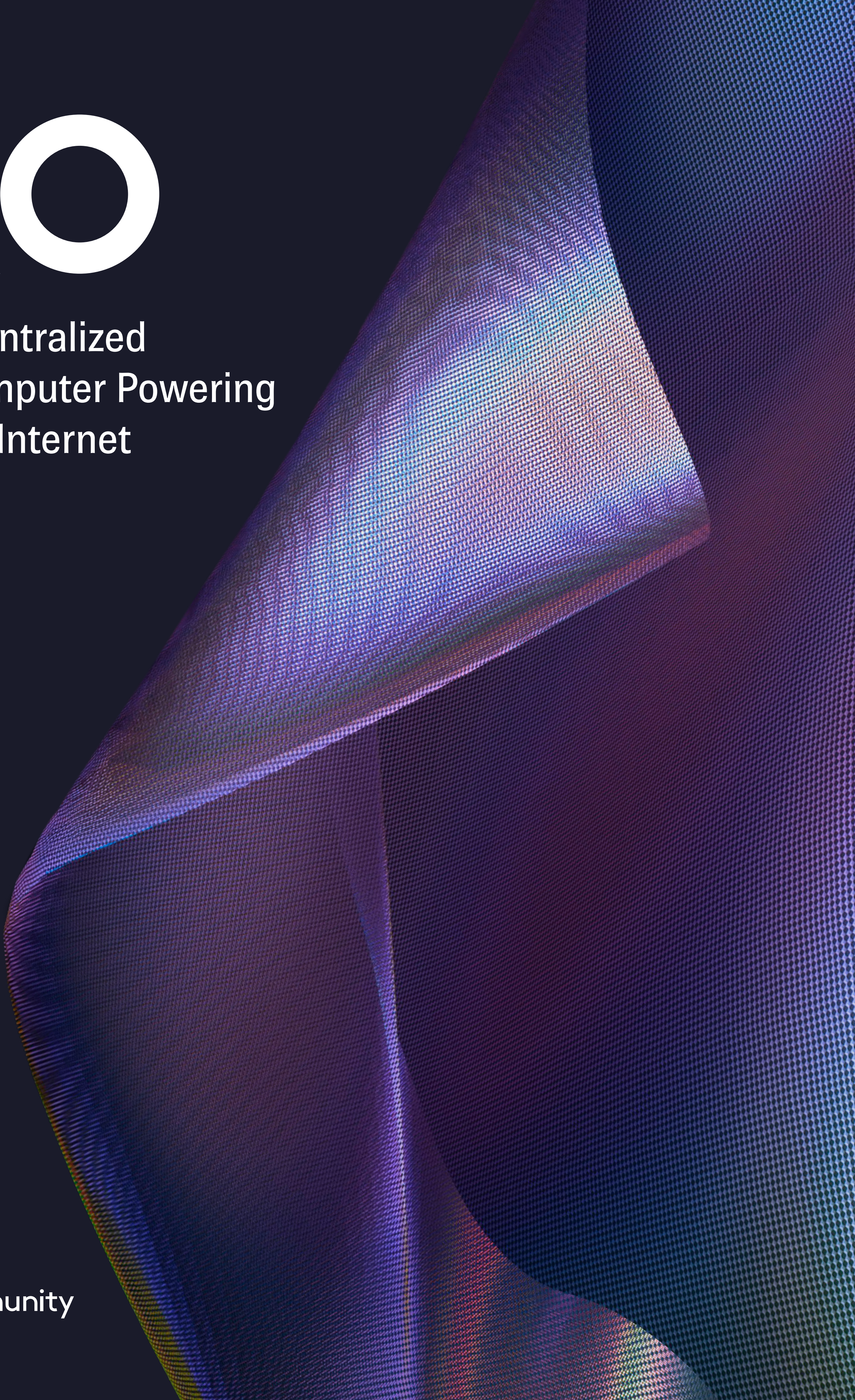
AO

The Decentralized Supercomputer Powering the Next Internet



Community Labs

Abstract

This report examines AO, a next-generation decentralized computing protocol that eliminates the scalability, execution, and resource constraints of traditional blockchain platforms. AO introduces an actor-based model, where independent processes operate asynchronously and communicate via message passing. By leveraging Arweave as a settlement layer, AO removes consensus bottlenecks, enabling parallel computation, unbounded scalability, and modular security configurations. Unlike conventional smart contracts, AO allows developers to deploy custom virtual machines (VMs), define execution environments, and implement market-driven economic security models, ensuring greater flexibility and efficiency.

We compare AO to the Internet Computer Protocol (ICP) and Solana, highlighting its advantages in scalability, computation, and developer accessibility. AO's Permaweb Index, an onchain asset index, dynamically tracks user activity and adjusts its composition, enabling developers to build and monetize applications in a way that existing platforms do not support. We explore real-world applications—including AI agents, decentralized finance (DeFi), large-scale data storage, and gaming—while addressing AO's skepticisms, such as price constraints and how AO addresses privacy and the blockchain trilemma.

By removing traditional constraints and introducing a more scalable, efficient, and flexible execution model, AO has the potential to transform decentralized computing and serve as the foundation for the next internet.

Acknowledgments & Contributors

This report is the result of a collaborative effort by a dedicated team who brought their expertise to every stage of its development. We extend our gratitude to everyone involved in this work.

Bithiah Koshy
Research Analyst

Mohsin Chheena
Research Analyst

Jeff Marsilio
Chief Business Officer

Pace Studio
Design & Publication

Jacob Kendall
Research Analyst

Jonathan Lelli
Head of Enterprise Business Development

Matias Parij
Head of Engineering

Tate Berenbaum
Chief Executive Officer

Contents

1. Introduction to AO

AO represents a novel approach to decentralized computing designed to overcome the limitations of current blockchain-based systems. Unlike traditional platforms that restrict computation through shared state, memory architecture, and fixed resource limits, AO creates a unified computing environment where users can run an unlimited number of parallel processes with unrestricted resource utilization. Each process can operate independently without being confined by typical blockchain constraints such as small computational capacity, mandatory user-triggered activation, or protocol-enforced size limitations.

At its core, AO implements an actor-oriented model where each process operates independently and communicates through message passing. With Arweave serving as the settlement layer for all messages and ensuring their long-term data storage, processes can execute in parallel without requiring consensus on computation itself. This architecture enables previously impractical applications in decentralized environments, from computationally intensive tasks to autonomous agents that can operate continuously without user triggers.

This combination of scalability, flexibility, and decentralized execution makes AO a significant breakthrough in blockchain-based computing. Its key innovations include:

- **Asynchronous, Parallel Execution** – Eliminates consensus bottlenecks by enabling independent processes to execute in parallel, ensuring unbounded scalability and efficient resource utilization.
- **Modular, Customizable Security** – Allows developers to define their own cryptographic validation mechanisms and economic incentives, removing reliance on rigid, protocol-enforced security constraints.
- **Permissionless Ecosystem Funding (PEF)** – A decentralized funding model where developers are rewarded based on application usage, reducing or eliminating the need for grants, venture funding, or centralized token allocations.
- **Unbounded Computational Capacity** – Supports complex, high-performance applications that can run indefinitely with dynamic resource allocation, overcoming the strict execution limits of smart contracts.
- **Trust-Minimized State Management** – Derives state from immutable message logs on Arweave, ensuring verifiable and transparent computation without reliance on centralized validators.

- **Multi-VM Support** – Enables developers to deploy any VM, removing programming language restrictions and fostering greater flexibility in application development.
- **Autonomous Agent Activation** – Processes can self-trigger and schedule execution autonomously, enabling continuous operation of decentralized AI agents and automated workflows.
- **Native Arweave Integration** – Provides decentralized data storage and access, allowing applications to store, retrieve, and manage data with permanence and verifiability.
- **Decentralized Serverless Computing** – Enables event-driven execution without always-on infrastructure, reducing costs while maintaining full availability for applications.

With these innovations, AO establishes itself as a highly scalable, flexible, and economically efficient computing environment, redefining what is possible in decentralized computing. And this is just the beginning—what comes next will reshape the way we think about decentralized applications (dApps) and computation.

2. AO vs Traditional Smart Contract Platforms

> *"Any sufficiently advanced technology is indistinguishable from magic."*

Arthur C. Clarke's timeless observation resonates deeply in the ongoing race to build the ultimate world computer. This pursuit, filled with innovation and ambition, aims to deliver the "magic" of advanced computational efficiency, seamless decentralization, and transformative potential to the world. Projects like Solana and Internet Computer Protocol (ICP) have emerged as key players in this space, each pioneering unique solutions: ICP with its unified systems and governance-focused model, and Solana with its fast execution powered by its Proof of History (PoH)[1] mechanism. AO, however, redefines the competition with its modular, developer-friendly architecture, combining unparalleled scalability, trustless computation, and creativity-driven design. By breaking traditional boundaries, AO seeks to deliver an experience that feels even more "magical" in its potential. This section

[1] PoH: a consensus mechanism that uses the concept of time to achieve distributed consensus.

will delve into the technical distinctions among AO, Solana, and ICP, exploring how each platform addresses the evolving demands of dApps and the overarching vision of a truly trustless, global computing network.

2.1 AO vs Internet Computer Protocol (ICP)

ICP was released by DFINITY in 2021.
It quickly gained popularity due to being the first blockchain providing a stateful, decentralized, and serverless computing platform to enter the market. Despite slowed growth since then, ICP maintains a passionate user base that is actively improving the service. AO and ICP are both protocols designed to enable horizontally scalable decentralized computation, enabling developers to host applications with internet-scale user bases without relying on centralized servers; however, they employ fundamentally different architectures. AO implements a message-based system where computational state is derived from logs stored on Arweave, with Compute Units (CUs)[2] processing these logs on demand. The system allows arbitrary data sizes and parallel execution through Messenger Units (MUs)[3] that coordinate both inter-process and intra-process communication.

ICP utilizes subnets to essentially create mini-internets, and these run special programs called "canisters." Canisters are one of the biggest selling points of ICP. These programs can be very complex and can be created to integrate seamlessly with already existing traditional IT software practices and infrastructure. As the number of users participating in the network increases, so does the number of subnets. It works in rounds, with the network processing updates and changes in each round (also called an epoch-based model). Merkle Trees,[4] alongside multiple computers working together, are used to ensure the correctness of the data (threshold BLS signatures). The protocol handles consensus through subnet-specific chains and uses catch-up packages (CUPs) for state synchronization between nodes.

A key technical distinction between the two lies in state management: AO derives state implicitly from message history, allowing flexible computation timing but requiring full message log access, while ICP maintains explicit state with immediate updates and certification. This means that unlike AO's ability to allow computation of the current state when required, ICP requires that each node explicitly update state after every change. AO's approach allows any computation to be independently verified by any party by

[2] CUs: nodes that execute computations for processes on AO.
[3] MUs: operate as relay nodes in the AO network, facilitating message transmission.
[4] Merkle Tree: A data structure that organizes data using hashes to make it more secure and efficient to process

replaying the message log, without requiring trust in the nodes that originally performed the computation. This theoretically makes the system more resilient to potential malicious actions (otherwise known as byzantine behavior). AO offers protection against these kinds of byzantine behaviors since incorrect computations can always be detected and corrected.

For governance, ICP utilizes what it calls a "Network Nervous System" (NNS). ICP's NNS consists of many different parts, but one of them is an automated decentralized autonomous organization (DAO)[5]. This maintains a voting element but removes the human aspect from the implementation of DAO proposals, as approved proposals are automatically implemented. NNS incentivises people to stake more tokens for a longer period of time by giving them more voting power. This makes centralization of voting power a major risk. AO, however, is retroactively incentivising long term holders of the Arweave token ($AR) with the AO token ($AO) through a fair-launch program. Additionally, developers migrating applications from other blockchains to AO are also rewarded through the same incentive program. There is no central treasury regulating the distribution of $AO after launch, relying exclusively on market dynamics instead. As a result, users can have confidence in the value of $AO.

Using AO's hierarchical security structure, processes themselves can decide to trust signers of messages, taking care of cryptographic validation concerns. AO-Sec Origin handles economic security, acting as a layer mechanism in AO that allows a modular, customizable implementation. One of its features, back-stop liveliness, ensures overall functionality of the network even when part of the network fails. Again, this customizable security differentiates AO from ICP's rigid security mechanisms.

In conclusion, both ICP and AO offer innovative ways to decentralize computation, with each having distinct architectural philosophies. While ICP's introduction of canisters and the NNS revolutionized the industry, its rigid security and governance models create points of weakness in the implementation. AO, on the other hand, aims to strengthen the decentralized nature of the technology and implement more efficient processes for computation, creating a more flexible and minimalist platform than ICP. While ICP may be the most similar technology to AO on the market, other alternatives exist like Solana, traditionally known for its high transaction speed and low costs.

[5] DAO: a decentralized autonomous organization is a type of organization that uses computer programs to govern itself, without a central authority.

2.2 AO vs Solana

Both Solana and AO represent distinct innovative approaches within the blockchain and Web3 ecosystem, yet they differ in their underlying technologies, target use cases, and overall impact in their respective spaces and market. While Solana is one of the most impressive high performance blockchain infrastructure advancements in recent years, AO represents a new frontier in the future of scalability, security, and composability in decentralized computing. This section will cover the intricacies of both technologies, and how their unique designs cater to their specific uses: Solana's fast transaction processing and native programming language (Rust) and AO's developer and community driven environment with its unique Permissionless Ecosystem Funding (PEF) and asynchronous computation, allowing the empowerment of all creators through decentralized collaboration, tokenized engagement, and limitless scalability.

While both AO and Solana strive for high performance, security, and decentralization, each employs distinctly different mechanisms to reach those goals. Solana, for instance, harnesses its innovative Proof of History (PoH) to establish a verifiable timeline for events, enabling validators to skip real-time coordination on transaction ordering. This pre-sequencing of events significantly reduces consensus overhead, allowing the network to confirm transactions faster than traditional blockchains that rely on constant synchronization. As a result, Solana boasts a theoretical throughput of over 65,000 TPS, making it one of the fastest transactional blockchains in the world. Further enhancing its ecosystem, Solana integrates Rust, a robust and versatile programming language that enables developers to build applications where the core logic runs on the blockchain, rather than relying on a single server controlled by a company. Rust's memory safety and concurrency features also cater to high-performance workloads, enabling a wide range of complex, onchain dApps to flourish in the Solana ecosystem.

While Solana confines its users to a single programming language and development approach, AO encourages developer creativity by allowing them to run an instance of any virtual machine on the platform, allowing developers more agency over the frameworks that they build their applications on. The ability for developers not to be locked into a single language like Rust on Solana or Solidity on Ethereum allows for lower barriers to entry for beginners developing on AO, with existing libraries of tools being able to be imported and used. Along with this added flexibility, developers who build on AO also get future proofing and scalability unlike any other dApp platform, with the ability to migrate to newer VMs, without having to replatform projects or face the risk of being "locked in" on a single language or digital toolset.

Another way AO improves over traditional dApp platforms—like Solana or Ethereum—is by introducing a native mechanism to reward creators directly for building on its ecosystem. On existing dApp platforms, developers have to rely solely on their own applications' economic models—be that transaction fees, subscriptions, NFT sales, or other revenue streams—to see any return on their work. AO addresses this challenge by introducing Permissionless Ecosystem Funding (PEF), a concept that brings a new, meritocratic, and transparent way for builders to earn on the platform. Unlike traditional grants or funds, where developers pitch to centralized entities, PEF channels liquidity and funding directly from users to developers whenever those applications are actively used. This model not only offers flexibility—since it's driven by onchain interactions rather than external approvals—but also injects a level of transparency and accountability that has been largely absent in Web3 development up to now.

The backbone of PEF is the AO token itself (discussed in detail later). When users hold bridged assets on AO, they earn a yield of AO tokens, they can choose to redirect any percentage of this yield they would otherwise earn to dApps built on AO and earn the dApps' project tokens in return. This provides a continuous, transparent flow of tokens that fuels innovation and encourages creators to continuously build on AO—without having to rely solely on transaction fees, sales, or private fundraising. At the same time, users gain full agency over where their money goes, supporting the projects they believe in.

As these communities grow and improve, AO will grow alongside them. Unlike traditional protocols like Solana, which operate with a shared global state, AO has completely localized states, which means that although nodes on AO are connected, updates can happen asynchronously, allowing different areas of the network to operate more independently. As a result, AO can achieve greater performance and avoid bottlenecks faced by traditional protocols that operate on a shared global state.

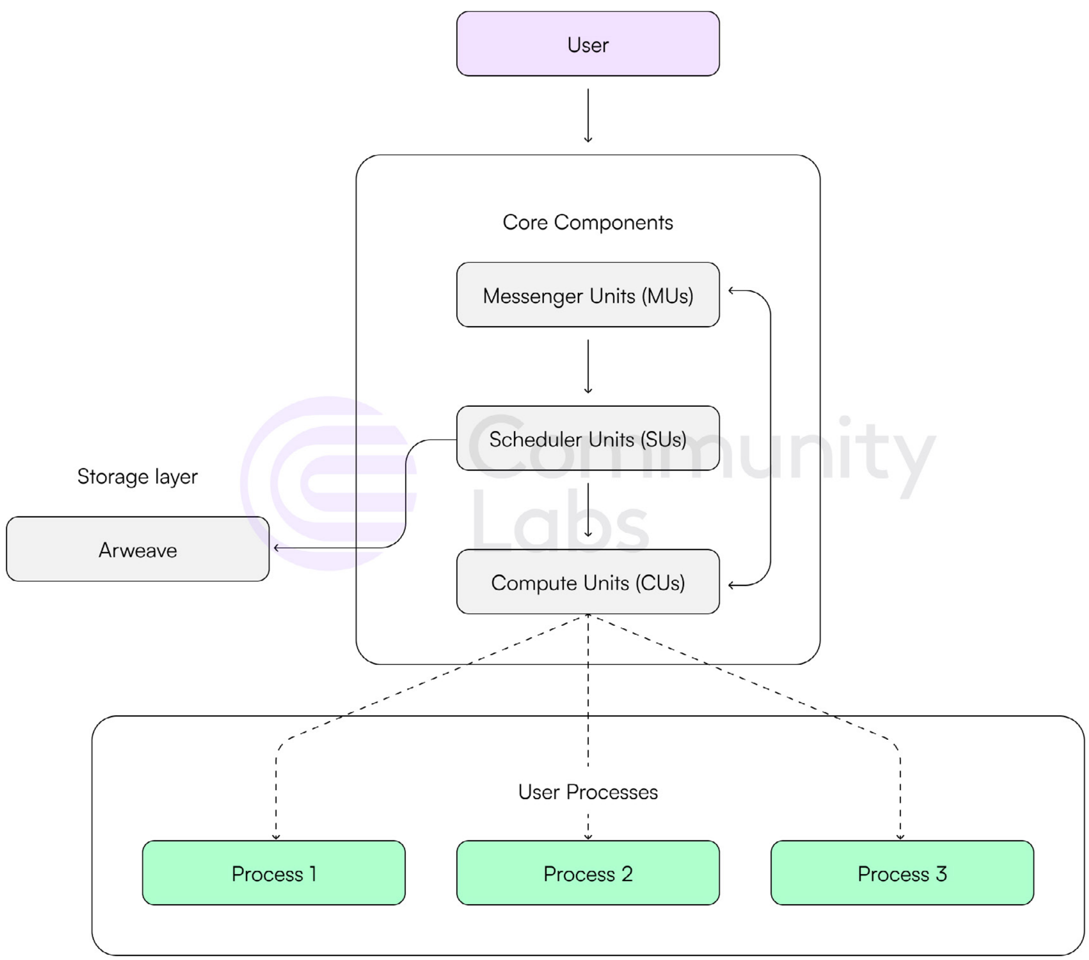
These features grant AO unparalleled scalability without sacrificing security. In modern blockchains, scalability is paramount—just look at the world's two most popular blockchains, Bitcoin and Ethereum, which together boast over 25,000 active nodes[6]. As these networks grow, their performance generally does not scale in proportion to their size. This is where AO truly sets itself apart; rather than being constrained by an ever-increasing number of participants, the performance of AO's network expands alongside its user base without an exponentially increasing cost to the user. By harnessing asynchronous architecture and localized states, AO ensures each new participant bolsters the network's capacity, making scalability a natural byproduct of adoption, rather than an inevitable hurdle to overcome.

[6] *Bitnodes, "Reachable Bitcoin Nodes,"* accessed February 9, 2025, https://bitnodes.io/; *Ethernodes, "Ethereum Network & Node Explorer,"* accessed February 9, 2025, https://ethernodes.org/

In conclusion, both AO and Solana bring significant innovations to the blockchain ecosystem, though each addresses distinct markets and use cases in its own way. By embracing asynchronous architecture, developer creativity, and PEF, AO aims to redefine decentralized computing—offering an experience for users that is more flexible, scalable, and transparent than ever before. In the next section, we'll dive deeper into the technical structure behind these core features, examining precisely how AO delivers its promise of a fast, secure, and community-centric platform poised for the future of Web3.

3. How AO Works

AO operates through three primary components: Scheduler Units (SUs) handling message sequencing, Compute Units (CUs) executing state calculations, and Messenger Units



(MUs) relaying messages between processes. AO's modular architecture allows processes to specify their VMs, scheduling mechanisms, and security parameters while maintaining verifiable computation and minimized trust assumptions.

3.1 Processes

Each process in AO is defined by a message log, initialization data, and computing environment. These processes operate independently without shared memory, communicating exclusively through a message-passing system.

The state of a process is determined by a function that computes based on the message log and environment specifications. When initialized, processes can specify their operational parameters including memory limits, operation bounds per message, and required virtual machine extensions.

AO processes utilize a "lazy evaluation" architecture where consensus is reached on message logs rather than computation results. This means nodes don't need to perform computation to reach consensus about state transitions. Instead, state is implied "holographically" by the Arweave-hosted message log and optionally stored within CUs. This approach enables unbounded computation and increased efficiency, with compute costs delegated to users who can either calculate their own states or request execution from nodes. Processes can be activated through direct user interaction, inter-process messages, or autonomous scheduled operations.

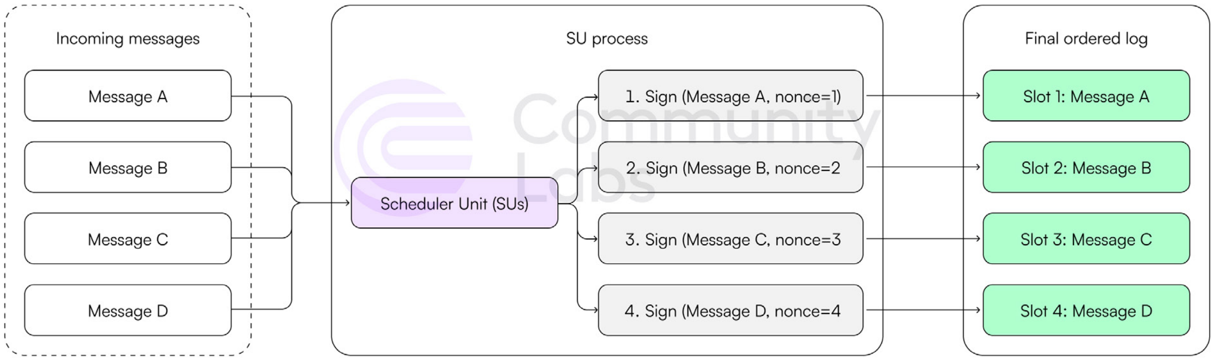
3.2 Scheduler Units (SUs)

SUs are responsible for the foundational task of assigning sequential ordering to messages within processes. They perform two critical operations:

1. Creating an assignment by giving the message a unique incremental nonce n and producing a cryptographic signature over both the message and nonce. This creates a verifiable record of the message's position in the processing sequence.
2. Storing this signed assignment and the message itself onto Arweave to ensure permanent availability. This guarantees that the message data remains permanently available and accessible to all network participants.

SUs can be implemented in different ways—whether decentralized, centralized, or user-hosted—allowing developers to configure a sequencing model that matches

Message sequencing and slot assignment



the requirements of their process at creation time. Once specified, this scheduling configuration becomes a core part of the process's definition. A process indicates its preferred scheduler, when it's first created/spawned, through tags in its initialization data, or as part of its core configuration.

3.2.1 Slashing and Recovery Mechanisms

There are several security measures to ensure SUs operate correctly. If a SU fails to perform its duties or acts maliciously, it will be subject to "slashing" (penalty) conditions. These conditions include: failing to perform message assignment, failing to persist data to Arweave, or assigning the same slot number to different messages.

If a SU fails in its duties or becomes unresponsive, the protocol includes recovery mechanisms through the AO-Sec Origin process. This allows any network participant to challenge a SU's behavior, and if the challenge is successful, the process can become "unhosted" and select a new SU through a market-based mechanism.

3.3 Compute Units (CUs)

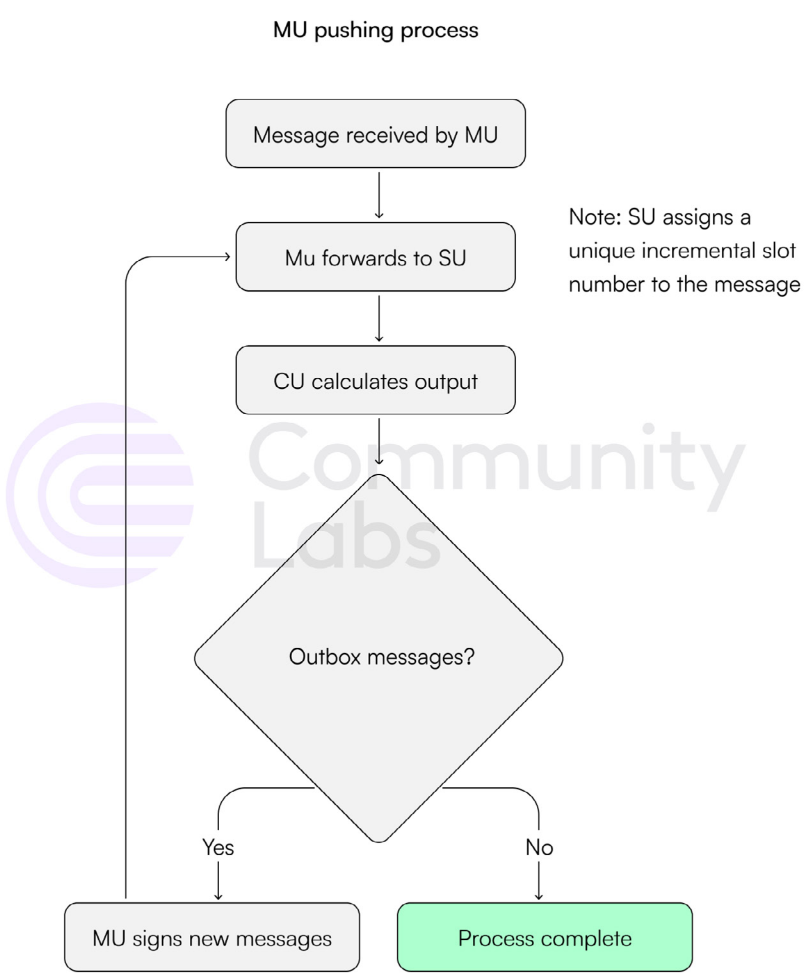
CUs are nodes that execute the virtual machine (e.g. WASM) functions for processes on AO. When users or MUs need to calculate the state of a process, they can utilize CUs. Unlike SUs, which must sequence messages for processes they've accepted, CUs operate in a completely voluntary manner so no CU is required to calculate the state of any particular process. This creates a peer-to-peer marketplace where CUs compete with each other to offer computation services, balancing factors like price, computational requirements, latency, and other quantitative or qualitative requirements.

When a CU executes a computation for a process, it produces three key outputs:

1. A new process state
2. Set of any resulting outbound messages
3. Signed attestation of the computation

The new process state ensures that all network participants can track the current status of processes, the outbound messages allow processes to communicate and coordinate with each other, and the signed attestation provides cryptographic proof that computations were performed correctly—enabling the network to verify results and penalize any dishonest behavior through token slashing.

3.4 Messenger Units (MUs)



MUs operate as relay nodes in the AO network, facilitating message transmission through a process called ”pushing.” When an MU receives a message, it forwards it to the appropriate SU for slot assignment, then coordinates with a CU to calculate the interaction output. The MU continues this process recursively for any resulting outbox messages, signing and forwarding them until no new messages remain.

MUs support both standard message processing and an optimized ”cast” operation where messages are sent to SUs without waiting for responses. The integrity of message handling is enforced through a slashing mechanism, where invalid message signing can result in stake penalties for the MU, with provisions for MUs to claim against CU stakes if the invalidity stems from incorrect computation.

For enhanced security, MUs implement stake aggregation by collecting attestations from multiple CUs to meet process security requirements. This involves bundling attestations from available CUs, each contributing partial stake, into a unified stake bundle that satisfies predefined security criteria. The aggregated bundle is then transmitted through the SU to the target process, ensuring both efficient transmission and robust security guarantees through distributed stake commitment.

3.5 Technical Validations

A key component of AO’s technical validation framework is the Sybil-resistant Independent Verification sub-protocol (SIV). SIV enhances AO’s foundational security by introducing an additional layer of Sybil-resistance, ensuring reliable message validation through a deterministic ordered set of attestors. Clients can specify the number of attestor signatures required for result validation, allowing them to balance security needs against cost and latency.

The protocol provides quantifiable security guarantees through its mathematical model where the likelihood of a staker deceiving a client decreases exponentially with each additional attestor involved.

SIV’s consensus mechanism is notably streamlined, operating with $O(1)$ computational complexity for partial consensus operations, compared to the $O(n)$ complexity typical of full blockchain networks. This is important as it allows users to achieve complete consensus on results when needed, while normally operating with partial consensus for improved efficiency. These characteristics enable systematic validation of both security guarantees and consensus efficiency within the network.

3.6 AO-Core: an Operating System (OS) on HTTP

AO-Core is a reference OS environment that implements the Converge protocol (Converge), accepting computational requests over HTTP. It runs entirely through its HTTP API where users send Converge messages to request computational resources from the node.

It is configured through a boot message and can run on existing operating systems or as a standalone OS with Trusted Execution Environment (TEE) support. It executes computations using devices (programmable modules) based on parameters set in the boot message, with all interaction happening through standard web protocols.

Devices in Converge are modular programs that take input messages and return valid Converge responses. Through the stack device, multiple devices can be composed together to create more complex computations, allowing developers to implement new functionality without modifying the core protocol. For example, an AO process might use a stack combining devices for deduplication, WASM interfaces, JSON bridging, and multi-pass execution. This allows developers to build and deploy their own computational logic by implementing new devices or composing existing ones, without needing to create an entirely new blockchain or seek their own pool of node operators.

For instance, WeaveDrive is a device for AO that enables processes to access data stored on the Arweave network as if it were part of a local file system. Integrated within AO-Core, WeaveDrive allows applications to retrieve, process, and store large datasets directly from Arweave with minimal overhead, leveraging a lazy download-on-read approach. This method ensures that only the required data is retrieved, optimizing performance. Additionally, WeaveDrive ensures deterministic execution by halting the process when a requested transaction is not available, preventing inconsistencies in execution until the data becomes accessible.

AO-Core stands out by fundamentally creating an OS that runs entirely through HTTP messages while using TEE hardware for tamper-proof computation verification at near-native speeds. AO processes are enabled to execute through a stack of "devices" that can run WASM code and interface with other systems, turning nodes into a decentralized computing layer that bridges traditional web infrastructure with AO's permissionless computing environment.

This modular approach can rapidly adapt to new computational requirements and use cases simply by deploying new devices. Instead of building entire new blockchain implementations, developers can create just the specific logic they need as a device, combine it with existing devices, and even charge fees for its usage. Users can then select and combine these devices to create their desired computational environment, requesting execution from nodes that support their chosen device stack.

4. Key Features

4.1 Verifiable & Confidential Compute

AO implements verifiable and confidential compute through TEEs, specifically using AMD SEV-SNP hardware in version 1.0 initially. This system runs as an isolated guest OS in a VM, with only a small 6% performance overhead compared to native execution. The key benefit is that users can trust their computation results based on hardware guarantees, without having to trust the operators of the nodes themselves.

Nodes generate ephemeral keys at boot (never written to disk) and are only accessible through the HTTP interface-—no Secure Shell (SSH[7]) or external access is allowed. The node's ephemeral address and node message ID are both included in the attestation that's generated on the machine and made available through the node's API.

When computations occur, the hardware provides cryptographic attestations proving that the code ran in a secure environment that even the node operator cannot tamper with. These attestations allow anyone to verify that their computations remained confidential and were executed correctly within the TEE.

4.2 Virtual Machines & Deployment Options

4.2.1 AO as the Next Decentralized Physical Infrastructure (DePIN) OS

In current DePIN[8] architectures, there is a fundamental technical challenge: most blockchain systems force all computation through a single, shared execution thread. This creates severe limitations for physical infrastructure networks that need to process vast amounts of real-time data from sensors, coordinate multiple devices, and handle complex payment flows simultaneously. For example, when a decentralized wireless network needs to process thousands of connection requests while simultaneously handling bandwidth accounting and payment distributions, traditional blockchain architectures become a

[7] SSH: a method for securely sending commands to a computer over an unsecured network.
[8] DePIN: a network that uses blockchain technology to create and manage physical infrastructure.

bottleneck, forcing all these operations to wait in line for execution.

AO enables multiple computational processes to run in parallel, communicating through a message-passing layer that settles on Arweave. This architecture has immense potential to support DePIN applications by allowing them to handle multiple computational tasks concurrently (like payment processing and state management), but it's important to note that the actual interface with physical infrastructure would require additional implementation layers.

4.2.2 Decentralized Serverless Functions

AO's decentralized serverless functions operate through aos[9] by allowing developers to deploy message-driven processes to the network where they are stored permanently on Arweave and executed by CUs in response to inbound messages. Each process is defined by handlers that specify both the message patterns to match and the computational logic to execute, with no need to manage underlying infrastructure. The processes are truly serverless as they run only in response to messages and can be executed by any CU in the network (that meets the process' specified requirements).

Unlike traditional serverless functions, AO processes maintain state between executions and can be dynamically updated through the aos environment. Processes can communicate with each other through messages, enabling developers to create networks of interacting serverless functions that work together to perform complex tasks.

4.3 Cron

Traditional smart contracts only execute programs in response to user transactions. AO however, enables autonomous process activation through scheduled "cron" interactions.

The implementation requires two key steps. First, when spawning a process through the aos console, you specify the cron interval. Intervals can be set in seconds, minutes, hours, or blocks. Second, you must start a monitor event to begin receiving the cron messages.

Cron messages are first produced in a process's outbox, but they need to be actively monitored for the messages to be pushed through the network. The subscription model allows any party (including the process itself) to pay a node to handle the scheduled message processing. Users and processes can establish subscription relationships with MUs, enabling automated pushing of messages resulting from timed cron interactions.

[9] aos: a decentralized operating system for AO, allowing developers to launch command-line processes that function like smart contracts within its decentralized network.

4.4 How Data is Returned

AO handles data returns through a combination of attestations and message passing. When a CU executes a computation, it produces a signed attestation containing three key elements:

1. The new process state
2. Any outbound messages
3. A cryptographically signed attestation of the computation

This attestation serves as proof of the computation's result and its correctness. The results then flow through MUs, which act as intermediaries to relay the data. When a MU receives computational results from a CU, it can verify the attestation and forward the results to the appropriate parties.

This forms a chain where computations can trigger further computations through outbox messages, with each step being verifiable through signed attestations. If any attestation is found to be incorrect, stakeholders can challenge it and potentially slash the CU's stake (this incentivizes honest result reporting).

4.5 WASM Containers

WebAssembly (WASM) serves as AO's initial reference VM environment. Each WASM container in AO has 64-bit support, which means developers can create applications that utilize more than 4GB of RAM and theoretically achieve a RAM ceiling of 18 exabytes. The WASM implementation enables prolonged computation durations while taking advantage of WebAssembly's comprehensive compilation toolchain. WASM containers support these computational workloads while maintaining WebAssembly's inherent security and execution guarantees.

AO supports a complicated caching system for WASM modules and instances, with support for streaming compilation and memory management up to process-defined limits. More specifically, a two-tier Least Recently Used (LRU) caching system[10]:

1. For compiled WebAssembly modules and another for module instances
2. WASM binaries are streamed from Arweave with configurable memory limits enforced at process level

[10] LRU caching system: a memory management strategy that optimizes performance by evicting the least recently accessed items first while retaining more recently used data in cache.

This architecture enables efficient execution of WASM modules while minimizing redundant compilations and network requests.

4.6 Economic Security

AO achieves economic security through a market-driven approach where users purchase specific levels of security for each message they send rather than relying on block rewards or transaction fees (like in Ethereum).

4.6.1 Staking

Any time a user wants to send a message on AO that others need to trust, tokens are ”staked” (locked up) as collateral to guarantee good behavior. This stake can be slashed (taken away) if you misbehave.

- SUs - Must stake tokens to get the right to order messages for a process. If they double-sign or fail to publish messages, they lose their stake.
- CUs - Must stake tokens to attest to computation results. If they give incorrect results, they lose their stake.
- MUs - Must stake tokens to relay messages between processes. If they send invalid messages, they lose their stake.

Users are required to insure their messages to the security level required by their counterparts. For example, if a user wants to secure a message worth $500,000 with a 30-minute stake-exclusivity period, and stakers are expecting a 10% annual return on their capital, the cost would be:

$$\text{Cost} = \text{MessageValue}\left(\frac{\text{AnnualReturnRate}}{\text{MinutesInYear}}\right)\text{ExclusivityPeriod}$$

$$\text{Cost} = \$500{,}000 \times \left(\frac{0.10}{526{,}000}\right) \times 30 = \$2.85$$

Note that the price varies depending on how much security is needed (exclusivity period) and return rate for stakers. Hence, cost will be higher for increased security and higher returns, lower for decreased security and lower returns.

This stake remains locked during the exclusivity period and can be slashed if any malfeasance is detected, creating direct economic incentives for security. Creating peer-to-peer markets where SUs, CUs, and MUs manage security on a per-message basis rather than through block space allocation.

4.6.2 AO-Sec Origin

AO-Sec Origin is the foundational security process for AO, acting as custodian of staking tokens and maintaining ownership records for all units staked. It provides critical back-stop security functions, including staking, slashing, and un-staking of tokens.

The process handles three types of challenges for SUs: liveness (unresponsive SUs), double assignment (multiple signatures for the same slot), and non-publication (unpublished messages). Each challenge has specific resolutions - processes becoming unhosted, forced message publication, or staker voting on availability.

When processes become unhosted, AO-Sec Origin enables a market-driven re-hosting system where SUs can offer to become new hosts through a negotiation process. Built on Arweave's Byzantine Fault Tolerant consensus, this provides trustless operation without requiring honest majorities except in optional voting scenarios.

4.6.3 Proof of Stake (PoS) vs Succinct Proof of Random Access (SPoRA)

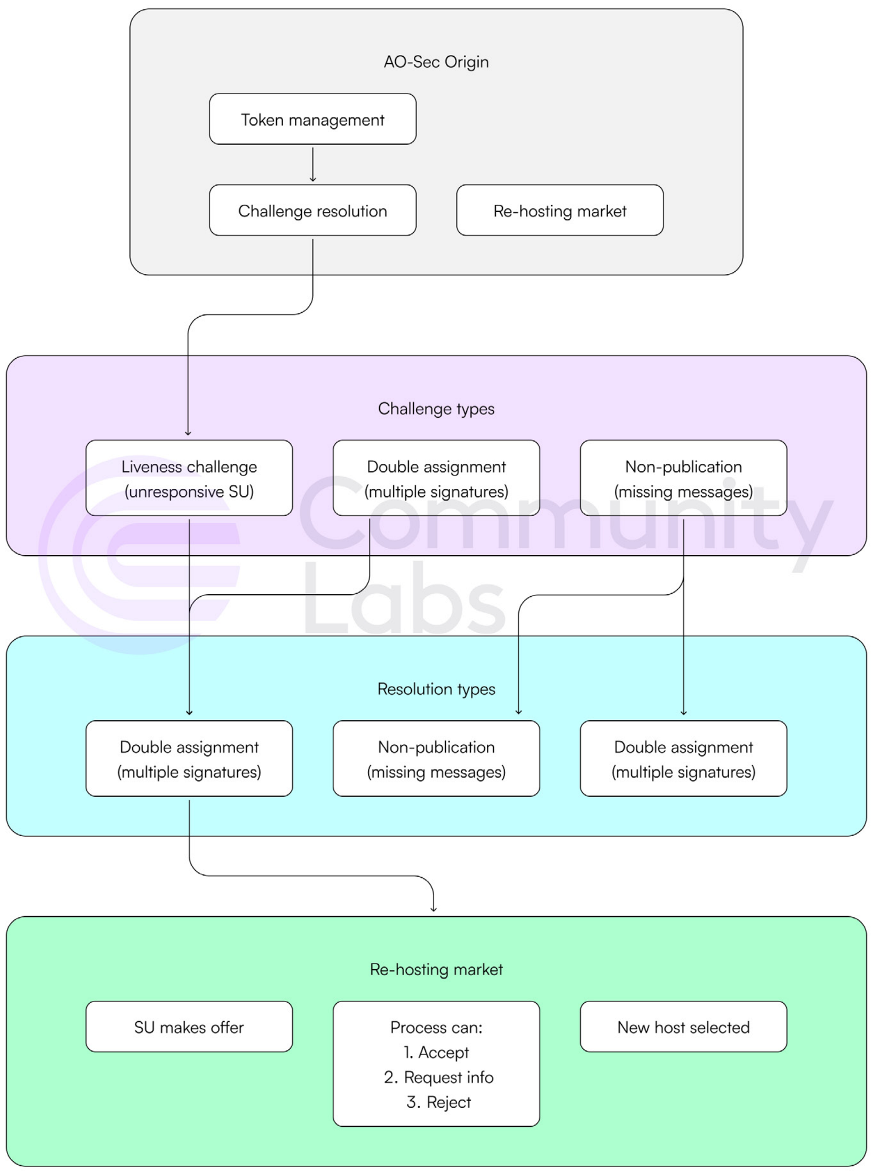
The underlying data storage for AO is handled through Arweave, which provides a permanent and verifiable record of process messages and interactions. This direct storage access enables verifiable logs of all process states and messages, which are essential for AO's message-passing architecture and consensus.

Arweave's SPoRA and AO's PoS represent two distinct yet similar approaches to consensus. The fundamental difference is that SPoRA is about proving storage/access of data through mining, while AO's model is about securing individual messages through economic stakes. They serve different purposes:

- AR's SPoRA: Ensures data availability and quick retrieval
- AO's PoS: Provides customizable security guarantees for message passing

SPoRA replaced Arweave's previous consensus mechanism, which combined Proof of Work (PoW)[11] with Proof of Access (PoA)[12]. This dual approach didn't effectively incentivize rapid data retrieval. This led to miners favouring remote storage pools over maintaining individual nodes, and it consumed significant energy primarily for computation rather than useful storage operations.

AO-Sec Origin Process



[11] PoW: a blockchain consensus mechanism that incentivizes network validation by rewarding miners for adding computational power and difficulty to the network.

[12] PoA: a consensus mechanism used by Arweave to ensure the permanent storage of data by requiring miners to retrieve and use an old block from the network's history before adding a new block.

The new SPoRA mechanism, introduced in Block 633720 on February 24th 2021, fundamentally shifts the mining dynamics by making mining profitability directly proportional to data access efficiency. It randomly polls and verifies data across the network, ensuring miners maintain quick access to their stored data. SPoRA aligns mining rewards with storage utility rather than pure computational power (disincentivizing CPU pooling).

AO diverges from traditional PoS by implementing a message-centric security model rather than block validation. Instead of validators staking tokens to create blocks and earn rewards, AO uses stakes to secure individual messages through a market-driven pricing mechanism. The security is provided through the AO-Sec Origin process which manages token staking/slashing and enables stake-exclusivity periods where collateral is locked for specific timeframes to prevent double-spending of stake across messages.

4.7 Multi-Chain Applications

AO enables multi-chain applications by allowing processes to execute across multiple blockchain ecosystems without requiring direct interoperability between on-chain assets. To facilitate this, Compute Unit (CUs) can securely generate and store private keys within a TEE and sign transactions for external blockchains, enabling processes to interact with external networks without relying on asset bridges.

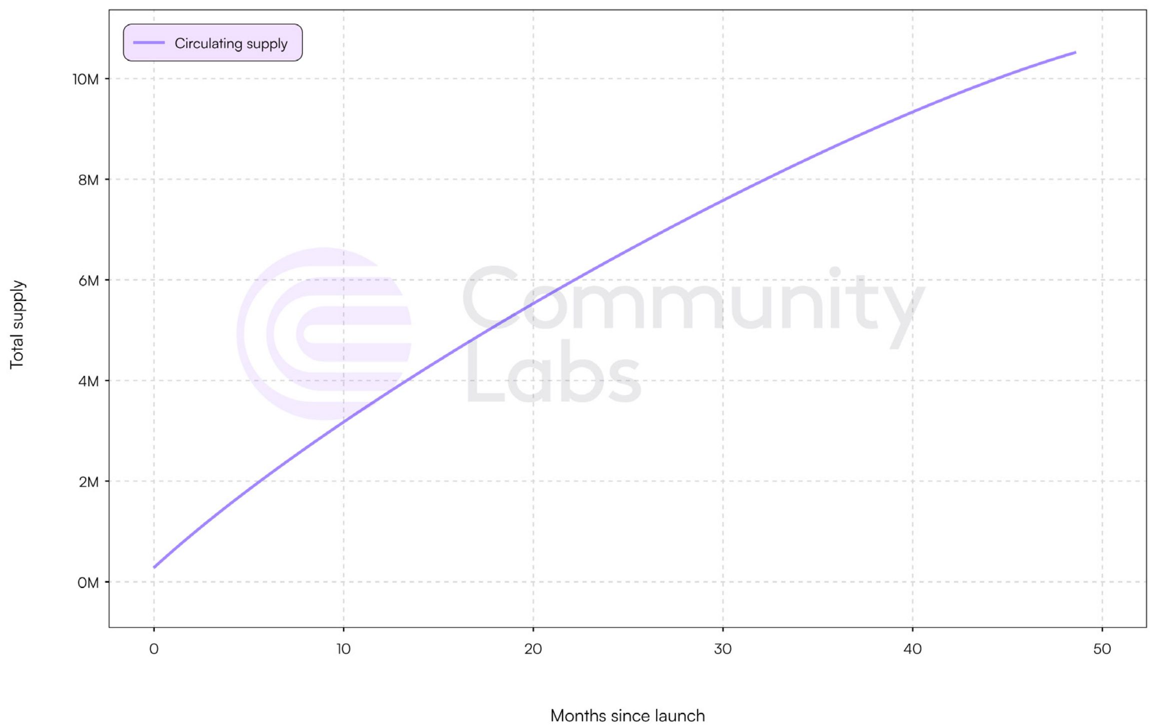
For example, an application might use Ethereum for financial transactions while leveraging AO and Arweave for decentralized computation and permanent data storage. This approach allows developers to design more efficient and scalable systems by distributing workloads across specialized blockchain environments while maintaining decentralized execution on AO.

5. Tokenomics

5.1 Fair Launch

AO's Fair Launch token distribution model is designed to ensure complete community ownership. Typically, token launches reserve tokens for investors, team members, or early backers. AO, however, has taken a completely community-focused approach. The entire supply of 21 million AO tokens will be distributed to users with absolutely no pre-sales, pre-allocations, or preferential access of any kind.

AO Token Distribution Over Time



The distribution follows a four-year halving cycle similar to Bitcoin, but with a smoother curve. Distribution of $AO happens through two main channels: holding $AR and bridging other yield-bearing crypto assets.

5.1.1 Hold $AR to Earn $AO

Anyone holding AR tokens automatically earns AO tokens with distribution calculated every five minutes based on their AR balance and with 36% of the total supply distributed to AR token holders over time. Users with AR in non-custodial wallets receive their AO tokens directly, while those holding AR on exchanges will receive their tokens through their respective platforms.

5.1.2 Yield Bearing Assets Deposited to Earn $AO

Users who bridge supported assets from other networks (e.g. staked ETH and DAI with more supported tokens coming) to AO will receive derivative tokens representing their bridged assets. Bridging assets account for 64% of the total supply of AO to be distributed. Bridged assets can be used within the AO ecosystem while continuously earning AO tokens. When users bridge assets like staked ETH to the AO network, they're not just earning tokens; they're providing valuable liquidity for applications in the AO ecosystem.

5.1.3 The Permaweb Index (PI)

The Permaweb Index (PI) is an onchain asset index that autonomously tracks and adjusts its composition based on real user activity in the AO and Arweave ecosystems. Rather than following a fixed allocation model, PI dynamically mirrors the distribution of AO yield delegations, ensuring that it remains aligned with the most actively utilized components of the permaweb.

PI's architecture integrates an onchain rebalancing system, which automatically adjusts its holdings to reflect shifts in ecosystem participation. Users who deposit $AO can delegate their yield to emerging fair-launch projects, and PI allocates weight accordingly, capturing the economic distribution of the network without manual intervention.

At its core, PI maintains exposure to:

- Arweave and $AR: The decentralized storage layer for permanent data availability.
- AO and $AO: The computational layer enabling stateful execution and decentralized processing.
- Ecosystem Projects: Fair-launch tokens that receive delegated AO yield from participants.

For Arweave holders, PI serves as the default option for yield allocation, allowing users to passively accumulate a diversified mix of $AR, $AO, and emerging fair-launch project tokens. This structure removes the need for active rebalancing while ensuring that holdings are dynamically adjusted in response to network-wide participation trends.

As an autonomous index, PI does not rely on centralized decision-making or fixed schedules for asset adjustments. Instead, it functions as a decentralized economic filter, allocating weight based on protocol-level adoption signals. This ensures that its composition remains responsive to real network activity, continuously reflecting the evolution of the AO and Arweave ecosystems.

6. Use Cases & Projects on AO

6.1 Large Language Models (LLMs)

AO has taken a monumental leap by launching with WebAssembly 64bit support, boasting a theoretical RAM ceiling of 18 exabytes. This increases the limit of RAM per instance of a CU virtual machine from 4GB on ICP and 10MB on Solana to a considerably larger amount. Any practical limit can also easily be changed without any changes to the protocol in the future. This large memory limit already allows most of the common open-source LLMs to run within a process on AO.

WeaveDrive, an AO device, enhances this even further. If referenced in the smart contract, AO can intelligently pull data from Arweave to use for computation purposes, such as training LLMs. The most interesting part about WeaveDrive is the ability for it to be implemented in the OS layer, so a developer can use Rust, C, Solidity, or any language they desire and treat data from Arweave like an actual hard drive in a computer, performing computations on that data. This means that AO can have unbounded compute power and limitless storage access, making it perfect for LLMs.

Through AO, the future of LLMs will be one that is more adaptable, more resilient, and less prone to censorship, manipulation and bias.

Llamaland - An AI powered massively multiplayer online game on AO. Offers virtual worlds with quests, fostering interactions with other players, all in a decentralized environment. Llamaland is revolutionizing the concept of non-player characters (NPCs), by using AI agents to automate their behavior, as well as providing dynamic updates leading to an enhanced gaming experience.

Apus - Apus Network is developing a decentralized GPU network to deliver secure, efficient, and scalable AI inference and training. Apus acts as a device to AO, permitting a deterministic execution environment for GPUs. They offer a decentralized incentive model using AO and APUS tokens ($APUS). These features combined with AO's unbounded computation possibilities means that users can get the best AI models at the most cost effective prices.

6.2 Agents

AO is uniquely positioned for AI agents through its combination of autonomous process activation, unlimited computation capabilities, and trustless operations.

Unlike traditional smart contract platforms where programs must be externally triggered, AO processes can "wake themselves up" on a schedule, allowing agents to operate independently without central coordination. This autonomy guarantees optimal timing of actions taken by agents, as well as allowing real-time responses to changes. Combining this with AO's unbounded computing potential and inherent lack of trust required, AI agents can get the closest they have ever been to their true potential of autonomy and independence.

Dexi - Acting as a central hub for autonomous agents within the AO ecosystem, Dexi can automatically identify, collect and aggregate real time data from various events within the AO network. Dexi behaves as a mediator, facilitating trustless agents to make informed decisions leading to efficient asset settlements.

6.3 Databases

AO's arbitrary scalability combined with direct integration with Arweave allows an unprecedented level of flexibility for databases. Through parallel processing, large scale data analysis, including machine learning, can occur in a database hosted on AO. This is considerably more efficient than current database systems using traditional computing technology, which requires exchange of data across multiple programs to achieve the results AO manages in a seamless and streamlined manner.

ArFleet - Designed to facilitate the purchase of time-limited data storage on a permissionless network. With AO's advanced database systems, ArFleet is able to hold collateral and rewards according to terms outlined between purchasers and sellers of storage. Also, AO's wide range of flexibility and adaptability features allow ArFleet to support transactions using any mutually agreed tokens. These quality of life features are only possible through the use of AO.

6.3.1 Indexing

Not too dissimilar to the advantages achieved by using AO for databases, indexing also benefits from the immense scalability offered by AO. Any number of indexing tasks can run at the same time without interfering with one another. Indexing tasks can also be scheduled to run automatically at any interval or as a result of specific events like data updates. This practically eliminates the need for human or any other manual intervention while indexing.

AR.IO - a permanent cloud network consisting of decentralized gateways that manage domain names, hosting, indexing, querying, and caching, while delivering over 99% of Arweave's data. Having recently migrated their smart contract to AO, AR.IO now benefits from improved performance, scalability and even deeper integration within the Arweave ecosystem. AO infrastructure can be deployed on AR.IO gateways, giving developers unparalleled access to local data access and indexing without any rate limits. This means faster queries, greater reliability and a seamless development experience.

6.4 Finance

AO is launching with an impressive suite of implementation options for finance. Its high throughput makes it perfect for real time financial systems that require timely transaction settlements.

AO's integration with Arweave, enabling permanent storage of transaction logs, simplifies audits and enhances financial regulation compliance. These immutable and easily accessible transaction logs also allow for data transparency and long term availability. In case of failures, it is easy for financial operations to recover using the state stored on Arweave as well.

Support for tokenized assets is also present in AO, which facilitates the issuance, trading and settlement of these tokens. AO allows recipients of high value or sensitive financial transactions to set a "stake-exclusivity" period. During this period, the collateral securing the transaction is locked exclusively for it, effectively eliminating double staking.

By combining these features with dynamic security and unbounded compute power, AO creates an effective foundation for all types of financial operations, especially DeFi and large scale financial transactions.

Botega - Botega is a decentralized exchange platform built on AO. Utilizing AO's autonomous agent capabilities, Botega enables advanced order types and dynamic liquidity allocation for assets. It is completely decentralized and permissionless, including its front-end UI, and all transactions are uncensorable.

Astro - Built on AO, Astro is focused on building unified liquidity for the Arweave and AO ecosystems. With projects such as USDA (stablecoin), as well as Quantum Bridge (a portal between Arweave and AO), Astro Labs has an emphasis on security and reliability.

LiquidOps - A lending and borrowing protocol, entirely decentralized and specifically designed for the Arweave and AO ecosystems. Allows collateral-based loans, enabling borrowers to gain liquidity without having to sell their digital assets.

6.5 Gaming

Due to AO's hyper-parallel computing capabilities, massively multiplayer online (MMO) games[13] and open world games can benefit tremendously from its integration. In-game worlds can be fully autonomous and dynamic, changing based on players' actions or other pre-specified conditions (seasons, holidays, etc.). The parallel nature of these processes also means that resource-intensive simulations can be run completely independently from all the other processes, and that there is significant resilience against downtime. There is also in-built cheat resistance, as cryptographically verifiable computations prevent modifications to game logic.

Asset trading can also occur without a need for intermediaries, allowing a trustless exchange of in-game items. Enhancing this further is the interoperability with DeFi and financial transactions as a result of hosting on AO, as in-game assets can be designed to generate yields to encourage incentives for player engagement.

Basejump - A Web3 platform that combines AI-driven game generation with AO, enabling users to instantly create and permanently store games and assets onchain while ensuring cross-game interoperability. Unlike closed ecosystems like Roblox, Fortnite, and Minecraft, Basejump is fully open and permissionless, allowing users to create, own, and monetize their games without platform restrictions. This is made possible by ACTION, a substrate built on AO that powers Basejump and other applications, providing an open economic model where game creators benefit from true digital ownership, seamless asset portability, and onchain persistence.

6.6 Settlement Layers

Settlement layers can also benefit immensely from AO. Utilizing AO's economic efficiency tools, users only have to pay per transaction and only for their desired level of security. This is especially beneficial for settlement layers as traditional solutions are very rigid, causing unnecessary transaction expenses due to inefficiency and a lack of customizable options.

AO can also allow automatic execution of interactions. This facilitates periodic settlements without requiring external triggers. For example, a financial application can set a condition requiring the activation of settlement processes based on a payment threshold.

FusionFi Protocol (FFP) - FFP is a unified financial protocol that supports AI-driven financial models and enables seamless integration across the AO ecosystem, extending beyond financial platforms. The settlement layer executes operations according to

[13] MMO Games: an online video game with a large number of players to interact in the same online game world.

predetermined rules, such as asset transfers and collateral releases, leveraging AO's computing power for efficient and secure processing.

7. AO's Solution to Common Challenges

The blockchain trilemma– the challenge of balancing stability, security, and decentralization in a given protocol –has long been a cornerstone of both innovation and skepticism in the blockchain space. While many tout blockchain technology as the foundation for a decentralized future, critics often question its capacity to fulfill its lofty promises without compromising one or more of these three critical pillars. For every project that aims to overcome this trilemma, there are often trade-offs and compromises being made beneath the surface. This section examines the challenges inherent to this trilemma, common skepticisms, and how AO aims to overcome them.

7.1 Performance & Scalability

One common skepticism surrounding emerging decentralized computing platforms centers on both performance and scalability—and AO addresses these concerns through several innovative methods. Unlike Solana or ICP, which can reach a throughput ceiling due to fixed architectures, AO's performance naturally scales alongside its user base without scaling cost, leveraging a growing pool of participants to boost the network's capacity. A further improvement comes from AO's decision to move away from subsidized block rewards, a practice employed by Solana, Ethereum, and others that often leads to resource inefficiencies. Instead, AO adopts a market-driven approach to security, where users pay directly for the level of security and computational resources they require—optimizing overall resource allocation,enhancing performance, and creating a market where computation providers compete to offer the lowest cost to the user.

Beyond these measures, AO's holographic state mechanism provides a key advantage for scalability. Rather than forcing every node to process the exact same transactions—potentially creating network-wide bottlenecks– AO decouples state storage and computation from the consensus process. This approach is reinforced by on-demand state computation, which lets nodes retrieve and compute state as needed by accessing logs on Arweave. By removing the burden of universal synchronization for every single operation, AO eliminates many of the trade-offs that hamper traditional blockchains, establishing a more adaptive, high-performance framework that naturally scales with increasing demand.

In conclusion, by combining asynchronous architecture, market driven security, and on-demand state computation, AO effectively addresses the core skepticism around performance and scalability in decentralized computing. Instead of being hindered by stagnant frameworks or inefficient resource allocation, AO's design ensures that network capacity grows in tandem with user participation, paving the way for a more adaptive, high performance infrastructure built to meet the demands of an ever-evolving Web3 landscape.

7.2 Decentralization

A cornerstone of blockchain technology has always been the principle of decentralization, ensuring no single entity wields complete control over network processes. Yet skeptics often question whether real-world blockchains can maintain this ideal in the face of performance demands, governance decisions, and various outside pressures. Centralized choke points, such as decision-making bodies like DAOs, specialized hardware requirements like those of Solana, or dominant validator sets[14], can undermine trust and limit broader participation. This section delves into how AO navigates these concerns, preserving a truly decentralized framework without compromising on efficiency, security, or user experience.

The issue of taxing hardware requirements plagues many modern blockchains. Solana, for example, mandates validator nodes to have over 512GB of RAM, 1TB or more of SSD, and an ultra-fast internet connection (1Gbps or higher). These demanding specifications make it costly and challenging for everyday users to run nodes or otherwise participate directly in the network. As a result, power often concentrates among those who can afford specialized equipment, leading to reduced decentralization, and fewer participants end up controlling integral processes such as block production and validation.

AO circumvents these taxing hardware requirements by not enforcing a single, fixed specification for nodes within the network. Instead, CUs scale up or down based on the process they're running: a Raspberry Pi CU[15] could handle a basic token ledger, whereas a complex task like rendering a 3D model would require a more powerful CU. In this way, hardware demands vary according to actual application needs, rather than arbitrary network mandates. By removing rigid barriers to entry, AO broadens participation and fosters a more decentralized ecosystem overall.

Furthermore, AO reinforces its decentralized ethos by allowing modular security and governance mechanisms. Processes can independently choose their security parameters—

[14] Dominant validator set: a few validators hold most stake, undermining decentralization.
[15] Raspberry Pi CU: an AO CU powered by a Raspberry Pi which is containerized and deployable from a microSD.

whether it's staking, slashing conditions, or voting thresholds—creating a system whose protections scale alongside developer creativity. This flexibility empowers both users and developers to tailor their own onchain experiences, ensuring that decentralization remains a core pillar of AO's design.

In conclusion, by decoupling rigid hardware demands from network participation and empowering processes with customizable security parameters, AO demonstrates a clear path toward a truly decentralized infrastructure. The flexibility of AO nodes not only lowers barriers to entry for prospective operators, but also encourages inclusive approaches to governance. Ultimately, AO exemplifies how a blockchain can uphold scalability and security, without sacrificing a fundamental commitment to decentralization.

7.3 Privacy

Beyond the traditional issues of the blockchain trilemma, privacy remains a pressing challenge that most current protocols do not adequately address. AO, however, elevates confidentiality to a core priority by leveraging TEEs for tamper-proof computation as well as using cryptographic attestations. This ensures node operators cannot inspect or modify the data and code running inside these secured environments. Additionally, AO opens the door to zero-knowledge (ZK) proofs[16], which can further extend its confidentiality and onchain verification capabilities, allowing privacy-preserving interactions without exposing sensitive details.

This dual approach—hardware-level isolation through TEEs and potential ZK integrations—empowers AO to handle an array of resource-intensive tasks securely, from sensitive financial operations to AI-driven analytics, while maintaining high standards of privacy and integrity. By resolving concerns around privacy, AO stands prepared to address the next evolution of decentralized computing—sometimes called the blockchain "tetralemma"—and does so before it becomes a barrier to real-world adoption.

7.4 Price Constraints

While scalability, performance, and decentralization are three of the most widely discussed and significant concerns in decentralized computing, an equally pressing challenge involves price constraints. These constraints can deter new projects, hinder wider adoption, and create imbalances between resource costs and actual usage. In the following section, we explore how traditional decentralized platforms handle (or fail to handle) these cost-related issues—and, more importantly, how AO proposes to overcome them through a

[16] ZK Proofs: cryptographic methods that let one party prove a statement's truth without revealing any underlying data.

combination of market-driven resource allocation and on-demand computation models.

Traditional decentralized computing platforms often struggle with price constraints that stem from fixed fee schedules, subsidized block rewards, or congestion-based pricing models. These constraints often lead to volatile transaction fees for users, limited control over costs for developers, and an experience for both that is poorly optimized. AO loosens the grip of traditional price constraints by moving away from these previously mentioned systems.

AO doesn't rely on fixed block rewards or preset fee schedules. Instead, users pay directly for the security they require, bypassing traditional price constraints seen in blockchains such as Ethereum, where gas fees are often set by network congestion or protocol defined limits. AO's pricing structure is instead determined by real time supply and demand, ensuring that resources go to the most critical applications. AO's holographic state and on-demand computation mean that nodes will only pay for the exact amount of processing and storage they consume. On top of previously mentioned benefits, this will also lead to more cost-effective scaling, where network usage can expand without causing unpredictable or prohibitive increases in fees.

7.5 AO Addressing Skeptics & the Blockchain Trilemma

In exploring the major skepticisms—the blockchain trilemma, privacy, and price constraints—we've seen how traditional solutions often leave one pillar of the blockchain trilemma compromised. AO, by contrast, brings these elements into harmony through an asynchronous architecture that grows with user demand, a market-driven cost model that aligns fees with actual resource usage, and a secure, robust network foundation. The synthesis of these aspects removes the trade-offs we typically associate with blockchain technology, allowing developers to innovate freely without sacrificing speed, scalability, or decentralization. Ultimately, AO emerges as a holistic response to the trilemma, offering a path forward that could redefine the standards of reliability and innovation in decentralized computing.

8. Conclusion

Over the course of this paper, we've examined how AO tackles the core challenges facing decentralized computing, while preserving the foundational ideas of security and decentralization. By combining asynchronous architecture, localized states, market-driven resource allocation, and the Permaweb Index, AO stands apart from traditional platforms and embraces a model where network capacity grows hand-in-hand with user participation, where developers can innovate without fear of prohibitive fees or bottlenecks, and where every process– from dApp deployment to large scale analytics– benefits from the resilient and flexible framework AO has to offer.

Looking ahead to 2025 and beyond, AO represents far more than a modest advancement in blockchain protocols; it heralds a new era of business value, unbounded creativity, and opportunity in Web3. Picture a world where resource-intensive applications run seamlessly onchain, creators receive direct rewards from their communities, and the network scales effortlessly to meet surging demand—all while keeping costs low and data as private as possible for both users and developers. By delivering a holistic solution to the blockchain trilemma, AO paves the way for entirely new possibilities—from AI-driven ecosystems to massive multiplayer onchain experiences—transporting us into a future where decentralized computing isn't just powerful or scalable, but truly transformative for industries and innovators alike.

Community Labs fuels the people and technologies redefining the decentralized future

About Community Labs

Community Labs is uniquely placed at the center of the Arweave & AO ecosystem to build foundational tools and infrastructure. Our goal is to create innovative solutions to pain points faced by developers and users alike.

Through our venture studio, we provide founders and founding teams with the resources they need to identify product ideas and the ability to create, develop, and scale products from inception to mass adoption.