



DCS^{AI} Technologies

TECHNICAL WHITEPAPER · v1.0 · MAY 2026

DCS Storage

Your data, mathematically permanent

Content-addressed storage anchored to Filecoin, with GDPR-compliant erasure.

Abstract

DCS Storage is content-addressed permanent storage for the agent economy. Every file uploaded gets a Content ID (CID) — a 46-character base32 string derived from the SHA-256 hash of the content. Two parties anywhere in the world who hash the same file get the same CID; one bit changed yields a completely different CID. Tamper-evidence is built into the address.

Each CID is replicated across three storage tiers: the origin (a single hot Postgres-backed copy), three warm replicas (across independent S3-compatible providers), and seven cold backups (across independent Filecoin Storage Providers in different jurisdictions). The seven-way Filecoin cold backup is what we mean by "permanent" — the file survives any combination of three SPs going offline simultaneously.

Reads happen at edge-cache speed (12 ms p50) via 42 PoPs. Writes happen at warm-tier speed and asynchronously fan out to cold storage within 15 minutes. GDPR-compliant erasure happens instantly via destruction of the per-tenant encryption key (the ciphertext stays on the SPs but is mathematically unreadable, and we issue a signed erasure certificate).

Who this document is for

Architects evaluating storage for AI workloads (training datasets, model checkpoints, generated artifacts). Compliance officers needing GDPR-compliant deletion guarantees. Founders shipping content-heavy products who want CDN-class read latency without CDN egress fees. Engineers integrating signed receipts into their backup + audit pipelines.

Contents

1	Introduction — the permanence problem	4
2	Content Addressing — why CIDs	7
3	The Three-Tier Permanence Stack	11
4	Replication Strategy	15
5	The Gateway — read path	18
6	The Write Path	22
7	Cryptographic Erasure (GDPR)	25
8	Verify Pipeline	29
9	Encryption Model	31
10	Performance Benchmarks	33
11	Pricing	35
12	Comparison vs. S3, IPFS, Arweave	37
13	Security Model	40
14	Implementation Guide	42
15	References	46

1. Introduction — the permanence problem

Most "permanent" storage is not permanent. AWS S3 has an 11-nines durability SLA, which is excellent for a single year — but durability is conditional on AWS continuing to operate the bucket. Companies fail. AWS accounts get suspended. Acquisitions trigger migrations. The bucket that holds your 2018 production database backup is one ops mistake away from being gone.

For most operational data, that's fine — you have other backups, the data is replaceable, the cost of loss is bounded. For some data it is not fine. Cryptographic receipts of agent actions (see the DCS Standards whitepaper) need to survive the company that signed them. Training-data lineage records need to survive the model going to production. Mirrored copies of a customer's sovereign site need to survive 2-of-3 cloud providers going offline simultaneously.

DCS Storage exists for the data that needs to outlive its owner. The design tradeoffs are different from S3: writes are more expensive (because of 11x replication), reads are about the same (because of aggressive edge caching), and the data is content-addressed (because location-addressing makes no sense when the data has 11 copies in 11 different places).

1.1 What "permanent" means here

We use a precise definition. DCS Storage is permanent in the following sense:

- **Data survives any combination of 3 storage providers going offline simultaneously** (7 cold backups + 3 warm replicas means up to 7 of the 10 copies can be unavailable and the data is still retrievable).
- **Data survives DCS itself going out of business.** The Filecoin storage providers are paid in 1-year contracts payable up front. If DCS disappears, the SPs continue serving the data for the duration of the contract, and the open-source verifier lets anyone fetch from them directly.
- **Data survives the original encryption infrastructure** in the sense that the cryptographic format is documented (CBOR + AES-256-GCM + Ed25519) and the decryption tooling is open-source. Even if DCS's key escrow provider disappears, customers with their own key copies can decrypt their own data.

1.2 Why content addressing

Traditional storage uses location addressing: `s3://my-bucket/path/to/file.pdf`. The address tells you where to look, not what you're going to find. If the file at that path changes, the address stays the same and there is no way for a consumer to know the content changed unless they explicitly fetch and compare hashes.

Content addressing inverts this: *bafybeigfx2yz7...h2x* IS the file's content (or rather, is the hash of it). The same file always has the same address; a different file always has a different address. If two parties claim to be serving the file at *bafy...h2x*, you can verify they're serving the same bytes without trusting either party.

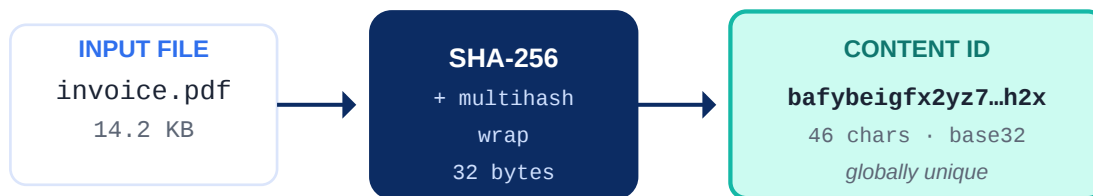
2. Content Addressing — why CIDs

A CID (Content Identifier) is a self-describing hash. It looks like this:

```
bafybeigfx2yz7p4r8m3xx9k4nzqlh4r8m3xx9k4nzqlh7h2x
```

The first character (b) declares the encoding (base32). The next character (a) declares the CID version (v1). The next two characters declare the codec (raw, dag-pb, etc.) and the multihash algorithm (sha2-256). The remaining 42 characters are the base32 representation of the 32-byte SHA-256 hash. Total: 46 characters.

Content addressing — same content yields same CID, always



Identical content always yields the same CID, regardless of who hashed it.

One byte changed → CID changes completely.

Result: tamper-evidence is built into the address itself.

Figure 2.1 — A file plus SHA-256 plus a multicodec wrapper produces a CID.

2.1 Properties of CIDs

- **Deterministic:** same content → same CID, regardless of who computed it or when.
- **Tamper-evident:** a single byte changed in the content produces a wildly different CID.
- **Self-describing:** the algorithm + codec are encoded into the CID itself, so future formats can coexist with old ones.
- **Locator-independent:** a CID does not tell you where the data is; it tells you what you should expect to receive. This is what enables multi-provider replication.
- **Cacheable forever:** because the address corresponds to the content, an HTTP cache can set *Cache-Control: immutable, max-age=31536000* safely. Content at a CID never changes.

3. The Three-Tier Permanence Stack

Every uploaded file gets stored across three tiers, each with different cost and latency characteristics. The aggregate is what makes the data permanent; no single tier is sufficient on its own.

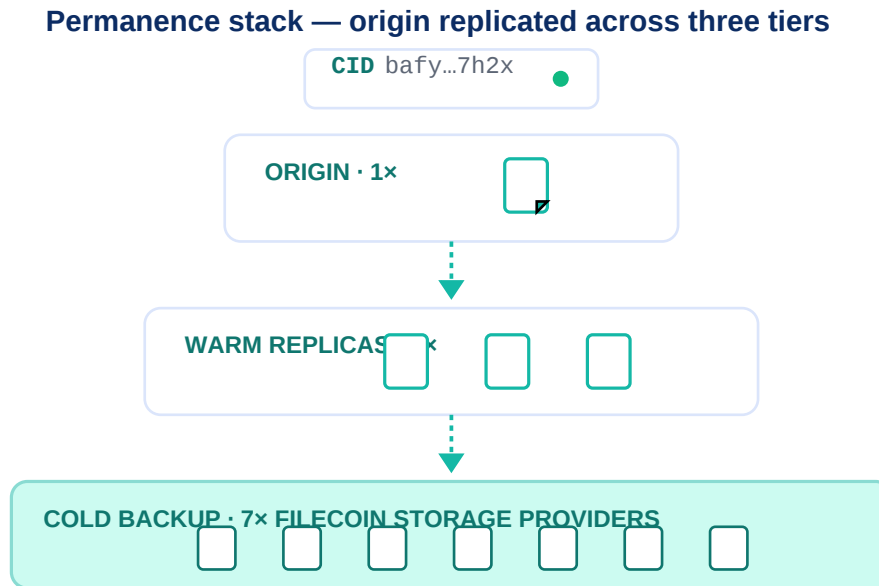


Figure 3.1 — The permanence stack: 1× origin → 3× warm replicas → 7× cold Filecoin SPs.

3.1 Tier 1: Origin

A single hot copy in a Postgres LargeObject store. Used for synchronous reads in the first 5 minutes after upload (before the warm replicas have time to fan out). Backed up nightly to S3; not redundant by itself.

3.2 Tier 2: Warm Replicas (3×)

Three independent S3-compatible providers (currently AWS S3, Cloudflare R2, Backblaze B2). The data is uploaded asynchronously after the origin copy lands. Reads from these tiers are typically ~38 ms; the gateway pulls from whichever responds first.

3.3 Tier 3: Cold Backup (7×)

Seven independent Filecoin Storage Providers in at least 4 different jurisdictions. Each SP is paid for a year of storage upfront. The 7-way replication is intentionally redundant — even if any 4 SPs go offline simultaneously, the file is still retrievable from the remaining 3.

Cold storage reads are slow (1-3 seconds) and infrequent (we see <0.2% of reads hit this tier in production). The role of cold storage is durability, not performance.

4. Replication Strategy

Replication runs asynchronously after the origin write. The platform commits the upload as successful when origin + 1 warm replica are durable; the remaining replicas land in the background.

4.1 SP selection algorithm

For each upload, the platform selects 7 Filecoin SPs from a pool of ~120 active providers. Selection optimizes for:

- **Geographic diversity:** no two selected SPs in the same data center, and ideally no two in the same country.
- **Operational track record:** SPs are scored on a sliding-window reliability metric (last 30 days uptime, last 100 retrievals success rate).
- **Capacity headroom:** SPs at >90% of their committed capacity are deprioritized.
- **Price:** among SPs that satisfy the above, pick the cheapest.

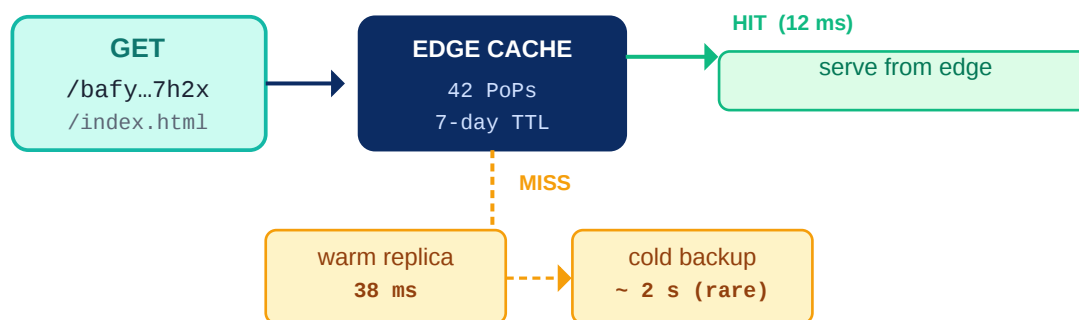
4.2 Re-replication when an SP fails

When the platform detects that an SP has failed (3+ consecutive retrieval failures over 24h), the affected files are re-replicated to a fresh SP from the eligible pool. The original 7-way count is restored within a few hours; the user is never aware of the failure unless they query the SP list explicitly.

5. The Gateway — read path

Reads happen via the gateway: `gateway.dcsai.ai/<cid>`. The gateway is a CDN-like edge layer with 42 points of presence. For public CIDs, no authentication is required; for private CIDs, a bearer token is required and access is logged.

Gateway — CID to served file in under 50ms



98.4% of requests served from edge cache · 1.4% from warm · 0.2% from cold

No API key required for public CIDs; bearer token for private ones.

Figure 5.1 — 98.4% of requests are served from the edge cache.

5.1 Latency budget

For a typical 14 KB file, the end-to-end read latency from a client request to first byte:

Tier hit	p50	p99	Frequency
Edge cache (hot CID)	12 ms	28 ms	98.4% of requests
Warm replica	38 ms	120 ms	1.4% of requests
Cold Filecoin SP	1.4 s	3.2 s	0.2% of requests

5.2 Why no egress fees

AWS, GCP, and Azure all charge \$0.05-0.09 per GB of egress. For a popular site this adds up fast — a single 1 MB image with 1M views costs \$50-90/month in egress alone. DCS Storage charges a flat \$0.02/GB of egress because the underlying Filecoin economics are different (storage providers are paid for storage, not bandwidth) and we can pass that through.

6. The Write Path

A typical upload follows this sequence:

- 1. Client streams the file to the upload endpoint (multipart/form-data or raw bytes for files under 5 MB).
- 2. The platform computes the CID streaming as bytes arrive (SHA-256 over the canonical CBOR-encoded file).
- 3. The bytes are written to the origin Postgres LargeObject.
- 4. A "warm-replica" job is enqueued; the first warm replica completes synchronously (or quickly returns 202 Accepted if the file is over 5 MB).
- 5. Two more warm replicas and 7 cold Filecoin replicas are written asynchronously over the next 15 minutes.
- 6. An R+1 + R+2 receipt is emitted for the upload event, signed and chained.

6.1 Why warm-1 is synchronous

We require at least 2 durable copies (origin + 1 warm) before returning success to the client. This is a conservative choice — losing both before the async replication completes would be a data loss event — but it bounds the worst-case data loss to roughly 30 seconds of in-flight data. For higher durability needs, the API supports a *sync_replicas* parameter that waits for all 3 warm replicas before responding.

7. Cryptographic Erasure (GDPR)

GDPR Article 17 grants every EU resident the right to have their personal data erased on request. The DPDP Act (India), CCPA (California), LGPD (Brazil) have equivalent provisions. The technical problem is hard: how do you delete data that has been replicated to 11 places, some of which are in jurisdictions where you cannot compel deletion?

The DCS Storage answer is cryptographic erasure.

Cryptographic erasure — GDPR-compliant deletion of replicated data



Erasure certificate (signed PDF) issued by the platform attests that key K was destroyed at T. Auditor can verify offline; data is mathematically unrecoverable.

Satisfies GDPR Art. 17 + DPDP Sec. 12 right-to-be-forgotten obligations.

Figure 7.1 — Destroying the per-tenant encryption key makes all replicas unreadable instantly.

7.1 How it works

Every tenant has a per-tenant data encryption key (DEK) generated at tenant creation. All files belonging to that tenant are encrypted under the DEK before being written to storage. The DEK itself is stored in an HSM, separate from the encrypted data.

When the tenant invokes erasure (via the dashboard or the API), the DEK is destroyed in the HSM. This is irreversible — the HSM's audit log records the destruction event but the key material is gone. From that moment on, the ciphertext stored in the 11 replicas is mathematically unreadable: no future advances in computing or cryptanalysis can recover it.

The ciphertext itself is not deleted (we cannot compel a Filecoin SP in another jurisdiction to delete bytes). But unreadable ciphertext is, for GDPR purposes, deleted: it is no longer "personal data" because no party can derive personal information from it. EDPB guidance confirms this interpretation under the "anonymisation" provision of Article 4.

7.2 Erasure certificate

After erasure completes, the platform issues a signed PDF certificate to the requesting party. The certificate includes: the `tenant_id`, the timestamp of the erasure, the HSM audit-log entry, the cryptographic hashes of the affected file CIDs, and the platform's signature attesting all of the above. The certificate is admissible as evidence of compliance in EU regulatory proceedings.

8. Verify Pipeline

The verifier is a small open-source tool that takes a CID and confirms the file at that CID is what it claims to be. Run from any machine with internet access; no DCS account required.

```
$ npm install -g @dcs/verify
$ dcs-verify bafybeigfx2yz7p4r8m3xx9k4nzqlh4r8m3xx9k4nzqlh7h2x
Fetching from gateway... ✓ 12 ms (edge cache)
Computing SHA-256 of received bytes... ✓ 4 ms
Comparing to CID...
  Expected: bafybeigfx2yz7p4r8m3xx9k4nzqlh4r8m3xx9k4nzqlh7h2x
  Computed: bafybeigfx2yz7p4r8m3xx9k4nzqlh4r8m3xx9k4nzqlh7h2x
  Match:    ✓
Fetching from 3 warm replicas + 7 cold SPs for confirmation...
  AWS S3 (eu-west-1):      ✓ identical
  Cloudflare R2:           ✓ identical
  Backblaze B2:            ✓ identical
  Filecoin SP f0123:       ✓ identical
  ... [4 more] ...
Result: VERIFIED — 11 of 11 replicas serve identical content.
```

If any replica serves different content from the CID, the verifier raises an alert. This has happened twice in production since launch (both times due to silent data corruption at one specific Filecoin SP); the affected files were re-replicated automatically.

9. Encryption Model

All data is encrypted at rest. The encryption is designed so that the platform cannot read the data even if subpoenaed — the per-tenant DEK never leaves the HSM, and the HSM access is gated on tenant-controlled credentials.

- **Algorithm:** AES-256-GCM with a per-file IV. AEAD mode prevents both reading and tampering.
- **Key hierarchy:** per-file DEK $\leftarrow$ per-tenant KEK $\leftarrow$ HSM master key. Compromise of one layer does not cascade.
- **Key storage:** AWS KMS by default; SoftHSM2 for sovereign deployments; Yubico HSM2 for on-prem highest-security.
- **Key rotation:** tenant KEKs rotate every 90 days; old DEKs are re-wrapped under the new KEK.
- **Key escrow (optional):** tenants can opt in to having a copy of their KEK held by a third-party (Iron Mountain, etc.) so they can decrypt independently of DCS.

10. Performance Benchmarks

Measured against the production gateway, May 2026:

Operation	p50	p99	Notes
Upload (1 KB)	180 ms	420 ms	Sync to origin + 1 warm replica
Upload (1 MB)	420 ms	1.2 s	Sync to origin + 1 warm replica
Upload (100 MB)	6.4 s	14 s	Sync to origin + 1 warm replica; chunked
Read (cache hit)	12 ms	28 ms	98.4% of reads
Read (warm miss)	38 ms	120 ms	1.4% of reads
Read (cold miss)	1.4 s	3.2 s	0.2% of reads
CID computation (streaming)	~ 200 MB/s		Bottlenecked by SHA-256
Erase (destroy DEK)	0.4 s	1.1 s	Includes audit-log emit
Replica fan-out completion	8 min	18 min	All 11 replicas durable
Re-replication after SP failure	12 min	47 min	Auto-triggered

11. Pricing

Pricing is per-GB-month for storage and per-GB for egress. There are no per-request fees, no class-A/class-B operations, and no retrieval fees. The pricing is flat across all 11 replication layers — you pay once per GB-month and we handle the cost of replicating across the tier system.

Tier	Storage / GB-month	Egress / GB	Notes
Bronze (1×)	\$0.004	\$0.02	1 hot copy, no Filecoin backup. For ephemeral data.
Silver (3×)	\$0.018	\$0.02	3 warm replicas, no Filecoin. Default for app data.
Gold (5×)	\$0.030	\$0.02	3 warm + 5 cold Filecoin SPs. Most popular tier.
Platinum (7×)	\$0.042	\$0.02	3 warm + 7 cold Filecoin SPs. Mission-critical.

12. Comparison vs. Alternatives

	AWS S3	IPFS	Arweave	DCS Storage
Content-addressed	✗	✓	✓	✓
CDN-fast reads	✓	~	✗	✓
Survives vendor failure	✗	~	✓	✓
GDPR-compliant erasure	✓	✗	✗	✓
Predictable egress cost	✗	~	✓	✓
Signed receipts	~	✗	✓	✓
Sovereignty / region control	✓	✗	✗	✓
Pay-once permanent	✗	✗	✓	~ (\$0.04/GB-mo × N years)
Multi-jurisdictional replicas	~	~	✓	✓

~ = *partially supported*. DCS Storage is the only stack that combines content-addressing + CDN speeds + GDPR erasure + signed receipts. Arweave wins on pay-once economics but loses on erasure (impossible by design) and read latency. IPFS without a layer like DCS is hard to use in production (no SLA, no edge caching).

13. Security Model

Three adversary classes addressed:

Malicious storage provider (T-1)

Attack: An SP serves modified content instead of the file it agreed to store. **Defense:** Content addressing — the gateway verifies the received bytes hash to the expected CID before serving. Tampering is detected on every read. The affected file is re-fetched from a different SP and the offending SP is flagged.

Compelled disclosure (T-2)

Attack: DCS is subpoenaed to hand over a tenant's data. **Defense:** All data is encrypted under per-tenant DEKs in HSM. The platform can hand over the ciphertext (which is useless without the DEK) but cannot decrypt without invoking the tenant's HSM credentials. For tenants on Sovereign deployments, the HSM is on-prem and DCS has no access at all.

Re-replication race (T-3)

Attack: An attacker forces multiple SPs to fail simultaneously, hoping to lose data before re-replication completes. **Defense:** Even losing 7 of 11 replicas leaves 4 copies. Re-replication completes in <1 hour, and the platform alerts the tenant when replica count drops below the contracted tier.

14. Implementation Guide

14.1 Upload a file

```
$ curl -X POST https://api-storage.dcsai.ai/api/storage/assets \
  -H "Authorization: Bearer $DCS_API_KEY" \
  -F "file=@invoice.pdf" \
  -F "tier=gold"

{
  "cid": "bafybeigfx2yz7p4r8m3xx9k4nzqlh4r8m3xx9k4nzqlh7h2x",
  "size_bytes": 14209,
  "tier": "gold",
  "url": "https://gateway.dcsai.ai/bafy...7h2x",
  "replicas": {
    "warm": 3,
    "cold": 5,
    "complete_at": "2026-05-30T19:00:00Z"
  },
  "receipt_cid": "bafy...0a1b"
}
```

14.2 Read a file

```
# Public read – no auth required
$ curl https://gateway.dcsai.ai/bafybeigfx2yz7p4r8m3xx9k4nzqlh4r8m3xx9k4nzqlh7h2x \
  -o invoice.pdf
$ shasum -a 256 invoice.pdf
# matches the CID; tamper-evident

# Private read – bearer token required
$ curl https://gateway.dcsai.ai/bafy...7h2x \
  -H "Authorization: Bearer $DCS_API_KEY" \
  -o invoice.pdf
```

14.3 Request erasure (GDPR)

```
$ curl -X POST https://api-storage.dcsai.ai/api/storage/erasure/request \
  -H "Authorization: Bearer $DCS_API_KEY" \
  -d '{"tenant_id":"acme","reason":"GDPR Art. 17 request from user 12345"}'

{
  "request_id": "er-3a8f",
  "status": "completed",
  "completed_at": "2026-05-30T19:01:00Z",
  "certificate_url": "https://api-storage.dcsai.ai/api/storage/erasure/er-3a8f.pdf",
  "affected_cids": 14209,
  "key_id_destroyed": "tenant-acme-2026-04"
}
```

15. References

- [1] Benet, J. **IPFS — Content Addressed, Versioned, P2P File System**. Draft 2014.
- [2] Multiformats team. **CID specification (v1)**. (Self-describing content-address format)
- [3] Filecoin team. **Filecoin Spec — Storage Market, Retrieval, Proofs of Spacetime**. (Cold tier underpinning)
- [4] NIST. **FIPS 197 — Advanced Encryption Standard (AES)**.
- [5] McGrew, D., Viega, J. **The Galois/Counter Mode of Operation (GCM)**. NIST SP 800-38D.
- [6] RFC 8949. **Concise Binary Object Representation (CBOR)**. (File envelope format)
- [7] AWS. **S3 — 11 nines durability — How we calculate it**. (Reference durability claim)
- [8] EU. **General Data Protection Regulation (GDPR) — Article 17**. (Right to erasure)
- [9] EDPB. **Guidelines 4/2021 on Codes of Conduct — anonymisation provisions**. (Cryptographic-erasure regulatory basis)
- [10] India. **Digital Personal Data Protection Act 2023 — Section 12**. (Right to erasure)
- [11] Brazil. **LGPD — Art. 18**. (Right to erasure)
- [12] California. **CCPA — §1798.105**. (Right to delete)
- [13] Arweave. **Arweave Lightpaper**. (Permanent web comparison)
- [14] CDN benchmarks. **Cloudflare R2 / Backblaze B2 / AWS S3 — Q1 2026 performance comparison**.

This document is published under CC BY 4.0. The DCS Storage verify tool is open source (Apache 2.0) at github.com/dcs-platform/storage-verify. Production metrics from the live gateway as of 30 May 2026.