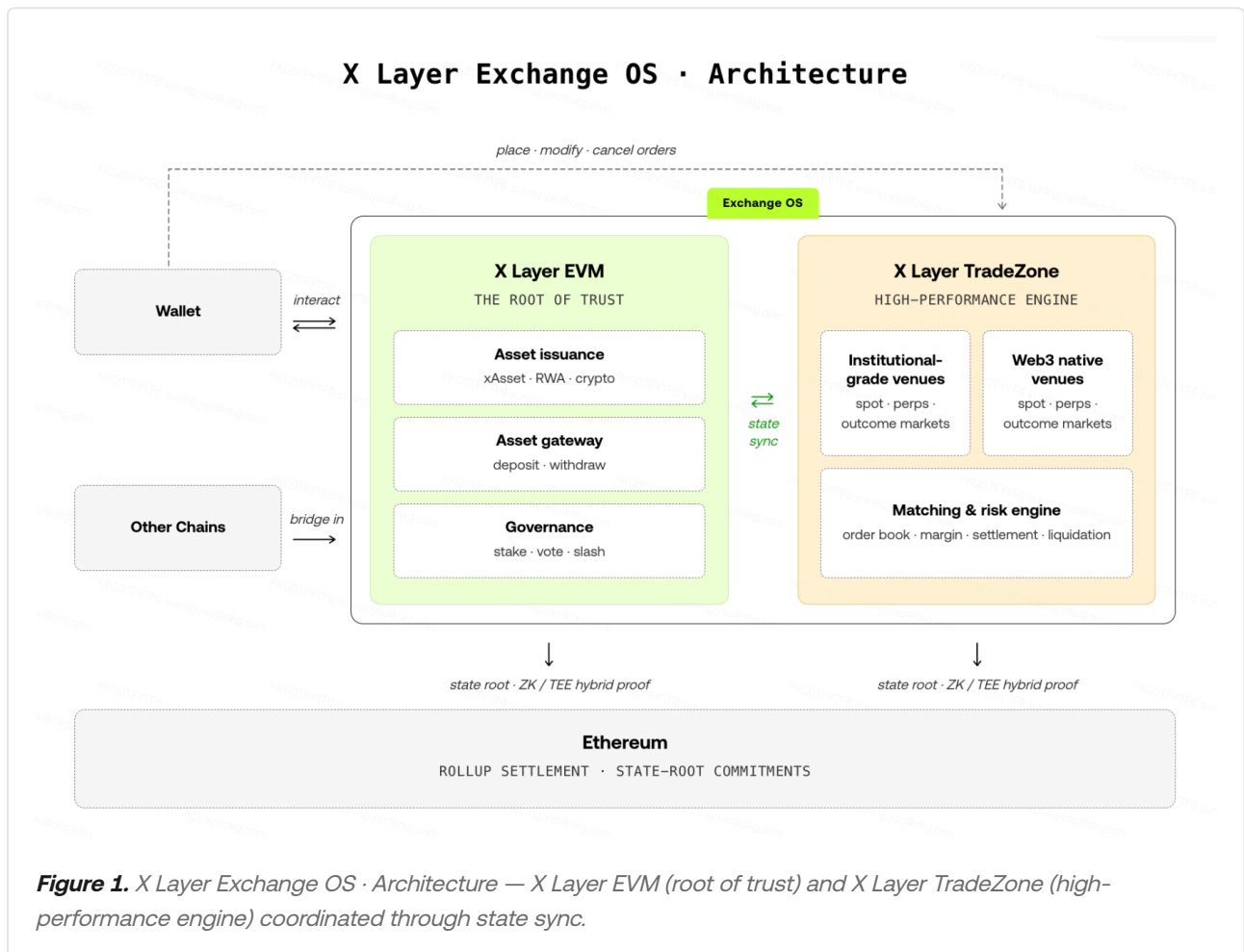


Figure 1. X Layer Exchange OS · Architecture — X Layer EVM (root of trust) and X Layer TradeZone (high-performance engine) coordinated through state sync.

X Layer EVM — The Root of Trust

The X Layer EVM handles all on-chain actions tied to asset security and governance rights, including user asset anchoring, deployer stake locking, release, and slashing execution.

Key properties:

- **Fully on-chain** — every action is transparent and verifiable.
- **The single source of truth for fund state** — all deposits, withdrawals, staking and slashing are executed here.
- **No privileged operators** — no party holds unilateral control over user funds.

X Layer TradeZone — The High-Performance Infrastructure

Order-book markets demand low-latency matching and real-time margin computation — workloads that exceed the practical throughput limits of general-purpose EVM execution. To meet these demands, X Layer provides a purpose-built native execution environment: **X Layer TradeZone**.

Key properties:

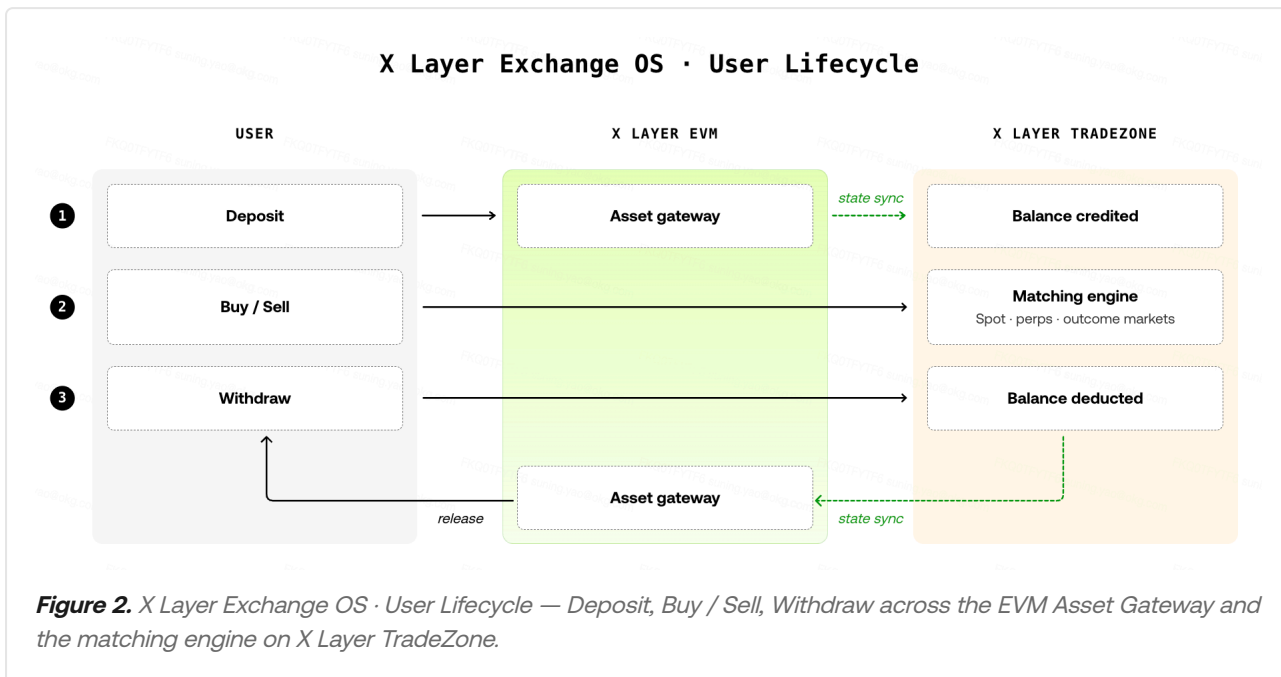
- **Millisecond matching latency**
- **300k TPS throughput**
- **State is publicly verifiable by all users**
- **Gas-free trading for users**

X Layer TradeZone is powered by four internal engines:

1. **Matching Engine** — enforces maker-friendly execution-order guarantees.
2. **Margin Engine** — performs real-time margin calculation; supports both isolated and cross-margin modes.
3. **Liquidation Engine** — continuously monitors account health, with triggers for position reduction, and ADL (Auto-Deleveraging).
4. **Settlement Engine** — handles trading settlement, PnL updates and funding-rate settlement.

Trading and Settlement Lifecycle

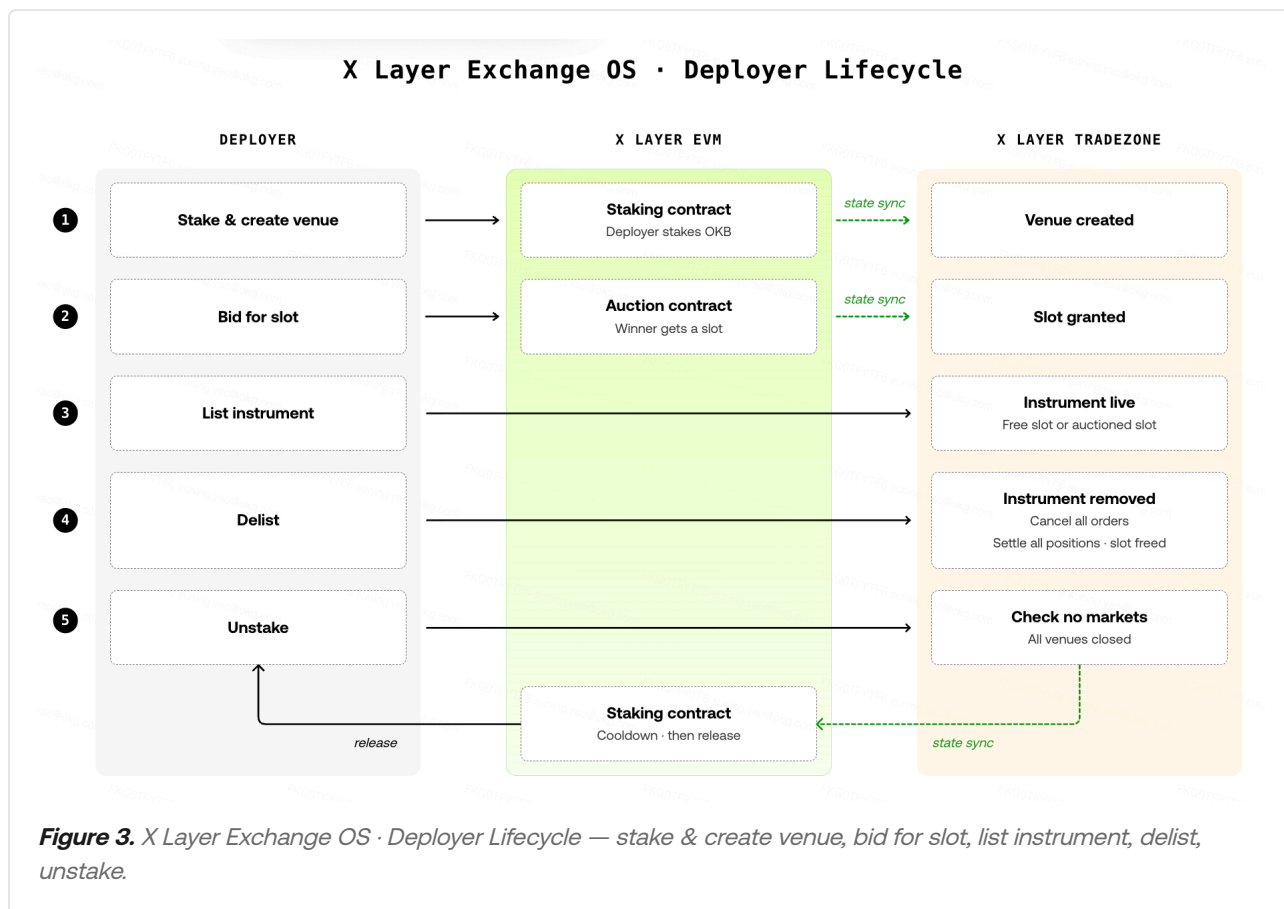
The end-to-end user flow on Exchange OS proceeds as follows:



Key boundaries:

- **Deposit** — The user deposits assets into the X Layer EVM deposit contract; the resulting state is synchronized to X Layer TradeZone, which then credits the user's account.
- **Order Matching** — All matching logic executes entirely within the X Layer TradeZone.
- **Withdrawal** — The user initiates a withdrawal on X Layer TradeZone; the state is synchronized back to the X Layer EVM contract, which releases the assets to the user's wallet.

Venue Creation and Market Listing via Staking



Venue Creation

A deployer must first stake OKB in the X Layer Staking Contract before creating a venue or listing instruments.

Flexible Staking Mechanism

Exchange OS introduces a flexible staking model designed for a diverse deployer ecosystem. With a single stake, a deployer can unlock the ability to list across spot, perps, and outcome markets, enabling different types of market operators to launch customized venues based on their own strategy, user base, and risk preferences.

Risk-isolated, deployer-defined venues

Within protocol-defined boundaries, each venue can set its own risk preferences and service scope, including region-specific user access and controls to share liquidity with or remain isolated from other types of trading venues where applicable. Deployers can flexibly choose their service model based on their own positioning — they can offer institutional-grade services to KYC-verified users, deliver a

permissionless on-chain experience to Web3-native users via self custody wallet, or launch games and simulated markets that deal only in assets with no real-world value, accommodating the distinct needs of traditional financial institutions and Web3 deployers alike.

Slots and Listings

- **Base Slots** — Each venue is granted a free allocation of Base Slots upon creation, sized to the instrument types it deploys (spot, perps, or outcome markets). These represent the venue's initial listing capacity.
- **Slot Auction** — Listings that exceed Base Slot capacity must acquire additional slots through a Slot Auction.
- **Deployer Discretion** — Within Base Slot capacity, deployers retain full discretion over which instruments to list.

Oracle

The role of oracles in X Layer TradeZone varies by instrument type.

- **Perps** rely on mark and index prices for position valuation and liquidation. Deployers have full discretion over data-source selection, subject to two protocol-level constraints:
- Oracle update frequency must meet a protocol-defined minimum.
- Abnormal price feeds are recorded on-chain and confirmed manipulation will result in slashing.
- **Outcome Markets** do not depend on continuous price feeds; event outcomes are resolved by designated third-party oracle providers.

Cross Margin

- Perps support two margin modes: **isolated** and **cross**.
- Once an instrument is approved for cross-margin, it joins the venue's cross-margin account pool, where margin is computed using a fixed, protocol-defined formula.
- Deployers are responsible for continuously monitoring and responding to risk within their venues. Failure to maintain adequate risk controls may result in slashing.

Liquidation

Liquidation is triggered when an account's assets fall below the maintenance margin. If the liquidated position cannot be absorbed by the market for an extended period, the final risk control mechanism of Auto-Deleveraging (ADL) will be activated, force-closing profitable positions.

Slashing

The deployer's stake serves as an economic guarantee against malicious or negligent conduct. If a deployer's actions harm user interests, the protocol may slash a portion of the stake — calibrated to the severity of the offense — with the final determination made by the Governance mechanism. See the **Governance Model** for full details.

4 Governance Model

On Exchange OS, any eligible entity can become a deployer by staking OKB, gaining autonomous control over its market's data sources, risk parameters, and listed assets. This open structure unlocks broad flexibility, but it also surfaces scenarios that demand protocol-level adjudication — deployer misconduct, and emergency response to major incidents, among others. Protocol-level governance exists to address precisely these scenarios: not to constrain openness, but to ensure that openness remains sustainable as the ecosystem scales.

4.1 Governance Boundary

Exchange OS governance adheres to a clear boundary: **governance secures the protocol; market operations are left to the protocol and to deployers, who manage their venues autonomously.**

Deployers decide how to run their own markets — which assets to list, which data sources to use, what leverage limits to set, and what revenue model to adopt — all configured within the parameter ranges the protocol permits.

The protocol enforces hard limits on every such parameter: strict upper and lower bounds on funding-rate movements, price-deviation thresholds for data sources, and hard caps on maximum leverage and open interest.

Governance intervenes only where human judgment is genuinely required — adjudicating deployer misconduct and authorizing emergency market interventions, among a narrow set of comparable situations.

4.2 How Governance Operates

Governance of Exchange OS is entrusted to a decentralized model. Decisions that require deliberation — protocol upgrades, slashing rulings, and similar matters — are resolved through the structured processes, with different procedures applied across the spectrum from rapid response to thorough deliberation.

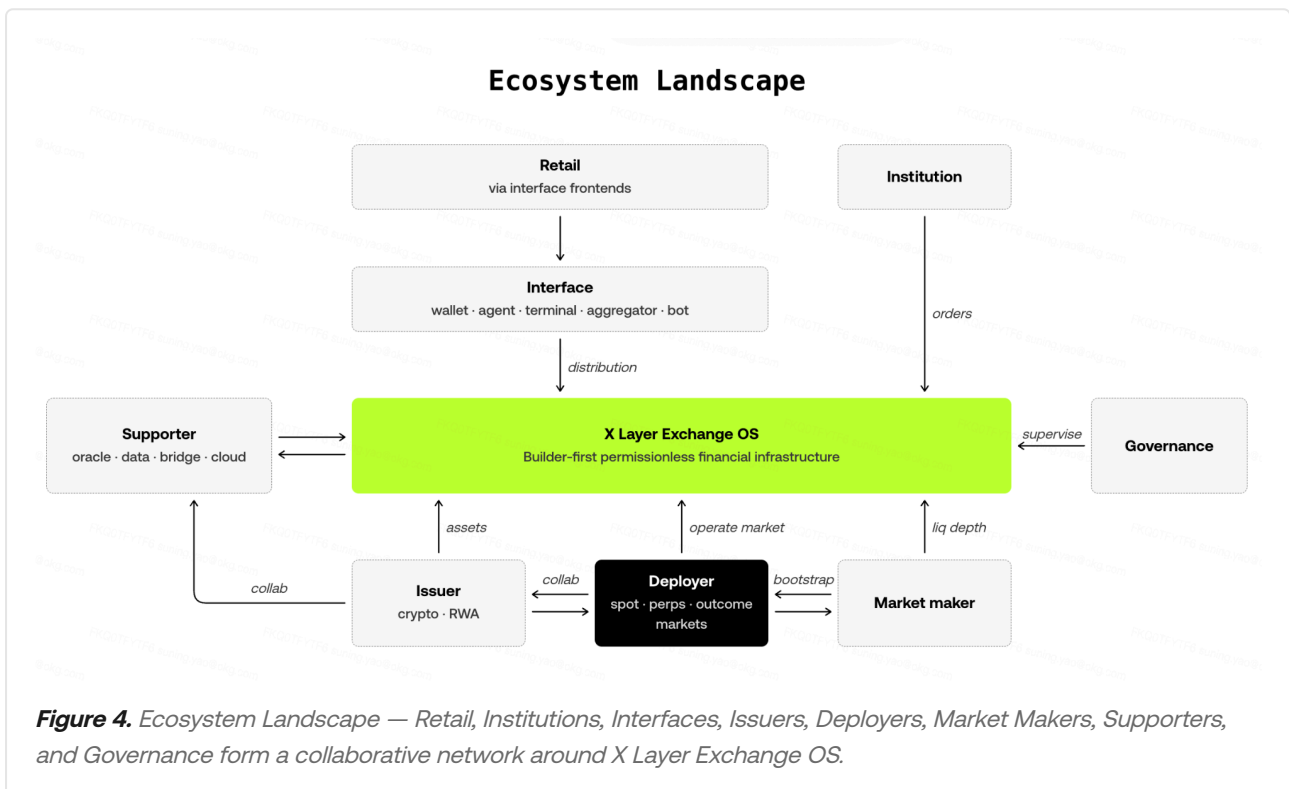
The governance model will prevent any single party from exerting unilateral control over governance outcomes, ensuring that decisions rest on consensus rather than power.

The governance model operates within a governance contract system on the X Layer. Proposals, votes, resolution execution, slashing rulings, and fund allocation — every action is transparent, publicly verifiable, and tamper-proof. Different categories of decisions are mapped to institutional processes balancing efficiency with prudence.

5 Ecosystem

5.1 Ecosystem Landscape

Exchange OS makes market creation permissionless at the protocol layer, with unified user accounts, balances, and risk systems across the network. Liquidity, order flow, assets, and strategies flow freely in the markets — turning Exchange OS into a collaborative ecosystem rather than a collection of isolated venues.



Trading markets should not be segregated. Deployers, asset issuers, users, distribution platforms, supporters, and market makers should collaborate, enabling liquidity and order flow to be distributed, and allowing trading strategies to be composed across markets.

5.2 Deployer — People Who Identify Market Needs and Turn Them Into Realities

Deployable market types include, but are not limited to:

- **Spot Markets** — native crypto assets, on-chain RWA markets, including tokenized treasuries, tokenized equities, or commodities.

- **Perps Markets** — assets that include crypto-native assets, TradFi tokens and pre-market assets.
- **Outcome Markets** — sports events, macro events, or on-chain event outcomes.

At the protocol layer, deployers gain access to:

- **Full trading infrastructure** — matching, margining, liquidation, and settlement engines.
- **Service model positioning** — institutional-grade services to KYC-verified users via regulated structure, or permissionless on-chain experience to Web3-native users.
- **Full market-parameter discretion** — oracle data sources, margin mode, leverage caps, and fee model.
- **Ecosystem collaboration supports** — interface, market maker liquidity, and supporter infrastructure services.

Access path:

Stake OKB → obtain venue creation rights and listing quota → configure market parameters → build a frontend or connect to third-party Interfaces.

5.3 Issuer — Enables Assets

Issuers provide the tradable assets that markets are built around, including tradable tokens.

At the protocol layer, issuers gain the benefit that they only need to issue once, then every venue can list it. The same asset can be listed by any deployer with permissionless integration.

Access path:

Issuers do not need to integrate directly with the protocol layer. Deployers can then independently select and assemble them into markets.

5.4 Interface / Distribution — Wherever the Users Are, That's Where the Market Is

Interfaces determine where users discover, access, and trade markets on Exchange OS. This layer directly determines the adoption of Exchange OS.

Interfaces include:

- **Wallets** — Web3 wallets that bring trading into users' everyday asset entry point.
- **Terminals** — trading terminals for high-frequency and institutional traders.
- **AI Agent Platforms** — automated interfaces where AI agents can place orders on behalf of users.

- **Bots / Social Trading Platforms** — copy trading and social trading products.
- **Aggregators** — optimal trading experience across multiple deployer markets.

At the protocol layer, interfaces gain:

- **One integration, full-network access** — a single protocol integration reaches all deployer venues, without adapting separately to each one.
- **Unified account** — users can see and manage their positions, margin, and PnL across all deployer markets directly inside the interface, without switching between products.
- **Zero infrastructure burden** — matching and liquidation are handled by the protocol layer, allowing interfaces to focus on the user-facing experience.

5.5 Supporter — Service Providers

Supporters are the service providers that every market on Exchange OS needs. Oracles, data services, bridges, and cloud services are all prerequisites for efficient market operation.

At the protocol layer, supporters gain:

- **Permissionless access** — The protocol does not pre-integrate any specific provider; all high-quality supporters have a fair opportunity to participate.
- **Cross-market reuse** — One integration can be selected by multiple deployers and is not bound to a single venue.
- **Market-based selection** — Deployers publicly evaluate and switch providers, creating a market-based signal for service quality.

Access path:

Supporters are directly selected by deployers as optional data-source, bridge, cloud, or other service providers. No protocol-side authorization is required.

5.6 Market Maker — The Lifeline of Financial Markets

Market makers are the liquidity foundation of financial markets.

At the protocol layer, market makers gain:

- **Maker-friendly matching** — The matching rule enforces a maker-friendly principle for execution-order guarantees, helping handle toxic-flow more efficiently.
- **Cross-market capital efficiency** — Unified accounts, balances, and margin let the same inventory quote spot, perps, and outcome markets.

Access path:

Market makers connect directly to deployer venues through the protocol's API service.

5.7 Open System of Exchange OS

Exchange OS redefines how markets are created, operated, and connected. It moves financial markets into an open generation initiated by deployers, co-built by the ecosystem, and secured by protocol-level fairness.

- **Deployers** no longer need to build from scratch. They build markets on top of high-performance infrastructure.
- **Issuers** no longer need to build their own exchanges. They only need to issue assets on X Layer.
- **Interfaces** no longer need to rebuild the full trading stack. They only need to build the frontend.
- **Supporters** no longer need to wait for approval from a centralized operator. They can freely support the ecosystem.
- **Market makers** no longer need to struggle with toxic flow on their own. The protocol's built-in mechanisms make it easier to handle.

The Exchange OS ecosystem isn't a partner network — it's a system of interdependent contributors.

Its value does not come from X Layer alone, but from every participant contributing. The protocol weaves these contributions into a complete financial market network: one that can continue to grow, self-expand, and operate independently of any single initiator.

6 Conclusion

Exchange OS is not only a protocol upgrade. It is an innovation to build the next generation of on-chain financial markets on X Layer.

By moving market creation from platform approval to permissionless deployment, Exchange OS gives builders and institutions a shared foundation to launch, operate, and distribute new markets. The protocol provides the neutral infrastructure — matching fairness, asset self-custody, risk enforcement, and final settlement — while each venue retains the flexibility to define its own market design, risk preferences, access model, and user scope.

This creates a new operating model for financial market participants. Deployers can launch markets without rebuilding the full exchange stack. Issuers can bring assets into an open network where they can be discovered and priced. Interfaces can distribute markets to users wherever they are. Supporters can supply the data, oracle, cloud, and infrastructure services that every venue needs. Market makers can provide liquidity in a shared market layer.

Exchange OS turns separate venues into a shared market network. One high-performance stack powers matching, margin, liquidation, and settlement. Unified accounts, balances, and margin allow capital to move across markets. Liquidity can be shared. Strategies can be composed. Markets can be created, connected, and scaled by the ecosystem.

For institutions, asset issuers, infrastructure providers, and application builders, Exchange OS offers a new path: build your own market, define your own risk preferences, serve your own users, and plug into a shared financial network.

Build with us on X Layer Exchange OS.

This paper is provided for information only. Nothing in this paper constitutes investment advice, a financial promotion, a recommendation to buy or sell any digital asset, or legal advice. Readers should do their own research and consult their own advisers before making any decision relating to the use of X Layer, Exchange OS, or any market deployed on Exchange OS.

*Authored by **OKX Exchange OS Working Group**. Released for feedback. Wire-format schemas, security analysis, and reference-implementation details are maintained in the Exchange OS specification. This paper does not constitute a security audit, and it does not constitute an offer of services.*