

#OMN Technical Specification

This project focuses on surfacing and organising ActivityPub-based object flows - the near real-time, federated movement of data (news, videos, reports, media, and historical objects) between instances within the Fediverse. It leverages rich, flowing metadata to create a commons layer, transforming disparate activity into structured news and shared community knowledge.

A more social overview can be found here: <https://hamishcampbell.com/lets-try-and-simplify-the-omn/>

Core Concepts & Directions

Surfacing Object Flows

The project acts as a second-layer aggregator, making sense of raw ActivityStreams data (videos, articles, posts) flowing between federated servers via ActivityPub.

Commons / News Layer

The goal is to transform decentralised feeds into coherent, public-interest news and media streams. It moves away from single-server thinking, instead embracing the idea of data as a shared "data soup", not confined to any one instance.

Trust and Moderation

The project develops a moderated flow model, where content is filtered and validated through trusted relationships, rather than opaque algorithms. It also introduces a commons/news/collection UX layer to support communities of use. This creates a meta-layer on top of the Fediverse, focused on community-owned (or institutional) news and shared knowledge, rather than just social networking.

Open Media Network

An ActivityPub-based public-facing architecture is built on top of emissary mature codebase: <https://github.com/EmissarySocial/emissary>

Crew

- **Hamish** - Project Lead. 40+ years in grassroots media and technology, Indymedia involvement, and 8+ years working with Fediverse/ActivityPub systems. Responsible for vision, coordination, network-building, UX and core documentation.
 - **Michael** - Lead Developer. 10 years developing #OMN concepts, projects and system logic. Experienced in front-end development and developing backend capabilities this project, working under Ben's technical mentorship.
 - **Ben** - Non-remunerated technical adviser. Core Emissary developer, providing architectural guidance and code review throughout the project. His contribution is in-kind and does not draw from the grant budget.
 - **Alex** - Potential DAT / distributed storage support.
 - **IKA** - Testing and rollout.
-

1. Core Concept

The Open Media Network (#OMN) is a distributed system for content publishing, syndication, and discovery. It connects independent media nodes using open standards, enabling:

- Decentralised publishing
- Federated distribution
- Trust-based filtering
- Commons-oriented governance

The system is designed as a DIY grassroots semantic web, built on the principles of the #4opens (open data, code, standards, and process).

2. The #5F Interaction Model

At the core of #OMN is a minimal interaction model: the #5F (Five Functions).

2.1 Publish

(Object → Stream) Create and distribute content into a stream.

- Input: content object (**Create**)
- Output: federated activity (**Announce**)

2.2 Subscribe

(Stream → Stream) Follow and aggregate content streams.

- Actors: users, groups, instances (`Follow`)
- Targets: actors, tags, collections (`Follow`)

2.3 Moderate

(Object / Stream Evaluation) Apply trust signals to content and sources.

States include:

- Trust / untrust
- Like
- Subscribe / unsubscribe
- Block (`Block`)
- Flag (`Flag`)

Moderation is distributed and local, not global.

2.4 Rollback

(Temporal Stream Filtering) Re-evaluate and reshape historical content visibility.

- Remove previously accepted content
- Filter by source, tag, or trust
- Recompute timelines

Enables time-aware moderation via local re-indexing.

2.5 Edit

(Metadata Mutation) Modify metadata associated with objects.

- Tagging
- Contextual linking
- Annotation

Edits are:

- Distributed
 - Multi-source
 - Non-destructive (versioned)
-

3. System Architecture

Each #OMN node is an independent federated instance.

Emissary provides core services such as:

- ActivityPub integration
- A subscribable search engine (used as the primary news feed)

Most #OMN features can be implemented via templates on top of Emissary. However, advanced features - such as distributed versioning of media require some work on the core codebase.

Node Responsibilities

- Host local content
 - Store metadata and trust relationships
 - Federate with other nodes
 - Apply local moderation rules
-

4. Data Layer

Storage Options

- Local filesystem
- Object storage (S3-compatible)
- Databases (e.g. MongoDB or compatible)

Data Format

- JSON / JSON-LD (for semantic extensibility)

Core Content Model

Field	Type	Description
id	UUID	Unique identifier
origin	URI	Source actor or instance
title	String	Content title
summary	Text	Short description

Field	Type	Description
body	Markdown/HTML	Main content
tags	Array	Keywords / categories
timestamp	DateTime	Publish time
license	URI	Open licence reference
media_attachments	Array	Linked media

5. Communication Layer

Primary Protocol

- ActivityPub (actor-based federation)

Secondary Protocols

- RSS / Atom (legacy interoperability)

Future Extensions

- Dat / Hypercore
 - IPFS
 - Solid pods
 - Nostr (experimental)
-

6. Trust and Moderation Model

Emissary supports basic moderation (site-wide and per-user blocks).

#OMN extends this into a flow-based moderation layer applied to:

- Individual content
- Search results
- Federated streams

Moderation becomes:

- Community-operated

- Distributed across trust networks
- Integrated into content flow itself

Trust is local, subjective, and composable.

7. Content Flow

Ingestion

Content is aggregated from:

- Local posts
- ActivityPub actors
- RSS feeds
- Other Emissary nodes

Normalisation

Metadata stored includes:

- Name, description, icon
- Location and dates
- Tags and publish date
- Ranking

A "trust trail" will be added to each object.

Search Actors

Each search can become an ActivityPub actor:

- Users define criteria
- New matching content is pushed to their inbox

This reverses traditional search logic:

Instead of searching for content, content finds you.

Federation

Search actors enable cross-instance aggregation:

- Instance A can create flows from search results from Instance B
- Communities build shared flows across servers

Indexing

Current capabilities:

- Full-text fuzzy search
- Tag matching

Planned extension:

- Complex boolean tag queries

Distributed Archive

Some nodes may cache full content (not just metadata), creating a distributed archive.

- Stores HTML + ActivityStreams
 - Maintains links to original sources
 - Notifies origin servers of updates
-

8. Front-End Flexibility

#OMN is backend-first. Front-ends are interchangeable.

Possible interfaces:

- Regional news hubs
 - Topic-specific news hubs
 - Indymedia-style sites
 - Custom apps and APIs
-

9. Reference Implementations

#makinghistory

Testing code: <https://unite.openworlds.info/Open-Media-Network/MakingHistory>

A narrative-focused layer:

- Persistent storytelling
- Versioned narratives
- Context-aware media flows

Focus: keeping history alive and connected across the #openweb.

#indymediaback

A reboot of Indymedia concepts:

- "Publish, then filter"
- Federation via ActivityPub + RSS
- Local autonomy with shared flows

Focus:

- Enabling grassroots journalism
- Reconnecting legacy sites via RSS
- Bridging older content and work traditions into the Fediverse

#OGB (Open Governance Body)

A governance layer:

- Distributed decision-making
- Trust-based coordination
- Federated governance clusters

Relationship:

- #OMN → publishing layer
 - #OGB → governance layer
-

10. Design Principles

- Decentralisation (no central control)
 - Interoperability (open standards first)
 - Resilience (redundant systems)
 - Transparency (open processes and data)
 - Human-scale simplicity (#KISS)
-

11. Conceptual Model

The data cauldron and the golden ladle

- Cauldron → shared pool of content
 - Ladle → trust, filtering, meaning-making
-

12. Summary

#OMN is not a platform - it is a protocol-driven ecosystem native to the Fediverse.

The #5F model provides the minimal functional core:

- Publish
- Subscribe
- Moderate
- Rollback
- Edit

Together, these enable a living, federated commons where:

- Content flows
- Trust is negotiated
- Meaning evolves

Without central control.

In simple terms: it's plumbing for #KISS digital flows. The semantic web for human beings.

Further reading: <https://hamishcampbell.com/a-minimal-governable-infrastructure-for-trust-based-media-flows/>

Funding draft: <https://unite.openworlds.info/Open-Media-Network/Open-Media-Network/wiki/Draft-budget>

Small-print-for-funding: <https://unite.openworlds.info/Open-Media-Network/Open-Media-Network/wiki/Answering-the-small-print-for-funding>