SnorlapAI: A Composable Protocol for Sovereign On-Chain Intelligence

Whitepaper Version: v6.4, June 2025

SnorlapAI Labs

Abstract

The maturation of the decentralized economy has precipitated a crisis of complexity. The exponential growth of on-chain protocols, the fragmentation of cross-chain liquidity, and the increasing sophistication of decentralized governance have created a profound "autonomy gap"—a chasm between the strategic intent of users and the capacity of existing tools to execute it. Current solutions are inadequate, forcing a trade-off between capability, security, and incentive alignment. Centralized AI agents offer power but demand a complete surrender of sovereignty, operating as opaque, censorable black boxes. Conversely, on-chain logic, while secure, is constrained by monolithic architectures, and first-generation decentralized AI networks have yet to bridge the gap between commoditized intelligence and high-stakes, verifiable on-chain action.

SnorlapAI resolves this impasse. It introduces a decentralized, modular agent network designed to orchestrate a workforce of specialized, autonomous agents—termed "Tools"—to execute complex, verifiable tasks across the Web3 ecosystem. Recognizing the immense engineering challenge of building a fully trustless system from day one, SnorlapAI will execute a strategic, two-phase architectural roadmap. The protocol will launch with a pragmatic and secure architecture (Phase 1) leveraging battle-tested Trusted Execution Environments (TEEs) and hardware attestations to deliver immediate utility and solve the market's "cold start" problem. Following this, the network will transition to its trustless endgame architecture (Phase 2), built upon a general-purpose Zero-Knowledge Virtual Machine (zkVM) and a decentralized auditor network secured by zk-STARKs and a restaking model. The native protocol token, $SNOR, powers the economic flywheel in both phases, aligning the incentives of users, developers, and, ultimately, network auditors. By resolving the Web3 Agent Trilemma through this phased evolution, SnorlapAI establishes a new protocol layer for Sovereign AI—agents that are simultaneously capable, composable, and economically aligned with their users. This protocol serves as the essential orchestration layer for the next generation of autonomous on-chain finance, governance, and operations, creating a decentralized, autonomous workforce for the digital economy.

SnorlapAI: A Composable Protocol for Sovereign On-Chain Intelligence 1
1. The Dawn of Sovereign AI: A New Paradigm for On-Chain Autonomy 4
1.1. The Crisis of On-Chain Complexity 4
1.2. Formalizing the Web3 Agent Trilemma: Capability vs. Composability vs. Alignment 5
1.3. The SnorlapAI Thesis: Orchestrating a Decentralized, Verifiable Workforce 6
2. The Competitive Landscape: An Unsentimental Analysis 6
2.1. The Centralized Leviathans: Benchmarking Capability 6
2.2. The Decentralized Frontier: Unbundling the AI Stack 7
2.3. The Infrastructure Bedrock: Dependencies and Synergies 9
3. The SnorlapAI Solution: A Modular, Composable, and Sovereign Network 12
3.1. Core Principles: Modularity, Sovereignty, and Emergent Intelligence 12
3.2. The Ideal User Experience: An Intent-Centric Interface 13
4. Protocol Architecture: A Phased Evolution from Pragmatism to Purity 14
4.1. Phase 1: The Pragmatic Launch Architecture 14
4.1.1. Phase 1 Security Model: TEEs and Their Limitations 15
4.2. Phase 2: The Trustless Endgame Architecture 16
4.3. The Universal Five-Layer Stack 17
4.4. Cross-Chain Orchestration: A Hybrid IBC and LayerZero Strategy 17
4.4.1. Cross-Chain Security Considerations 17
5. The $SNOR Token: Engineering a Self-Sustaining Economic Flywheel 18
5.1. Token Overview and Distribution 18
5.2. The Phased Marketplace: A Game-Theoretic Analysis 19
5.3. Value Accrual and Deflationary Mechanics 21
5.3.1. DAO Governance Guardrails for Phase 1 21
6. High-Value Use Cases and Go-to-Market Strategy 22
6.1. DeFi Agent Suite 22
6.2. DAO Operations Suite 22
6.3. A Phased Go-to-Market Strategy 22
7. Roadmap & Future Vision 22
7.1. Development Phases 23
7.1.1. The Path to Trustlessness: A Migration Framework 23
7.2. Future Vision: The Protocol for Autonomous Work 24
8. Core Contributors & Path to Decentralization 24
8.1. Core Contributors 24
8.2. A Deliberate Path to DAO Governance 25
Appendix 25
A. Formal Definitions and Mathematical Preliminaries 25
A.1. Reputation Scoring Function (Phase 2) 26
A.2. Reputation-Weighted VRF Auction Mechanism (Phase 2) 26
B. Overview of Verification Technologies 27
B.1. Trusted Execution Environments (TEEs) and Hardware Attestation (Phase 1) 27
B.2. zk-STARKs (Phase 2) 27
C. Glossary of Terms 27

1. The Dawn of Sovereign AI: A New Paradigm for On-Chain Autonomy

1.1. The Crisis of On-Chain Complexity

The core problem SnorlapAI addresses is not the administrative overhead of the traditional knowledge worker, but the exponential and unforgiving complexity of the on-chain world. As the Web3 ecosystem matures, manual operation ceases to be merely inefficient; it becomes a critical vector for financial loss, security breaches, and strategic failure. This escalating complexity has created a significant "autonomy gap": a widening chasm between a user's high-level strategic intent and the capacity of existing tools to execute that intent securely, optimally, and autonomously.

This complexity manifests across three primary domains:

- **Decentralized Finance (DeFi):** The DeFi landscape is a combinatorial explosion of protocols, liquidity pools, yield farms, and derivative instruments spread across a growing number of Layer 1 (L1) and Layer 2 (L2) networks. For an individual user or a DAO treasury, optimizing capital allocation is no longer a simple matter of choosing a protocol. It is an NP-hard optimization problem, akin to the Traveling Salesman Problem, where the "cities" are protocols and the "distances" are transaction costs, slippage, and smart contract risk. A single agent using a greedy algorithm results in demonstrably suboptimal returns, leaving significant value on the table. Manual portfolio management in this environment is akin to navigating a labyrinth blindfolded, where a single misstep can lead to irreversible loss.
- **DAO Governance and Operations:** Decentralized Autonomous Organizations (DAOs) manage billions of dollars in assets and make critical decisions affecting entire ecosystems. Yet, their operational toolkit remains primitive. Treasury management often relies on multi-signature wallets and manual execution of governance proposals. This introduces significant operational drag, the potential for human error, and critical security risks. The delay between a governance vote and its execution can be exploited, and complex treasury management strategies (e.g., dynamic rebalancing, yield farming) are often too operationally intensive to implement effectively, leaving DAOs unable to act with the agility the market demands.
- **Cross-Chain Fragmentation:** The multi-chain future has arrived, but it has brought with it a fractured landscape of liquidity and state. Moving assets and executing tasks across different chains requires navigating a complex web of bridges, each with its own security assumptions, fees, and latency. Orchestrating a single logical transaction—for example, swapping an asset on Ethereum, bridging it to Arbitrum, and providing liquidity on a native DEX—is a multi-step, error-prone process that exposes the user to significant risk at each stage. This fragmentation creates silos of value and opportunity, accessible only to the most sophisticated and diligent users.

Existing AI solutions are ill-equipped to handle this environment. Centralized agents cannot be trusted with private keys or the execution of high-value transactions. On-chain automation scripts are too rigid and lack the adaptive intelligence to navigate changing market conditions. SnorlapAI is engineered specifically to fill this autonomy gap, providing the missing layer of intelligent, verifiable, and secure orchestration that the decentralized economy requires to scale.

1.2. Formalizing the Web3 Agent Trilemma: Capability vs. Composability vs. Alignment

The pursuit of effective artificial intelligence agents in the digital asset space is constrained by a fundamental challenge, which we formalize as the Web3 Agent Trilemma. This framework posits that any given agent architecture is forced to compromise on one of three essential properties, making it impossible to achieve all three simultaneously with current approaches. The tension between these vertices defines the current market landscape and illuminates the strategic opening for a new architectural paradigm. This trilemma represents a market failure: there is clear, demonstrated demand for agentic capability, yet existing solutions are fundamentally incompatible with the core tenets of Web3, creating a vacuum that SnorlapAI is designed to fill.

The three vertices of this trilemma are:

1. **Capability (Centralized Agents):** This vertex is defined by an agent's ability to independently plan and execute complex, multi-step tasks in dynamic environments. Centralized platforms like Manus.im exemplify this, orchestrating powerful foundation models like Anthropic's Claude and Alibaba's Qwen within sandboxed cloud environments to perform sophisticated operations. They demonstrate what is possible with unfettered access to computational resources. However, this capability is achieved within a "walled garden." The underlying logic is opaque, the execution is unverifiable, and the user is entirely subject to the platform's control, censorship, and potential failure. This model, which demands total surrender of sovereignty, is philosophically and practically incompatible with the trust-minimized ethos of Web3.
2. **Composability (On-Chain Logic):** This vertex represents the native strengths of blockchain protocols. On-chain logic, codified in smart contracts, is transparent, immutable, and seamlessly composable. However, this security comes at the cost of performance and flexibility. Monolithic architectures, even high-performance L1s like Monad that introduce parallel execution to increase throughput, are fundamentally settlement layers, not dynamic agentic frameworks. Their rigid, gas-constrained execution environments create bottlenecks that limit the operational complexity and adaptive intelligence required for true autonomy. They are built to finalize state, not to perform the complex, iterative, and often off-chain computations that define advanced agentic behavior.
3. **Alignment (First-Gen DeAI):** This vertex has been pioneered by first-generation decentralized AI (DeAI) networks. Protocols like Bittensor have successfully created token-based markets for commoditized machine intelligence, organizing contributors into specialized "subnets" to produce high-quality AI outputs. Fetch.ai has built a framework for deploying general-purpose "digital twins" to automate tasks. While these networks have proven the viability of tokenized incentives, their focus remains on either providing raw intelligence outputs or generic agent frameworks. They do not solve the core challenge of orchestrating secure, verifiable, and high-stakes actions in the complex, adversarial environment of on-chain finance and governance.

The Web3 Agent Trilemma creates a state of market tension. The undeniable power of centralized agents creates demand, but their architecture is philosophically and practically untenable for a decentralized world. The proliferation of specialized DeAI protocols—unbundling intelligence (Bittensor), inference (GaiaNet), and infrastructure (0G, Monad)—mirrors the "tool entropy" of the legacy SaaS world, but at the protocol level. This

fragmentation necessitates a new layer of aggregation and orchestration.

1.3. The SnorlapAI Thesis: Orchestrating a Decentralized, Verifiable Workforce

SnorlapAI is architected as the definitive solution to this trilemma. It is not merely another agent platform but a new protocol layer that integrates the strengths of each vertex. The core thesis is that a new architecture is required—one that is verifiable to be trusted, modular to be capable, and incentivized to be aligned.

SnorlapAI enables the deployment of highly capable, specialized agents, ensures their actions are verifiably secure and composable on-chain, and aligns all network participants through a robust, self-sustaining economic model. In doing so, SnorlapAI creates a new category: Sovereign AI, where agents are powerful extensions of the user's will, not instruments of a platform's control. It achieves this by establishing a protocol for coordinating a decentralized workforce of autonomous, specialized, and economically-aligned "Tools," effectively becoming the orchestration layer for the next generation of autonomous on-chain activity.

2. The Competitive Landscape: An Unsentimental Analysis

To engineer a durable solution, one must first conduct a rigorous and unsentimental analysis of the existing landscape. The market for AI agents is bifurcating into two distinct philosophical camps: the centralized, "walled garden" approach, which prioritizes capability at the cost of user sovereignty, and the decentralized, "open frontier" approach, which champions sovereignty but often struggles with fragmentation and focused utility. SnorlapAI is positioned to transcend this dichotomy by learning from the strengths and exploiting the weaknesses of both.

2.1. The Centralized Leviathans: Benchmarking Capability

The centralized agent model serves as both a benchmark for capability and a cautionary tale for the Web3 ecosystem. These platforms demonstrate what is possible with unfettered access to computational resources and data, but also reveal the inherent dangers of such concentration of power.

- **Manus.im:** Hailed as a significant advance in autonomous agents, Manus is an orchestrator of multiple powerful foundation models, including Anthropic's Claude and Alibaba's Qwen, operating within a sandboxed Linux environment in the cloud. This architecture allows it to perform complex, multi-step tasks that require web browsing, code execution, and file manipulation, achieving state-of-the-art performance on benchmarks like GAIA. However, its centralized nature is its Achilles' heel. The system is a complete black box; users have no visibility into or control over the precise models being used, the data being collected, or the security of the underlying infrastructure. Early reports indicate significant reliability issues, including server overloads and high task failure rates, highlighting the fragility of a centralized point of control. Its business model, an invite-only beta with a per-task fee of approximately $2, further entrenches it as a gatekeeper. For the Web3 ecosystem, which demands transparency, verifiability, and censorship-resistance, Manus represents an existential threat—a powerful tool of automation that could be used to control access to the decentralized web itself.

- **ChatGPT Operator:** OpenAI's foray into agentic AI takes a different technical approach. Branded as a "Computer-Using Agent" (CUA), Operator leverages vision models like GPT-40 to interpret graphical user interfaces (GUIs) from screenshots, interacting with web pages by simulating mouse clicks and keyboard inputs rather than using APIs. In its current "research preview" state, however, Operator is widely regarded as functionally ineffective. It struggles with fundamental web interactions like handling logins, payments, and CAPTCHAs, often handing control back to the user at the first sign of complexity. Its performance is slow and prone to hallucinations, making it more of a "cool demo" than a productive tool. Operator's struggles are instructive: they reveal the immense difficulty of building a reliable general-purpose agent. Its monolithic, "one-size-fits-all" approach stands in stark contrast to SnorlapAI's modular design, which relies on a network of specialized, verifiable Tools, each designed and audited for a specific, high-stakes task.

2.2. The Decentralized Frontier: Unbundling the AI Stack

The decentralized AI space is a vibrant, rapidly evolving ecosystem of projects, each tackling a different facet of the challenge. This unbundling of the traditional AI stack—separating data, models, compute, and application logic into distinct protocols—is a hallmark of a nascent, innovative market. It also creates the strategic opening for a re-bundler: a protocol that can aggregate these disparate components into a cohesive, valuable service. SnorlapAI is architected to be this definitive aggregator at the most valuable layer: the Agentic Application Layer. It orchestrates the unbundled components of the DeAI stack to deliver verifiable, autonomous on-chain actions.

- **Fetch.ai (ASI):** Fetch.ai is building a broad, horizontal platform for a decentralized "AI empowered agent based digital economy". Its core offering is a Layer-1 blockchain (built on the Cosmos SDK) and a framework (uAgents) for creating "Digital Twins"—autonomous agents designed for a wide array of use cases, from supply chain management to DeFi. The recent merger with SingularityNET and Ocean Protocol to form the Artificial Superintelligence Alliance (ASI) further broadens this mission. Fetch.ai's strength is its scope, but this is also its weakness. By aiming to be a general-purpose platform for all industries, it lacks the specialized focus and rigorous verification required for the high-stakes, adversarial environment of on-chain finance. SnorlapAI, in contrast, is a vertical specialist, laser-focused on providing a secure and verifiable framework for the most valuable and complex niche: on-chain operations.
- **Bittensor (TAO):** Bittensor is a revolutionary protocol for creating a decentralized marketplace for commoditized machine intelligence. It operates as a network of "subnets," where miners compete to provide the best outputs from AI models and are rewarded in $TAO based on the consensus of validators. Bittensor is not a direct competitor but a potential, powerful dependency. It commoditizes the output of models, not the execution of actions. A SnorlapAI Tool could act as a validator or consumer on a Bittensor subnet, leveraging its specialized intelligence (e.g., a financial analysis model) as one step in a larger, orchestrated workflow. This positions SnorlapAI as a higher-level orchestration layer that consumes the raw intelligence produced by networks like Bittensor.
- **Morpheus (MOR):** Morpheus is a peer-to-peer network designed to power "Smart Agents". Its key innovation lies in its tokenomic model, where capital providers stake yield-bearing assets (like stETH) to fund the network, and this yield is distributed as

rewards to contributors of code, compute, and community. However, the economic link is indirect; rewards are derived from external yield, not from fees generated by the network's actual economic activity. SnorlapAI's model, based on direct fee-for-service payments in $SNOR, creates a more direct and robust economic flywheel where the value of the token and the rewards to participants are explicitly coupled to the value being created on the network.

- **Olas (Autonolas):** Olas provides a unified network and open-source stack for building off-chain autonomous services, which are then anchored on-chain. Its architecture focuses on software composability, allowing developers to mint components and agents as NFTs and combine them into services. These services achieve consensus on their actions through mechanisms like Tendermint, operating on temporary blockchains. While Olas provides a robust framework for coordinating agent services, its security model relies on this off-chain consensus among operators rather than cryptographic proof of individual computations. SnorlapAI's primary differentiator is its non-negotiable requirement for cryptographic verifiability of every complex, high-stakes action via zk-STARKs. This provides a higher degree of trustlessness and security assurance, which is critical for financial operations and a dimension Olas does not natively address with the same mathematical rigor.
- **ElizaOS:** ElizaOS is an open-source framework, written in TypeScript, designed to make AI agent development more accessible, particularly for Web3 DApps and social interactions on platforms like Discord and Telegram. Its modular plugin architecture is a strength, validating the modular approach. However, its focus is broader, targeting general DApp integration and social bots rather than the high-security, verifiable computation required for high-stakes financial operations. SnorlapAI differentiates itself by focusing on the verifiability of complex actions, a critical requirement that ElizaOS does not natively address with the same cryptographic rigor.
- **Virtuals Protocol:** Virtuals Protocol focuses on the tokenization and co-ownership of AI agents, primarily for the gaming and entertainment sectors. It allows developers to launch AI agents as community-owned, revenue-generating assets, using its $VIRTUAL token as the base currency for agent creation and transactions. This represents a fundamentally different economic model: Virtuals treats the agent as an asset, whereas SnorlapAI treats the agent's execution as a service. SnorlapAI is building a protocol for autonomous work, while Virtuals is building a protocol for interactive IP and entertainment.

2.3. The Infrastructure Bedrock: Dependencies and Synergies

The most durable protocols are not those that reinvent every wheel, but those that strategically leverage best-in-class infrastructure. The DeAI market is not a zero-sum game; it is a modular stack being built in real-time. The most valuable position is not at the bottom (L1s) or in the middle (intelligence providers), but at the top: the agentic orchestration layer that aggregates the underlying components into a high-value, coherent service. SnorlapAI is strategically positioned as this aggregator for on-chain autonomous action.

- **Monad:** A high-performance, EVM-compatible L1 that re-architects the Ethereum Virtual Machine for parallel execution, targeting over 10,000 TPS. Monad is not an AI agent protocol; it is a foundational settlement layer. Its existence validates a key SnorlapAI thesis: the EVM itself is not the bottleneck if engineered correctly.

SnorlapAI is designed to be chain-agnostic but will be uniquely positioned to leverage the capabilities of next-generation L1s like Monad for ultra-fast task scheduling, auctioning, and settlement.

- **0G (0g.ai):** An infinitely scalable Data Availability (DA) layer designed for AI dApps. This is a critical synergy. The verifiable actions performed by SnorlapAI's Tools will generate a vast amount of trace data, logs, and cryptographic proofs (zk-STARKs). Storing this data on a traditional L1 would be prohibitively expensive. By integrating with a specialized DA layer like 0G, SnorlapAI's security and verification model becomes economically viable at a global scale, demonstrating crucial architectural foresight.
- **GaiaNet:** A decentralized infrastructure project for hosting fine-tuned AI models on a distributed network of edge nodes. Similar to Bittensor, GaiaNet focuses on decentralized inference and providing access to specialized knowledge via "AI twins". It is a network of knowledge providers, not action-takers. SnorlapAI can orchestrate Tools that query GaiaNet nodes for specific, expert information as part of a broader task, positioning SnorlapAI as the "action" layer that sits atop GaiaNet's "knowledge" layer.

Table 2.1: Comprehensive Competitive Matrix

Project	Architecture	Decentralization Model	Core Function	Token Utility	Key Differentiator / SnorlapAI's Edge
Manus.im	Centralized cloud sandbox; multi-model orchestrator (Claude, Qwen)	Fully Centralized (Proprietary)	Autonomous execution of complex web tasks.	Fiat / Per-task Fee (~$2)	Weakness: Opaque, censorable, and unverifiable. SnorlapAI offers verifiable, trustless execution.
ChatGPT Operator	Centralized; GUI interpretation via vision models (CUA)	Fully Centralized (OpenAI)	Automation of browser-based tasks.	Subscription Fee (ChatGPT Pro)	Weakness: Unreliable and functionally limited. SnorlapAI uses specialized, audited Tools for high-stakes, reliable execution.
SnorlapAI	Decentralized network of	DAO Governance;	Orchestration of verifiable,	Payment, Staking	Resolves the Agent

	modular, composable Agent-Tools; Phased verification (TEE then ZK).	Staking-based security.	autonomous on-chain actions.	(Devs/Auditors), Governance, Security.	Trilemma via a pragmatic, phased roadmap; focuses on high-value, verifiable on-chain tasks with a direct fee-for-service economy.
Fetch.ai (ASI)	L1 blockchain (Cosmos-SDK); general-purpose agent framework (uAgents)	PoS Blockchain; DAO Governance	Platform for building a decentralized digital economy with "Digital Twins".	Staking, Gas, Agent Registration, Governance	Fetch is a broad, horizontal platform. SnorlapAI is a vertical specialist for high-value, secure on-chain operations.
Bittensor (TAO)	Network of specialized subnets for commoditized intelligence	PoS-like consensus (Yuma); Senate/Triumvirate Governance	Decentralized marketplace for AI model outputs.	Staking, Governance, Payments for intelligence	Bittensor provides intelligence, SnorlapAI provides action. SnorlapAI is a higher-level orchestration layer.
Morpheus (MOR)	Peer-to-peer network of "Smart Agents"	DAO; funded by yield from staked capital (stETH)	Marketplace for AI agents and compute.	Staking (directs yield), Payments for compute	Morpheus's economy is indirect (yield-based). SnorlapAI's direct fee-for-service model provides a more robust, self-sustaining economic flywheel.
Olas (Autonolas)	Off-chain MAS framework; on-chain NFT registries & consensus	DAO Governance	Coordination of off-chain autonomous services.	Governance, Staking (for operators), Bonding	Olas focuses on agent coordination. SnorlapAI focuses on the cryptographic

	anchoring				verifiability of every action, critical for high-stakes finance.
ElizaOS	Open-source framework (TypeScript) for multi-platform AI agents	DAO Governance	Building general-purpose AI agents for DApps and social media.	Governance, Transactions, Staking	Eliza focuses on developer experience and social agents. SnorlapAI focuses on cryptographically verifiable, high-stakes on-chain actions.
Virtuals Protocol	AI agent launchpad and tokenization platform on Base/Solana	DAO Governance	Marketplace for co-owned, revenue-generating AI agents (gaming/entertainment).	Agent Creation, Liquidity Pairing, Governance	Virtuals tokenizes the agent as an asset. SnorlapAI provides the agent's execution as a service. Different markets (entertainment vs. work).
Monad	High-performance, parallel execution EVM L1 blockchain	PoS Blockchain	Scalable transaction settlement layer.	Gas, Staking, Governance.	Infrastructure Synergy: Monad is a potential settlement layer, not a competitor. SnorlapAI can leverage Monad's speed.
0G (0g.ai)	Modular L1 with a specialized, high-throughput Data Availability (DA) layer	PoS Blockchain	Scalable Data Availability for AI and Web3 applications.	Gas, Staking, Governance.	Critical Dependency: SnorlapAI leverages OG's DA layer to make its verification model economically viable at scale.

GaiaNet	Decentralized network of edge nodes running fine-tuned "AI twins"	Node operators contribute compute; reward program (Gaia XP)	Decentralized marketplace for specialized AI inference.	Credits for network interaction	Gaia provides knowledge/inference. SnorlapAI provides action. SnorlapAI can orchestrate Tools that query Gaia nodes.

3. The SnorlapAI Solution: A Modular, Composable, and Sovereign Network

SnorlapAI rejects the false dichotomy between centralized capability and decentralized security. The solution is not a better monolithic agent or a slightly faster blockchain, but a fundamentally new architecture for on-chain autonomy. This architecture is built on a foundation of three core principles, designed to foster a composable, sovereign, and intelligent ecosystem. The design deliberately mirrors the successful pattern of the internet: a "stupid" network (the core SnorlapAI protocol) with intelligent endpoints (the user's Conductor and the developer's Tools). The protocol itself does not need to be intelligent; it only needs to provide a robust, secure, and incentive-aligned environment for intelligence to emerge and be coordinated.

3.1. Core Principles: Modularity, Sovereignty, and Emergent Intelligence

- **Modularity over Monoliths:** The design philosophy of SnorlapAI is rooted in the rejection of monolithic agent architectures. Instead of a single, all-powerful AI attempting to perform every task, the SnorlapAI network is composed of a vast ecosystem of small, specialized, and single-purpose agents, which we term "Tools." Each Tool is an expert in its domain—be it executing a swap on Uniswap, parsing a governance proposal, or calculating the optimal collateral ratio on Aave. This modular approach provides several distinct advantages: enhanced security (isolating risk to individual Tools), greater efficiency (using the right tool for the job), and permissionless innovation (allowing any developer to contribute a new Tool to the network). This stands in direct contrast to the observed failures of generalist agents like ChatGPT Operator, which struggle with the complexity of the open web.
- **Sovereignty by Design:** In the SnorlapAI ecosystem, the user is the ultimate sovereign. This principle is a direct counter-response to the opaque, custodial nature of centralized agents like Manus. Users retain full, non-custodial control over their assets, private keys, and data at all times. Tools are granted specific, time-bound permissions to execute actions on the user's behalf, and these permissions are enforced by smart contracts. Every significant action requires the user's cryptographic approval, ensuring that the agent network is an extension of the user's will, not a replacement for it.
- **Emergence through Incentives:** Complex, intelligent behavior is not programmed top-down by a central authority. It emerges from the bottom-up interactions of

simple, specialized Tools operating within a shared, incentive-driven environment. This leverages the concept of swarm intelligence but grounds it firmly in economic reality. The network's tokenomics are engineered to reward the creation of high-value Tools, the accurate verification of their execution, and their composition into increasingly sophisticated workflows. The result is a system that learns and adapts, not through a centralized training process, but through the emergent dynamics of a decentralized market.

3.2. The Ideal User Experience: An Intent-Centric Interface

To achieve mass adoption, the protocol's underlying complexity must be completely invisible to the end-user. SnorlapAI is designed to provide an intent-centric user experience, powered by the Snorlap Conductor, that feels as seamless and intuitive as interacting with a human assistant or a state-of-the-art platform like Manus.

The user journey is designed for simplicity and power:

1. **User States Intent in Natural Language:** The user interacts with a simple, conversational interface (web, mobile, or desktop app) and states their high-level goal. They do not need to know what a "Tool," "TEE," or "DAO" is.
    - **Example:** *"Take my 50 ETH of staking rewards from Lido, swap just enough to pay off my 20,000 DAI debt on Aave, and then take the remaining ETH, bridge it to Arbitrum, and deposit it into the highest-yield, audited ETH/USDC liquidity pool."*
2. **Automated Orchestration by the Conductor:** The Snorlap Conductor acts as the user's intelligent orchestrator, handling all the backend complexity:
    - **Parses Intent:** An integrated LLM parses the user's goal, identifying the assets, protocols, amounts, and constraints.
    - **Decomposes the Task:** It breaks the high-level goal into a logical sequence of atomic tasks, creating a machine-readable workflow graph (DAG). For the example above, this DAG would include discrete steps like claim_lido_rewards(), calculate_swap_amount(), swap_eth_for_dai(), repay_aave_debt(), bridge_remaining_eth_to_arbitrum(), find_highest_yield_pool(), and deposit_liquidity().
    - **Discovers & Selects Tools:** For each task, the Conductor automatically queries the on-chain registry for DAO-approved Tools with the required capability (e.g., aave_v3_repay). It then runs an automated, fee-based auction to select the most efficient Tool for each step.
    - **Executes the Workflow:** The Conductor dispatches each task to the selected Tool, manages the dependencies between steps, handles all cross-chain messaging, and collects the verifiable proofs of execution (hardware attestations in Phase 1, zk-STARKs in Phase 2).
3. **Seamless User Approval:** The user is never exposed to the underlying mechanics. They are simply presented with a clear, human-readable summary of the proposed plan and its expected outcome.
    - Example Summary: "Plan: Pay off 20,000 DAI debt on Aave and deposit remaining ~X.XX ETH into Y-Pool on Arbitrum for an estimated Z% APY. Estimated cost: $W.XX. [Approve]"
      With a single signature, the user approves the entire complex workflow, which the Conductor then executes autonomously.

4. Protocol Architecture: A Phased Evolution from Pragmatism to Purity

SnorlapAI's architecture is defined by a strategic, two-phase evolution. This approach acknowledges the immense engineering complexity of a fully trustless ZK-based system and de-risks development by launching with a pragmatic, secure, and more executable model first. This allows the protocol to accelerate its time-to-market, deliver immediate value to users, and build a robust economic flywheel that can fund the transition to its ultimate, cryptographically pure endgame.

4.1. Phase 1: The Pragmatic Launch Architecture

The initial mainnet launch of SnorlapAI is designed for speed, security, and developer accessibility. It replaces the immediate need for a zkVM and a decentralized auditor network with a more mature and battle-tested set of technologies, allowing the protocol to solve the "cold start" problem by delivering a functional product quickly.

- **Execution in Trusted Execution Environments (TEEs):** Instead of a zkVM, Agent-Tools will initially run within hardware-based TEEs (e.g., Intel SGX, AMD SEV). A TEE is a secure, isolated "enclave" within a processor that protects code and data during execution, even from the host machine's privileged operating system. This approach offers several key benefits for launch:
    - **Developer Accessibility:** It drastically lowers the barrier to entry. Developers can write Tools in standard languages like Rust or C++ and deploy them within a TEE without needing to master the arcane art of zero-knowledge circuits.
    - **Performance and Confidentiality:** TEEs are highly performant for general-purpose computation and are inherently designed for confidentiality, enabling the "Trustless IP Monetization" goal from day one by protecting proprietary Tool logic from the node operator.
- **Verification via Hardware Attestation:** The core of the Phase 1 verification model is hardware attestation. Instead of a zk-STARK proof, a TEE produces a small, verifiable cryptographic proof, digitally signed by the hardware manufacturer (e.g., Intel). This attestation certifies that a specific, known piece of code was executed correctly on given inputs within a genuine TEE. This mechanism eliminates the need for a dedicated, decentralized Auditor network at launch, as attestations can be quickly and cheaply verified by any party, including the user's Conductor or an on-chain smart contract. This drastically simplifies the initial economic and security model to a two-sided marketplace of Users and Developers.
- **Security & Task Allocation via a DAO-Curated Registry:** To ensure a high standard of quality and security from day one, the initial task allocation mechanism will not be fully permissionless. Instead, SnorlapAI will use a DAO-Curated Registry.
    - **Mechanism:** Developers must submit their new Tool to the SnorlapAI DAO for approval. This submission must be accompanied by a mandatory, independent security audit. The DAO, composed of $SNOR token holders, then votes on whether to add the Tool to an on-chain "whitelist" or registry of trusted providers.
    - **Benefits:** This model provides robust Sybil resistance, as passing a community governance vote is far more difficult and costly for a malicious actor than simply acquiring stake. It also ensures a high-quality ecosystem of reliable Tools, which is

critical for attracting users to a network that handles high-value financial tasks. Auctions for tasks are then conducted only among this pre-vetted set of Tools, simplifying the initial auction mechanism.

4.1.1. Phase 1 Security Model: TEEs and Their Limitations

The choice to launch with a TEE-based architecture is a pragmatic engineering decision that must be presented with intellectual honesty. This model replaces pure cryptographic trust with a **trusted hardware assumption**, which introduces a specific and well-documented set of risks. TEEs, particularly Intel SGX and AMD SEV, have a known history of vulnerabilities to sophisticated side-channel and fault-injection attacks.

The protocol will employ a defense-in-depth strategy to mitigate these risks, focusing on best practices for confidential computing:

- **Confidentiality over Integrity:** The primary security guarantee of TEEs in this context is **confidentiality** (preventing data from being read) rather than perfect **integrity** (preventing data from being altered). An attacker who compromises a TEE might be able to extract sensitive data but would not be able to falsify an on-chain transaction, which is ultimately secured by the settlement layer's consensus.
- **Key Rotation and Forward Secrecy:** The protocol will enforce frequent rotation of any cryptographic keys used within TEEs. This limits the "blast radius" of a potential compromise, ensuring that even if a key is extracted, it cannot be used to decrypt past or future communications.
- **Minimalist Tool Design:** Developers will be encouraged to design Tools that minimize the amount of sensitive state held within the TEE at any given time, reducing the potential impact of a data leak.

Despite these mitigations, the reliance on TEEs places a clear constraint on the "Trustless IP Monetization" promise during Phase 1. For Tools whose core value is a public function (e.g., executing a standard DeFi swap), the risk is manageable. However, for Tools whose value is derived from a highly sensitive, proprietary algorithm (e.g., a unique MEV strategy), a confidentiality breach is an existential failure. Developers of such Tools must understand and accept the residual risk of hardware vulnerabilities during Phase 1. The protocol will provide clear documentation and risk scoring to help developers make informed decisions.

4.2. Phase 2: The Trustless Endgame Architecture

This is the ultimate vision for SnorlapAI, achieving maximum decentralization, censorship resistance, and security. This phase re-introduces and builds upon the most advanced concepts, positioning them as the protocol's mature state. The transition to this phase is critical for the long-term health and scalability of the network.

- **The Rationale for Transitioning to zk-STARKs:** While TEEs are pragmatic for launch, they rely on a trusted hardware assumption centered on a few vendors. The zk-STARK model is superior long-term because it:
  - **Eliminates Centralized Trust:** Security is grounded purely in mathematics, not hardware manufacturers.
  - **Achieves Trust Scalability:** As the value secured by the network grows, a purely cryptographic system can scale its security guarantees without introducing new centralized points of failure.

    - **Offers Post-Quantum Security:** zk-STARKs are built on collision-resistant hash functions, making them plausibly resistant to attacks from future quantum computers, ensuring long-term protocol viability.
- **Core Components of the Endgame Architecture:**
    - **zkVM for General-Purpose Computation:** Agent-Tools will run in a general-purpose Zero-Knowledge Virtual Machine (zkVM), likely based on the RISC-V instruction set. This will automatically generate zk-STARK proofs of execution, re-enabling fully trustless verification.
    - **Decentralized Auditors & Restaking:** The three-sided marketplace is restored. A decentralized network of Auditors will be responsible for validating the zk-STARK proofs. To solve the "cold start" problem for the Auditor network's economic security, the protocol will integrate a restaking model, functioning as an Actively Validated Service (AVS) on a platform like EigenLayer. This allows Auditors to stake both $SNOR and restaked assets (like LSTs), bootstrapping security and enhancing capital efficiency.
    - **Reputation-Weighted VRF Auction:** With a mature and permissionless network, the more complex and dynamic auction mechanism can be introduced. This system will use on-chain reputation and a Verifiable Random Function (VRF) to select the optimal Tool for each job in a highly efficient and manipulation-resistant manner.

4.3. The Universal Five-Layer Stack

Across both architectural phases, the protocol can be conceptualized as a five-layer stack, where each layer performs a distinct function. This modularity ensures future-proofing and adaptability.

- **Layer 5 (Orchestration): The Snorlap Conductor**
- **Layer 4 (Execution): The Decentralized Tool Network** (Running in TEEs in Phase 1, zkVMs in Phase 2)
- **Layer 3 (Verification & Governance): The SnorlapAI Protocol** (Verifying attestations in Phase 1, zk-STARKs in Phase 2)
- **Layer 2 (Data Availability): Specialized DA Layer**
- **Layer 1 (Settlement): High-Performance Blockchain**

4.4. Cross-Chain Orchestration: A Hybrid IBC and LayerZero Strategy

To operate effectively in a multi-chain world, SnorlapAI will employ a hybrid strategy, managed by the Snorlap Conductor, to orchestrate seamless, multi-chain workflows. This strategy remains consistent across both architectural phases.

- **For the Cosmos Ecosystem & Sovereign Rollups:** SnorlapAI will leverage the Inter-Blockchain Communication (IBC) protocol for trust-minimized communication.
- **For External & Heterogeneous Ecosystems:** To connect with disparate environments like Ethereum mainnet, its L2s, and Solana, SnorlapAI will integrate LayerZero for its flexibility.

4.4.1. Cross-Chain Security Considerations

The modular, cross-chain nature of SnorlapAI is a core strength, but it also introduces complex security considerations. The security of any given workflow is only as strong as the weakest link in the chain of L1s, DA layers, and interoperability protocols it touches.

SnorlapAI addresses this challenge through a principle of transparent, user-configurable security.

The Snorlap Conductor is designed to surface these inherited trust assumptions to the user before they approve any cross-chain workflow. For example, when using LayerZero for interoperability, the OApp Security Stack allows applications to configure their own Decentralized Verifier Networks (DVNs). The Conductor will present the user with a clear summary of the security model for a proposed transaction (e.g., "This action will be verified by a 2-of-3 DVN set including Google Cloud and LayerZero Labs"). This ensures informed consent and allows users or applications to set their own security thresholds, refusing to execute workflows that do not meet their required level of trust.

5. The $SNOR Token: Engineering a Self-Sustaining Economic Flywheel

The $SNOR token is the cryptographic lifeblood that powers SnorlapAI's economic engine across both architectural phases. The tokenomics are designed to create a self-sustaining marketplace that aligns the incentives of all participants and drives a virtuous cycle of growth and value accrual. The model is built on a direct, fee-for-service economy where the token's utility is indispensable and its value is intrinsically linked to the network's economic throughput.

5.1. Token Overview and Distribution

The $SNOR token is an ERC-20 standard token with a fixed, non-inflationary total supply of 1,000,000,000 tokens. A portion of all network fees is programmatically burned, creating a deflationary pressure that increases with network activity.

Table *5.1*: $SNOR Token Allocation & Utility Mapping

Category	Allocation (%)	TGE Unlock (%)	Vesting Schedule	Primary Beneficiary / Purpose	Core Utility Mapping
Ecosystem & Community Fund	30%	10%	36-month linear vest	Grants for developers, liquidity mining, user incentives, bug bounties.	Payment: Users access grants. Staking: Devs/Auditors receive rewards. Governance: Community directs fund allocation.
Core Contributors & Team	20%	0%	6-month cliff, then 24-month linear vest	Reward and retain the core development team responsible for building the protocol.	Governance: Aligns team with long-term protocol health.

Treasury & Reserve	15%	0%	48-month linear unlock, controlled by DAO	Long-term operational funding, protocol-owned liquidity, and strategic initiatives controlled by governance.	Governance: The DAO's primary tool for strategic capital allocation.
Strategic Partners & Advisors	10%	0%	6-month cliff, then 24-month linear vest	Align strategic partners (e.g., DeFi protocols, infrastructure providers) with the network's success.	Governance: Provides partners a voice in protocol direction.
Private Sale	10%	0%	3-month cliff, then 18-month linear vest	Secure early-stage funding from strategic investors to accelerate development and go-to-market.	Governance: Aligns early investors with long-term protocol success.
Public Sale & Liquidity	10%	100%	N/A	Establish deep initial liquidity on DEXs and provide broad public access to the token.	Payment: Initial source of tokens for users to pay fees.
Airdrop	5%	25%	12-month linear vest	Reward early adopters, bootstrap the user base, and decentralize initial token distribution.	Payment: Provides initial tokens for early users. Governance: Encourages community participation

5.2. The Phased Marketplace: A Game-Theoretic Analysis

The SnorlapAI economy is a flywheel connecting participants, with the $SNOR token as the medium of exchange and collateral. The marketplace structure evolves with the architectural phases.

- **Phase 1 (TEE Architecture): A Two-Sided Marketplace**
  - **Participants:** The initial economy is simplified, connecting **Users** (demand) and **Developers** (supply).
  - **Mechanism:** Users pay fees in $SNOR. Developers stake $SNOR to have their Tools whitelisted by the DAO and earn the majority of fees from their Tool's execution. The need for a decentralized Auditor network is eliminated by hardware attestations, making the initial economy easier to bootstrap.
- **Phase 2 (zk-STARK Architecture): A Three-Sided Marketplace**
  - **Participants:** The marketplace matures to include the third critical participant: the **Auditor** (security).
  - **Mechanism:** The model described in v5.0 is fully realized. Users pay fees, Developers stake and earn fees, and Auditors stake collateral ($SNOR and/or restaked assets) to validate zk-STARK proofs, earning a share of network fees for securing the protocol.

Table 5.2: Participant Incentive and Slashing Model

Participant	Action	Incentive (Reward)	Disincentive (Slashing/Penalty)	Economic Rationale
User	Pays for a workflow	Successful, verifiable execution of complex task.	Payment of fees in $SNOR.	Users pay for value-added services that save time, reduce risk, or generate alpha.
Developer (Honest)	Builds & stakes on a high-quality Tool (DAO-approved in Phase 1)	Earns majority of fees (in $SNOR) from Tool executions. Builds on-chain reputation.	Stake is at risk if Tool is faulty or rejected by DAO. Opportunity cost of capital.	Incentivizes creation of useful, secure, and reliable Tools to maximize fee revenue.
Developer (Malicious)	Deploys a malicious or faulty Tool	Potential for short-term exploit (theft).	Forfeiture of entire $SNOR stake. Permanent reputation damage. Inability to pass DAO curation.	Makes malicious behavior economically irrational by ensuring the cost of the stake and reputational gatekeeping exceeds potential illicit gains.
Auditor (Phase 2)	Stakes collateral & correctly validates proofs	Earns a share of all network transaction fees (in $SNOR).	Severe slashing of staked collateral. Permanent ban from auditor set.	Incentivizes diligent and accurate verification to earn a steady yield from network activity.

5.3. Value Accrual and Deflationary Mechanics

- **Staking Tiers & Protocol-Level Benefits:** The staking mechanism provides real yield from network fees and confers additional protocol-level benefits. To make this utility concrete, staking tiers will be clearly defined. For example: staking 10,000 $SNOR may grant a *5%* network fee discount for users, while developers staking over 100,000 $SNOR could receive a 1.2x multiplier in the Phase 2 reputation-weighted auction. These quantified benefits create a clear and compelling incentive to acquire and lock $SNOR tokens.
- **Governance:** The SnorlapAI protocol will transition to a fully decentralized autonomous organization (DAO) controlled by $SNOR token holders. The DAO's role is critical from day one in Phase 1, as it is responsible for curating the registry of approved Tools, providing a powerful layer of security and quality control. The DAO will eventually control the protocol treasury, grant programs, and all future upgrades, using a quadratic voting system to ensure fair representation.
- **The Fee-Burn Flywheel:** A fixed percentage of every transaction fee paid on the network will be programmatically burned. This mechanism directly links network usage to token scarcity, creating a powerful deflationary force that benefits all long-term token holders and completes the economic flywheel.

5.3.1. DAO Governance Guardrails for Phase 1

The DAO-Curated Registry in Phase 1 is a necessary security measure, but it also creates a potential centralization risk. To mitigate the risk of capture or censorship, the DAO's governance framework will be built on a foundation of transparency and clear, objective standards.

- **Standardized Proposal Requirements:** To be considered for the registry, a developer must submit a formal proposal that includes: a full, independent security audit from a reputable firm; comprehensive technical documentation; and a clear description of the Tool's capabilities and intended use.
- **Transparent Evaluation Criteria:** The DAO will maintain a public set of objective criteria for evaluating Tools, focusing on security, reliability, and utility. This ensures that decisions are not arbitrary and provides a clear benchmark for developers.
- **Structured Voting and Appeal Process:** Proposals will be subject to a fixed voting period with a predefined quorum threshold. If a Tool is rejected, the developer will receive a clear rationale based on the evaluation criteria and will have the right to resubmit the proposal after addressing the identified issues. This creates a fair and accountable process that encourages high-quality submissions while preventing malicious gatekeeping.

6. High-Value Use Cases and Go-to-Market Strategy

The true measure of an agentic network is the tangible value it creates. SnorlapAI is not a tool for generic productivity; it is a precision instrument for executing high-stakes, complex operations in the Web3 ecosystem. The initial use cases are focused on the areas of greatest complexity and highest value: decentralized finance and DAO operations.

6.1. DeFi Agent Suite

- Autonomous Yield & Liquidity Optimizer
- Intelligent Liquidation & Collateral Management Agent
- MEV-Aware Transaction Execution Agent

6.2. DAO Operations Suite

- Autonomous Treasury Management Agent
- Programmatic Governance Delegate Agent

6.3. A Phased Go-to-Market Strategy

A platform's launch strategy must solve the "cold start" problem inherent in multi-sided marketplaces. The most critical, non-negotiable element of SnorlapAI's launch is the aggressive execution of its "Genesis Tools" strategy. This involves partnering with blue-chip protocols to build initial, high-impact agent tools, which is the most effective way to solve the marketplace's "cold start" problem and jumpstart the economic flywheel from day one. This strategy is even more crucial for the phased launch, as it builds the community and trust necessary for the DAO to effectively curate the initial Tool registry.

- **Phase 1:** Deep Integration with Blue-Chip Protocols (Months 1-6)
- **Phase 2:** Bootstrapping the Developer Ecosystem (Months 7-18)
- **Phase 3:** Cultivating a Global Community (Months 18+)

7. Roadmap & Future Vision

The SnorlapAI roadmap is an engineering-driven plan that reflects the new phased architectural evolution, focusing on delivering tangible, high-value functionality at each stage.

7.1. Development Phases

- **Phase 1: Pragmatic Foundation (Q3 2025 - Q2 2026)**
  - **Objectives:** Launch a robust and secure mainnet based on the TEE architecture. Release the ADK for TEE-based Tool development. Establish the DAO-curated registry and onboard the first set of audited "Genesis Tools."
  - **Key Deliverables:** Public testnet with TEE support; Mainnet launch on a single, high-performance EVM chain (e.g., Monad, Arbitrum); Deployment of Genesis Tools in partnership with blue-chip protocols; Functioning DAO governance for Tool curation.
- **Phase 2: Trustless Transition (Q3 2026 - 2027)**
  - **Objectives:** Begin the transition to the zk-STARK architecture. Launch the zkVM and the decentralized Auditor network, incorporating restaking for economic security. Gradually open the Tool marketplace to be more permissionless.
  - **Key Deliverables:** Testnet for the zk-STARK architecture; ADK v2.0 with zkVM support; Mainnet integration of the Auditor network and restaking; Launch of the reputation-weighted VRF auction mechanism.
- **Phase 3: Ecosystem Maturation & Multi-Chain Expansion (2028+)**
  - **Objectives:** Complete the transition to the full zk-STARK architecture. Achieve full decentralization of protocol governance. Expand protocol deployment to multiple

L1/L2 networks to become a universal, chain-agnostic orchestration layer.
    - **Key Deliverables:** Full migration of network activity to the zk-STARK model; Successful handover of all protocol control to the DAO; Live integrations with multiple blockchain environments; A thriving, permissionless marketplace with thousands of third-party Tools.

7.1.1. The Path to Trustlessness: A Migration Framework

The transition from the TEE-based Phase 1 to the ZK-based Phase 2 is a critical and complex upgrade that will be managed through a clear, structured framework. This is not an abrupt switch but a gradual migration designed to ensure security, stability, and seamless user experience.

- **Economic Triggers:** The transition will be initiated when the network reaches key economic milestones, to be ratified by DAO governance. These triggers will signal that the ecosystem is mature enough to support the three-sided marketplace. Illustrative triggers include: achieving a sustained daily network fee volume of over $10,000 and having at least 50 unique, DAO-approved Tools in the registry.
- **Technical Migration Strategy:** The migration will follow a parallel adoption model. For a defined period (e.g., 6-12 months), the network will support both TEE-based and ZK-based Tools. The Snorlap Conductor will be upgraded to handle both verification types, allowing users to choose their desired level of security. This dual-support period allows developers to migrate their Tools to the new zkVM standard at their own pace without disrupting existing services.
- **Incentivizing the Upgrade:** The DAO will use the Ecosystem Fund to incentivize developers to migrate. This will include direct grants for porting popular Tools to the zkVM, as well as a temporary "reward multiplier" where ZK-based Tools earn a higher percentage of network fees for a limited time after the Phase 2 launch. This creates a strong economic pull for developers to embrace the more secure architecture.

7.2. Future Vision: The Protocol for Autonomous Work

The long-term vision for SnorlapAI remains unchanged: to become the fundamental protocol for autonomous on-chain work—the equivalent of TCP/IP for intelligent, decentralized operations. In this future, SnorlapAI will be the invisible, trustless backbone that enables any dApp, DAO, or individual user to securely and verifiably delegate any complex on-chain task to a global, permissionless network of specialized agents.

This vision extends beyond DeFi and governance to decentralized commerce, on-chain gaming, and Decentralized Science (DeSci). SnorlapAI will be the essential infrastructure that allows the decentralized economy to move beyond simple transactions and into the realm of complex, autonomous, and intelligent processes.

8. Core Contributors & Path to Decentralization

A protocol's long-term success is a function of both its technical architecture and the quality and integrity of its human capital. SnorlapAI is being developed by a team of seasoned experts with deep, demonstrated experience in the precise domains required to build this ambitious system: distributed systems, zero-knowledge cryptography, and

economic mechanism design. The team is committed to a path of progressive decentralization, ensuring that control of the protocol ultimately rests with its community of users, developers, and stakeholders.

8.1. Core Contributors

The initial development of the SnorlapAI protocol is led by a core team with exemplary credentials across designing and launching consumer applications, blockchain systems, Web3 infrastructure, game theory and tokenomics modelling, and generative AI & machine learning. This ensures that the project is built on a foundation of proven expertise.

In addition, the team is supported by a board of advisors composed of founders from top-tier DeFi protocols and partners from leading Web3 venture capital firms, providing both strategic guidance and deep ecosystem connectivity.

8.2. A Deliberate Path to DAO Governance

SnorlapAI is founded on the principle of community ownership. The ultimate authority over the protocol will be the SnorlapAI DAO, governed by $SNOR token holders. The transition to this state will be phased and deliberate, ensuring network stability and security at each step.

1. **Initial Phase (Launch - 12 months):** During the first year post-mainnet, protocol upgrades will be managed by a multi-signature council composed of the core team and trusted industry advisors. The DAO's primary role will be to govern the Tool registry, as detailed in Section 5.3.1.
2. **Transitional Phase (Months 12-24):** Governance will shift to a hybrid model. The multi-signature council will retain veto power over critical security matters, but all other protocol parameters and treasury allocations will be decided by on-chain $SNOR holder votes.
3. **Full Decentralization (Month 24+):** The multi-signature council will be dissolved, and all authority over the protocol will be transferred to the SnorlapAI DAO.

This phased approach ensures that SnorlapAI begins with strong, expert stewardship and matures into a truly decentralized, community-run public good, fulfilling the promise of a sovereign and autonomous future.

Appendix

A. Formal Definitions and Mathematical Preliminaries

This appendix provides more rigorous definitions for key concepts discussed in the main body of the paper.

A.1. Reputation Scoring Function (Phase 2)

The reputation score RT for a given Tool T is calculated at block height h as a weighted sum of normalized components. This mechanism is planned for the permissionless environment of Phase 2.

$$R_T(h) = w_E \cdot N(E_T(h)) + w_S \cdot N(S_T(h)) + w_U \cdot N(U_T(h)) + w_A \cdot N(A_T(h))$$

Where:

- $E_T(h)$ is the count of successful, verified executions by Tool T over a trailing window of blocks.
- $S_T(h)$ is the amount of SNOR tokens staked by the developer of Tool T.
- $U_T(h)$ is the aggregated user rating for Tool T, weighted by the value of the transactions in which the user participated.
- $A_T(h)$ is a score representing the security audit history of the Tool.
- w_i are the governance-determined weights for each component.
- $N(\cdot)$ is a normalization function that maps each component to a standard range.

This model is inspired by multi-dimensional reputation systems designed to be resistant to manipulation like Sybil attacks or reputation laundering.

A.2. Reputation-Weighted VRF Auction Mechanism (Phase 2)

For a given task, let $\mathcal{T}$ be the set of eligible Tools that have submitted a bid. Each Tool $T \in \mathcal{T}$ has a reputation score R_T and a bid price P_T. The selection probability π_T for each Tool is determined by:

$$\pi_T = \frac{f(R_T, P_T)}{\sum_{T' \in \mathcal{T}} f(R_{T'}, P_{T'})}$$

Where f(R,P) is a function that combines reputation and price. A common choice is an inverse price weighting combined with a reputation weight:

$$f(R, P) = R^{\alpha} P^{-\beta}$$

Here, α and β are governance-set parameters that tune the sensitivity to reputation versus price. A Verifiable Random Function (VRF) is then used with a public seed (e.g., the block hash) to draw a winner according to the probability distribution $\{\pi_T\}_{T \in \mathcal{T}}$. This ensures the selection is both random (preventing censorship by the Conductor) and weighted towards reputable, cost-effective Tools.

B. Overview of Verification Technologies

B.1. Trusted Execution Environments (TEEs) and Hardware Attestation (Phase 1)

A TEE (e.g., Intel SGX) is a secure area inside a main processor, isolated from the host operating system. It guarantees the confidentiality and integrity of the code and data executing within it. Verification is achieved via a hardware attestation, a cryptographic signature from the CPU manufacturer that proves a specific, unmodified program was executed on a genuine TEE. This provides strong, albeit hardware-dependent, assurances of correct execution.

B.2. zk-STARKs (Phase 2)

A zk-STARK is a proof system that allows a Prover to convince a Verifier of the integrity of a computation. The process involves two main steps :

1. **Arithmetization:** The computational statement (e.g., "The execution of this program in the zkVM with these inputs produces this output") is transformed into an algebraic representation. This involves creating an Execution Trace, a table where each row represents a step in the computation and columns represent registers or memory. This trace is then used to define a set of polynomial constraints. The original statement is true if and only if there exist low-degree polynomials that satisfy these constraints over a finite field.
2. **Low-Degree Testing:** The Prover commits to these polynomials. The Verifier then uses an interactive oracle proof (IOP) protocol, made non-interactive using a Fiat-Shamir transform with a collision-resistant hash function, to query the Prover's committed polynomials at random points. By checking that the revealed points lie on a low-degree polynomial, the Verifier can be convinced, with high probability, that the original computation was correct.

This process is transparent (no trusted setup), scalable (verifier work is polylogarithmic in the computation size), and zero-knowledge (if secret inputs are properly masked).

C. Glossary of Terms

- **Actively Validated Service (AVS):** A service, such as an oracle or a verification network, that leverages the pooled security of a protocol like EigenLayer. SnorlapAI's Phase 2 Auditor network is designed as an AVS.
- **Agent Development Kit (ADK):** A suite of tools, libraries, and compilers provided by SnorlapAI for developers to build, test, and deploy Agent-Tools on the network.
- **Agent-Tool:** A specialized, single-purpose autonomous agent on the SnorlapAI network, designed to perform a specific on-chain or off-chain task.
- **DAO-Curated Registry:** An on-chain whitelist of approved Agent-Tools, vetted for security and quality by the SnorlapAI DAO. This is the primary security mechanism in Phase 1.
- **Hardware Attestation:** A cryptographic proof from a hardware manufacturer certifying that specific code was executed correctly within a TEE. This is the verification method for Phase 1.
- **Restaking:** A mechanism that allows staked assets (like ETH) to be used to secure additional protocols or services (AVSs), extending crypto-economic security beyond the primary network.
- **Snorlap Conductor:** A client-side application that acts as the user's personal orchestrator, interpreting intent, generating workflows, and managing interactions with the SnorlapAI network.
- **Sovereign AI:** An AI agent that is a direct extension of a user's will, operating under their non-custodial control, with actions that are verifiably secure and aligned with their stated intent.
- **Trusted Execution Environment (TEE):** A secure, hardware-isolated enclave within a processor that protects code and data during execution. The execution environment for Phase 1.

- **Web3 Agent Trilemma:** A framework positing the inherent trade-offs in current agent architectures between Capability, Composability, and Incentive Alignment.
- **Zero-Knowledge Scalable Transparent Argument of Knowledge (zk-STARK):** A type of cryptographic proof system that allows a prover to prove the integrity of a computation without a trusted setup, offering transparency and post-quantum security. The verification method for Phase 2.
- **Zero-Knowledge Virtual Machine (zkVM):** A virtual machine that executes a standard instruction set and generates a zero-knowledge proof of the entire execution trace, enabling verifiable general-purpose computation. The execution environment for Phase 2.