Federated Content Routing

A Three-Tier Architecture for Resilient Content Routing in IPFS

Lucas Dias de Espindola

IEEE ICC 2026
Glasgow, Scotland — 24–28 May 2026

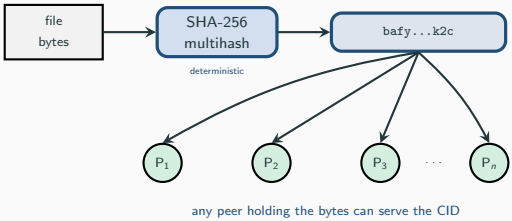
Pinata Technologies Inc.
lucas@pinata.cloud

What is IPFS?

InterPlanetary File System — a peer-to-peer network for *content-addressed* storage.

Key idea
Files are identified by the **hash of their content** (a CID), not by where they live.

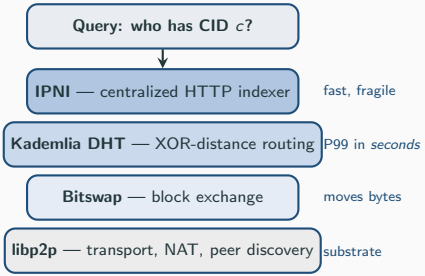
- Same file ⇒ same CID, anywhere.
- Anyone holding the bytes can serve them.
- Tampering is detectable by construction.
- Used by NFTs, dApps, archives, Filecoin.



The hard part isn't *storing* the data — it's *finding who has it*.

Content Routing in IPFS Today

One question to answer: *"Who has CID c?"*



- **Kademlia** is the principled answer. Elegant; P99 measured in *seconds* at scale.
- **IPNI** is the pragmatic shortcut. A centralized HTTP indexer. Fast — but a single point of failure.
- Production silently migrated to IPNI for latency. **It failed on April 15, 2025.**

The trade production made
Speed via centralization — breaking the decentralization promise IPFS was built on.

The Decentralization Paradox

April 15, 2025 — A cascading failure in the InterPlanetary Network Indexer (IPNI) made **billions of CIDs effectively undiscoverable**.

The data was *still there*. The routing layer was not.

Why? A supposedly decentralized network had converged on a single, centralized lookup path to meet production latency SLOs.

Core tension

- Centralized indexers $\Rightarrow$ fast, but single point of failure
- Vanilla DHTs $\Rightarrow$ resilient, but P99 in seconds

Can we have both?

When the Routing Layer Fails — the Scale at Stake

What goes dark when IPNI stalls and the DHT can't catch the load:

Indexed by `cid.contact`

- **1.3 T+** CIDs from hundreds of providers
- Tens of billions added/expired daily via megaprovider ingress

Filecoin onchain — Q3 2025

- **1,110 PiB** active-deal data
- **35.2 M** active deals
- **925** datasets > 1 PiB

The bytes are still on the disks. The find step is what failed.

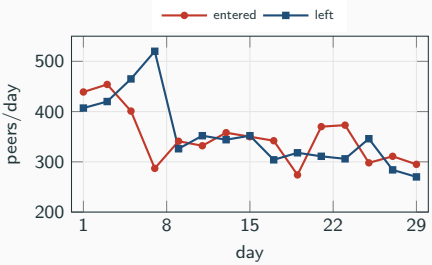
Why Existing DHTs Break in Production

Three failure modes at scale

1. **Membership churn** — auto-scaling and rolling updates destabilize Kademlia routing tables; lookups fail during convergence.
2. **Cold start** — minutes of DHT traversals to bootstrap; incompatible with elastic infra.
3. **Megaprovider saturation** — a few operators publish *billions* of CIDs through a small set of peer IDs; pigeonhole guarantees DHT-server overflow.

DHT peers entering/leaving (per day)

measured by ProbeLab, last 30 days



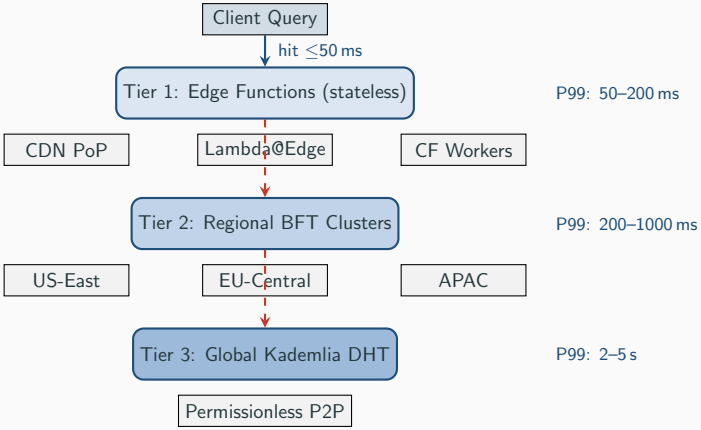
Mean: **350** in, **360** out *every day*.

Root cause: query serving and routing-state control have different requirements — current designs conflate them.

Contributions

1. A **three-tier content-routing architecture**: edge query plane + regional BFT control plane + global DHT fallback.
2. A **snapshot-based lookup path** that decouples consensus from queries while preserving control-plane linearizability.
3. An **SLO-gated reconfiguration protocol**: dual-write, progressive cutover, automatic rollback — with provable latency, error, and freshness budgets.
4. **Implementation & evaluation**: $\geq$99.95% availability under Tier-1 outage; P99 $\leq$300 ms normal / $\leq$950 ms degraded; 500M ads/day at $\leq 0.5\times$ centralized cost.

Three-Tier Architecture



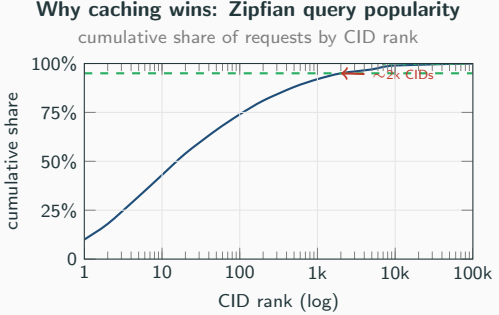
Dashed = failover path Solid = common case (cache hit)

Tier 1 — Edge Functions

Goal: sub-100 ms P99 for the common case.

- **Stateless** Workers / Lambda@Edge at CDN PoPs.
- Cache hits served in $<$50 ms; misses fall through to Tier 2.
- Caches *only* entries attested by a Tier-2 quorum-signed snapshot $\Rightarrow$ no trust placed in edge.
- TTL bounds staleness ($\tau_{\text{ttl}} = 30\,\text{s}$).
- Zipfian popularity ($\theta \approx 0.9$) means caching absorbs the vast majority of queries.

Why caching wins: Zipfian query popularity



Top $\sim$**2,000 CIDs** absorb **95%** of queries $\Rightarrow$ tiny edge cache wins.

Why this works

Verifiable attestations let an untrusted edge serve the hot path without weakening consistency.

Tier 2 — Regional BFT Control Plane

Each region runs $3f+1$ replicas of a HotStuff-style BFT protocol.

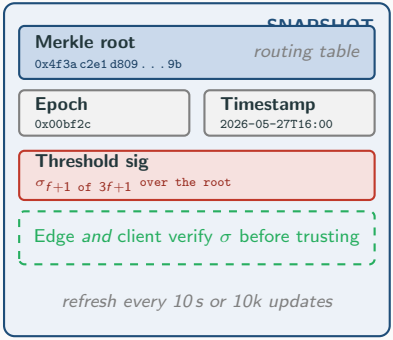
Consensus applies *only* to:

- membership changes
- advertisement-chain updates

Queries do **not** block on consensus — they read from quorum-signed snapshots refreshed every **10 s** or **10k updates**.

Snapshots are pushed to Tier 1 via CDN object stores. Inter-region sync is eventual via gossip; freshness lag bounded by SLO.

Anatomy of a signed snapshot



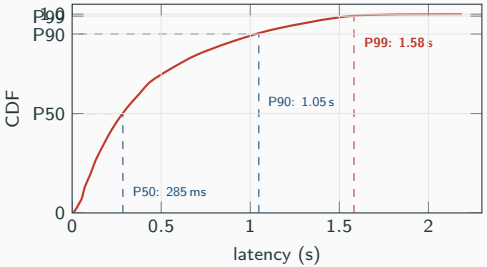
Tier 3 — Global Kademlia Fallback

- Standard permissionless DHT, IPFS-compatible.
- Regional clusters periodically `PROVIDE` routing-table summaries after BFT commit.
- Used for:
  - cross-region lookups (Tier 2 misses ⇒ ask Tier 3 which region holds it)
  - universal fallback if both upper tiers degrade
  - bootstrapping new regions
- Intentionally slow (P99: 2–5 s) but **always reachable**.

Liveness over **latency**.

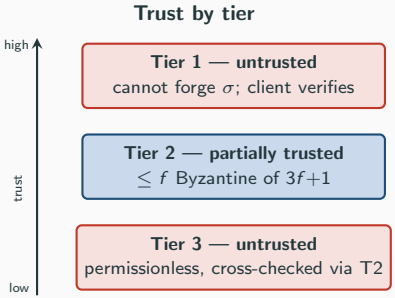
Real-world DHT lookup CDF

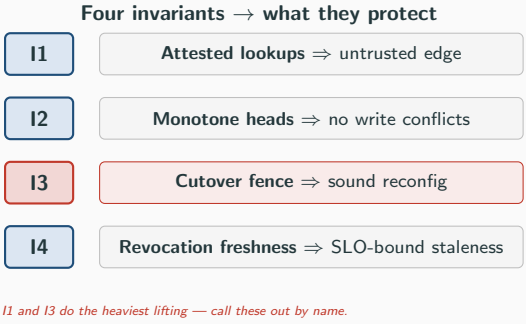
measured by ProbeLab (n=3,513 lookups)



Even at the median, a vanilla DHT lookup is ∼**285 ms**.

Threat Model & Key Invariants

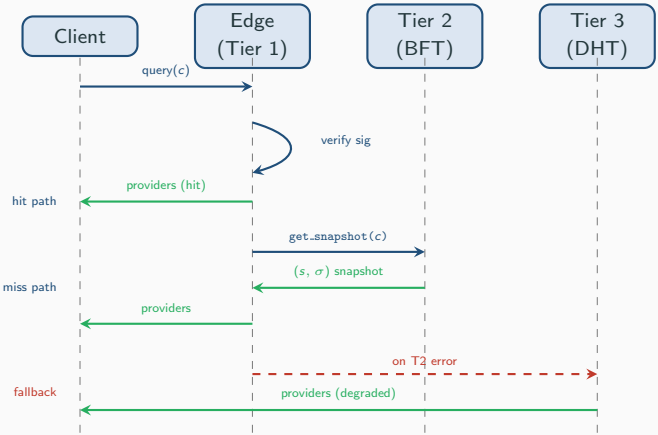




I1 and I3 do the heaviest lifting — call these out by name.

Algorithm — Attested Snapshot Lookup

1. Edge receives query for CID c.
2. Cache hit $\Rightarrow$ verify threshold sig against quorum pubkey.
   - Pass $\Rightarrow$ return providers.
   - Fail $\Rightarrow$ evict, treat as miss.
3. Miss $\Rightarrow$ ask Tier 2, verify, cache (TTL τ_{ttl}), return.
4. Tier-2 error $\Rightarrow$ Tier 3.

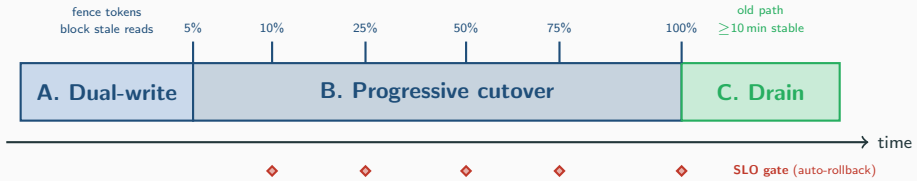


Property

Cryptographic attestation lets an *untrusted* edge serve hot lookups — the trust boundary is the signature quorum, not the cache.

SLO-Gated Reconfiguration

Zero-downtime topology change in three phases — progressive cutover is automatically rolled back on SLO violation.



Enforced budgets throughout reconfig:

$L_{99} \leq 300\,\text{ms}$ Error $\leq 0.1\%$ Freshness $\leq 30\,\text{s}$

- Step violates SLO for $>1\,\text{min} \Rightarrow$ rollback one step automatically.
- Production use: entire BFT replica sets swapped without paging an operator.

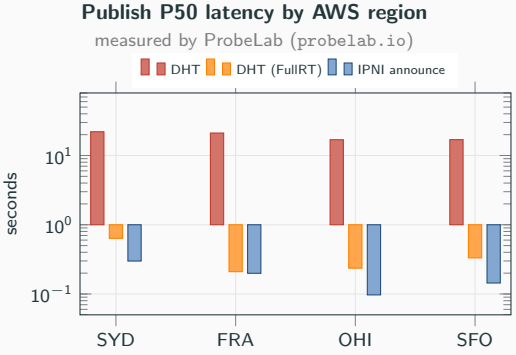
The Megaprovider Problem

Billions of CIDs from a small number of peer IDs. Provider records expire after **24 h** — continuous re-announcement is mandatory.

Sequential reprovide breaks when total reprovide time exceeds 24 h. Pigeonhole: too few DHT servers, too many records.

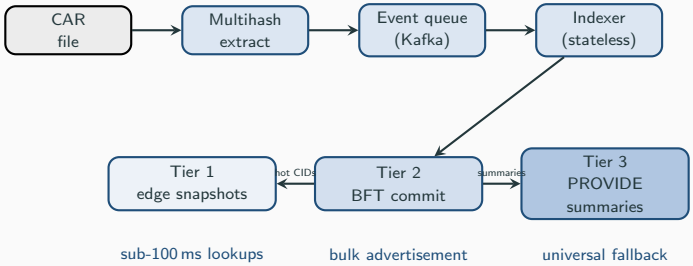
How federation fixes it

Tier 2 absorbs bulk writes via batched, BFT-committed ad chains. Tier 1 gets hot snapshots only. Tier 3 gets summaries. Announcement volume **decoupled** from query latency.



DHT publish **17–22 s** vs IPNI **100–300 ms** — a ∼**80**× gap.

CID Ingestion Pipeline

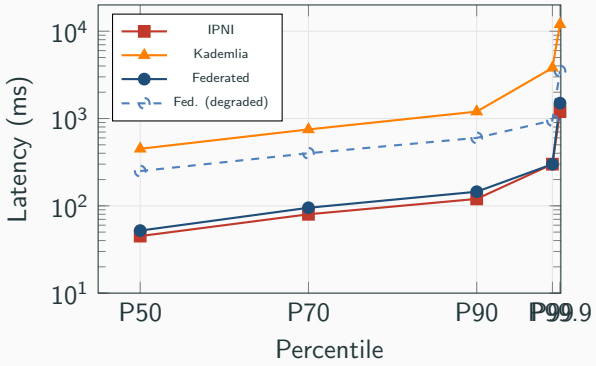


- **CAR files** (Content Addressable aRchives) bundle IPLD blocks + their CIDs.
- Stateless indexers extract multihashes and produce provider records — *"this peer stores this CID"*.
- Tier 2 commits via BFT, then propagates **selectively**: hot snapshots to edge, summaries to global DHT.
- Resumable; horizontally scalable; handles **billions of CIDs/day** without saturating any single tier.

Evaluation — Query Latency

System	P50	P90	P99	P99.9
IPNI	45	120	300	1200
Kademlia	450	1200	3800	12000
Federated	**52**	**145**	**300**	**1500**
Fed. (T1 fail)	250	600	950	3500

- P50 within 7 ms of centralized IPNI.
- P99 matches IPNI at 300 ms.
- Even *with Tier-1 down*, sub-second P99.
- Vanilla Kademlia: 3.8 s P99 — why production avoided pure DHTs.

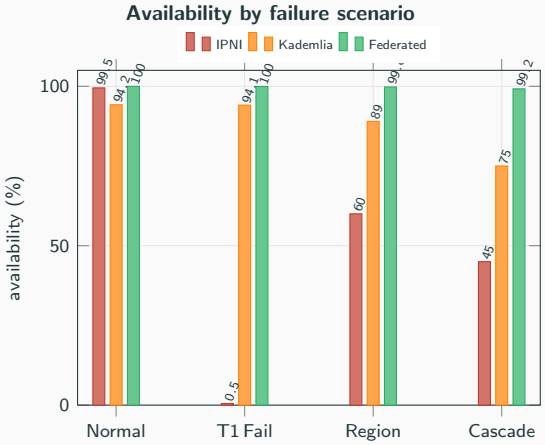


Evaluation — Availability Under Failures

System	Norm.	T1	Reg.	Casc.
IPNI	99.5	0	60	45
Kademlia	94.2	94.1	89	75
Federated	**99.99**	**99.95**	**99.8**	**99.2**

values in %

- **T1 outage:** IPNI → 0%. Federated stays at 99.95% via Tier-2 failover.
- **Region partition:** federated 99.8% via cross-region snapshots + DHT.
- **Cascading:** tier isolation caps blast radius (99.2% vs IPNI 45%).



IPNI T1-fail bar is 0 % — shown at 0.5 to remain visible.

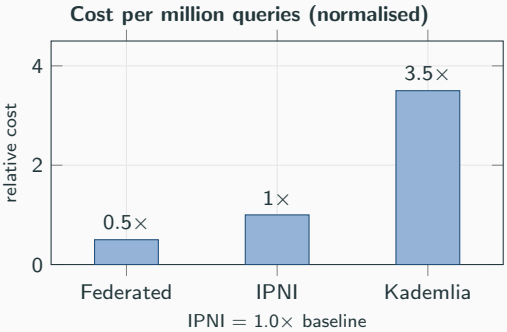
Evaluation — Scale & Cost

Scale (sustained)

- **500M** advertisements/day.
- ∼5.8 k ads/s peak.
- ≈ 10× current production IPFS.
- Control plane: ≥2 k ops/s, commit P99 ≤200 ms under 2%/min churn.

Why cost halves

Edge caching absorbs the long tail. Geographic distribution amortises egress. Cost dominated by Tier-2 capacity, not query volume.



Federated ≈ **0.5**× the IPNI baseline. Kademlia is expensive because of fan-out traffic per lookup.

Key result

Better latency *and* better resilience *and* lower cost — the trilemma narrows substantially when consensus is removed from the query path.

Related Work

DHTs Kademlia, Chord, CAN, Pastry — strong theoretical foundations but lack cloud-native tiering and BFT for control planes.

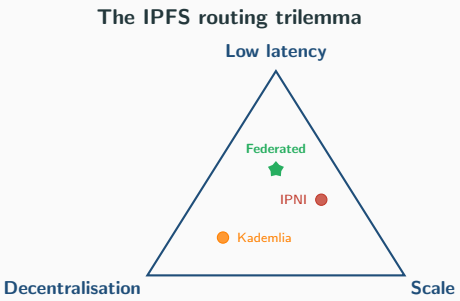
CDNs Akamai-style edge deployment proves caching wins, but relies on centralized control. We bring the pattern to a permissionless substrate.

BFT PBFT, HoneyBadger, HotStuff — we apply *scoped* BFT only to the control plane, so consensus latency never blocks reads.

IPFS measurement Trautwein et al., Henningsen et al., Wei et al. quantified the performance/resilience gap; we propose an architectural fix rather than a protocol tweak.

Takeaways

1. **The query path doesn't need consensus.** Snapshot-based attested reads give cache-fast lookups without weakening consistency.
2. **Scope BFT to the control plane.** Membership + advertisement updates need linearizability; reads don't.
3. **Use a DHT as a safety net, not a hot path.** Tier 3 is slow on purpose — it guarantees liveness when everything else fails.
4. **SLO-gated reconfiguration works.** Dual-write + progressive cutover + automatic rollback keeps latency, error, and freshness budgets through topology changes.



Federation reaches all three corners.

Federation, not centralization, resolves the IPFS routing trilemma.

Thank you

Questions?

Lucas Dias de Espindola
Pinata Technologies Inc.
lucas@pinata.cloud

IEEE ICC 2026 — Glasgow

Backup — Availability Bound

Let p_1, p_2, p_3 be per-tier availabilities and $\rho \in [0, 1)$ be the correlation between Tier-1 and Tier-2 failures (Gaussian copula approximation):

$$A \geq 1 - (1 - p_3)\Big[(1 - p_1)(1 - p_2) + \rho\sqrt{p_1(1 - p_1)p_2(1 - p_2)}\Big]$$

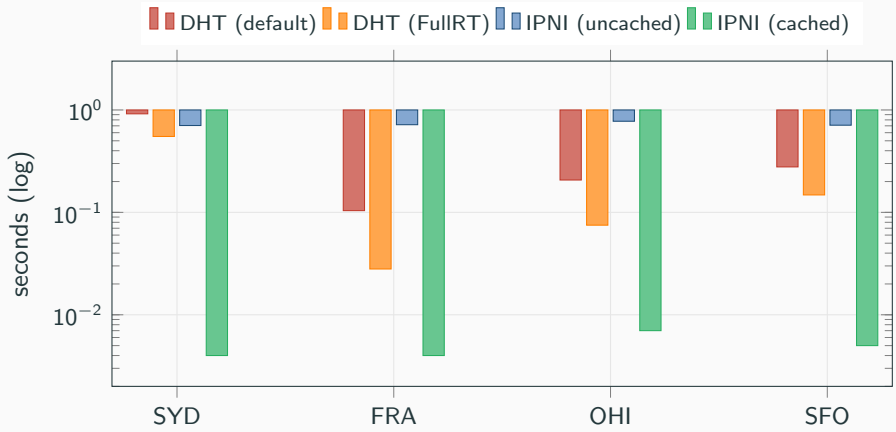
- Independence ($\rho = 0$): availability multiplies as expected.
- Worst-case correlated failure: $\rho \to 1$ collapses the upper bound to a two-tier system.
- Empirically fit from production telemetry.

Backup — Parameters

Symbol	Meaning
τ_{ttl}	Cache TTL / freshness budget (30 s)
T_{snap}	Snapshot interval (10 s or 10k updates)
$\mathcal{Q}$	Threshold quorum ($f+1$ of $3f+1$)
A	End-to-end availability
L_p	Latency at percentile p
ρ	Failure correlation parameter
θ	Zipf popularity parameter

Backup — ProbeLab Lookup Measurements

Lookup P50 latency by AWS region, measured by ProbeLab (`probelab.io`).



DHT default 100 ms (FRA) – 900 ms (SYD). **IPNI cached** 4–7 ms across all regions.

Backup — Experimental Setup

- **Regions:** 4 (us-east-1, eu-central-1, ap-southeast-1, +1 for failover).
- **Tier 1:** Workers / Lambda, warm concurrency 1 k per region.
- **Tier 2:** 7 nodes/region on m6i.large (f=2).
- **Tier 3:** 200 Kademlia nodes, 20 FullRT.
- **Workloads:** G1 (95% reads, θ=0.9); G2 (80% reads, θ=0.7, hourly skew); G3 (tail stress, θ=1.1).
- **Bursts:** 30 k ads/s for 15 min.
- **Failure matrix:** DB slowdown, regional partition, cascading kill 10%/hr, 30% Byzantine (drop / equivocate / spam).