

Sylo Network

Update Paper

Contents

3	Introduction
4	The Problem: Walled Gardens in the Metaverse
5	The Solution: Sylo Network
6	Key Features & Use Cases
8	The Sylo Protocols: Their Role
9	Sylo Token & Tokenomics
13	The Roadmap

1

Introduction

The Sylo Network is built to power the next evolution of digital ownership within the open metaverse. Evolving from its roots as a decentralized communication protocol, Sylo has grown into a robust data layer designed to power true interoperability of assets and agents across metaverse ecosystems. This paper highlights the progress since joining forces to form Futureverse, its expanded use cases, and the revamped tokenomics of the Sylo Token (\$SYLO) – a utility token fueling real, practical applications for creators, game developers and asset owners across the open metaverse.

Imagine a metaverse echoing Ready Player One – one where assets aren't just tokens but fully portable, data-rich entities, able to be used anywhere. Built as a core pillar of The Root Network, Sylo is the solution to a metaverse without barriers, powered by *interoperable, ownable data*.

See how Sylo has become an integral part of the open metaverse vision, from its underlying protocols to its economic model, and why it's poised to redefine digital interoperability...

2

The Problem: Walled Gardens in the Metaverse

The promise of the metaverse – a boundless, interconnected digital universe – remains fractured by a persistent problem: walled gardens. In the Web2 paradigm, every game or platform operates as an isolated island. A character or item earned in one experience – like a skin in Fortnite – can't cross into another, trapped by proprietary data structures and centralized servers. This fragmentation locks assets into single ecosystems, stifling creativity, limiting true ownership, and blocking the seamless, cohesive metaverse users expect.

Web3 steps in with blockchains, offering decentralized ownership through NFTs and transparent ledgers. But scalability remains a hurdle. Storing metaverse-scale data – 3D models, metadata, game states, and more – on-chain demands full replication across every node, a costly and impractical burden as the digital universe grows. Ethereum, for example, shines as a transaction layer but struggles to host a character's render files efficiently. The result? Developers and creators face soaring costs and complexity, unable to make assets easily discoverable, permissioned, or functional across diverse platforms and experiences.

Without a shared, scalable data layer that behaves like an API, the metaverse stays a patchwork of disconnected worlds – walled gardens with a blockchain sheen. Assets – and even agents like AI-driven NPCs – can't reach their full potential, held back by inaccessible, isolated data. Sylo tackles this head-on, blending Web2's efficient storage with Web3's portability and ownership to create an ownable, shared data layer. It's about more than transactions; it's about enabling a metaverse where assets flow freely and agents benefit too, all under the control of their rightful owners, no matter who's building, playing, or interacting.

3

The Solution: Sylo Network

The Sylo Network redefines the metaverse by delivering a federated, scalable data layer that powers true interoperability for assets and agents alike. Unlike blockchains constrained by on-chain storage limits, Sylo bridges Web2's efficient cloud infrastructure and scalability with Web3's ownership and portability, creating a shared data layer accessible by experiences and agents alike. Sylo operates as a LayerZero-like service but for any and all data, providing the standards and protocols for the entire digital ecosystem, not just one chain. It's the backbone of the open metaverse: a shared data layer on which developers, creators, and IP holders can build games, assets, agents, and experiences that work together, now and into the future.

At its core, the Sylo Network comprises four interlocking components:

Sylos: Storage nodes that anyone can run, empowering data owners – whether creators or individual consumers – to choose where their data resides. A Sylo is lightweight software layered over common cloud solutions (e.g., AWS S3), enabling owners to host and serve content to the open metaverse. Flexible enough to let data owners self-host their own data or entrust it to a third-party Sylo, collectively they ensure autonomy over data associated with assets, agent behavioral scripts, interaction histories, and more. Decentralized yet accessible, Sylos form the foundation that keeps data portable and functional across a myriad of games and experiences.

Sylo Protocols: The rules and tools that ensure data can be interoperable across the metaverse. These handle discoverability (locating data), permissions (who can access or modify what data), and usage pricing (what access costs). The Sylo Protocols remove the need for prior coordination or setup between creators, agents, and experiences, enabling seamless integration and functionality without central oversight; the open metaverse in action.

Sylo Spaces: Modular web experiences for brands and projects. A Space blends plugins like social feeds, chat, asset pages, and wallets, pulling data directly from Sylos to create web3 spaces. Web presences that sync with in-game ecosystems and agent-driven interactions – all customizable for games and brands alike.

Sylo Token (\$SYLO): The utility token fueling the system. \$SYLO powers protocol interactions as gas, facilitates data access payments for both assets and agents, and incentivizes contributors through staking, tying the network's economy together. Whether registering a Sylo, verifying data, or querying state, \$SYLO ensures the system remains fluid and incentivized.

Together, these tools establish the building blocks for a truly open metaverse, where a shared, API-like data layer powers seamless experiences across web and 3D engines. By breaking down fragmented ecosystems, Sylo puts assets front and center – portable, interoperable, and fully owned – while letting agents tap into the same rich data to support gameplay, commerce, and more, all under the control of their creators and users.

4

Key Features & Use Cases

Sylo redefines ownership in the open metaverse by tying NFTs to rich, portable data stored in Sylos, enabling assets and any associated data to thrive across ecosystems. Ownership now extends beyond the on-chain NFT itself – it encompasses the full scope of an asset’s or agent’s existence, from 3D render files and behavioural logic to interaction histories and learned skills, all usable anywhere in the metaverse. Here’s the kind of data that could be stored in Sylos:

- **Asset Metadata:** Names, descriptions, ownership records – core details for any NFT.
- **3D Files & Render Instructions:** UBF Blueprints ensuring assets display correctly across engines.
- **Murmur Matrix:** An asset’s or agent’s acquired context – its “memories” or history. As a character or agent moves through experiences, games can append data (e.g., battles fought, skills earned), which become a part of the NFT itself.
- **Real-Time Data:** Dynamic info like pricing or availability.
- **Agent-Specific Data:** Behavioral scripts, learned datasets, or states, sharable across platforms or between agents.

By storing this data in Sylos, assets and agents gain true interoperability, freed from fragmented ecosystems. Ownership becomes control over living, evolving datasets that carry their utility – be it a character’s gear or an agent’s skills – wherever they go, breaking the walls that once confined them. This dynamic system ensures creators, agents, and users alike can operate seamlessly across the metaverse, their digital creations evolving naturally within a connected, decentralized framework.

Use Case 1: Interoperable Characters

A creator designs a character NFT for the open metaverse and uploads its data – 3D model, metadata, and usage rules – to a Sylo. Once registered on the Sylo Network, this character becomes:

- **Discoverable:** Any game or experience can find and integrate it.
- **Usable:** It renders perfectly in Unreal, Unity, or beyond in real-time using the UBF Standard.
- **Editable:** Experiences and users can equip it with another asset (e.g., a sword).
- **Persistent:** That sword stays equipped as the character moves to another experience, embedded at the data layer – not just within a single game. This continuity, impossible in siloed Web2 ecosystems, creates a cohesive metaverse where an asset’s relevance persists beyond the boundaries of a specific game.

This use case showcases how Sylo enables assets to transcend individual platforms, delivering a seamless, user-owned experience that mirrors the boundless vision of the open metaverse.

Use Case 2: User Profiles

The Sylo Network powers user-owned profiles, bundling a user’s identity – preferences, achievements, histories, and social graph – into a portable dataset they control. Unlike platform-bound profiles, these Sylo-powered User Profiles travel across any game or experience, breaking free from single-game boundaries.

Stored in a Sylo, data like avatar details, usernames, and social graphs (e.g., friends or guilds) can be self-hosted or kept in a managed Sylo, all governed by Sylo Protocols:

- **Portable:** Games access and render the profile and social graph seamlessly across engines like Unreal or Unity.
- **Ownable:** Users set permissions (e.g., “share social data with co-op games”) via the Attestation Protocol, recorded with \$SYLO on The Root Network.
- **Interoperable:** Standardized formats ensure data – like titles or friend lists – works anywhere.

Developers use these profiles for personalization, paying \$SYLO via Sylo Tickets to hosts, while users carry a persistent identity and social ties across the metaverse. User Profiles make the open metaverse user-centric, driven by Sylo's decentralized framework.

Use Case 3: Enabling Agents with Data Access

The Sylo Network supercharges agents – AI-driven entities like companions, guides, or merchants – by granting them access to rich data stored in Sylos, whether permissioned by owners or openly available. This access can unlock smarter, more dynamic functionality across the open metaverse, leveraging datasets like the Murmur Matrix, environmental details, or user histories.

Imagine an AI companion granted permission to a user's Sylo-stored adventure log (e.g., "survived the Lava Pits"). It uses this to tailor dialogue or suggest quests, growing more personalized with each interaction. Alternatively, a merchant agent taps open real-time pricing data from a Sylo to adjust offers on the fly, no developer coordination needed. The Sylo Network ensures this is seamless:

- **Permissioned Access:** Owners define who can use their data (e.g., "companions only read my history"), paid with \$SYLO, keeping control in their hands.
- **Open Access:** Public datasets (e.g., map layouts) could let any agent – like a guide plotting paths – plug in instantly and tailor quests across different experiences.

Developers and agents acquire access via \$SYLO payments or staking, rewarding data hosts while enabling agents to adapt, reason, and serve users across games. This turns agents into responsive, data-driven allies, amplifying their roles in storytelling, commerce, and exploration.

Sylo's power lies in its simplicity: any developer or creator, on any blockchain or engine, can plug into this layer by adhering to its standards. From a Ready Player One-style avatar to an AI companion driving decentralized commerce, Sylo makes the metaverse a connected, user-owned world.

5

The Sylo Protocols: Their Role

The Sylo Network's interoperability hinges on three core protocols – Permissions, Verification, and The Asset Register – working together to manage data across the Open Metaverse. These protocols enable creators and developers to safely access and use interoperable data, delivering seamless experiences to end-users without exposing the underlying mechanics. Here's how they function and why they're essential.

Permissions Protocol: Defining Access Rules

This protocol governs who can use specific data and where.

Manages Access: Creators set granular permissions when uploading an asset and its associated data – whitelisting specific games (e.g., “only in The Third Kingdom”), blacklisting others, or using broad categories (e.g., “PG-rated experiences”). These rules are recorded on The Root Network via a \$SYLO transaction, and Sylos enforce them, ensuring data is served only to approved applications. This gives IP holders (like brands) precise control everywhere in the metaverse while keeping integration straightforward for developers.

Verification Protocol: Ensuring Trust

This protocol guarantees data authenticity, recommended when creators use third-party Sylos.

Ensures Trust: Upon upload, an asset's data gets an on-chain hash – a unique fingerprint – stored on The Root Network. When the API returns data, developers can trigger an optional, automated verification process to confirm the served data matches the original, eliminating tampering risks. While not needed when creators host their own Sylos, it's a vital layer for premium IP or high-stakes assets and agents, ensuring the data being accessed can be trusted.

The Asset Register: Tracking and Portability

This protocol manages the rules around what assets can be equipped to other assets, as well as what assets are currently equipped.

Tracks Connections: The Asset Register logs equipped items – like a character's sword – or agent states – like an AI's active skills. Updates (e.g., swapping items or updating skills) are recorded instantly. This ensures no detail is lost as assets or agents move.

Asset Rules: It also dictates which items an asset can pair with – like a character can wield a specific sword but not also wield a shield. This is important data that needs to be commonly understood so the asset always conforms to its creator's specifications. Creators define allowable combinations, and the Asset Register validates link requests. Approved links are finalized with a \$SYLO transaction, maintaining consistency across the metaverse.

Together, these protocols make Sylo a plug-and-play layer. Creators configure rules and enhancements, developers integrate assets and agents, and \$SYLO gas keeps it all working – all without burdening end-users with the mechanics.

6

Sylo Token & Tokenomics

The Sylo Token (\$SYLO) serves as the utility token of the Sylo Network, enabling the decentralized data layer for the open metaverse. \$SYLO facilitates interactions, rewards contributors, and ensures interoperability among creators, developers, hosts, and network supporters. This section outlines its current functionality and its intended evolution as the ecosystem scales.

Functionality

\$SYLO operates through two primary mechanisms: transactional spending and staking for network support.

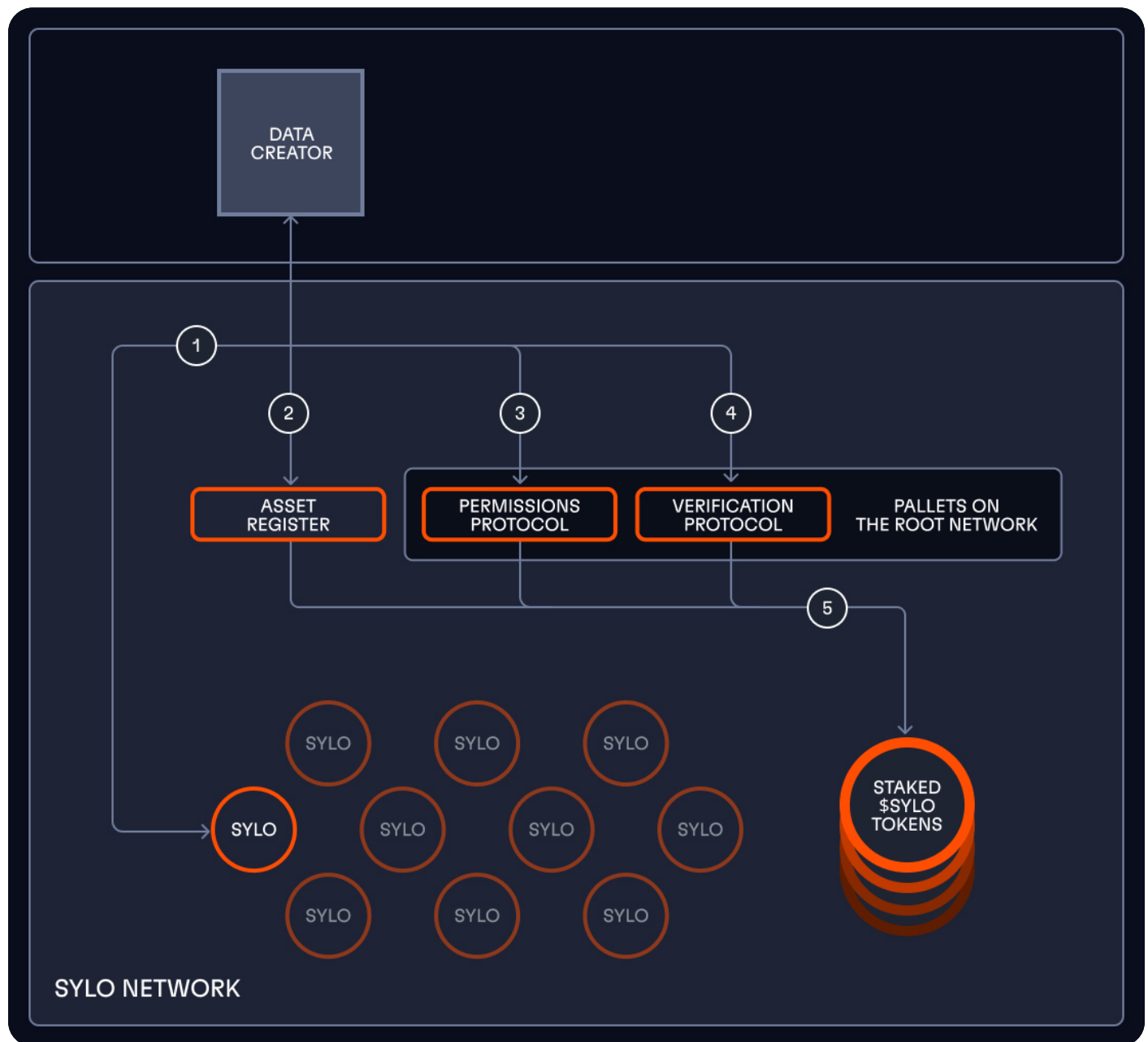
Spending \$SYLO:

- Games and Applications utilize \$SYLO to access (and pay for) licensed data stored in Sylos – such as 3D models associated with IP. This kind of access is paid for utilizing a novel off-chain micropayments mechanism we've coined Tickets.
- Users, Games and Applications transact against the Asset Register in \$SYLO to equip and unequip assets. These fees flow to Stakers on the network.
- Creators spend \$SYLO to leverage the Permissions Protocol to define access permissions for their data. These transactions occur on The Root Network, in \$SYLO, with a percentage allocated to Stakers on the network.
- Creators spend \$SYLO to leverage the Verification Protocol, providing an additional layer of trust. These transactions occur on The Root Network, in \$SYLO, with a percentage allocated to Stakers on the network.
- Sylo Hosts pay a one-time \$SYLO fee to register their Sylo node. These transactions occur on The Root Network, in \$SYLO, with a percentage allocated to Stakers on the network.

Staking \$SYLO:

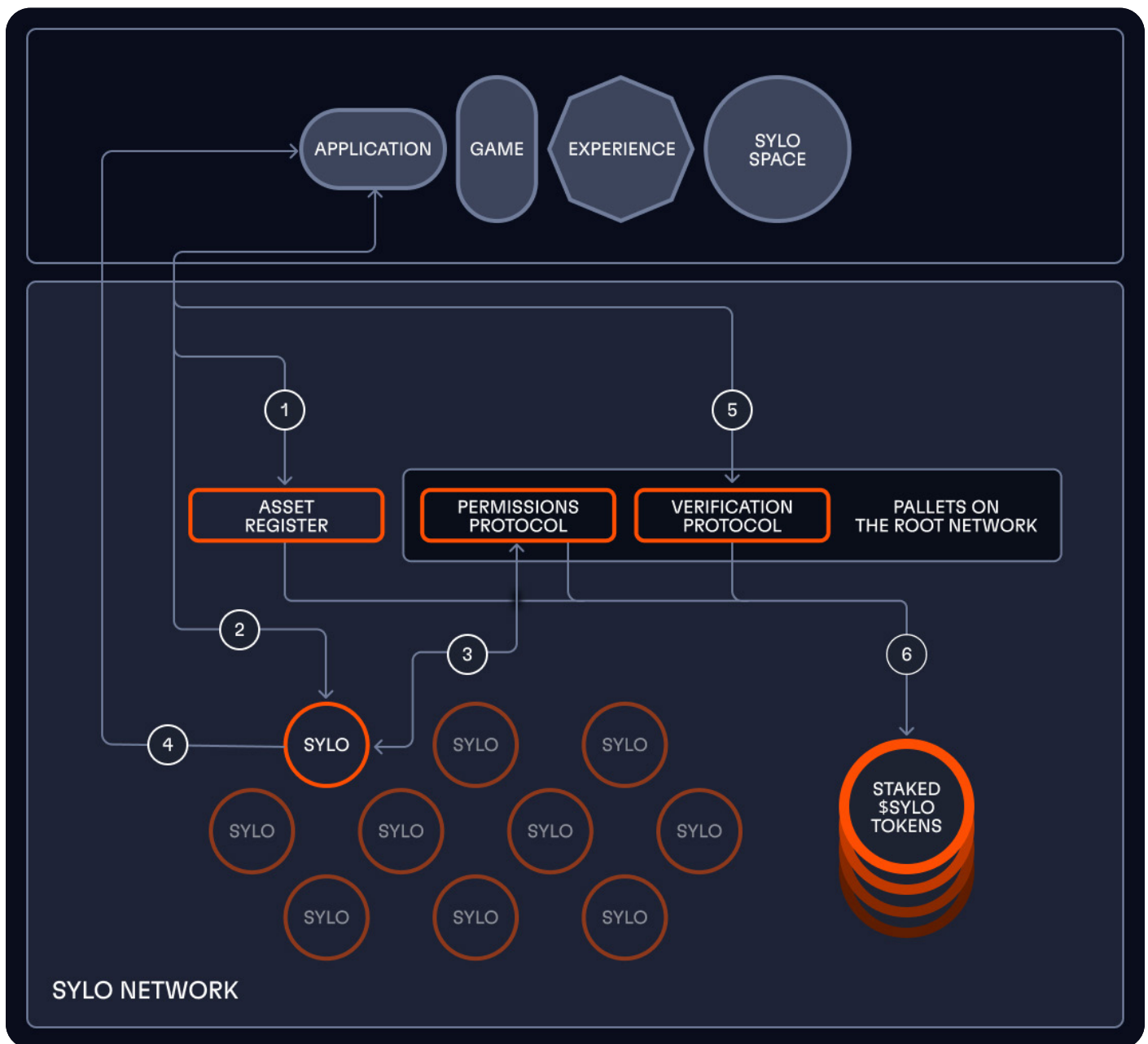
- Network Supporters stake \$SYLO to enhance system stability. They receive rewards for use of the Protocols as explained above.

Adding Data to the Network:



1. A Data Creator either sets up a Sylo for their Data, or chooses a Sylo they would like to use.
2. The Asset Profile is stored on the Asset Register, holding the location of specific files relating to the Asset.
3. Any permissions for the data are registered using the Permissions Protocol and enforced by the Sylos.
4. The Data Creator can utilise the Verification Protocol so that users can verify that the data they are served is the exact data that the Creator uploaded.
5. Fees flow to stakers based on Sylo Token staked.

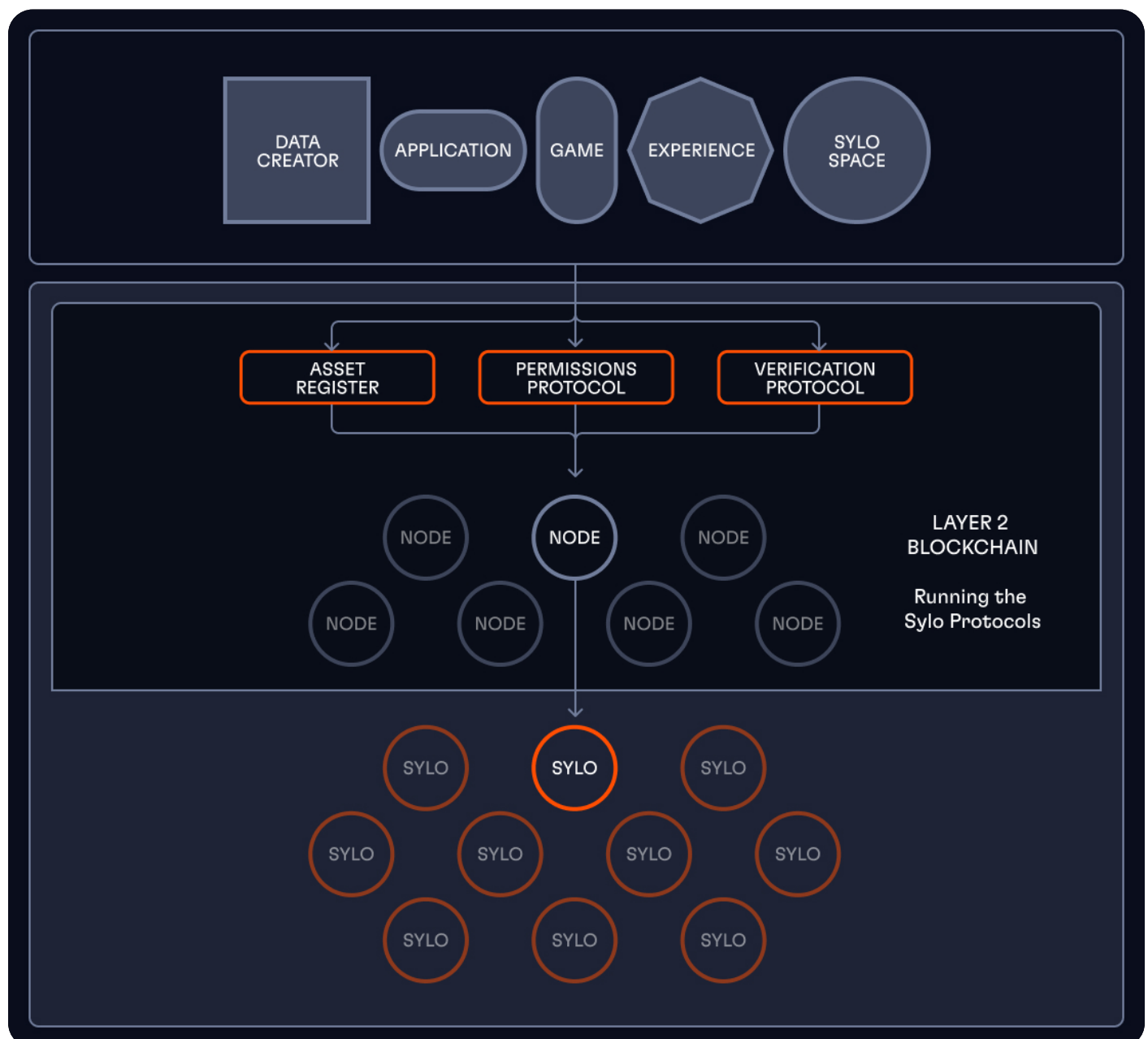
Fetching Data:



1. An application makes a call to the Asset Register to fetch relevant data for an asset in a users wallet. The AR returns the location of all relevant data for the asset, and any equipped assets.
2. The Application then calls the required Sylo directly.
3. For each request, the Sylo leans on the Permissions Protocol to ensure the request adheres to the permissions over that data.
4. The Sylo returns the data in exchange for payment (if required).
5. The application (or user) can check that the data returned by the Sylo is correct using the Verification Protocol.
6. Fees flow to stakers based on Sylo Token staked.

Future Evolution: As Sylo scales

In its next phase, the Asset Register itself will become a blockchain; a Layer 2 of The Root Network run by Seeker Nodes. As scale dictates, the Sylo Protocols will also migrate to the same Layer 2 blockchain. Under this architecture, the functionality of the network will not change, nor will the underlying tokenomics, but the network will become more efficient. The \$SYLO staked currently will become the network Stake of the Layer 2. More details on this will be released when the network grows to require this evolution.



7

The Roadmap

The Sylo Network is on a path to redefine the open metaverse, with key milestones designed to enhance its decentralized data layer, empower contributors, and scale its infrastructure. Below are the upcoming steps and their significance in realizing this vision.

Pricing Sylo:

A dedicated Sylo providing up-to-date pricing for tokens on The Root Network. This enables real-time market data access for dApps, experiences and agents.

Status: Released

Data Verification Protocol:

A protocol ensuring the authenticity of data stored in Sylos. By generating on-chain hashes verifiable by anyone, this builds trust in the network, critical for high-value assets and agents operating across ecosystems.

Status: Released

Profile Sylo:

A Sylo housing user profile data. This creates a portable, user-owned profile layer, enhancing personalization and social features in metaverse experiences. Data currently included in the Profiles is usernames and avatars. This will be extended over time to include preferences and achievements.

Status: Released

Sylo Graph:

Release users interoperable social graph, enabling users to bring their social circles with them as they move across the open metaverse.

Status: Launching Q2/Q3 2025

Incentivize the Asset Register:

Introduce \$SYLO fees for updating the Asset Register, which tracks asset and agent connections, incentivizing contributors to maintain its accuracy.

Status: Launching Q2/Q3 2025

Data Permissions Protocol:

Defining access rules for a Sylo's data. Built on the ideas from our doughnuts patent, Creators set permissions recorded on The Root Network, ensuring control and interoperability without centralized gatekeepers.

Status: Launching Q2/Q3 2025

Transition Asset Register to Layer 2 architecture:

This improves decentralization and further aligns the network to the Staking model, aligning with the networks decentralization goals.

Status: Launching when scale requires

Port Protocols to the Layer 2 network:

Move the Verification, Permissions, and Asset Register protocols to a Layer 2 blockchain powered by a community of Nodes (Seekers). This enhances scalability and efficiency, with Seekers validating transactions and earning \$SYLO fees.

Status: Launching when scale requires

The Seeker Network (Open Beta):

Enabling any community member to run a Seeker as a fully functional Node.

Status: Launching when scale requires

Launch Network Governance:

Establish a governance model for the Sylo Network, enabling \$SYLO holders to vote on fee structures, protocol upgrades, and more. This final step empowers the community to shape the network's future, ensuring it remains decentralized and adaptive.

Status: Launching 2025

These milestones mark the Sylo Network's journey toward a fully decentralized, scalable, and community-driven metaverse backbone, with \$SYLO at its core.

