

ODA3 INSTITUTE

GOV — GOVERNANCE & DECISION RECORD

Normative Errata: UAIF™ v1.0

TCR-STD-002 / TCR-STD-003

Document ID: ODA3-2026-08-ERR-UAIF-001

Classification: Standards Development — Governance Record

Licensed under GAISSEF Ecosystem Licence (GEL v1.0)

Status: CLOSED

CLOSED — INTEGRATED INTO UAIF™ v1.0 FINAL PUBLICATION

- This is a normative correction, not a market-version increment. The market-facing version remains UAIF™ v1.0.
- Disposition, closure date, and residual scope are recorded in §9 Closure Record.

Authority chain governing this errata: TCR-STD-002 (semantics & conformance architecture) → TCR-STD-003 (schema specification) → canonical JSON Schema (uaif-core-v1.0.json) → validation fixtures (fixtures.json, proved via run_conformance.py, 15/15 passing) → COM-012 and other practitioner material.

Reconciliation basis: four intertwined defects identified during multi-pass architectural review — see decision log below.

1. Defects Corrected

- record_type was assigned to L4 while required by Core (which prohibits L4) — direct conformance contradiction.
- Observation was defined in TCR-STD-003's record_type enum but never defined semantically in TCR-STD-002 — undefined normative value.
- severity_justification was unconditionally required by TCR-STD-003 but never defined in TCR-STD-002, and duplicated authority with severity_rationale.
- Core prohibits L3 while every Incident record conditionally required the L3 field realized_harm_categories — a Core Incident could not conform.

A fifth, structural gap was identified independently: profile prohibitions were documented in prose but not machine-enforced by the schema (declaring a property optional does not prohibit it in JSON Schema).

2. Locked Decisions

Element	Final decision
record_type	Reassigned L4 → L0
record_type enum	Incident, Vulnerability (Observation removed)
Pre-classification signals	Remain in upstream detection/case-management workflow; not a UAIF record type
profile	New required L0 field. Base values: Core, Enterprise, Regulatory
extensions	New optional L0 array, additive only. Values: SOC_SIEM, AI_Security
L0 name	"Incident Identity & Workflow" → "Record Identity & Workflow"
L4 name	"Incident vs Vulnerability" → "Incident–Vulnerability Relationship"
realized_harm_categories	Stays in L3; required for Incident only under profile ∈ {Enterprise, Regulatory}; prohibited under Core
severity_justification	Deleted from schema and all documentation
severity_rationale	Confirmed sole canonical severity explanation
Core profile enforcement	Machine-enforced prohibition of L3/L4 fields via allOf/not/anyOf
Extension enforcement	Machine-enforced via allOf/if/contains/else, gated explicitly on extensions to avoid absent-property ambiguity
classification_status (Observation successor concept)	Deferred — not introduced in v1.0; flagged for a future minor version once vocabulary is fully designed
Incident definition	Reworded to drop "confirmed impact" (see §5)

3. Exact Replacement Text — TCR-STD-002

§1.2 Primary Outputs table

No new row for severity_justification (it is not adopted). Table otherwise unchanged.

§2 UAIF 7-Layer Architecture — replace the L0 and L4 rows

L0 (was “Incident Identity & Workflow”):

L0 — Record Identity & Workflow

- Record identity, classification, conformance-profile declaration, and lifecycle/workflow state.
- Fields: incident_uuid, record_type, profile, extensions, reporting_timestamp, reporter_confidence_level, remediation_status, deduplication_hash

L4 (was “Incident vs Vulnerability”):

L4 — Incident–Vulnerability Relationship

- Links classified incidents to associated vulnerabilities; captures exposure context.
- Fields: linked_vulnerability_id, exposure_duration_hours

New §2.1 — Record Type Semantics (insert after the layer table)

record_type SHALL take one of two normative values:

- Incident — a realized event involving an AI system that has produced, or is reasonably assessed to have produced, an adverse operational, security, safety, privacy, societal, or other relevant effect.
- Vulnerability — a weakness, exposure, or susceptible condition that could contribute to an adverse event but is not itself evidence that such an event has occurred.

Unclassified signals, alerts, telemetry items, or investigative leads are not a UAIF record_type. They remain in the originating detection or case-management workflow until sufficient information exists to classify the record as Incident or Vulnerability. Observation is not a UAIF record_type value.

New §2.2 — Profile and Extension Declaration

Every UAIF record SHALL declare profile, identifying its base conformance profile. The normative base values are Core, Enterprise, and Regulatory, where Enterprise extends Core with L3/L4 fields and Regulatory extends Enterprise with extended L6 obligations.

A record MAY additionally declare one or more extensions. UAIF v1.0 defines two extension identifiers: SOC_SIEM (approved STIX/ATLAS-mapping and security-tooling fields) and AI_Security (selected L5 generative/agentive/control-plane fields). Extensions are additive and SHALL NOT reduce or override the requirements of the selected base profile. Extensions are independent of base profile: any base profile MAY declare either, both, or neither extension.

§3 Adoption Profiles and Conformance — replace table

Type	Name	Layer coverage / mandatory fields	Prohibited fields
Base profile	Core	L0, L1, L2, L6 baseline; for record_type = Incident, the L2 severity bundle only	L3, L4 fields in primary payload
Base profile	Enterprise	Core + L3, L4; for record_type = Incident, L2 bundle + realized_harm_categories	—
Base profile	Regulatory	Enterprise + extended L6 obligations	—
Extension	SOC_SIEM	atlas_technique_id and approved security-tooling mapping fields	Requires explicit extensions declaration
Extension	AI_Security	Selected L5 fields (agent_escalation_chain, mcp_endpoint_ids, oauth_pivot_chain, policy_enforcement_point, ai_system_type, non_adversarial_failure_type)	Requires explicit extensions declaration

Remove the former “SOC/SIEM” and “AI Security Extension” rows as peer profiles — they are now extensions, not base profiles.

Appendix B — replace rule 2 and add new rule

Rule 2 (revised) and Rule 2a (new)

- Rule 2 (revised): record_type = Incident (any profile) → severity_analytical_score, severity_presentation_score, severity_level, severity_rationale SHALL be present.
- Rule 2a (new): record_type = Incident AND profile ∈ {Enterprise, Regulatory} → realized_harm_categories SHALL additionally be present.

§7 Governance table — no change required; version display rule already correctly states UAIF™ v1.0 in market-facing text.

4. Exact Replacement Text — TCR-STD-003

- §1 Status and Authority — no change; conformance to TCR-STD-002 already stated and now satisfied.
- §2 7-Layer Architecture Alignment — update L0/L4 rows to match TCR-STD-002 §2 exactly.
- §3 Profile Conformance — replace table to match TCR-STD-002 §3 (Core/Enterprise/Regulatory as base profiles; SOC_SIEM/AI_Security as extensions).

§4 Normative JSON Schema — replace in full with uaif-core-v1.0.json

Key changes from the prior version:

- record_type enum: ["Incident", "Vulnerability"] (Observation removed)
- New required field profile, enum ["Core", "Enterprise", "Regulatory"]
- New optional field extensions, array of ["SOC_SIEM", "AI_Security"]
- severity_justification removed entirely (not in properties, not in required); blocked automatically by unevaluatedProperties: false
- allOf conditional rules replaced per the gate logic in uaif-core-v1.0.json (universal Incident rule; Enterprise/Regulatory harm-categories rule; Core L3/L4 prohibition; SOC_SIEM gate on atlas_technique_id; AI_Security gate on L5 fields)

§5 Conditional Validation Rules table — replace to match Appendix B of TCR-STD-002

Rule	Enforcement
Core profile	L3/L4 fields prohibited regardless of extensions
atlas_technique_id	Requires extensions to contain SOC_SIEM
L5 agentic/control-plane fields	Require extensions to contain AI_Security

- §8 Sample Valid Incident Payload — update to include "profile": "Enterprise" (the existing sample already contains realized_harm_categories, making it an Enterprise-level example, not Core) and remove severity_justification (not present in the corrected schema).
- §9 Invalid Legacy Payload Patterns — add rows for record_type = "Observation" and severity_justification present.
- §10 Regulatory Translation Boundary — no change required.

5. Wording Refinement — Incident/Vulnerability Definitions

Adopted per §3 of this errata (see TCR-STD-002 §2.1). Rationale: “confirmed impact” would exclude Incidents that are correctly classified but still under active investigation, which UAIF already models via remediation_status = Investigating and root_cause_confidence.

6. Deferred Item

classification_status (or equivalent triage/epistemic-state field) was proposed as the correct home for pre-classification uncertainty that Observation had informally served. Not adopted in this errata — the vocabulary and semantics were not fully designed during reconciliation, and introducing an underspecified field during a

normative correction pass would recreate the class of defect this errata exists to close. Flagged for a future UAIF minor version.

7. Deliverables Accompanying This Errata

- uaif-core-v1.0.json — canonical corrected schema
- fixtures.json — 15-case conformance matrix (8 positive, 7 negative)
- run_conformance.py — gate-logic test harness

8. Validation History (Build-Integrity Record)

Pass 1

Hand-implemented gate-logic harness only. 15/15 against the harness's own mirrored rules. Not independently proven against the actual Draft 2020-12 schema (no jsonschema library available in the offline build sandbox).

Independent check (external, jsonschema 4.26.0, Draft 2020-12)

Found 3 of 15 fixtures failing against the real canonical schema despite passing the harness:

- core_plus_soc_siem_valid, core_plus_ai_security_valid, regulatory_incident_valid — failed multipleOf: 0.1 on severity_presentation_score due to binary floating-point representation of decimal fractions (2.8, 5.1, 9.1).
- regulatory_incident_valid additionally had a malformed deduplication_hash (65 characters, not 64).
- Separately flagged: several negative fixtures carried the same 65-character hash defect, meaning they failed for an unintended reason in addition to (or instead of) the architectural rule each was designed to isolate — weakening the negative proof.

Corrections applied in response

- multipleOf: 0.1 removed from severity_analytical_score and severity_presentation_score in the canonical schema. One-decimal ROUND_HALF_UP precision is now a normative semantic/conformance rule (TCR-STD-002 §4.4) checked by dedicated tooling (run_conformance.py::precision_check), not a JSON Schema constraint — avoiding validator-dependent floating-point false negatives in a machine-interchange standard.
- Every fixture's deduplication_hash regenerated as a clean 64-character SHA-256 hex digest, eliminating the unrelated-failure contamination in the negative fixtures.
- Harness re-run post-correction: 15/15 passing, precision-check clean on all fixtures.

15 passed, 0 failed, 15 total

Final acceptance re-run — CLOSED (14 August 2026)

The corrected canonical schema was re-run directly through jsonschema 4.26.0 using Draft 2020-12 semantics. Draft202012Validator.check_schema() passed, and all 15 fixtures produced their expected valid/invalid outcomes (15/15 PASS). The purpose-built run_conformance.py harness independently produced 15/15 PASS, including the separate semantic one-decimal precision check.

The schema/conformance blocker is therefore closed. The corrected TCR-STD-002 and TCR-STD-003 full publication artifacts were rebuilt, rendered and visually inspected, and the final source set was promoted to UAIF™ v1.0 Final Publication. Residual sector calibration, independent inter-rater studies, and broad platform-integration validation remain explicitly outside the claim of specification finality.

9. Closure Record

CLOSURE

- Closure date: 14 August 2026
- Disposition: Normative errata fully integrated into TCR-STD-002, TCR-STD-003, and the canonical JSON Schema.
- Market version: UAIF™ v1.0 (unchanged).
- Publication status after closure: Final Publication v1.0.

Change control: this errata is recorded as a normative correction, not a version increment. Downstream propagation to COM-012 and other practitioner material now follows this errata per the authority chain in the header, unconditionally — no further validation gate is pending.

NOTABLY ABSENT FROM THIS CLOSURE CLAIM

- No assertion of sector-wide empirical calibration.
- No assertion of regulator endorsement.
- No assertion of production deployment at scale.
- No assertion of completed independent inter-rater reliability validation.