Overcoming the Visibility Crisis in Web3: Designing a Search Engine for the Decentralized Web

Artyom Harutyunyan
Blockstars LLC
Yerevan, Armenia
e-mail: artyomharutyunyans@gmail.com

***Abstract*—The rapid growth of decentralized web technologies, such as IPFS, ENS, and Arweave, has enabled the creation and hosting of censorship-resistant, open-access websites. However, these systems suffer from a fundamental usability problem: decentralized websites are effectively invisible to the average user due to the absence of an indexing and discovery infrastructure. This paper introduces Web3 Compass, a search engine purpose-built for the decentralized internet. Unlike traditional search engines that rely on centralized servers and behavioral tracking, Web3 Compass discovers and indexes content from decentralized domains through real-time blockchain monitoring, resolver contract interactions, and a custom IPFS infrastructure. It outlines the visibility problem, examines failed or insufficient past solutions, and presents the architectural design of a hybrid, privacy-preserving search tool optimized for the decentralized web. The contribution aims to address the core bottleneck in Web3 usability by making decentralized content discoverable and accessible.**

***Keywords*—Decentralized Search, Web3, IPFS, ENS, Content Discovery, Blockchain Indexing, Pinning, Distributed Hash Table.**

I. Introduction

The decentralized web (Web3) promises to transform how users interact with content online — offering resistance to censorship, independence from centralized hosting, and enhanced privacy. Technologies like the InterPlanetary File System (IPFS), Ethereum Name Service (ENS), and Unstoppable Domains (UNS) allow websites to be hosted across peer-to-peer networks and referenced via blockchain-based domain names. Yet despite these advancements, Web3 remains practically invisible to mainstream users. This is not due to technical immaturity, but to a structural gap: the absence of a functioning search infrastructure. Without the ability to index and retrieve decentralized websites, even high-quality content remains undiscovered and unused.

This paper presents Web3 Compass, a dedicated search engine that addresses this core problem. By integrating blockchain event monitoring, resolver contract querying, decentralized content fetching and a lightweight indexing engine, Web3 Compass creates a functional layer of visibility for decentralized sites.

II. Related Work and Literature Review

While the potential of decentralized web hosting is widely acknowledged, efforts to make this ecosystem usable and searchable have been sparse and largely ineffective. Existing initiatives fall into two categories:

Failed or Incomplete Search Engines: Projects like Presearch and Aleph.im were early attempts to build decentralized or Web3-aligned search tools. Presearch is a federated search engine with token incentives, but it does not index decentralized content directly, nor does it resolve domain names or parse content from systems like IPFS or ENS. Aleph.im focuses on decentralized storage and indexing of on-chain data but does not operate as a general-purpose web search engine. These tools either rely on traditional Web2 sources or are too limited in scope to function as practical discovery layers for Web3. They do not interact with resolver smart contracts or decentralized peer-to-peer content networks.

Centralized Gateways and Name Services: Many decentralized websites are only accessible through centralized gateways like ipfs.io or .limo. These services act as HTTP proxies for IPFS content but are centrally hosted, which makes them vulnerable to takedowns and content filtering. For example, ipfs.io has a known policy for rate-limiting and blocking content under certain conditions. ENS, UNS, and BNB Name Service offers blockchain-based domain registration, but they do not provide mechanisms for discovering domains or crawling their content. Limitations of Traditional Web Search Engines Conventional search engines like Google and Bing do not crawl or index decentralized content by default. Without a traditional HTTP-accessible URL and supporting metadata (e.g., sitemaps), IPFS-hosted content remains invisible to these engines. While content may become searchable when served via public gateways, this reintroduces the same problems of centralization and censorship.

Name and Content Resolution: ENS and IPFS: The Ethereum Name Service (ENS) maps human-readable names to blockchain addresses and content hashes (e.g., IPFS CIDs) using resolver smart contracts [1]. However, ENS lacks a built-in search interface; users must already know the domain name. IPFS uses a distributed hash table (DHT) to allow nodes to advertise which content they host. While this enables peer-to-peer retrieval by CID, it does not support keyword-based content search or semantic discovery. Academic studies confirm that IPFS content availability depends on nodes actively pinning or requesting the data. Without sufficient

https://doi.org/10.51408/csit2025_80

replication, unpopular or stale content may vanish from the network [2], [1].

In short, existing naming systems (ENS), storage layers (IPFS), and search tools (Presearch, Aleph) address different parts of the Web3 stack, but none offer a comprehensive, usable, and censorship-resistant search experience. This fragmentation and the lack of fully decentralised, end-to-end content discovery solutions are well-documented in recent academic surveys, which conclude that while decentralised file systems and name registries have matured, decentralised search remains an open challenge, with no complete systems that achieve true decentralisation, usability, and performance comparable to centralised solutions [1]. Web3 Compass addresses this by bridging the gap between decentralized naming and content networks, and creating a usable, indexed discovery layer.

III. System Overview: WEB3 Compass Architecture

The Web3 Compass system is designed to detect, resolve, retrieve, and index decentralized websites registered on blockchain-based naming services and stored on peer-to-peer content networks such as IPFS. The goal of the system is to provide usable search and access functionality for decentralized websites, without relying on centralized crawling infrastructure, SEO practices, or user tracking. The architecture is composed of four interdependent layers:

Domain Discovery Layer: This layer continuously monitors decentralized naming systems to detect new domain registrations. Web3 Compass currently supports the following:

- ENS (Ethereum Name Service)
- UNS (Unstoppable Domains)
- BNB Name Service (Space ID)
- CNS, Tomi Domains, SID

Each naming system is monitored through its corresponding smart contract registry. Web3 Compass runs event listeners to capture events such as domain creation and updates. When a domain is detected, the system stores metadata such as the domain name, block timestamp, and ownership data. This process uses tools such as the ENS subgraph and Alchemy API to extract on-chain events in real-time.

Domain Resolution Layer: After a domain is discovered, Web3 Compass queries the domain's resolver contract to extract the associated content hash — the address pointing to the actual website content stored in a decentralized network (e.g., IPFS). Resolvers may be:

- Public (default) — widely adopted and predictable in behavior
- Custom — require interface compatibility (e.g., support for contenthash())

The content hash (CID) can reference storage networks like IPFS, Swarm, or Skynet. Currently, Web3 Compass supports IPFS-based sites only.

Content Retrieval Layer: Once the content hash (CID) is resolved, the system attempts to fetch the corresponding site from the IPFS network. To ensure availability and reliability, Web3 Compass:

- Hosts its own IPFS node
- Implements a pinning policy for important content (files under 100MB)
- Falls back to public gateways (e.g., ipfs.io) if necessary

To avoid censorship and improve load times, the system also provides its own access layer through a custom gateway: dweb3.wtf, replacing centralized options like .limo, .link, and ipfs.io.

Content Filtering and Indexing Layer: Not all content retrieved from IPFS is suitable for indexing. Web3 Compass filters and indexes only relevant formats, specifically:

- HTML files
- Text-based pages
- Client-side rendered Single Page Applications (SPA)

Files such as images, videos, PDFs, and archives are excluded from indexing. To handle SPAs, the system uses a headless browser to render and scrape the final HTML after JavaScript execution. This ensures that pages that generate their content client-side are fully captured and indexed. The filtered content is indexed using Meilisearch, a lightweight and fast full-text search engine that supports custom ranking logic. The full system flow, from domain discovery to content indexing and access, is illustrated in Figure 1.

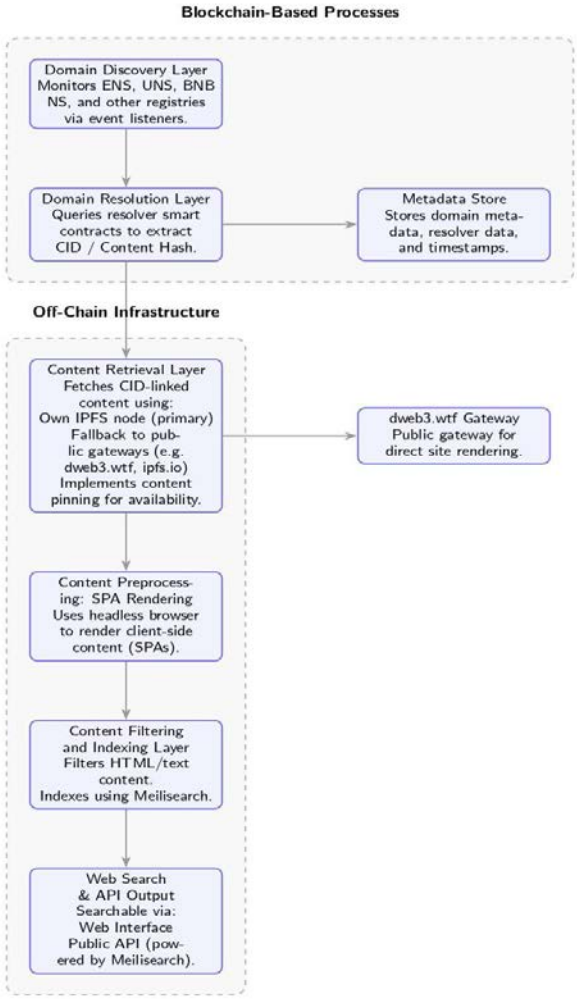

Figure 1. Web3 Compass system architecture

The pipeline consists of four main stages: domain discovery via on-chain registry monitoring, resolver-based CID resolution, IPFS-based content retrieval (with fallback to dweb3.wtf), and client-side rendering with content filtering and indexing. The final output is exposed through a web interface and a public API.

IV. Methodology and Implementation

The implementation of Web3Compass follows a modular architecture, where each layer of the system, from domain discovery to search indexing, operates as a distinct service. The system is built to handle the decentralized nature of content and naming systems without introducing centralized behavioral analytics or storage of user data.

A. Monitoring and Discovery of Domains

To detect new decentralized websites, the system continuously listens to events emitted by the smart contracts of supported naming services. This is done via a dedicated monitoring service for each domain system (e.g., ENS, UNS, BNB Name Service), which uses tools such as:

- Alchemy API for Ethereum network access
- ENS subgraph for structured query access to ENS records

Whenever a new domain is registered or updated, the system logs:

- The domain name
- The resolver address
- Timestamp and transaction metadata

All domain entries are added to an internal metadata store for further processing.

B. Resolver Query and Content Hash Extraction

After a domain is logged, Web3Compass initiates a resolution phase by querying its associated resolver smart contract. The resolver may be either:

- A default public resolver
- A custom resolver, as long as it implements the standard contenthash() function

The system calls this function to retrieve the CID (Content Identifier) that points to the website content, typically hosted on IPFS. This CID is stored for content retrieval and indexing.

C. Content Access via IPFS

Web3Compass operates its own IPFS node to retrieve content using the CID. This node ensures that:

- Access is not dependent on external public gateways (e.g., ipfs.io)
- Important content is pinned, avoiding garbage collection
- Retrieval is fast and reliable

As a fallback, public gateways are still used in cases where content is not pinned. The custom gateway dweb3.wtf is used as the default access point, providing censorship-resistant rendering of decentralized sites.

D. Rendering and Filtering of Content

To support Single Page Applications (SPAs) and client-side rendered sites, the system uses a headless browser (Puppeteer). This allows:

- Execution of site JavaScript
- Waiting for DOM readiness
- Extraction of the final rendered HTML

The retrieved HTML is then parsed, and only content relevant for indexing (i.e., text-based information) is retained. Non-text files like images, videos, PDFs, or ZIP archives are ignored.

E. Indexing Engine and Search Configuration

For indexing, Web3Compass uses Meilisearch, an open-source full-text search engine. The system stores metadata and content fields such as:

- CID
- Domain name
- Raw and rendered content
- Extracted text sections

The ranking algorithm is configured based on:

- Word match count
- Typo tolerance
- Term proximity
- Attribute weightings
- Exactness
- Synonym matching (e.g., mapping "DeFi" to "decentralized finance")

The search engine is accessible via a web-based UI and through a public API (with rate limits and authentication keys).

V. Results and Evaluation

Web3Compass has been implemented as a working prototype and has demonstrated successful indexing and search capabilities across a significant volume of decentralized websites. This section presents the outcomes of the system's deployment, focusing on functionality, reach, and system performance.

A. Indexed Content Volume

As of the current version, the system has:

- Indexed over 700,000 decentralized domains, including ENS, UNS, CNS, and others. This scale demonstrates the feasibility of automated large-scale discovery and indexing in decentralized ecosystems.
- Resolved and indexed content from associated IPFS-hosted websites.

Domains pointing to traditional (Web2) endpoints or non-resolvable addresses are automatically filtered and excluded from indexing.

B. Content Availability

By hosting its own IPFS nodes and enforcing a content pinning policy (for content under 100MB), the system maintains high availability of indexed websites. Even when public IPFS gateways are down, Web3Compass continues to serve content reliably through:

- Its internal node network
- A fallback to the custom censorship-resistant gateway: dweb3.wtf

The gateway has proven to be more consistent than external options like ipfs.io, .limo, or .link, especially when accessing content restricted or throttled by central providers.

C. Dynamic Rendering Accuracy

Many decentralized sites use SPAs that require client-side rendering. By employing a headless browser (Puppeteer), Web3Compass can accurately:

- Detect SPA structures
- Render final page content after JavaScript execution
- Extract and index meaningful HTML text

This approach significantly improves search result relevance and completeness, compared to traditional crawlers which would capture only empty or template HTML shells.

D. Search Engine Responsiveness

Using Meilisearch as the indexing engine provides:

- Fast, near-instantaneous search response times
- Support for typo tolerance, proximity-based ranking, and synonym matching
- Privacy-preserving operation with no tracking of user behaviour

The engine operates effectively with the current data volume and remains scalable as more domains are registered and indexed.

E. Real-World Access Outcomes

Users accessing Web3 content via traditional browsers (e.g., Chrome, Firefox) often fail to resolve decentralized sites unless they use extensions or dedicated tools. Web3Compass bridges this gap by:

- Allowing users to search for decentralized content
- Redirecting them through a gateway that fetches and renders the site on their behalf

This increases accessibility for non-technical users, expanding the usability of the decentralized web.

VI. Discussion and Challenges

The development of Web3Compass surfaced several architectural and systemic challenges that are not merely technical, but foundational to the way the decentralized web operates. These issues, while partially mitigated in the current system, remain open areas for future enhancement.

A. Lack of Standardization Across Name Services

Each decentralized domain system — ENS, UNS, BNB NS, and others — has its own smart contract architecture, resolver logic, and field naming conventions. The absence of a unified standard for domain resolution necessitated custom logic for each registry. Maintaining compatibility with evolving contract versions and emerging TLDs requires ongoing manual adaptation and protocol-specific monitoring.

B. Dynamic Content and Non-Standard Web Practices

Many decentralized websites use SPAs and client-side JavaScript to generate content, diverging from conventional static site structures. These sites often lack sitemaps, robots.txt files, or predictable metadata. This presents two key problems:

- Dynamic rendering is resource-intensive, requiring browser emulation for every page.
- Content extraction becomes fragile, depending on DOM timing and non-standard markup.

While Web3Compass uses headless browsers effectively, the method is heavier than traditional indexing approaches and may not scale linearly without optimization.

C. Content Authenticity and Quality

Web3Compass currently does not verify the authenticity or trustworthiness of content. While Hexens provides static security audits for known CIDs, there is no built-in spam detection, duplicate filtering, or trust ranking. The decentralized web's openness is both a strength and a weakness: anyone can publish, but not all content is useful or safe. Without behaviour-based filtering or community signals, maintaining search quality over time may require new relevance models.

D. Infrastructure Limitations and Downtime

The system hosts its own IPFS node and provides an open API. However, the API has experienced downtime, indicating the need for improved infrastructure resilience and monitoring. As user demand grows, performance and availability under load will become more critical.

E. No Historical Version Tracking

The system only indexes the latest version of a decentralized website, based on its current content hash. Older versions are not stored or retrievable through Web3Compass. This limits potential use cases for version-aware research, forensic analysis, or content rollback. These challenges illustrate that the decentralized web is not just a technical space, but a fragmented and under-standardized environment where basic usability infrastructure is still emerging. Web3Compass addresses a critical piece — discoverability — but its development reveals the broader ecosystem's immaturity and the need for further foundational work.

VII. Conclusion

Web3Compass is a dedicated search engine designed to solve one of the most critical issues facing the decentralized web: the lack of a usable discovery layer. By integrating real-time blockchain monitoring, decentralized domain resolution, custom IPFS infrastructure, and privacy-preserving indexing, Web3Compass enables users to find and access decentralized websites that were previously hidden from view. Unlike traditional search engines, Web3Compass does not rely on centralized web crawling, SEO indexing, or behavioural tracking. Instead, it operates within the unique constraints of Web3 — indexing only relevant decentralized content, resolving domains through smart contracts, and rendering dynamic sites through headless Browse. While the system currently indexes and renders over 700,000 decentralized domains, several challenges remain:

- Protocol fragmentation across domain name systems
- Scalability limitations in rendering SPAs
- Absence of content authenticity filtering
- Infrastructure constraints and a lack of historical version tracking

By solving discoverability and access, Web3Compass lays foundational groundwork for Web3 adoption. It transforms isolated decentralized websites into part of a visible, searchable network — a necessary step for turning technical potential into practical utility.

References

[1] N. V. Keizer, O. Ascigil, M. Król, D. Kutscher, G. A Pavlou, "Survey on Content Retrieval on the Decentralised Web", *ACM Computing Surveys*, 2024.

[2] M. Chen, L. Wang, F. Zhang, "An Empirical Study of Content Availability and Retrieval Performance in IPFS", *Computer Networks*, vol. 235, 2024.