Hive Protocol

Decentralised Infrastructure for a User-Sovereign Internet

Hive Protocol Contributors

v0.1 — March 27, 2026

Abstract

Hive is a decentralised peer-to-peer content sharing network where any node can store, discover, and serve content with no central authority. Built on Ethereum Swarm for persistent content discovery and Pear/Holepunch for NAT-traversing peer-to-peer transport, Hive bridges existing decentralised networks under a unified protocol. Every fetch creates a new provider, making the network self-healing and increasingly resilient with use. The protocol is extensible by design, economics-ready, and ships today as an open-source reference implementation.

Contents

1 The Problem

The internet was designed as a decentralized, peer-to-peer network where no single entity controlled the flow of information. Over time, it has been progressively captured by a small number of corporate data silos — cloud providers, social platforms, search engines — that:

- **Control access** to users' own data
- **Monetise** user data without consent or compensation
- **Create single points of failure** and censorship
- **Lock content** into walled gardens and proprietary formats
- **Concentrate power** in ways the internet's architects never intended

Existing decentralized networks like IPFS and Swarm represent steps in the right direction, but in practice they largely rely on gateway intermediaries for reliability, scalability, and performance. They don't interoperate, and fragment the decentralized ecosystem rather than unifying it.

The original vision of the internet — an open, peer-to-peer network of equals — has been betrayed by the platform era. Hive exists to reclaim it.

2 The Vision

Hive is a return to first principles. It restores the original promise of the internet by building infrastructure where:

1. **User control is default** — Users own, control, and can move their data freely. No platform holds it hostage.
2. **Decentralization is a first principle, not an afterthought** — No central authority, index, or gateway is required. The directory lives on a blockchain; the network is pure P2P.
3. **The network strengthens with use** — Every fetch creates a new provider. Content naturally flows to where demand exists, making the network more resilient the more it is used.
4. **Protocol-agnostic openness** — Hive bridges existing decentralized networks (IPFS, Swarm, and beyond) rather than creating yet another silo.
5. **A decentralised data bridge** — Through caching and proxying, Hive enables seamless data migration and offloading from centralised services to decentralised networks, where users have real control and sovereignty over their data.
6. **Privacy by design** — Node identities are never exposed publicly. Only cryptographic hashes appear on the network.

Hive is spiritually aligned with the early internet ethos — the world of personal homepages, open protocols, and the end-to-end principle. Before the platform era captured and centralised the web, the internet was a network of peers. Hive builds the infrastructure to make that vision viable again at modern scale.

> *"Our identities have no bodies, so, unlike you, we cannot obtain order by physical coercion. We believe that from ethics, enlightened self-interest, and the commonweal, our governance*

will emerge. Our identities may be distributed across many of your jurisdictions. The only law that all our constituent cultures would generally recognize is the Golden Rule. We hope we will be able to build our particular solutions on that basis. But we cannot accept the solutions you are attempting to impose."

— John Perry Barlow, *A Declaration of the Independence of Cyberspace*

3 How It Works

3.1 Swarm + Pear: Complementary Halves

Hive's core protocol is built on two complementary ecosystems that together form a complete decentralized content-sharing system.

Layer	Technology	Role
Discovery	Ethereum Swarm (Feeds / SOC)	The "Yellow Pages" — persistent, blockchain-backed registry mapping content hashes to provider peer hashes. Answers "where is it?"
Transport + Storage	Pear / Holepunch	The "plumbing" — NAT-traversing P2P connections, encrypted streams, content-addressable local storage. Answers "connect and get it."
Provider Bridge	IPFS, BZZ (extensible)	On-ramps — bridge content from existing decentralized networks into the Hive network via a three-tier cache-aside proxy.

Table 1: Hive's three-layer architecture.

Why both are needed.

What Swarm brings
: Persistent, globally-readable content-to-provider mappings that survive node restarts. Hyperswarm's DHT discovers peers on a topic but has no concept of "who has what content." Swarm Feeds solve this with a decentralized, append-only database of content-to-provider mappings. Swarm also enables decentralized payments and economic mechanisms (e.g. postage stamps) — economic mechanisms for storage and bandwidth that Pear's transport layer does not natively address.

What Pear brings
: Direct, real-time peer-to-peer connections with NAT traversal, efficient binary content transfer via RPC, and a local content-addressable filesystem. Swarm Feeds store small payloads (peer hashes), not large content.

3.2 The Core Loop

The handoff between layers is clean and creates a viral propagation effect. Figure 1 illustrates the store–announce–fetch–re-announce cycle.

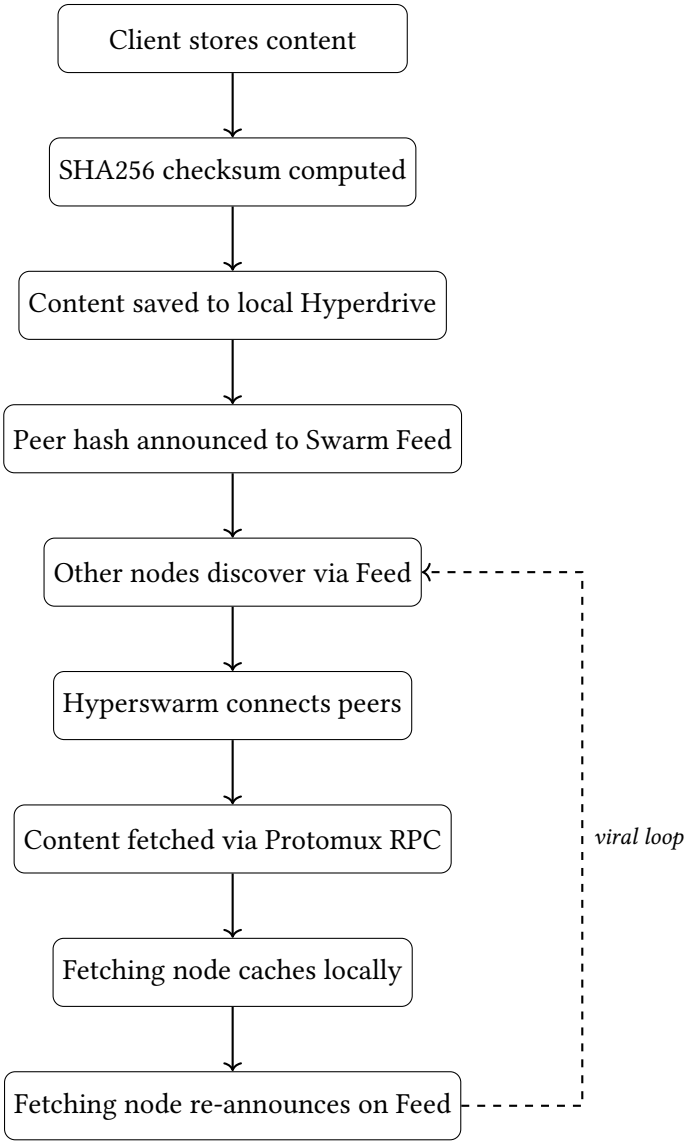


Figure 1: The core store–announce–fetch–re-announce loop. The dashed arrow represents viral propagation: every fetch creates a new provider.

Viral propagation: Every fetch creates a new provider. The more a piece of content is requested, the more nodes serve it — making the network self-healing and increasingly resilient.

3.3 Provider Protocols: IPFS, BZZ, and Beyond

Provider protocols are **on-ramps** — they bridge content from existing decentralized networks into the Hive network via a three-tier cache-aside proxy. Figure 2 shows the resolution flow.

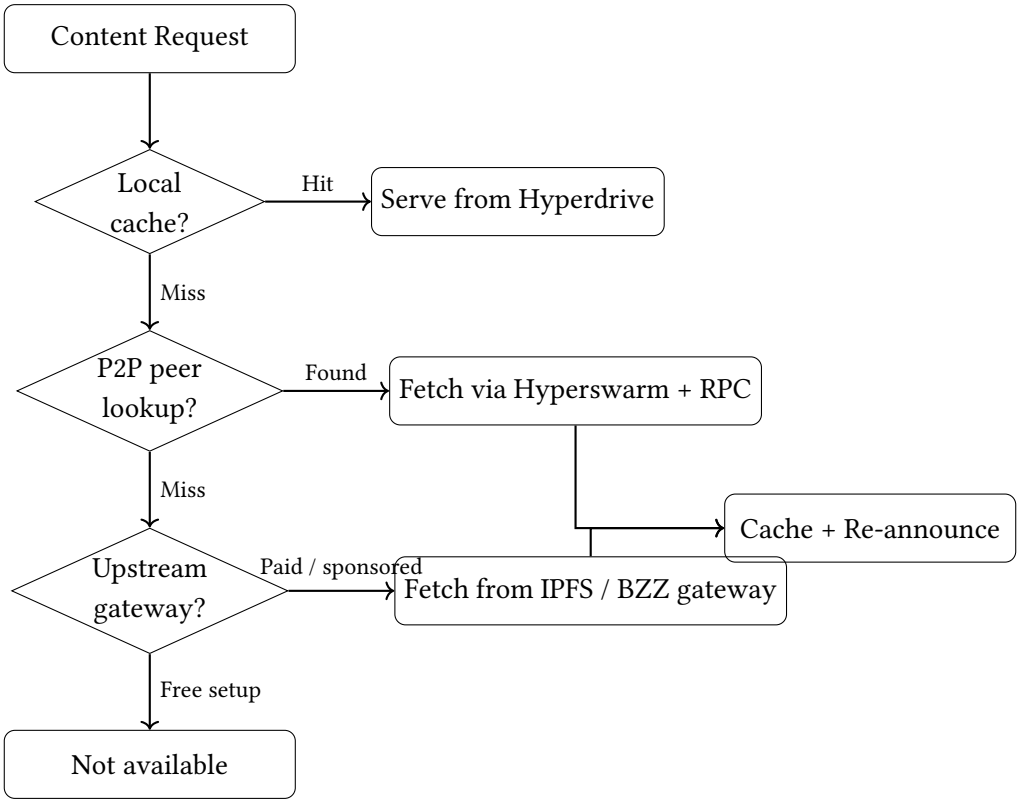


Figure 2: Three-tier cache-aside proxy resolution. Content is resolved locally first, then via P2P peers, and finally through an upstream gateway as a last resort.

1. **Local cache** (Hyperdrive) — free, instant.
2. **P2P peer lookup** (Swarm Feed registry → Hyperswarm fetch) — free, uses the DHT.
3. **Upstream gateway** (IPFS / BZZ gateway) — paid / sponsored setup.

Once content enters Hive from any provider, it is stored locally, announced on the registry, and served peer-to-peer. The upstream gateway is never needed again for that content. Each proxy fetch is a **one-time import** that permanently enriches the network.

Swarm plays a **dual role**: it is both core infrastructure (Feeds as the discovery registry for all content) and a provider protocol (BZZ content can be proxied into Hive like IPFS content). This is natural — Swarm's content layer and its feed layer serve architecturally distinct functions.

The provider system is **extensible by design** — new providers can be added with minimal effort, making Hive open to any decentralised storage network.

4 Key Protocol Properties

Content-addressable
: SHA256 as the universal content identifier, enabling efficient storage and automatic deduplication across the network.

Multi-provider discovery
: Sequential feed indexing finds all providers for any content hash.

Privacy-preserving addressing
: Peer hashes (SHA256 of public keys), never raw identities.

Protocol-agnostic bridging
: IPFS, Swarm, and beyond via extensible provider system.

Zero-cost P2P tier
: Free peer-to-peer content sharing with optional paid gateway fallback.

Deterministic identity
: Shared seed derives node identity via secp256k1, enabling permissionless participation.

5 Network & Economics

5.1 Permissionless Participation

Any node with a shared seed can join the network. There is no registration, no approval process, no gatekeeping. The network's activity feed serves as a global heartbeat — new content announcements and provider registrations flow across the network in real time.

When a node fetches content, it races multiple peers simultaneously for reliability and speed. The first responder wins; the content is cached locally and re-announced, adding redundancy to the network.

5.2 Configurable Cost Architecture

Aspect	Cost	Mechanism	Purpose
Local cache	Free	Hyperdrive	Repeat access, instant retrieval
P2P fetch	Free	Hyperswarm DHT + Protomux RPC	Direct peer-to-peer content transfer
Peer discovery	Cheap / Sponsored	Swarm Feeds	Discover which peers hold specific content
Gateway proxy	Paid / Sponsored	IPFS/BZZ HTTP gateways or self-hosted nodes	First import of external content
Full Swarm retrieval	Paid	Native Swarm fetch with postage stamps	Best performance, bandwidth, and retrieval speeds

Table 2: Configurable cost aspects. Operators mix and match to create custom tiers.

These aspects can be mixed and matched by node operators to create custom free and paid tiers tailored to their use case. A community node might offer everything free with sponsored discovery, while a commercial operator could bundle gateway and Swarm retrieval into a premium tier. Once content enters the Hive network through any path, it is served free from local cache and P2P thereafter.

5.3 Economics-Ready Design

Hive is **economics-ready infrastructure** — designed so that decentralized incentive mechanisms can be layered on top without requiring protocol changes:

- Micro-payment rails for bandwidth and storage contributions
- Reputation and peer scoring for quality of service
- Intersection with Swarm's existing postage stamp model
- Community governance for protocol evolution

6 The Hive Node

The reference Hive node is a fully functional, open-source implementation shipping today:

- **NestJS reference node** — Production-ready TypeScript implementation with Fastify
- **REST API** — 15+ endpoints for content CRUD, discovery, feeds, and proxy operations
- **Transparent gateway proxy** — Drop-in replacement for Bee and IPFS API consumers
- **IPFS + BZZ bridging** — Content flows freely between protocols through the proxy layer
- **Peer-to-peer content sharing** — NAT-traversing connections via Hyperswarm
- **Blockchain-backed registry** — Ethereum Swarm Feeds for persistent content discovery

6.1 Use Cases

Decentralized CDN
: Community-operated content caching network that strengthens with use.

Censorship-resistant publishing
: Content that cannot be taken down by any single actor.

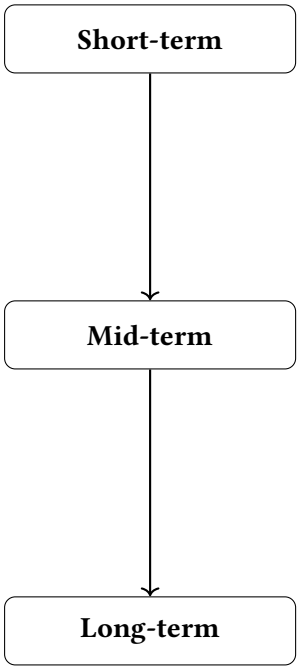
Cross-protocol aggregation
: Unified access to IPFS, Swarm, and future networks.

Community data commons
: Shared datasets maintained by their communities.

Resilient archival
: Content that survives individual node failures through viral replication.

7 Roadmap

Figure 3 summarises the development trajectory.



Core / Node

— Open-source reference node

— IPFS and BZZ proxy providers

— REST API with full content lifecycle

— Swarm Feed-based content registry

— Peer-to-peer content sharing via Hyperswarm

UX / Utilities

— Additional proxy providers

— Improved peer scoring and reputation

— Content pinning and persistence guarantees

— Web dashboard for Hive node

DX / Ecosystem

— Decentralised identity, accounts, and sub-accounts

— Decentralised databases and data sharing primitives

— Integrated payments and micro-transactions

— Hive app store and marketplace

— Protocol standardization and cross-chain discovery

Figure 3: Development roadmap.

Hive is infrastructure for a user-sovereign internet.
The more nodes that join, the stronger the network becomes.
Every fetch is an act of decentralization.