DAS

Documentary Accountability Substrate

Introduction and Relationship to DDRP and CAAP

Quantum Inquiry | quantuminquiry.org

1. The Problem the Stack Leaves Open

DDRP answers the structural question: what obligation-creating language appeared in this document, was it detectable, and was each obligation structurally resolved or left open. CAAP answers the accountability question: who acted on the obligations in that artifact, when, under what authority, and against which cryptographically verified version of the record.

Together the two layers produce a complete record for a single document, across a single extraction cycle, logged in a single event chain. That is sufficient for an isolated transaction. It is not sufficient for an organization.

Organizations produce documents continuously. The same actors appear across dozens of artifacts. The same obligations recur in successive versions of the same instrument. Authority chains shift over time. Artifacts accumulate. Event logs multiply. Without a substrate to hold the stack together, the record is technically complete but operationally inaccessible: a collection of isolated proofs that cannot be queried, compared, or reconstructed at scale.

> *The record exists. It cannot be found. It cannot be compared. It cannot be reconstructed across time. That is a different kind of failure than the ones DDRP and CAAP were designed to prevent — and it is the one DAS is designed to close.*

This is the problem the Documentary Accountability Substrate is designed to close. Not by modifying DDRP. Not by extending CAAP. By sitting beneath both.

2. What DAS Is

DAS — the Documentary Accountability Substrate — is the persistence and indexing layer beneath DDRP and CAAP. It receives completed DDRP artifacts and CAAP event logs. It stores them in a durable, queryable, retention-governed ledger. It enforces the immutability guarantees that DDRP and CAAP assert within a single cycle across the full organizational record.

DAS is	A persistence layer. It stores what DDRP found and what CAAP recorded — not to interpret either, but to ensure both survive.

DAS does not extract obligations. It does not attribute actions. It does not interpret documents or assess compliance. Those functions belong to DDRP and CAAP and are not repeated. DAS is the ledger that makes the record reconstructible at organizational scale.

Three properties define DAS's design:

- **Append-only:** Entries are never modified or deleted. The ledger grows in one direction. Past states are permanently retrievable.
- **Hash-chained:** Every artifact and event log is accepted by hash reference. DAS verifies the declared hash against the received payload before writing to the ledger. A hash mismatch is a rejection, not an error to be resolved.
- **Retention-governed:** DAS enforces configurable retention rules per document class, jurisdiction, or organizational policy. The ledger does not expire records silently. Retention actions are themselves logged as events.

3. The Three-Layer Architecture

DDRP, CAAP, and DAS are not versions of the same system. They are three adjacent layers in a single accountability architecture, each separated from the others by a published interface contract. The sequence is fixed: DDRP must complete its work before CAAP can operate. CAAP must produce a complete event log before DAS accepts it. The pipeline flows in one direction.

LAYER C	**DAS**	Documentary Accountability Substrate — where artifacts and event logs persist, for how long, and across what organizational scope
INTERFACE	**Ledger Contract v1.0**	Hash-bound submission schema. Append-only write. DAS consumes. CAAP produces. No shared state.
LAYER B	**CAAP**	Contextual Accountability Attribution Protocol — who acted, when, under what authority, against which artifact
INTERFACE	**Artifact Contract v1.0**	Hash-bound JSON schema. Read-only boundary. CAAP consumes. DDRP produces. No shared state.
LAYER A	**DDRP**	Deterministic Document Review Protocol — what obligations exist, whether they are structurally resolved, and whether the record is inspectable

3.1 Directional Flow

The interface between DAS and the layers it depends on is strictly directional:

Direction	What Moves
DDRP → CAAP	Immutable artifact (hash-stable JSON). Read-only. CAAP consumes it.

Direction	What Moves
CAAP → DAS	Append-only event log plus artifact hash reference. Hash-verified on receipt.
DAS → Ledger	Accepted submissions written to append-only ledger. No modification path exists.
DAS → CAAP	Nothing. DAS has no write access to CAAP event logs.
DAS → DDRP	Nothing. DAS has no write access to DDRP artifacts.
DDRP / CAAP → DAS awareness	None. Both layers are explicitly designed to be unaware of DAS internal operation.

DDRP and CAAP are explicitly designed to be unaware of DAS. This is not an oversight. It is the architectural guarantee that neither extraction nor attribution is contaminated by persistence concerns. A DDRP implementation does not need to know whether DAS exists to produce a valid artifact. A CAAP implementation does not need to know how DAS stores events to produce a valid event log.

4. What DAS Enforces

DAS enforces four structural guarantees across the organizational record. Each is a ledger-level property, not a feature of DDRP or CAAP individually.

4.1 Artifact Retention

DDRP artifacts are immutable within a single extraction cycle. DAS extends that immutability across time. Once a DDRP artifact is accepted into the ledger, it cannot be removed, overwritten, or silently expired. Retention rules govern how long a record must be held, but the decision to retire a record is itself logged as an event in the ledger. There is no silent deletion.

4.2 Cross-Artifact Queryability

An organization that has run DDRP on fifty contracts and logged CAAP events against all of them has fifty isolated proofs. DAS makes those proofs queryable as a corpus. The same actor appearing across multiple artifacts can be located. The same obligation type recurring across successive document versions can be traced. Authority chains spanning multiple instruments can be reconstructed. The ledger is the index that makes individual proofs collectively useful.

4.3 Hash-Chain Integrity

DAS does not accept the declared artifact_hash on trust. It recomputes the hash against the received payload using the same canonical serialization rules the DDRP Artifact Interface Contract defines: UTF-8 encoded, lexicographically sorted keys, no insignificant whitespace. A mismatch is treated as evidence of mutation and results in rejection. The rejected submission is logged with its declared hash and the computed hash, creating a tamper-evidence record even for submissions DAS refused.

4.4 No Write-Back

DAS receives. It does not modify. It has no write path to DDRP artifacts or CAAP event logs. It cannot add fields, correct errors, or append notes to records it has accepted. If a DDRP artifact is incorrect, the resolution is a new DDRP extraction producing a new artifact with a new hash. DAS accepts the new artifact as a separate ledger entry. The original remains permanently associated with whatever CAAP events were logged against it.

5. The Interface Contract

The only formal connection between CAAP and DAS is the DAS Ledger Interface Contract v1.0. This document defines the schema a CAAP submission must satisfy, the hash semantics DAS uses to verify integrity, and the append-only rules DAS enforces on every accepted record.

5.1 Required Submission Fields

Before DAS accepts a CAAP submission, it validates the following fields:

Field	Purpose
submission_schema_version	Structural compatibility check. DAS rejects unsupported versions without coercion.
artifact_hash	SHA-256 hash of the DDRP artifact this event log was produced against. DAS recomputes and must match.
log_hash	SHA-256 hash of the complete CAAP event log payload. Verified on receipt.
document_id	Stable identifier of the source document. Used for cross-artifact indexing.
submission_timestamp	UTC timestamp of when the CAAP event log was submitted to DAS.
actor_id	Identifier of the submitting actor. Required. Submission without a declared actor is structurally prohibited.
events	Array of CAAP event objects. May be empty at submission; a ledger entry is still created.
retention_class	Document class identifier that maps to a retention policy in the DAS configuration. Required.

5.2 Hash Semantics

DAS verifies two hashes on every submission: the artifact_hash, which must match the DDRP artifact the events reference, and the log_hash, which must match the CAAP event log payload as received. Both are SHA-256 hashes computed over canonical JSON using the same deterministic serialization rules the DDRP Artifact Interface Contract defines. A mismatch on either hash is a rejection. The two checks together guarantee that DAS has accepted the specific artifact the events reference and the specific events that were declared at submission time.

5.3 The Append-Only Rule

Once a submission is accepted into the ledger, it is immutable. DAS enforces this structurally: the ledger has no update operation. New CAAP events against the same artifact are submitted as new entries. Superseded artifacts produce new ledger entries. The previous entry remains permanently in place, associated with the events that were logged against that version of the artifact.

6. What DAS Is Not

Precision about what DAS does requires equal precision about what it does not do.

DAS does NOT...	Why this matters
Extract obligations	Extraction belongs to DDRP. DAS receives artifacts it did not produce and cannot evaluate whether they are correct.
Attribute actions	Attribution belongs to CAAP. DAS receives event logs it did not generate and has no basis to assess whether the declared authority was valid.
Interpret compliance	DAS records that artifacts and events exist in the ledger. It does not assess whether obligations were satisfied or whether actions constituted compliance. That determination belongs to the reviewing authority.
Modify accepted records	The ledger is append-only. DAS cannot correct, extend, or annotate records after acceptance. Corrections require new submissions with new hashes.
Perform probabilistic inference	DAS applies deterministic hash verification. No inference is performed at any point in the submission or retrieval path.
Replace DDRP or CAAP	DAS is not a shortcut to the lower layers. A document that was not processed by DDRP cannot be submitted to DAS as a DDRP artifact.
Provide a compliance dashboard	DAS is a ledger, not a reporting product. Query interfaces and visualization layers are consumers of the ledger, not components of it.

7. Why the Substrate Is a Separate Layer

The instinct when two systems produce outputs is to co-locate their storage: one database, one schema, one administrative boundary. DDRP and CAAP are explicitly designed to resist this instinct, and DAS continues the resistance at the persistence layer. The reason is functional, not aesthetic.

7.1 The Contamination Argument

Every DDRP artifact is a structural observation: this language appeared at these coordinates in this document at this time. It has no author. It has no intent. Every CAAP event log is an attributed action chain: this actor, under this authority, took this action against this obligation. Every DAS ledger entry is a persistence record: this submission was received at this time, its hashes verified, and its contents written to the ledger under this retention class.

These three categories of record answer different questions and are subject to different forms of challenge. A challenge to a DDRP record is a challenge to whether the extraction was correct. A challenge to a CAAP record is a challenge to whether the actor had authority. A challenge to a DAS record is a challenge to whether the submission was received intact and stored without modification. Merging the layers merges these challenges and makes all three harder to resolve.

7.2 The Stability Argument

DDRP is finalized extraction infrastructure. CAAP is finalized attribution infrastructure. Neither changes to accommodate DAS logic. DAS can evolve its retention policies, query interfaces, submission schema versioning, and UI layer without touching the protocols on which it depends. The historical artifact and event log record remains clean regardless of how the ledger layer develops. If DAS were modified to accommodate new functionality, every record in the ledger generated before the modification would have been accepted by a different version of the substrate. The evidentiary chain for those records would require qualification.

By treating DDRP and CAAP as stable and building DAS as a separate consumer, the historical record remains clean. DAS can evolve without disturbing the foundation on which it rests.

7.3 The Scope Argument

DDRP answers the structural question. CAAP answers the accountability question. DAS answers the organizational question: where does the record live, how long must it be kept, and how can it be reconstructed when required. These are not sub-problems of a single larger question. They arise at different points in the lifecycle of a professional obligation, require different evidence, and implicate different parties.

A DDRP operator reviewing a contract is asking what the document requires. A CAAP reviewer examining an event log is asking who acted and under what authority. A DAS administrator responding to a regulatory inquiry is asking which version of the artifact was in the ledger at a specific date, whether the hash chain is intact, and whether the retention policy was followed. The three-layer architecture reflects that sequence structurally.

8. Summary

DDRP, CAAP, and DAS address three adjacent but distinct problems in the management of professional obligations under document.

Protocol	Question Answered
DDRP	What obligations exist in this document, were they detectable, and were they structurally resolved?
CAAP	Who acted on the obligations in this artifact, when, under what authority, and against which version of the record?
DAS	Where does the record persist, for how long, across what organizational scope, and can it be reconstructed intact when required?

DDRP produces the evidentiary foundation. CAAP produces the provenance chain. DAS produces the organizational record that holds both across time. None is sufficient without the others in environments where the what, the who, and the where of obligation management are all subject to external scrutiny.

The relationship between all three layers is enforced by the same three constraints that govern the DDRP-to-CAAP boundary: the interface contract defines the handoff, the hash mechanism verifies integrity, and the separation rule prevents any layer from contaminating the others. Together these constraints produce a three-layer accountability architecture in which every claim is traceable to a structural observation, an attributed action, or a verified persistence event — and none of the three are conflated.

> *DDRP proves the record was made. CAAP proves the record was acted on. DAS proves the record survived.*