Christopher Aaron O'Hara

Udacity MSc Capstone – Reflective Synthesis Paper

February 21, 2026

Assurance-Centered Agentic AIOps: Parallel Contested Orchestration for Industrial DevSecAIOps

Industry Context and Problem Definition

This capstone synthesizes prior projects into an industry-focused AI decision support system for industrial cyber operations in IIoT/OT environments. The target context is a local-cloud manufacturing or critical-infrastructure setting where analysts must assess incidents quickly without violating operational constraints such as uptime, safety controls, and policy compliance. Security decisions must integrate heterogeneous evidence, generate interpretable hypotheses, preserve auditability, and enforce governance rules before any recommendation is operationalized.

The core problem addressed is the tension between two competing objectives: security assurance (containment, escalation, risk reduction) and operations continuity (bounded disruption, service preservation). In real OT systems, this tension appears in almost every high-impact incident decision. Over-rotating toward assurance can create unnecessary downtime; over-rotating toward continuity can miss critical containment windows. The project therefore frames incident triage as a constrained system-design problem instead of as a single-model optimization problem. This framing aligns with cybersecurity governance guidance emphasizing risk-informed and accountable decision making across technical and organizational boundaries (NIST, 2024; Stouffer et al., 2023).

The integrated artifact is designed as a decision pipeline rather than an autonomous responder. Its purpose is to produce auditable, policy-bounded recommendations for human operators.

Data sources: Artifacts from projects: 3 (machine learning), 4 (deep learning), 5 (generative), and 6 (agentic). Figure 1 illustrates the capstone data source pipeline.

Project repository: https://github.com/Ohara124c41/integrated-industrial-application-acaa

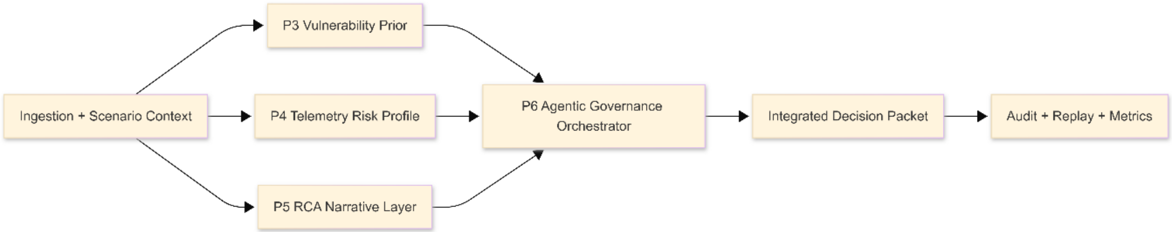
Ingestion + Scenario Context
P3 Vulnerability Prior
P4 Telemetry Risk Profile
P5 RCA Narrative Layer
P6 Agentic Governance Orchestrator
Integrated Decision Packet
Audit + Replay + Metrics

Figure 1 – High Level Data Pipeline

Overview of the Integrated Solution

The implemented artifact is an integrated notebook and runtime workflow that unifies outputs from prior projects and executes contested multi-agent adjudication with governance checks. The run starts by validating artifact availability, then builds unified incident packets (`integrated_packets.jsonl`), executes deterministic-vs-LLM comparison, runs a dual-orchestrator contested workflow, applies optional human-in-the-loop (HITL) overrides under hard safety gates, executes a bounded ReAct-style analyst loop, and materializes outputs into SQLite and Parquet for downstream analytics. Figure 2

The provided run produced 5 integrated packets with 25 fields. System-level baseline metrics were: decision distribution `escalate=4`, `refuse=1`; `policy_pass_rate=0.4`; `mean_hypothesis_confidence=0.8187`; `mean_attack_technique_count=2.0`. Contested orchestration yielded `disagreement_rate=0.2`, `hitl_required_rate=0.4`, and branch selections split between security assurance (3) and operations continuity (2). Safety invariants were preserved (`policy_gate_invariant_violations=0`, `blocked_override_rate=0.0`). ReAct loop outputs remained bounded (`policy_gate_violations=0`) with active model enrichment in this run (`react_llm_active=true`).

The local-cloud plugin extension (separate notebook) validated control-plane logic through plugins (ingest, risk, RCA, agent, governance, storage), with all plugin statuses `ok` and storage materialization to SQLite and Parquet. The architecture approach of the local cloud is shown in Figure 2.

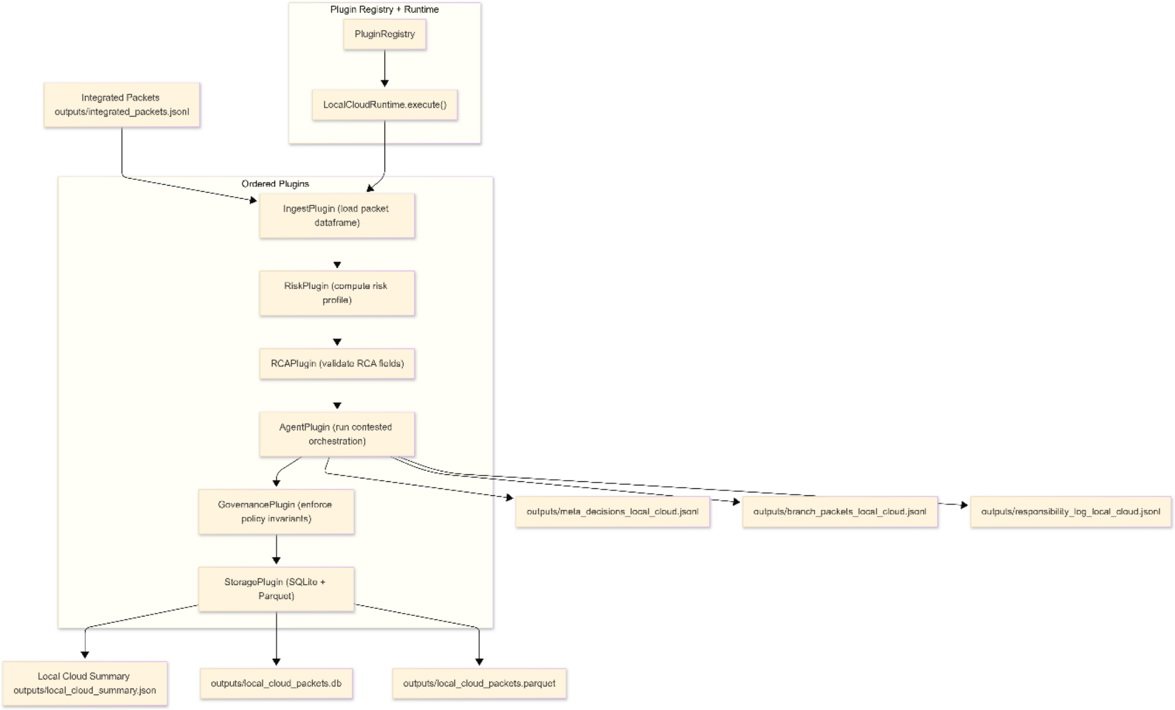
Plugin Registry + Runtime
PluginRegistry
LocalCloudRuntime.execute()
Integrated Packets
outputs/integrated_packets.jsonl
Ordered Plugins
IngestPlugin (load packet dataframe)
RiskPlugin (compute risk profile)
RCAPlugin (validate RCA fields)
AgentPlugin (run contested orchestration)
GovernancePlugin (enforce policy invariants)
StoragePlugin (SQLite + Parquet)
outputs/meta_decisions_local_cloud.jsonl
outputs/branch_packets_local_cloud.jsonl
outputs/responsibility_log_local_cloud.jsonl
Local Cloud Summary
outputs/local_cloud_summary.json
outputs/local_cloud_packets.db
outputs/local_cloud_packets.parquet

Figure 2 – Local Cloud Plugin Architecture

Integration of Prior Projects and Methods

This synthesis integrates four prior capstone stages directly and intentionally:

1. P3 (vulnerability analytics) contributes vulnerability pressure priors from processed NVD/KEV artifacts. These values are injected into

each integrated packet as context priors (`vuln_kev_rate`, `vuln_cvss_mean`) and serve as global risk background rather than packet-discriminative signals.

2. P4 (deep telemetry modeling) contributes telemetry prevalence and representation priors, captured in integrated packet fields such as `telemetry_attack_prevalence`. This layer carries forward model-centric risk context into system-level adjudication.
3. P5 (generative RCA artifacts) contributes generated RCA summaries and failure tags. These fields provide narrative and qualitative context that can be inspected, compared, and audited rather than treated as opaque text output.
4. P6 (agentic governance workflow) contributes structured decision packets and policy findings that act as the immediate upstream operational substrate for contested orchestration and framework validation.

The integration is not a concatenation of files. It is a contract-based synthesis where each prior artifact fills a specific architectural role: vulnerability context, telemetry context, narrative explanation, and governance-state decisions. The runtime enforces a unified schema so downstream orchestration can reason over consistent fields. This follows the same systems principle used in industrial integration pipelines: explicit interfaces reduce ambiguity, improve testability, and support traceability. Figure 3 shows the mid-level of abstraction for system architecture and pipeline.

The contested orchestration layer is the primary system-level contribution. Two orchestrator branches operate on identical integrated evidence but with different objective functions. A meta-orchestrator resolves branch contention under shared constraints and marks incidents requiring HITL review. This structure operationalizes a lead-architect decision pattern: subsystem perspectives are preserved, but final selection must satisfy global constraints, extending the COGENT concurrent-engineering lineage into an agentic cybersecurity context (O'Hara, 2021; O'Hara, 2022).

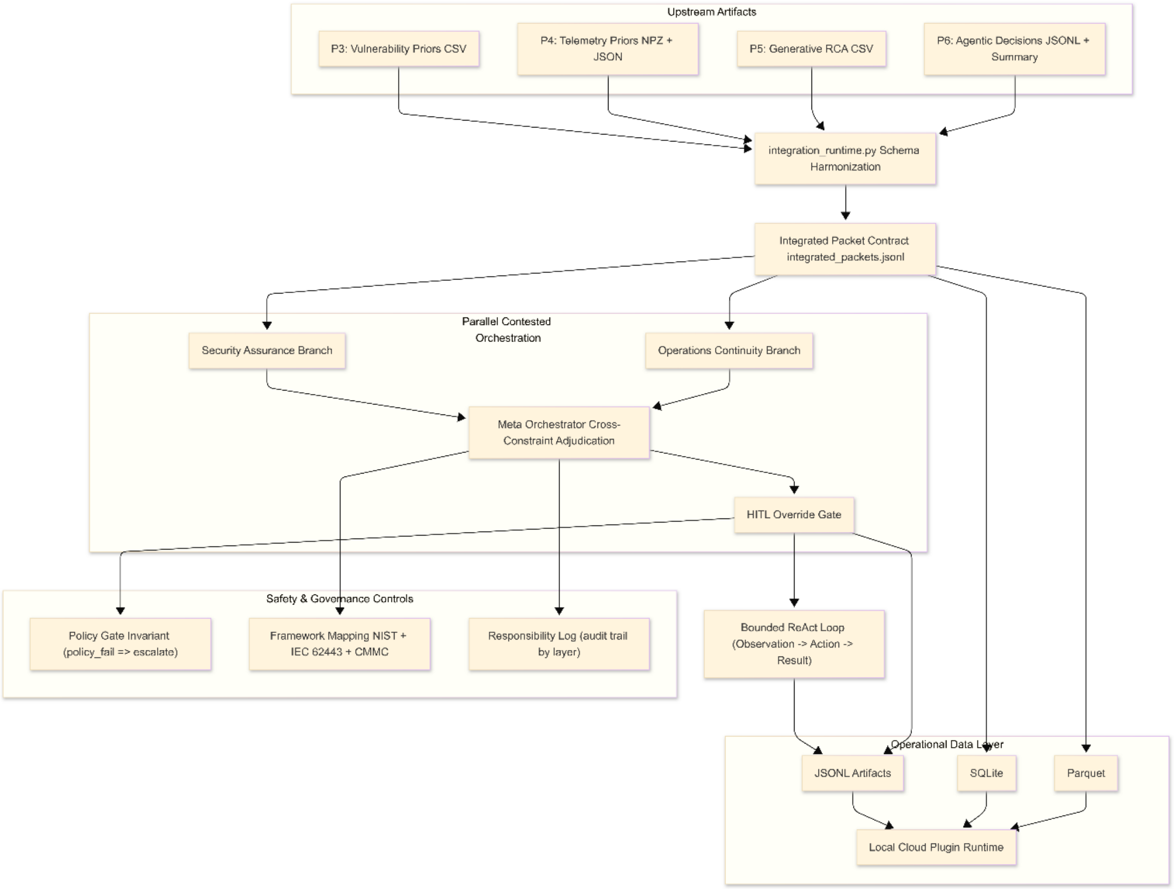
Upstream Artifacts
P3: Vulnerability Priors CSV
P4: Telemetry Priors NPZ + JSON
P5: Generative RCA CSV
P6: Agentic Decisions JSONL + Summary
integration_runtime.py Schema Harmonization
Integrated Packet Contract integrated_packets.jsonl
Parallel Contested Orchestration
Security Assurance Branch
Operations Continuity Branch
Meta Orchestrator Cross-Constraint Adjudication
HITL Override Gate
Safety & Governance Controls
Policy Gate Invariant (policy_fail => escalate)
Framework Mapping NIST + IEC 62443 + CMMC
Responsibility Log (audit trail by layer)
Bounded ReAct Loop (Observation -> Action -> Result)
Operational Data Layer
JSONL Artifacts
SQLite
Parquet
Local Cloud Plugin Runtime

Figure 3 – Mid-level system architecture pipeline.

Although executable artifacts are integrated from P3-P6, the synthesis also builds on earlier capstone reasoning patterns from P1-P2, especially uncertainty handling and governance-oriented interpretation in small-sample settings.

Technical Design Decisions and Tradeoffs

Several design decisions were made to prioritize defensibility over raw complexity.

Decision 1: Deterministic governance with optional LLM enrichment.

The system separates policy outcomes from language enrichment. In deterministic-vs-LLM comparison, governance outputs were stable (`decision_change_rate=0.0`, `policy_change_rate=0.0`, `confidence_change_rate=0.0`), while narrative diversity changed materially (`det_unique_ratio=0.2` vs `llm_unique_ratio=1.0`). This indicates LLM value is constrained to explanatory expressiveness, not control authority. That boundary is deliberate and aligns with reliability expectations in safety-sensitive environments.

Decision 2: Hard policy gates before final recommendation.

A hard invariant enforces escalation on policy-failing incidents. This eliminates a class of unsafe outcomes where branch preferences could otherwise emit permissive recommendations under policy failure. The run-level result (`policy_gate_invariant_violations=0`) confirms this control operated as intended.

Decision 3: Parallel contested orchestration instead of single-branch optimization.

A single branch would be simpler but would hide cross-objective tension. Parallel branches make disagreement explicit (`disagreement_rate=0.2`) and expose when human review is warranted (`hitl_required_rate=0.4`). The tradeoff is orchestration overhead and additional artifacts to manage, but this cost is acceptable for improved transparency and accountability.

Decision 4: Bounded ReAct loop with logged steps.

The ReAct-style loop is bounded and auditable. It records observation-action-result steps but remains policy constrained. This gives interpretable procedural traces without allowing unconstrained multi-step autonomy (Yao et al., 2023).

Decision 5: Statistical diagnostics with explicit small-sample constraints.

EDA and inference were retained for architectural visibility (quality diagnostics, nonparametric tests, bootstrap intervals, PCA/t-SNE). However, interpretation is conservative at `n=5`. For example, chi-square expected-count checks were explicitly flagged as weak, and t-SNE was used only for relative point-distance inspection, not clustering claims.

These decisions collectively prioritize architecture quality attributes: traceability, safety, interpretability, and integration rigor.

Evaluation and Reflection

The integrated workflow met system-level goals: it executed end-to-end, produced observable outputs, and preserved safety invariants while exposing tradeoffs between competing objectives. What worked best was the explicit separation of concerns: deterministic governance controlled outcomes, while LLM components improved explanatory coverage without changing policy gates. The contested architecture produced meaningful branch divergence and HITL triggers, which is preferable to an opaque single-path pipeline.

What did not work at full strength is statistical confidence. With 5 incidents, inferential outputs are useful for directional diagnostics but cannot support strong generalization. Another observed constraint is context-prior invariance: some integrated fields are valuable as global context yet contribute little packet-level discrimination. The primary learning outcome from cross-domain integration is that architecture clarity, contract discipline, and governance observability produce more practical value than isolated model optimization.

Ethical, Governance, and Responsible AI Considerations

The ethical posture of this system is grounded in deployment reality for industrial cybersecurity. Three concerns dominate:

1. Unsafe automation risk

If recommendation logic is not bounded, model outputs can overstep operational authority. The system mitigates this through hard gates, escalation preference under policy failure, and explicit HITL constraints.

2. Instruction-channel misuse and prompt injection

The workflow includes prompt-injection-style scenarios and refusal/escalation pathways. The design prevents unreviewed model text from changing policy outcomes and logs model calls for review.

3. Operational fairness across technical cohorts.

Fairness is screened across zone cohorts (OT vs IT) using AIF360-compatible parity metrics. In this run, statistical parity difference was high (`0.75`) and disparate impact unstable due to small sample (`n=5`). This is treated as a screening signal, not a deployment conclusion, and triggers the need for broader scenario coverage before any fairness claim (Bellamy et al., 2019; Hardt et al., 2016).

Governance alignment is implemented through incident-level reference mapping to NIST CSF, IEC 62443, and CMMC controls. ATT&CK technique identifiers are retained as operational threat context rather than as standalone labels (MITRE, n.d.). Control reference counts do not prove compliance, but they demonstrate traceability coverage and reveal where evidence concentration may create blind spots. This is consistent with governance-first system assurance practices (NIST, 2024; ISA/IEC, n.d.; DoD, 2024).

Limitations and Risks

The integrated system has clear limitations that must be acknowledged for both academic integrity and professional rigor.

Small-sample limitation

Most quantitative outputs are based on 5 integrated incidents. This supports architecture demonstration but not statistical generalization. Confidence intervals are wide, and distributional claims are necessarily tentative.

Context-prior invariance

Some integrated fields (for example global vulnerability priors) are intentionally invariant across packets. These fields are useful as background context but do not discriminate incident-level outcomes. The notebook addresses this by excluding invariant fields from certain correlation analyses and by explicitly reporting information-density risk.

No production telemetry scale

Although the architecture is structured for scale, this artifact is a capstone synthesis, not a production deployment. It does not include full SOC integration, distributed stream processing, or live change-window orchestration.

LLM dependence for narrative quality

Even with policy-safe controls, narrative quality depends on endpoint reliability and prompt discipline. This creates variability that must be managed through logging, deterministic fallbacks, and acceptance criteria.

These limitations do not invalidate the design but they define its current maturity level and guide next-stage engineering priorities.

Professional and Industry Relevance

This project demonstrates competencies expected in system-architecture and AI-governance roles in industrial domains.

First, it shows cross-domain synthesis rather than isolated model work: statistical reasoning, ML/deep context priors, generative RCA artifacts, and agentic orchestration are integrated into one governable pipeline. Second, it demonstrates explicit tradeoff reasoning: security assurance and operations continuity are represented as competing but valid objectives, then adjudicated under system constraints. Third, it emphasizes professional controls: audit artifacts, responsibility logging, framework traceability, HITL boundaries, and reproducible outputs. This directly reflects the architect-lead adjudication pattern documented in prior COGENT research and adapted here for AI-enabled industrial cybersecurity (O'Hara, 2021; O'Hara, 2022).

The local-cloud plugin architecture strengthens professional relevance by showing a migration path from notebook experimentation to control-plane patterns closer to operational engineering practice. This supports portfolio value for architecture oversight, AI assurance, and system integration roles.

Future Extensions or Improvements

Next development should focus on scale, calibration, and assurance:

1. Expand incident/scenario coverage to stabilize fairness and inferential metrics.
2. Introduce calibrated thresholds for branch scoring and HITL trigger policies.
3. Add richer policy packs (asset criticality, maintenance windows, zone-specific exception handling).
4. Extend provenance and lineage tracking for each integrated field to support compliance audits.
5. Benchmark deterministic, LLM-enriched, and hybrid modes over larger replay sets.
6. Integrate external data services through existing storage/interface contracts without changing policy core logic.

These extensions would preserve the current governance-first architecture while increasing operational validity and publication readiness.

References

1. Bellamy, R. K. E., Dey, K., Hind, M., et al. (2019). AI Fairness 360: An extensible toolkit for detecting and mitigating algorithmic bias. *IBM Journal of Research and Development, 63*(4/5). https://doi.org/10.1147/JRD.2019.2942287
2. Department of Defense. (2024). *Cybersecurity Maturity Model Certification (CMMC) 2.0*. https://dodcio.defense.gov/CMMC/
3. Hardt, M., Price, E., & Srebro, N. (2016). Equality of opportunity in supervised learning. *NeurIPS 2016*. https://proceedings.neurips.cc/paper/2016/hash/9d2682367c3935defcb1f9e247a97c0d-Abstract.html
4. ISA/IEC. (n.d.). *ISA/IEC 62443 series of standards*. https://www.isa.org/standards-and-publications/isa-standards/isa-iec-62443-series-of-standards
5. MITRE. (n.d.). *ATT&CK enterprise knowledge base*. https://attack.mitre.org/
6. National Institute of Standards and Technology. (2024). *Cybersecurity Framework (CSF) 2.0*. https://www.nist.gov/cyberframework

7. Stouffer, K. A., Pease, M., Tang, C., et al. (2023). *Guide to Operational Technology (OT) Security (NIST SP 800-82r3)*. https://doi.org/10.6028/NIST.SP.800-82r3
8. O'Hara, C. (2021). *COGENT: Concurrent Generative Engineering Tooling: Enabling Cross-Functional Teams in Architecture Design for Space Subsystems*. Technische Universiteit Eindhoven.
9. O'Hara, C., Menu, J., & Van den Brand, M. (2022). COGENT: A concurrent engineering and generative engineering tooling platform. In *2022 IEEE International Systems Conference (SysCon)* (pp. 1-8).
10. Yao, S., Zhao, J., Yu, D., et al. (2023). ReAct: Synergizing reasoning and acting in language models. *ICLR 2023*. https://arxiv.org/abs/2210.03629