DDRP + CAAP

A Two-Layer Accountability Stack

Comparative Market Analysis

Quantum Inquiry | quantuminquiry.org

1. The Stack

Before comparing the DDRP + CAAP stack to commercial systems, the stack itself needs to be stated precisely. It is a two-layer architecture separated by a published interface contract. Each layer answers a different question and is designed to remain ignorant of the other's internal operation.

LAYER B	**CAAP**	Attribution — who acted, when, under what authority, against which artifact version
CONTRACT	**Interface Contract v1.0**	Hash-bound JSON schema. Read-only boundary. No shared state between layers.
LAYER A	**DDRP**	Extraction — what obligations exist, were they detectable, were they structurally resolved

Protocol	The Question It Answers
DDRP	What obligation-creating language appeared in this document, was it detectable at the time of production, and was each obligation structurally resolved or left open?
CAAP	Who acted on the obligations in this artifact, when, under what authority, and was the action taken against a cryptographically verified version of the record?

These are sequential questions, not simultaneous ones. DDRP must complete its work before CAAP can operate. CAAP cannot exist without a completed, hash-stable DDRP artifact. The separation is enforced architecturally, not by convention.

2. The Commercial Landscape

Three categories of commercial software currently address some portion of the problem space that DDRP and CAAP jointly cover: Contract Lifecycle Management (CLM) platforms, Governance, Risk, and Compliance (GRC) platforms, and regulatory telemetry frameworks. Each approaches the problem from a different direction. None implements both layers with the architectural discipline that DDRP and CAAP require.

2.1 Contract Lifecycle Management Platforms

What They Are

CLM platforms — including Ironclad, Icertis, Agiloft, JAGGAER, Concord, and HyperStart — manage the end-to-end lifecycle of contracts from drafting through execution, obligation tracking, and renewal. They are the commercial systems most commonly deployed in environments where document obligations must be tracked and acted upon.

What They Get Right

- Audit trail generation: Most CLM platforms record who accessed a contract, who made changes, who approved it, and when. This is a form of attribution.
- Obligation tracking dashboards: Platforms like JAGGAER assign obligations to named individuals and send alerts on upcoming or overdue items.
- Version control: CLM systems maintain historical versions of documents and track changes between versions.
- Role-based access and delegation of authority: Some platforms enforce approval thresholds and record who was authorized to act at what level.

The Structural Gap

CLM platforms merge extraction and attribution into a single mutable system. This is the architectural failure that DDRP and CAAP are specifically designed to prevent.

- Extraction is probabilistic: Modern CLM platforms use AI to identify obligations. The extraction result is not deterministic, not hash-stable, and not reproducible. The same document run through the system twice may produce different outputs.
- The record is mutable: Obligations extracted by a CLM platform can be edited, relabeled, reassigned, or deleted within the same system. There is no cryptographic separation between the fact of extraction and the act of attribution.
- No authority basis requirement: CLM platforms track approval status — who clicked approve — but they do not structurally require a declaration of the authority under which the approval was made. Approval and authorization are treated as the same thing.
- No hash-binding of action to artifact version: When a CLM user marks an obligation as resolved, the system does not cryptographically verify that the resolution was recorded against the same version of the document from which the obligation was extracted. Artifact mutation is possible between extraction and resolution.
- No append-only enforcement at the protocol level: CLM systems permit administrators to modify or delete records. The audit trail is a feature, not a structural guarantee.

> *CLM platforms answer the question 'what is the current state of this contract?' DDRP + CAAP answers the question 'what was the state of this document at the moment of extraction, and who acted on that specific record?'*

2.2 Governance, Risk, and Compliance Platforms

What They Are

GRC platforms — including ServiceNow GRC, RSA Archer, MetricStream, and Hyperproof — manage compliance obligations at the control and policy level. They are designed for organizations that must demonstrate ongoing adherence to regulatory frameworks such as SOC 2, ISO 27001, HIPAA, GDPR, and NIST. They are the primary commercial systems for the kind of accountability chain that CAAP formalizes.

What They Get Right

- Control-level obligation tracking: GRC platforms maintain a registry of compliance controls, assign owners, and track testing status — a functional analog to CAAP's event log for the compliance domain.
- Evidence attachment: Platforms like Archer allow evidence artifacts to be attached to control records, creating a chain from obligation to proof.
- Workflow-enforced approval chains: ServiceNow GRC automates escalation and approval routing, creating a documented path from risk identification to remediation.
- Multi-framework mapping: Controls can be tagged to multiple regulatory frameworks simultaneously, enabling a single compliance record to satisfy several obligations.

The Structural Gap

GRC platforms operate at the control level, not the document obligation level. They assume that the obligations being tracked are already known, defined, and entered into the system by a human or a connected data source. They do not perform deterministic extraction from source documents. And they share the same mutable-system problem as CLM platforms.

- No deterministic extraction layer: GRC platforms do not scan source documents for obligation-creating language. Obligations are entered manually or imported. There is no equivalent of DDRP's lexical scan.
- AI-assisted evidence collection is probabilistic: Platforms like Archer and Hyperproof use AI to collect and categorize evidence. This is probabilistic inference, not deterministic observation.
- Evidence is stored in the same mutable system: A control record in ServiceNow GRC can be modified after the fact. The evidence attachment is not hash-bound to a specific version of the control record.
- No authority basis field: GRC platforms track who approved a control test result, but they do not require a declaration of the legal or organizational authority basis for that

approval. Approval is recorded as a workflow step, not as a structured authority assertion.

- Append-only is not structurally enforced: GRC audit logs are designed for review, not for evidentiary proof of immutability. Administrators retain modification rights.

2.3 Regulatory Telemetry Frameworks (FedRAMP / OSCAL)

What They Are

FedRAMP RFC-0024 and the OSCAL standard represent the most architecturally sophisticated commercial-adjacent framework in this space. RFC-0024 explicitly mandates the transition from human-written compliance narratives to machine-generated deterministic telemetry. OSCAL provides the structured data model for that telemetry.

What They Get Right

- Deterministic telemetry mandate: RFC-0024 explicitly prohibits generative AI outputs as evidence. Only machine-generated, reproducible data from authoritative sources qualifies. This aligns with DDRP's no-probabilistic-inference rule.
- Hash-bound artifact references: OSCAL assessment results reference specific, versioned artifacts. This is the closest commercial analog to DDRP's artifact_hash mechanism.
- Structured evidence chain: OSCAL's layered model — catalog, profile, SSP, assessment plan, assessment results, POA&M — creates a traceable chain from control definition to evidence. This parallels the DDRP-to-CAAP pipeline.
- Explicit prohibition of narrative compliance: RFC-0024's framing — moving from 'human-written narratives' to 'machine-generated deterministic telemetry' — is the same epistemological position that underlies both DDRP and CAAP.

The Structural Gap

OSCAL merges Layer A and Layer B into a single format. The same schema that defines what controls must be satisfied also records whether they were satisfied and by whom. There is no architectural separation between the extraction of obligations and the attribution of actions against them.

- No separation of extraction from attribution: OSCAL's System Security Plan defines control requirements and records implementation status in the same document. The boundary between normative obligation and attributed action is a field distinction, not a system boundary.
- Domain-constrained: OSCAL is purpose-built for federal cloud security authorization. It has no generalized application to contract obligations, regulatory documents, or professional instruments outside the FedRAMP context.
- No authority basis field in CAAP's sense: OSCAL records who submitted an assessment result, but it does not require a structured declaration of the authority under which the action was taken. Submission identity and authority basis are different claims.

- Adoption gap: As of early 2026, FedRAMP processed over 100 Rev5 authorizations without a single OSCAL submission. The framework exists but is not yet operationally standard.

3. Comparative Matrix

The following matrix compares the DDRP + CAAP stack against the three commercial categories across the dimensions that define the architectural gap. A check indicates the property is present as a structural guarantee, not merely as a feature or optional configuration.

Property	DDRP+CAAP	CLM Platforms	GRC Platforms	FedRAMP/ OSCAL	DocuSign	eSignature
Deterministic obligation extraction	Yes	No (AI/ML)	No (manual entry)	Yes (telemetry)	No	No
Hash-stable immutable artifact	Yes	No	No	Partial	No	No
Layer separation (extraction / attribution)	Yes	No	No	No	N/A	N/A
Append-only event log (structural)	Yes (CAAP)	No	No	No	No	No
Authority basis required per action	Yes (CAAP)	No	No	No	No	No
Hash-bound action to artifact version	Yes (CAAP)	No	No	Partial	No	No
No probabilistic inference in verification	Yes	No	No	Yes (mandated)	N/A	N/A
Domain agnostic	Yes	Yes	Yes	No (FedRAMP only)	Yes	Yes
Open protocol / non-commercial	Yes	No	No	No (regulatory)	No	No

Green = structural guarantee present. Red = property absent. Blue = partial or conditional.

4. The Structural Gap No Commercial System Fills

The matrix above identifies the dimensions on which DDRP + CAAP diverges from the commercial stack. But the most important gap is not any single property. It is the gap produced by their combination.

Every commercial system in this analysis can produce an audit log. None of them can produce an audit log in which every entry is cryptographically bound to the specific version of the document from which the underlying obligations were deterministically extracted, with a required declaration of the authority under which the action was taken, in an append-only structure that cannot be retroactively modified by any actor including administrators.

> *CLM and GRC platforms record that something happened. DDRP + CAAP proves what happened, to which record, at what time, and under whose authority — in a form that cannot be altered after the fact.*

This gap matters in three specific scenarios:

4.1 Legal Dispute Over Whether an Obligation Was Reviewed

In contract litigation, the question is often not whether an obligation existed but whether the responsible party knew it existed and took action. A CLM audit trail can show that a user accessed the contract. It cannot show that a specific obligation was identified by a deterministic process, that a named actor reviewed that specific obligation entry, that the review was recorded against a hash-verified version of the artifact, and that the actor declared the authority basis for their resolution decision.

DDRP produces the first three elements. CAAP produces the fourth. Together they constitute a provenance chain that a CLM audit trail cannot replicate because CLM systems do not enforce the separation between the record of what was found and the record of what was done.

4.2 Regulatory Audit Requiring Reconstructibility

Regulators in high-assurance domains increasingly require that compliance demonstrations be reconstructible, not merely assertible. FedRAMP RFC-0024's explicit rejection of 'human-written narratives' and 'machine-generated probabilistic text' as valid compliance evidence is the clearest statement of this requirement in current regulatory frameworks.

GRC platforms can produce reports showing that controls were tested. They cannot produce a deterministic, hash-verified record proving that the specific version of the control definition in effect at the time of testing was the version the tester acted against. CAAP's hash-binding requirement closes this gap directly.

4.3 Multi-Party Accountability Under a Shared Document

When multiple parties act on obligations in the same document — a common scenario in complex contracts, regulatory submissions, and multi-agency federal procurement — the

question of who was responsible for which obligation under what authority becomes legally and operationally critical.

CLM and GRC platforms handle multi-party workflows through assignment and approval routing. They do not require each actor to declare the specific authority basis for their action, and they do not maintain a per-obligation, append-only event log that is cryptographically bound to a single version of the source artifact. CAAP does.

5. Where Each System Belongs

The analysis above is not an argument that CLM and GRC platforms are inadequate. They are purpose-built for workflow management, organizational visibility, and operational efficiency. They serve those purposes well. The argument is that they are architecturally insufficient for environments where procedural care and authority chain must be provable, not merely reportable.

Environment	Appropriate System
Contract drafting, negotiation, approval routing, renewal tracking	CLM platform (Ironclad, Icertis, JAGGAER)
Enterprise risk management, policy compliance, control testing across frameworks	GRC platform (ServiceNow GRC, RSA Archer)
Federal cloud security authorization under FedRAMP	FedRAMP/OSCAL framework
Electronic signature with timestamped attribution	DocuSign or equivalent eSignature
Deterministic obligation extraction from any professional document with full audit provenance	DDRP
Append-only, hash-bound, authority-declared action logging against a DDRP artifact	CAAP
Environments requiring reconstructible proof of both what was found and who acted on it under what authority	DDRP + CAAP stack

The final row is the only cell in this table that no commercial system currently fills. That is the market position of the DDRP + CAAP stack.

6. Summary

The commercial obligation management landscape divides into systems that track workflow state (CLM), systems that track control compliance (GRC), and frameworks that mandate deterministic telemetry in specific regulatory contexts (FedRAMP/OSCAL). Each category fills a real operational need.

The gap across all three is the same: none enforces the architectural separation between the deterministic extraction of what a document obligates and the attributed, hash-bound, authority-declared record of what was done about it. This gap is not a missing feature that could be added to an existing platform. It is a consequence of architectural choices — specifically, the decision to merge extraction and attribution into a single mutable system — that are now load-bearing in those platforms.

DDRP closes the extraction gap. CAAP closes the attribution gap. The interface contract between them enforces the separation that makes both claims independently verifiable. The combination produces an accountability architecture that the commercial stack, as currently constituted, does not offer.

> *The commercial stack is built for workflow. DDRP + CAAP is built for proof.*