WHITE PAPER

Koinos AI

A Local-First, Decentralized Artificial Intelligence Network

Your hardware first. The network when you need it.

LOCAL-FIRST
Run private AI on hardware you control.

EARN
Contribute unused compute and earn KAI.

SCALE
Burst into decentralized compute on demand.

Working Draft v0.2 | August 2026

Core Thesis
Koinos AI combines private local inference, an open compute marketplace, and a self-hosted developer API. Users own the first layer of compute; the network supplies additional intelligence only when it is needed.

Abstract

Artificial intelligence is increasingly delivered as a centralized cloud service. Koinos AI proposes a different architecture: useful AI begins on hardware controlled by the user, while a decentralized network supplies additional compute when local resources are insufficient. Computer owners can contribute otherwise idle CPU and GPU capacity and earn KAI for verified useful work. Developers can self-host an OpenAI-compatible API, use local or private GPUs first, and burst into decentralized capacity on demand. Koinos provides the economic, ownership, and settlement layer without forcing ordinary users to understand blockchain mechanics.

The protocol is designed to progress from decentralized inference toward model evaluation, fine-tuning, distributed training, and eventually community-funded Koinos-native models. The project deliberately separates long-term ambition from the minimum product required to prove the network: install, run useful AI locally, contribute compute with one action, earn KAI, and spend KAI on stronger network AI.

1. The Problem

Modern AI is powerful, but the dominant delivery model concentrates compute, data, pricing, and access inside a small number of centralized providers. At the same time, millions of modern computers contain capable GPUs, CPUs, memory, and storage that remain underutilized for much of the day. Developers face a related tradeoff: self-hosting offers control but lacks elasticity, while cloud APIs offer elasticity but require dependence on third-party infrastructure.

- Centralized inference can require sensitive prompts and files to leave infrastructure controlled by the user.
- Self-hosted AI is attractive for privacy and cost, but a single workstation cannot satisfy every workload or traffic spike.
- Idle consumer and professional compute is fragmented and difficult to monetize as useful AI infrastructure.
- Blockchain-based compute projects often expose protocol complexity instead of presenting a usable AI product.

2. The Koinos AI Approach

Consumer
Run AI locally. Earn KAI with unused compute. Use the network when you need more.

Developer

> Self-host your AI API. Use your hardware first. Scale with decentralized compute.

Koinos AI treats decentralized compute as an extension of personally owned and privately operated infrastructure. The desktop application is useful even if a user never owns KAI or contributes compute. The developer gateway is useful even if every request remains local. The public network becomes valuable when the local or private layer is not enough.

Layer	Purpose	Typical Cost
Local	Inference on the user's own hardware	No KAI network charge
Private pool	Organization-controlled or trusted compute	Internal economics
Verified network	Public decentralized provider network	KAI settlement
Confidential (future)	Attested confidential-compute providers	Premium KAI settlement

3. How the Network Works

Each Koinos AI installation can act as a client, a local inference server, and - when enabled - a compute worker. A scheduler matches network jobs to eligible workers according to model availability, measured capability, price, latency, reputation, privacy class, region, and current capacity. AI execution occurs off-chain. Koinos records economic ownership and settlement state rather than prompts, responses, model weights, or individual inference steps.

Step	Network action
1	A user or API submits an AI request.
2	The local router applies privacy and spending policy.
3	If local/private compute is sufficient, the request stays there.
4	If network compute is permitted and needed, the scheduler selects an eligible provider.
5	The provider executes the standardized workload and returns the result.
6	Receipts, reputation, challenges, and verification establish

	confidence in useful work.
7	Usage is aggregated and settled in KAI through Koinos.

4. The KAI Economy

KAI is the native economic coordination asset of the Koinos AI network. Local inference does not require KAI. KAI is used when participants consume resources supplied by other participants and for related protocol services such as model royalties, verification, training, and future agent spending.

- Compute providers earn KAI for verified useful work.
- Model creators may receive bounded royalties for paid use of registered models.
- Verifiers and future schedulers can be compensated for useful protocol services.
- Providers can earn KAI without first purchasing KAI or KOIN.
- Staking may unlock higher-value workloads, but is not intended as passive-yield inflation.

Economic Principle

AI compute is economically priced independently of KAI. KAI is the settlement quantity. If KAI's market price changes, the amount of KAI needed for the same underlying workload can adjust by pricing epoch instead of making AI services arbitrarily more or less expensive.

5. Token Supply and Bootstrap Economics

The current working genesis supply is 1,000,000,000 KAI. Allocation percentages, bootstrap subsidy budgets, market-distribution levels, burn rates, fees, and any future useful-work tail issuance remain provisional until the economic model is stress-tested across GPU classes, utilization, electricity costs, AI demand, KAI market prices, and liquidity scenarios.

Working parameter	Current direction	Status
Genesis supply	1,000,000,000 KAI	Provisional
Liquidity & market bootstrap	50M KAI reserve; up to ~30M contemplated for initial price discovery	Simulation required
Models & research	~80M KAI working reserve	Provisional
Compute bootstrap	Large useful-work reserve; targeted/capped subsidies, not fixed-per-job inflation	Simulation required
Usage burn	Bounded mechanism; exact rate TBD	Provisional

Tail issuance	Only if later needed for useful work; tightly capped	Provisional

6. Market Bootstrap and Liquidity

Public market bootstrapping should follow demonstrated product utility. The preferred sequence is working alpha, working beta, providers earning KAI, users consuming network AI, and a functioning developer API before a public KAI price-discovery event. A staged on-chain order-book distribution can establish an initial market price without requiring the protocol to fully fund a large AMM pair at an arbitrary valuation.

A working concept reserves 50M KAI for liquidity and market bootstrap, with up to approximately 30M KAI offered through transparent rising price levels. Assets accumulated through price discovery can help supply the paired side of subsequent KoinDX liquidity. Exact price levels, amounts, and use-of-proceeds rules remain subject to simulation and legal review.

7. Provider Experience

Provider onboarding is intentionally consumer-grade. A compatible user should be able to install Koinos AI, allow hardware detection and benchmarking, and enable earning without choosing inference engines, model quantizations, CU prices, wallets, or blockchain nodes.

Default UI	Meaning
Start Earning	Enable the compute worker.
Balanced	Yield hardware to personal use and favor safe background operation.
Maximum Earnings	Use available resources more aggressively.
Low Power	Reduce energy and thermal impact.
Automatic pricing	Use network-recommended compensation rates.
Automatic models	Install/manage models appropriate to hardware and network demand.

Personal use receives priority by default. Gaming or other demanding foreground activity can pause new jobs. Battery, thermal, bandwidth, and resource limits protect consumer hardware. Entry-level providers require neither KAI stake nor KOIN/MANA ownership.

8. Verification and Reputation

Koinos AI uses layered proof of useful computation rather than a single universal proof mechanism. Providers do not self-report payable Compute Units. Standardized workload profiles, signed receipts, benchmarks, hidden challenge jobs, randomized audits, reputation, and selective redundant execution establish confidence in completed work. Verification strength increases with job value and risk.

Reliability failures such as power loss or an internet outage are not treated as fraud. They primarily reduce reliability reputation or payment eligibility. Slashing is reserved for sufficiently evidenced malicious behavior such as forged receipts, deliberate fake compute, or protocol manipulation.

9. Models and the Creator Economy

The network is model-agnostic. Mainstream users interact with capability aliases such as Koinos Fast, Koinos Smart, Koinos Code, and Koinos Vision. The underlying open model can evolve while advanced users and developers retain the ability to pin exact packages and versions.

Class	Meaning
Koinos Models	Curated defaults selected and packaged for broad network use.
Verified Models	Third-party packages that pass identity, licensing, compatibility, benchmark, and security checks.
Community Models	Open ecosystem models that users may choose without implying official verification.

Every network-compatible model is represented by versioned cryptographic commitments to weights, tokenizer, quantization, runtime, configuration, license, and creator metadata. Model files remain off-chain. Creators may optionally receive bounded KAI royalties for paid network usage, subject to licensing and protocol rules.

10. Self-Hosted API and Developer Platform

> **Primary Developer Selling Point**
> Self-host an OpenAI-compatible Koinos AI API on infrastructure you control, use your own GPUs first, and automatically access decentralized overflow capacity when local resources are not enough.

The self-hosted gateway is not merely a convenience around the public network. It is a primary product. A developer can run compatible applications entirely against local compute, combine multiple private GPUs into a trusted pool, or selectively fall back to the public network according to explicit cost, privacy, and availability policies.

Routing mode	Behavior
Local First	Prefer the machine running the gateway; use permitted fallback only when needed.
Private Pool First	Prefer organization-controlled workers before public providers.
Local Only	Never send inference to remote providers.
Network Only	Use the public network for consistent remote execution.
Lowest Cost / Fastest	Optimize routing according to developer policy.

The API targets OpenAI compatibility wherever practical, including familiar chat/model endpoints, streaming, embeddings, structured output, and tool/function calling over time. API keys are scoped application credentials rather than blockchain private keys. Projects support budgets, limits, model permissions, usage reporting, and later batch/spot compute.

11. One Runtime, Two Networks

Koinos AI Core can also become a simple gateway into the infrastructure that secures the blockchain beneath the AI economy. The full Koinos node and block-producer components are optional modules rather than requirements for using local AI or earning KAI. This creates a second native participation path without conflating the economics of KAI and KOIN.

Two Complementary Earning Paths

AI providers perform verified AI workloads and earn KAI. Koinos block producers secure consensus through Proof-of-Burn and earn KOIN under the Koinos protocol. Koinos AI should surface both opportunities in one interface without paying duplicate KAI rewards merely for producing blocks.

Module	Contribution	Native Reward
AI Compute Provider	Supplies useful inference and future AI workloads	KAI
Koinos Full Node / RPC	Strengthens blockchain access and can serve measurable Koinos AI infrastructure traffic	Potential KAI only for measurable AI-network services
Koinos Block Producer	Participates in Proof-of-Burn consensus and secures Koinos	KOIN

The consumer experience can expose these as optional modules under a unified Earn area: Share AI Compute - Earn KAI; Support the Koinos Network - Run a Full Node; Produce Koinos Blocks - Earn KOIN. Running a full node does not itself imply KAI compensation. A future Infrastructure Provider class may earn KAI only when it delivers measurable services needed by Koinos AI, such as approved RPC capacity, verification, or other protocol infrastructure.

This preserves clean economic roles: KAI coordinates useful AI and Koinos-AI infrastructure services, while KOIN remains the native consensus and resource asset of Koinos. It also strengthens decentralization at both layers: community hardware can supply AI compute while optional local nodes reduce dependence on centralized blockchain RPC infrastructure.

Block production remains an advanced opt-in because it requires KOIN/VHP and carries different operational and security considerations. The first public AI release does not depend on this module; one-click full-node and block-producer management fit naturally into the Open Compute era.

12. Identity, Smart Accounts, and MANA

Mainstream onboarding should use familiar authentication such as passkeys, Google/OpenID, or email-based flows while preserving a Koinos-capable account underneath. Smart-account architecture can separate owner authority from worker, API, and future agent permissions. A background worker should never require unrestricted authority over a user's KAI.

Koinos MANA sponsorship is a major UX advantage. Approved network operations can be sponsored so a new participant can create or use a network account and earn KAI without first purchasing KOIN merely to pay transaction resources. Sponsored actions should be allowlisted, rate-limited, and protected against resource abuse.

13. Privacy and Security

Koinos AI does not claim that ordinary remote inference is automatically confidential. Local execution provides the strongest simple privacy guarantee. Private pools restrict execution to explicitly trusted hardware. Verified public providers use encrypted transport and network controls, while future confidential-compute providers may use hardware attestation and protected execution environments.

- Prompts, responses, documents, embeddings, and conversation content are not stored on Koinos.
- Schedulers should receive only metadata needed to match a job wherever practical.
- Official workers should not intentionally persist or expose customer prompt/response content.
- Worker sandboxes should not receive unrestricted access to provider files, credentials, wallets, or arbitrary shell/network resources.
- User chat history and local RAG remain local by default; optional sync should be encrypted.

14. Agents and Machine Permissions

Agents are a major future layer, but they are not a prerequisite for the first network. Koinos AI Core is designed so tools execute through explicit permission boundaries. Sensitive credentials remain in a local secure vault. Smart-account permissions can later give agents narrowly scoped, revocable, and time-limited AI spending authority without exposing unrestricted wallet keys.

15. Distributed Model Development

Decentralized inference is the first step, not the endpoint. Koinos AI is intended to progressively support useful model-development workloads: evaluation, synthetic-data generation, data classification, fine-tuning, distillation, cluster training, and eventually experimental decentralized training. Consumer GPUs are not treated as substitutes for tightly interconnected superclusters, but they can contribute substantial parallelizable work around model creation and improvement.

Long-term governance may direct an AI Models & Research reserve toward network providers to create openly accessible Koinos-native models. User conversations are not automatically training data; any future contribution of private interaction data requires explicit opt-in.

16. Governance and Progressive Decentralization

The protocol should not launch as an unrestricted token-voting DAO. Early administration should use role-separated multisignature authorities, transparent parameters, timelocks for meaningful economic changes, and narrowly scoped emergency powers. Over time, governance can expand toward public proposals, community signaling, bounded on-chain execution, and competing schedulers/gateways.

Some economic parameters should have constitutional bounds. Governance may adjust within those ranges but should not be able to arbitrarily redefine fundamental issuance, fees, burns, or user custody protections without an explicit protocol migration.

17. Roadmap

Era	Focus
I - Network Genesis	Local AI, desktop chat, provider earning, KAI settlement, self-hosted API, decentralized overflow.
II - Open Compute	Public providers, additional models/hardware, private pools, batch/spot compute, creator marketplace.
III - Autonomous AI	Persistent agents, tools, smart-account budgets, machine payments, confidential compute.
IV - Distributed Intelligence	Synthetic data, fine-tuning, distillation, distributed training, Koinos-native models.

18. V1 Success Criteria

- A normal user can install Koinos AI and immediately run a useful local model.
- A compatible user can enable earning with essentially one action.
- A real decentralized inference job can be routed to that provider and completed reliably.
- Useful work produces reliable KAI settlement.
- A user can spend KAI on stronger network AI.

- A developer can self-host an OpenAI-compatible endpoint, use local compute, and transparently overflow into decentralized compute.

19. What Remains Provisional

The architecture is substantially defined, but a responsible public white paper must distinguish the protocol thesis from parameters that still require evidence. Exact token allocations, provider subsidy budgets, staged market prices, KAI reference-price methodology, CU formulas, protocol fees, burn rates, verification frequencies, launch model choices, and some bridge/payment integrations remain subject to simulation, engineering validation, and legal review.

> **Long-Term Thesis**
> Koinos AI aims to create an open AI infrastructure layer in which users own the first layer of intelligence, developers retain control of their infrastructure, idle computers become useful on-demand capacity, and the network can eventually help finance and create the models it serves.

Conclusion

The internet connected computers to information. Cloud AI connected applications to centralized intelligence. Koinos AI proposes a third model: personally controlled AI that can participate in an open economic network when more compute is needed. The blockchain is not the product users are asked to operate; it is the settlement and ownership layer underneath a familiar AI experience. If successful, Koinos AI turns the world's fragmented idle compute into an extension of every user's and developer's own infrastructure.

Koinos AI - Your hardware first. The network when you need it.