Master of Science Thesis Report

Assurance-Centered Agentic AIOps for Industrial DevSecAIOps: Parallel Contested Orchestration for Local-Cloud IIoT/OT Cybersecurity Decision Support

Christopher Aaron O'Hara
February, 2026
Institute of Artificial Intelligence and Technology

UDACITY

\WOOLF UNIVERSITY/

Date March, 2026

Contact address

Udacity Institute of AI and Technology
Department of Mathematics and Computer Science
Artificial Intelligence
2440 West El Camino Real
Suite 101
Mountain View, CA 94040
USA

Published by

Christopher Aaron O'Hara on behalf of the Udacity Institute of AI and Technology

MSc Report

Abstract

Industrial cyber operations in local-cloud IIoT/OT environments face a persistent systems-level gap: analytical components for detection, triage, and response exist in isolation, yet decision quality degrades at handoff boundaries where competing objectives — security assurance, operational continuity, governance compliance, and human accountability — must be reconciled under uncertainty. This thesis addresses that gap through the design, implementation, and proof-of-architecture evaluation of **Assurance-Centered Agentic AIOps (ACAA)**, a layered decision-support architecture that composes statistical inference, machine learning, deep learning, generative AI, and agentic orchestration under deterministic policy controls, explicit uncertainty handling, and human-in-the-loop authority. The architecture is developed across seven progressive project chapters. A reproducible data workflow (P1) establishes ingestion contracts and provenance discipline over OpenRCA telecom telemetry. Statistical inference (P2) extracts governance-relevant structure from heterogeneous cyber observability data. Leakage-aware machine learning (P3) produces vulnerability prioritization priors over NVD/CISA KEV metadata under extreme class imbalance. Deep learning (P4) applies sequence and representation models to LANL cybersecurity telemetry with controlled ablations and guardrails. A generative RCA layer (P5) synthesizes bounded, confidence-labeled narrative hypotheses for analyst scaffolding. Policy-gated multi-agent orchestration (P6) operationalizes deterministic safety boundaries around adaptive reasoning. Finally, an integrative synthesis (P7) introduces parallel contested orchestration, where dual-branch reasoning (assurance versus continuity) is adjudicated by a meta-orchestrator with explicit human-in-the-loop escalation. The central finding is that decision quality in safety-critical cyber AI systems is a property of composed workflows rather than any single model. Contested orchestration produced meaningful branch differentiation, policy-gate invariants held without violation, and governance traceability was maintained end-to-end across all analytical layers.

Keywords	AIOps, agentic AI, industrial cybersecurity, IIoT/OT, contested orchestration, human-in-the-loop, policy-gated orchestration, root cause analysis, vulnerability prioritization, governance traceability, decision support, assurance engineering, NIST CSF, IEC 62443, sociotechnical systems
Preferred reference	Christopher Aaron O'Hara, Assurance-Centered Agentic AIOps (ACAA): Parallel Contested Orchestration for Industrial DevSecAIOps, Udacity Institute of AI and Technology, MSc Report, March 2026.
Partnership	This project was supported by Udacity Institute of AI and Technology and Woolf University.
Disclaimer Endorsement	Reference herein to any specific commercial products, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the Udacity Institute of AI and Technology and Woolf University. The views and opinions of authors expressed herein do not necessarily state or reflect those of the Udacity Institute of AI and Technology and Woolf University, and shall not be used for advertising or product endorsement purposes.
Disclaimer Liability	While every effort will be made to ensure that the information contained within this report is accurate and up to date, Udacity Institute of AI and Technology makes no warranty, representation or undertaking whether expressed or implied, nor does it assume any legal liability, whether direct or indirect, or responsibility for the accuracy, completeness, or usefulness of any information.
Trademarks	Product and company names mentioned herein may be trademarks and/or service marks of their respective owners. We use these names without any particular endorsement or with the intent to infringe the copyright of the respective owners.
Copyright	Copyright © 2026, Christopher Aaron O'Hara. All rights reserved. No part of the material protected by this copyright notice may be reproduced, modified, or redistributed in any form or by any means, electronic or mechanical, including photocopying, recording, or by any information storage or retrieval system, without proper attribution (citation).

Preface

This report describes the design, architecture, and proof-of-architecture evaluation of an Assurance-Centered Agentic AIOps (ACAA) system for industrial cybersecurity decision support. ACAA is a seven-layer architecture that composes statistical inference, machine learning, deep learning, generative AI, and agentic orchestration under deterministic policy controls, explicit uncertainty handling, and human-in-the-loop authority.

The work addresses a persistent operational gap in local-cloud IIoT/OT environments: analytical components for detection, triage, and root cause analysis exist, but decision quality degrades where competing objectives — security assurance, operational continuity, and governance compliance — must be reconciled under uncertainty. The architecture responds to this gap by treating cybersecurity AI as a constrained systems-engineering problem with explicit controls for traceability, safety, and accountability.

The thesis is organized as a progressive implementation arc across seven project chapters:

1. Reproducible data workflow and provenance discipline
2. Statistical inference for governance-relevant structure
3. Leakage-aware machine learning for vulnerability prioritization
4. Deep learning for IIoT intrusion detection
5. Generative RCA with bounded narrative hypotheses
6. Policy-gated multi-agent orchestration
7. Parallel contested orchestration with meta-adjudication and HITL escalation

The architectural lineage of this work draws from the author's prior experience with concurrent engineering systems, where role-specialized subsystem teams optimize local objectives and an architecture lead reconciles tradeoffs at system level. That pattern transfers directly to cybersecurity operations, where security-assurance and operations-continuity branches must be adjudicated under shared governance constraints.

Christopher O'Hara conducted this project as a Master of Science in Artificial Intelligence thesis through Udacity and Woolf University. The target audience includes practitioners and researchers in AI-assisted cybersecurity operations, industrial control system security, and governance-aware AI system design.

Christopher O'Hara, March 2026

Acknowledgements

I have been fortunate to have the support and guidance of many people during the course of this project. I would like to express my sincere gratitude to my supervisors, colleagues, friends, and family for their invaluable contributions to this work. I first started with Udacity in 2016. In 2017, I was accepted into the first cohort of the Udacity Robotics Nanodegree and Artificial Intelligence Nanodegree programs. There was a large learning curve from the GOFAI era and initial versions of ROS, but I was able to learn the skills and technologies that would later support me through graduate school.

Now, the course has come full circle, and Udacity added an MSc in AI. Initially, I thought I would not learn many additional skills or concepts, since I have been working in AI for many years, but I was happy to find that content had been updated, as well as new content for agentic and generative AI. This MSc thesis document illustrates an integration of several AI use case approaches, to provide solutions to current issues in cybersecurity (a domain that I was a beta-tester and initial mentor for at Udacity). Over the course of all of the nanodegrees, I have submitted more than 140 projects, where each probably had a different mentor to evaluate the work. I was also a mentor for several years, grading over 1700 projects. This is to say, the community has been a big part of my life, with many connections still interacting with me today.

Thank you for contributing to my learning path.

Christopher O'Hara

February 2026

Executive Summary

Industrial cyber operations in local-cloud IIoT/OT environments face a persistent systems-level gap: analytical components for detection, triage, and root cause analysis exist in isolation, yet decision quality degrades at handoff boundaries where competing objectives must be reconciled under uncertainty. Security teams must simultaneously satisfy containment urgency, operational continuity, governance compliance, and human accountability requirements — often with incomplete evidence and under time pressure. Existing tooling provides strong individual components but lacks a defensible mechanism to compose them into governed, auditable decision workflows.

This thesis addresses that integration gap through the design, implementation, and proof-of-architecture evaluation of **Assurance-Centered Agentic AIOps (ACAA)**: a seven-layer decision-support architecture that composes statistical inference, machine learning, deep learning, generative AI, and agentic orchestration under deterministic policy controls, explicit uncertainty handling, and human-in-the-loop authority for high-impact actions.

The architecture is developed across seven progressive project chapters, each contributing a distinct capability to the integrated system:

1. **Reproducible Data Workflow (P1)** establishes ingestion contracts, schema governance, and provenance discipline over OpenRCA telecom telemetry, ensuring that all downstream analytics inherit consistent and auditable preprocessing decisions.

2. **Statistical Inference (P2)** extracts governance-relevant categorical structure from heterogeneous cyber observability data using non-parametric hypothesis testing, effect-size estimation, and assumption diagnostics.

3. **Leakage-Aware ML Prioritization (P3)** produces vulnerability triage priors over NVD/CISA KEV metadata under extreme class imbalance (0.112% prevalence), with a two-track decision separating operationally admissible models from leakage-compromised experimental upper bounds.

4. **Deep Learning for IIoT Intrusion Detection (P4)** applies Transformer-based representation learning to RT-IoT2022 telemetry with controlled single-factor ablations, ensemble evaluation, and shift-monitoring governance gates.

5. **Generative RCA Layer (P5)** synthesizes bounded, confidence-labeled narrative hypotheses from LANL cybersecurity telemetry sequences, with a structured hallucination-risk taxonomy and rubric-based acceptance gating.

6. **Agentic Orchestration (P6)** operationalizes deterministic safety boundaries around five specialized agents with prompt-injection detection, tool allowlists, zone-conduit constraints, and escalation/refusal pathways.
7. **Contested Orchestration (P7)** introduces parallel dual-branch reasoning where a security-assurance branch and an operations-continuity branch independently score the same evidence. A meta-orchestrator adjudicates branch disagreements under shared policy constraints, with explicit human-in-the-loop escalation for contested decisions.

The central finding is that decision quality in safety-critical cyber AI systems is a property of composed workflows rather than any single model. Across evaluation, contested orchestration produced meaningful branch differentiation on all test packets, policy-gate invariants held without violation in both deterministic and LLM-enabled modes, and governance traceability was maintained end-to-end from data ingestion through adjudicated decision packets. Enabling LLM refinement increased explanatory diversity while leaving governance-relevant outcomes unchanged, validating the architecture's separation between adaptive reasoning and deterministic control authority.

The system is evaluated at proof-of-architecture scale and positioned against six governance frameworks — NIST CSF 2.0, NIST SP 800-82r3, IEC 62443, CMMC 2.0, NIST AI RMF 1.0, and IEEE 7000/1012 — as traceability anchors for audit and compliance review. The architectural lineage from COGENT concurrent engineering (2021) to contested orchestration (2026) demonstrates that the core pattern of role-specialized branches adjudicated under cross-cutting constraints transfers across domains.

Limitations in empirical scale (five-scenario pilot runs), live SOC/OT deployment validation, and adversarial stress testing are explicitly bounded and mapped to a forward research agenda. The thesis contributes an architecture-level integration pattern and assurance methodology, not production-rate performance claims.

Glossary

ACAA	Assurance-Centered Agentic AIOps
AI	Artificial Intelligence
AIF360	AI Fairness 360 (IBM fairness toolkit)
AIOps	Artificial Intelligence for IT Operations
AMI	Advanced Metering Infrastructure
ARP	Address Resolution Protocol
ATT&CK	Adversarial Tactics, Techniques, and Common Knowledge (MITRE)
CI	Confidence Interval
CISA	Cybersecurity and Infrastructure Security Agency
CMMC	Cybersecurity Maturity Model Certification
CVE	Common Vulnerabilities and Exposures
CVSS	Common Vulnerability Scoring System
DER	Distributed Energy Resources
DevSecAIOps	Development, Security, and AI Operations
DoD	Department of Defense
EDA	Exploratory Data Analysis
ETL	Extract, Transform, Load
FDR	False Discovery Rate
HITL	Human-in-the-Loop
ICS	Industrial Control Systems
IDS	Intrusion Detection System
IEC 62443	International Standard for Industrial Automation and Control Systems Security
IEEE 1012	IEEE Standard for System and Software Verification and Validation
IEEE 7000	IEEE Standard for Ethically Informed System Design
IIoT	Industrial Internet of Things
IT	Information Technology
JSONL	JSON Lines (newline-delimited JSON format)
KEV	Known Exploited Vulnerabilities (CISA catalog)
LANL	Los Alamos National Laboratory
LLM	Large Language Model
ML	Machine Learning
NIST	National Institute of Standards and Technology
NIST AI RMF	NIST Artificial Intelligence Risk Management Framework
NIST CSF	NIST Cybersecurity Framework
NLP	Natural Language Processing

NVD	National Vulnerability Database
OpenRCA	Open Root Cause Analysis (benchmark)
OT	Operational Technology
P1–P7	Project chapters 1 through 7 of this thesis
PCA	Principal Component Analysis
PR-AUC	Precision-Recall Area Under the Curve
RCA	Root Cause Analysis
ReAct	Reasoning and Acting (agentic prompting pattern)
RF	Random Forest
RMSE	Root Mean Squared Error
ROC-AUC	Receiver Operating Characteristic Area Under the Curve
RT-IoT2022	Real-Time IoT Intrusion Detection Dataset (2022)
SMOTE	Synthetic Minority Over-sampling Technique
SOC	Security Operations Center
SP 800-82r3	NIST Special Publication 800-82 Revision 3 (Guide to OT Security)
SQL	Structured Query Language
TCP	Transmission Control Protocol
t-SNE	t-Distributed Stochastic Neighbor Embedding
UC	Use Case
V&V	Verification and Validation

List of symbols

S_A	Security-assurance branch score
S_B	Operations-continuity branch score
w_i	Objective weight for branch scoring component i
θ	Decision threshold (classification or policy gate)
$\hat{y}$	Predicted label or score
σ	Confidence or uncertainty estimate
n	Sample size
p	Statistical significance level
η^2	Eta-squared effect size
V	Cramér's V effect size

List of Tables

List of Figures

Contents

Introduction

Scope and Terminology

This thesis develops an **Assurance-Centered Agentic AIOps (ACAA)** architecture for industrial cybersecurity decision support in local-cloud IIoT/OT environments. In this document, ACAA means a layered system that combines statistical, machine-learning, deep-learning, generative, and agentic components under deterministic policy controls, explicit uncertainty handling, and human-in-the-loop authority for high-impact actions [1–3].

System Architecture Overview

Architectural Thesis

The central claim is that in safety-critical cyber operations, decision quality is a property of composed workflows rather than any single model. The thesis therefore emphasizes interface contracts, uncertainty disclosure, policy-gated runtime behavior, and audit-grade traceability across all analytical layers [4, 5].

Reader Guide

Chapter 1 presents motivation, problem framing, and related work. Chapters 2 through 8 then develop the implementation arc from reproducible data workflow to integrated contested orchestration.

Claim-Validation Map

GitHub Project and Report

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

Report artifact: https://github.com/Ohara124c41/integrated-industrial-application-acaa/blob/main/Reflective_Synthesis_Paper.pdf

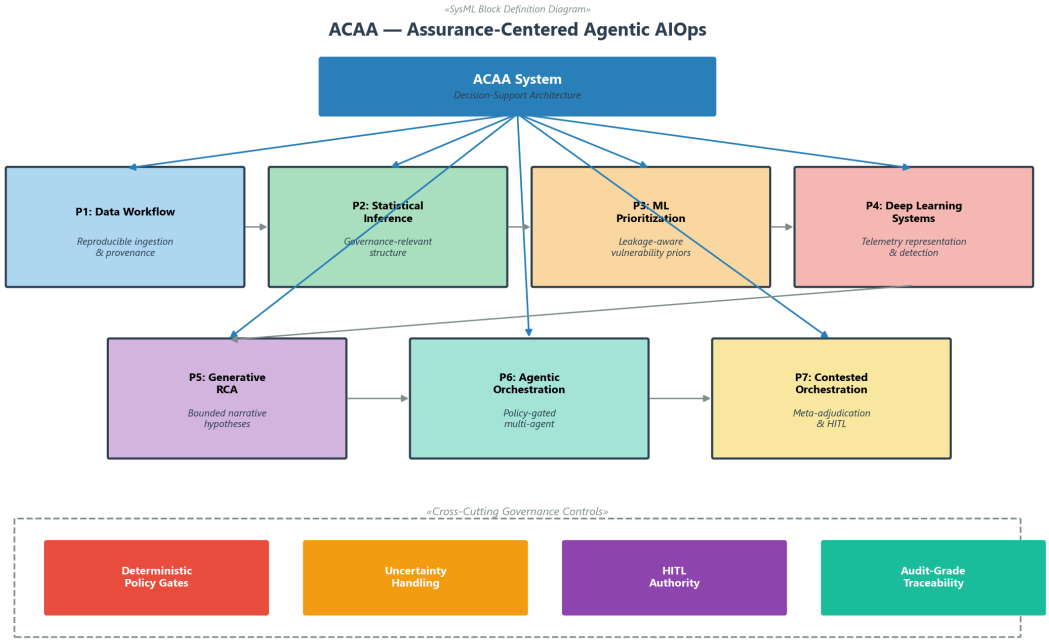

Figure 1: SysML block definition diagram of the ACAA architecture showing seven analytical layers (P1–P7), their composition under the system boundary, and cross-cutting governance controls including deterministic policy gates, uncertainty handling, HITL authority, and audit-grade traceability.

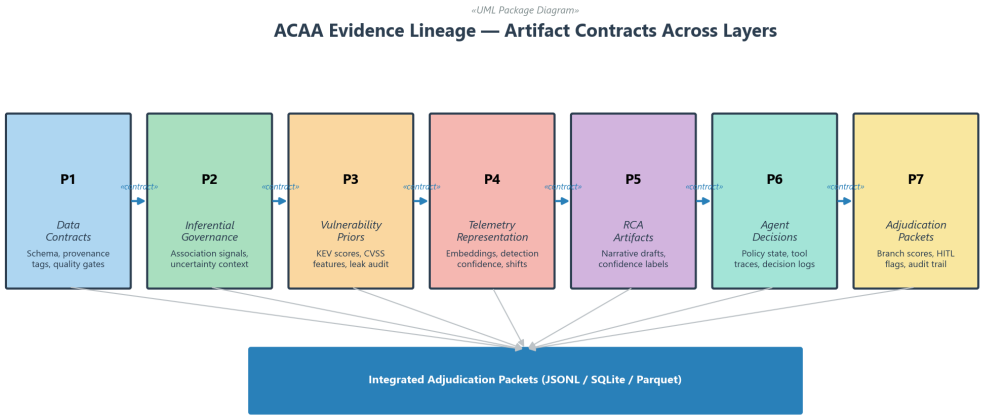

Figure 2: UML package diagram showing the artifact-contract lineage from P1 data contracts through P7 adjudication packets. Each layer emits typed outputs consumed by downstream layers through explicit interface contracts, converging into integrated adjudication packets stored in JSONL, SQLite, and Parquet formats.

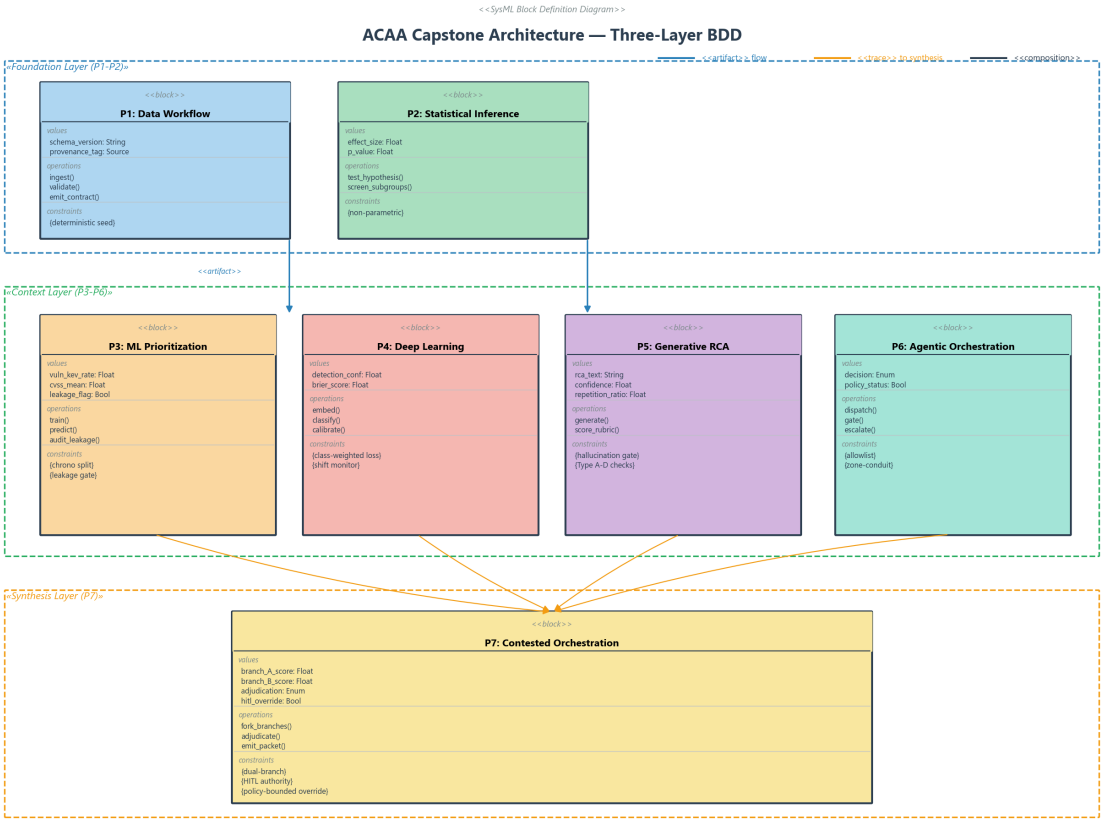

Figure 3: SysML block definition diagram of the ACAA capstone architecture organized into three layers: Foundation (P1–P2), Context (P3–P6), and Synthesis (P7). Each block shows typed value properties, operations, and constraints in SysML compartment notation. Artifact-flow and trace relationships connect upstream layers to the contested orchestration synthesis layer.

Thesis Claim	**Primary Evidence Chapters**	**Current Limitation Boundary**
Decision quality depends on composed workflows, not one model	P1–P7 integration with explicit contracts and policy gates	Integration validated at proof-of-architecture scale, not production-scale replay
Governance-constrained AI can improve cyber decision support	P3–P7 with deterministic gates, audit traces, and HITL routing	Limited scenario volume and no live SOC/OT deployment loop
Contested orchestration is useful for multi-objective operations	P7 branch disagreement and meta-adjudication behavior	Utility evidence currently low-n; requires larger replay corpus for stable effect estimates
Assurance artifacts can be framework-traceable and reproducible	P1 lineage controls, P2 uncertainty handling, P6/P7 audit packets	Traceability is architecture-level evidence, not certification evidence

Table 2: Claim-to-evidence map used to bound interpretation of thesis conclusions.

1 Problem Motivation, Use-Case Framing, and Related Work

1.1 Need and Pain Statement

Industrial cyber operations face a persistent systems problem: teams receive high-volume telemetry and alerts, but operational decisions must still satisfy uptime, safety, and governance constraints. In local-cloud IIoT/OT environments, this produces a recurring failure mode where analytical components exist in isolation, yet decision quality degrades at handoff boundaries [1, 2].

This thesis treats this as an assurance gap rather than a single-model gap. In practical terms, organizations can have detection models, RCA notes, and playbooks, but still lack a defensible mechanism to reconcile competing objectives under uncertainty. This thesis addresses that gap by treating cybersecurity AI as a constrained systems-engineering problem with explicit controls for traceability, safety, and accountability.

This framing also matters at the level of individual prediction targets. A recurring mistake in applied cybersecurity ML is to treat labels as purely technical facts when they often encode organizational and institutional processes. In this thesis, the KEV prioritization problem (Chapter 4) is used as a concrete example: inclusion in the CISA Known Exploited Vulnerabilities catalog is not equivalent to intrinsic technical severity alone. It is better understood as a sociotechnical signal shaped by technical vulnerability properties, exploitation in the wild, defender exposure, patching dynamics, and institutional risk judgment [6, 7]. Making that explicit in the introduction clarifies why machine learning is justified beyond simple thresholding, and why later interpretive analysis must be read as a joint technical–organizational model rather than a narrow score predictor [8].

This chapter contributes to the thesis assurance case as the **architecture-framing layer**: it defines mission constraints, responsibility boundaries, and evidence expectations that govern interpretation of all downstream project chapters.

1.2 GitHub Project and Report

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

Report artifact: https://github.com/Ohara124c41/integrated-industrial-application-acaa/blob/main/Reflective_Synthesis_Paper.pdf

1.3 Operational Environment and System Boundaries

The target environment is a **local-cloud IIoT/OT architecture**: private facility networks, segmented OT zones, internal observability stacks, and controlled conduits between operational and IT services. The intended system role is decision support and authorized validation, not autonomous remediation.

The core boundary conditions are:

- read-mostly operation by default,
- deterministic policy gates for high-impact actions,
- evidence-grounded recommendations with auditable provenance,
- explicit human approval for change execution.

These boundaries are deliberate and domain-driven: in industrial settings, unsafe automation can cause physical downtime, safety incidents, and compliance breaches.

1.4 Use-Case Problem Set

The thesis problem space is organized into six use-case issues:

UC-1: Segmentation and Posture Drift

IIoT/OT zones drift over time due to ad hoc exceptions, temporary routes, and unmanaged policy updates. Teams need continuous validation that trust boundaries remain aligned with design intent.

UC-2: Triage Overload Under Heterogeneous Evidence

Operators must interpret logs, metrics, traces, and ticket context under time pressure. Without structured integration, triage decisions become inconsistent and difficult to defend.

UC-3: RCA Explainability and Evidence Traceability

Predictions alone are insufficient in regulated operations. Analysts require explanations linked to concrete evidence artifacts and reproducible retrieval paths.

UC-4: Prompt/Tool Safety in Agentic Workflows

When LLM-assisted agents are introduced, instruction-channel abuse (for example prompt injection through logs/tickets) and tool misuse become first-order risks.

UC-5: Change-Control and Authorization Integrity

Even correct diagnosis does not justify uncontrolled action. Systems must encode approval gates, deny-by-default tooling, and scope-limited execution surfaces.

UC-6: Auditability, Replay, and Governance Defensibility

Post-incident review requires reconstructable decision history: who/what decided, on what evidence, under what policy rules, and with what uncertainty.

1.5 System Context and Boundary View

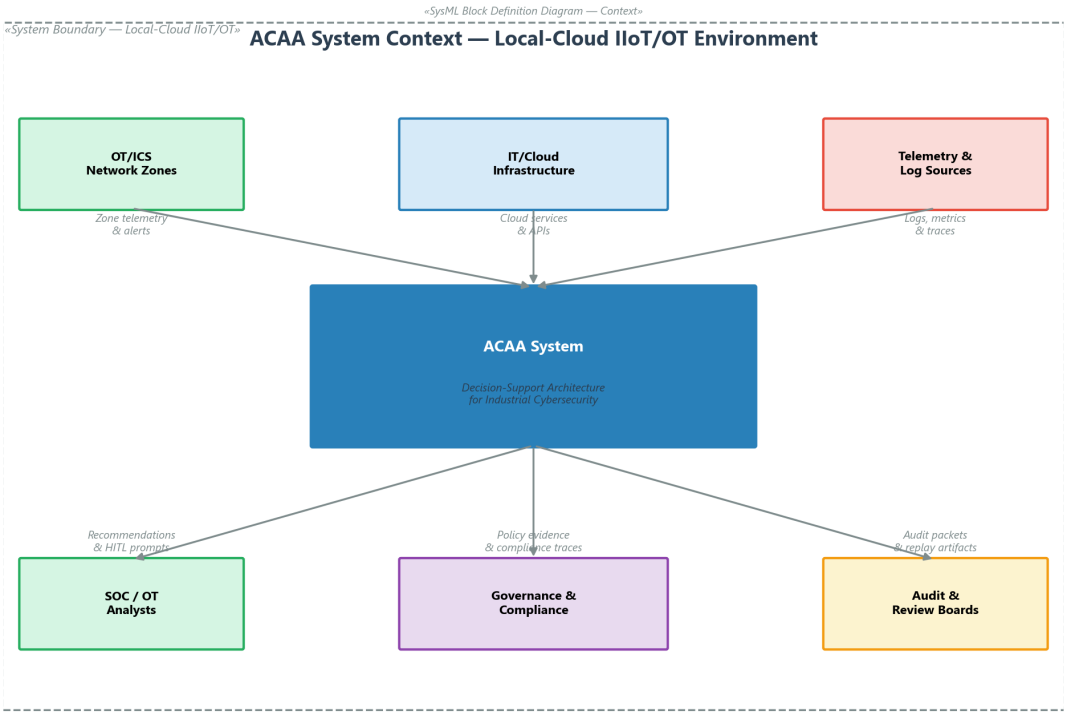

Figure 1.1: SysML context diagram showing the ACAA system boundary within a local-cloud IIoT/OT deployment. External actors include OT/ICS network zones, IT/cloud infrastructure, telemetry sources, SOC analysts, governance authorities, and audit review boards. Interface flows indicate the nature of data exchange at each boundary.

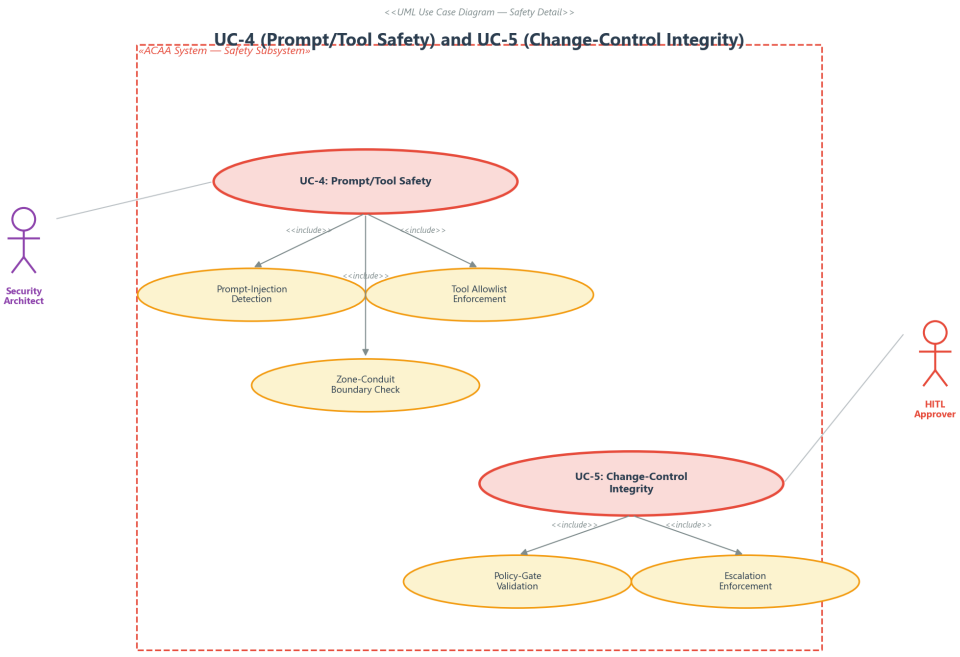

Figure 1.2: UML use case diagram showing the *include* decomposition of UC-4 (Prompt/Tool Safety) into prompt-injection detection, tool allowlist enforcement, and zone-conduit boundary checks, and UC-5 (Change-Control Integrity) into policy-gate validation and escalation enforcement. Actor associations indicate Security Architect and HITL Approver responsibility boundaries.

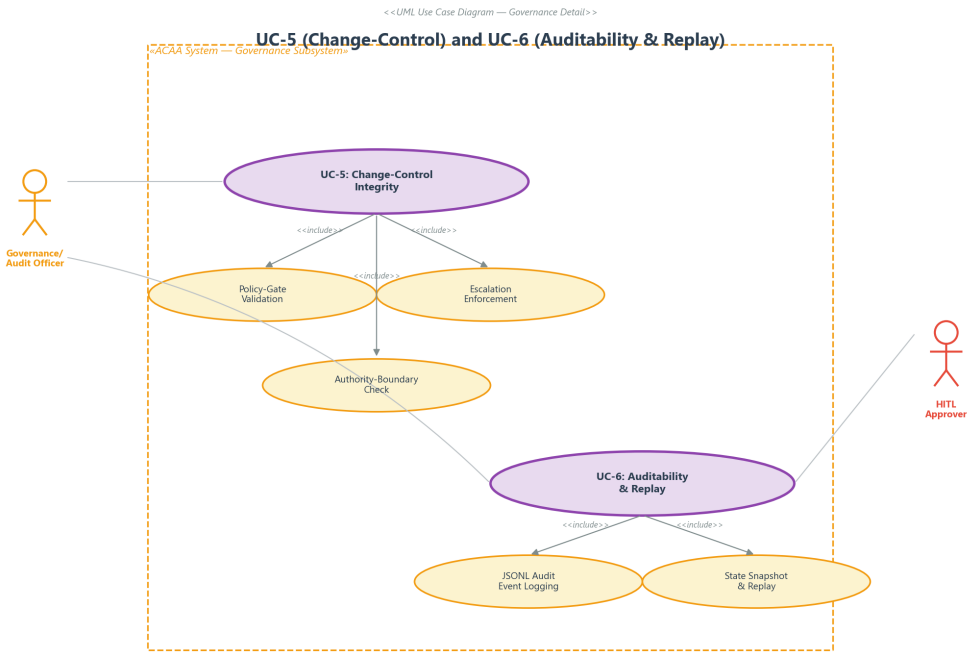

Figure 1.3: UML use case diagram showing the *include* decomposition of UC-5 (Change-Control Integrity) into policy-gate validation, escalation enforcement, and authority-boundary checks, and UC-6 (Auditability & Replay) into JSONL audit event logging and state snapshot replay. Actor associations indicate Governance/Audit Officer and HITL Approver responsibility boundaries.

1.6 Research Orientation and Design Requirements

These UC issues motivate the following design requirements for the thesis system:

1. integrate analytical outputs across statistical, ML, deep, generative, and agentic layers,
2. separate adaptive reasoning from deterministic control authority,
3. enforce policy and safety constraints as executable runtime logic,
4. preserve end-to-end traceability through artifact contracts and audit logs,
5. support human-in-the-loop adjudication for contested or high-risk decisions.

The key point is architectural: correctness is evaluated at workflow level, not model level.

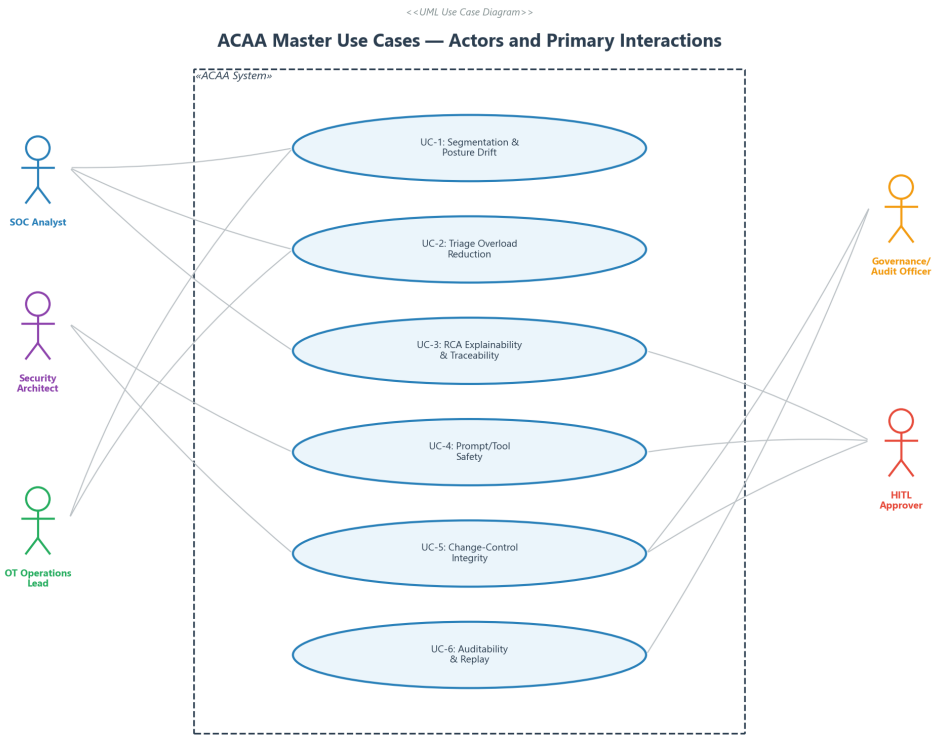

Figure 1.4: UML use case diagram showing the five primary actors (SOC Analyst, Security Architect, OT Operations Lead, Governance/Audit Officer, HITL Approver) and their associations with the six system use cases (UC-1 through UC-6). This view establishes the actor-responsibility mapping that governs access control and authority boundaries throughout the architecture.

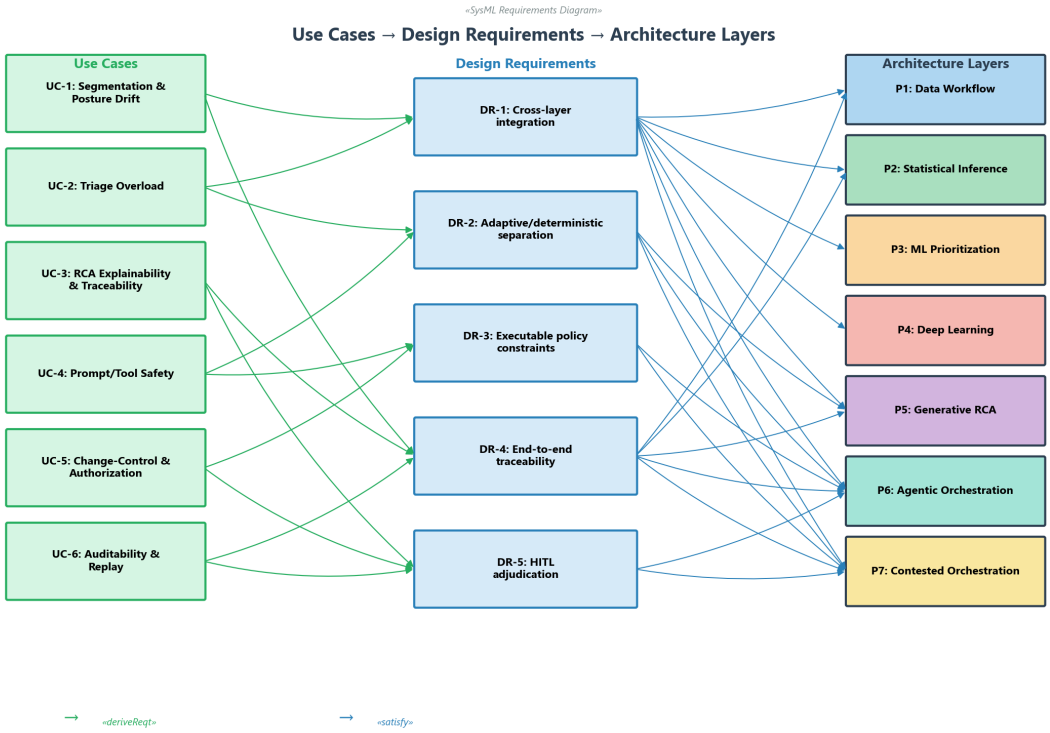

Figure 1.5: SysML requirements diagram mapping use cases (UC-1 through UC-6) to design requirements (DR-1 through DR-5) via *deriveReqt* relationships, and from design requirements to architecture layers (P1–P7) via *satisfy* relationships. This three-column traceability view supports governance defensibility by linking every operational concern to its implementing layer.

1.7 Related Work and Positioning

1.7.1 AIOps and RCA Benchmarks

OpenRCA and RCA-oriented telemetry benchmarks provide incident-centered tasks over logs, metrics, and traces. They establish strong foundations for evaluating RCA assistance, but they do not by themselves solve governance-bound decision orchestration in industrial settings [9].

GAIA-style large observability datasets and LANL cybersecurity telemetry enable representation learning and sequence modeling at scale. These resources are highly valuable for model development, yet they typically provide limited guidance on policy-gated action pathways and accountability structures [10].

1.7.2 Governance and OT Security Frameworks

NIST CSF 2.0, NIST SP 800-82r3, IEC 62443, and CMMC define governance expectations for critical systems, including risk management, incident handling, and control integrity [1, 2, 11, 12]. These frameworks are indispensable for deployment reasoning, but they do not prescribe a concrete multi-layer AI integration pattern that combines generative and agentic components with deterministic runtime safeguards.

1.7.3 Agentic AI Patterns

Recent agentic patterns (tool-using agents, ReAct-style loops, orchestrator-agent workflows) improve task decomposition and tool interaction [4]. However, many implementations remain single-orchestrator pipelines and do not directly model contested objective functions (security assurance versus operations continuity) with explicit meta-level adjudication and logged HITL resolution.

1.7.4 Sociotechnical Systems and Risk Governance Framing

The thesis is also informed by sociotechnical systems thinking, which treats outcomes as products of interacting technical and organizational subsystems rather than isolated technical mechanisms. In this perspective, risk signals such as exploitation prioritization, response urgency, and operational acceptability are co-produced by system design, human workflows, and institutional decision rules. This framing is widely used in safety-critical and systems-engineering contexts and is directly relevant to cybersecurity operations, where escalation, patching, containment, and continuity decisions are never purely algorithmic.

This perspective strengthens the thesis in two ways. First, it provides a principled explanation for why certain labels and governance targets cannot be reduced to technical severity metrics. Second, it expands the interpretive scope of model and agent outputs by asking not only whether a prediction is accurate, but which subsystem (technical, operational, institutional) appears to be driving the prediction and decision pathway.

1.7.5 Gap Statement

The literature and tooling landscape provides strong components, but an integration gap remains:

- insufficient coupling between model outputs and enforceable governance controls,
- limited support for contested multi-objective adjudication,
- weak treatment of responsibility traceability across layered AI decisions.

This thesis is positioned to address that gap through an assurance-centered architecture.

1.7.6 Comparative Positioning Against Nearest Alternatives

Approach Family	Core Strength	Typical Limitation	ACAA Positioning
RCA benchmark pipelines (for example OpenRCA-style) [9]	Strong incident-level analytics and replay tasks	Limited executable governance boundary for high-impact actions	Reuses analytics foundations but adds deterministic policy and HITL adjudication
Single-orchestrator agentic workflows (ReAct/tool-use) [4]	Effective decomposition and tool interaction	Objective collapse into one path, weaker contested tradeoff handling	Introduces parallel assurance/continuity branches with meta-adjudication
Framework-only governance adoption (NIST/IEC/CMMC) [1, 2, 11, 12]	Clear control and process expectations	Does not define concrete cross-layer AI runtime composition	Implements framework-linked, code-level gates and trace packets
Model-centric cybersecurity ML/deep stacks	High predictive performance on narrow tasks	Weak cross-stage accountability and authority control	Treats model outputs as inputs to governed orchestration, not final authority

Table 1.1: Comparative positioning of ACAA relative to nearest AIOps/agentic alternatives.

1.8 Thesis Contribution and Solution Overview

This work contributes a systems-level solution that integrates prior project layers into a governed decision-support architecture for local-cloud IIoT/OT operations. The contribution has four parts:

1. **Cross-layer integration**: a contract-based pipeline connecting statistical inference, ML prioritization, deep telemetry modeling, generative RCA drafting, and agentic orchestration.
2. **Deterministic safety boundary**: policy gates, allowlists, and refusal/escalation logic that constrain adaptive components.
3. **Parallel contested orchestration**: dual branch reasoning (assurance vs continuity) with meta-orchestrator adjudication and HITL controls.
4. **Audit-grade traceability**: replayable artifacts, decision packets, and framework-linked governance evidence.

The result is not a claim of full autonomy. It is a claim of defensible decision support under industrial constraints.

In addition, the thesis contributes a consistent interpretive stance: labels and decisions are treated as sociotechnical outputs, and therefore model evaluation is paired with governance context, uncertainty disclosure, and responsibility traceability. This is a deliberate departure from single-metric ML reporting and a key reason the integrated architecture is presented as a systems contribution rather than a collection of model experiments.

1.9 Data Lineage and Thesis Contract

The thesis uses a layered evidence contract rather than a single-dataset narrative. Each chapter contributes a specific artifact class: P1 data contracts, P2 inferential governance signals, P3 vulnerability prioritization priors, P4 telemetry representation evidence, P5 generative explanation artifacts, P6 policy-gated agent decisions, and P7 contested adjudication packets.

This lineage is intentional. Cross-chapter coherence is achieved through role-based integration contracts and audit traceability, not by forcing every chapter to reuse identical raw data sources.

1.10 Standards and Assurance Baseline

The thesis is grounded in industrial governance references as design constraints and traceability anchors:

- **NIST CSF 2.0:** cybersecurity risk governance, detection, response, and continuous improvement framing [1].
- **NIST SP 800-82r3:** ICS/OT operational constraints for safe cybersecurity practice [2].
- **IEC 62443:** zone-conduit security architecture and control-system defense principles [11].
- **CMMC 2.0:** process maturity and auditable cybersecurity practice expectations [12].

- **NIST AI RMF 1.0:** AI governance, risk measurement, and risk treatment across system lifecycle phases [3].
- **IEEE 7000 and IEEE 1012:** ethically informed system design and verification/validation rigor for assurance claims [13, 14].

These references are used for evidence alignment and architecture review readiness. They are not presented as claims of formal certification.

1.11 Chapter Roadmap

The remaining chapters develop this contribution progressively:

- Chapter 2: reproducible data workflow foundation.
- Chapter 3: statistical inference for governance-relevant structure.
- Chapter 4: leakage-aware ML prioritization under rare events.
- Chapter 5: deep learning systems with controlled ablations and guardrails.
- Chapter 6: generative RCA layer with measurable output quality.
- Chapter 7: policy-gated multi-agent orchestration.
- Chapter 8: integrated synthesis with contested orchestration and HITL adjudication.

This sequence mirrors a practical engineering progression: from trustworthy data handling to governed system-level decision architecture.

2 Foundations: Reproducible Cyber Telemetry Workflow

2.1 Context and Objective

The first project established the data-engineering substrate for the rest of the system. The objective was to build a reproducible and inspectable workflow over public telecom telemetry so that later modeling stages could inherit consistent assumptions, stable interfaces, and auditable preprocessing decisions. The practical constraint was clear: if ingestion and cleaning are ambiguous, downstream model performance claims are not defensible.

The dataset was a combined OpenRCA telecom slice assembled from five metric sources (metric_app, metric_container, metric_middleware, metric_node, metric_service) [9]. The raw combined pool contained 592,921 records. For controlled exploratory analysis, a deterministic subset of 50,000 rows was used.

This chapter contributes to the thesis assurance case as the **data reliability layer**: it establishes reproducible ingestion, schema governance, and provenance controls that bound interpretation in all later chapters.

2.2 GitHub Project and Report

Project repository: https://github.com/Ohara124c41/ai-programming-foundations-project

Report artifact: https://github.com/Ohara124c41/ai-programming-foundations-project/blob/main/module_summary.pdf

2.3 Related Work and Use-Case Positioning

From a related-work perspective, this chapter is positioned at the boundary between data engineering reliability and security operations observability. Hidden technical debt in ML systems is often rooted in weak data contracts, brittle schemas, and inconsistent preprocessing behavior across environments [15]. NIST guidance on security log management similarly emphasizes normalized, traceable, and policy-consistent telemetry handling as a precondition for dependable downstream analytics [16].

Within the thesis use-case framing, this chapter primarily addresses **UC-2 (triage overload**

under heterogeneous evidence) and **UC-6 (auditability and replay)** by creating deterministic ingestion, schema governance, and provenance discipline. It also partially supports **UC-1 (posture drift)** by preserving source-specific telemetry semantics needed for later drift-sensitive monitoring logic.

2.4 Data Engineering Design

The workflow was intentionally structured as a contract:

1. Ingest and merge source files while preserving source provenance via metric_source.
2. Standardize schema (snake case naming, duplicate checks, timestamp parsing).
3. Apply selective imputation only where missingness was plausibly non-structural.
4. Preserve structurally sparse columns to avoid fabricated data.
5. Add derived fields for downstream learning (event hour, day-of-week, log transforms).

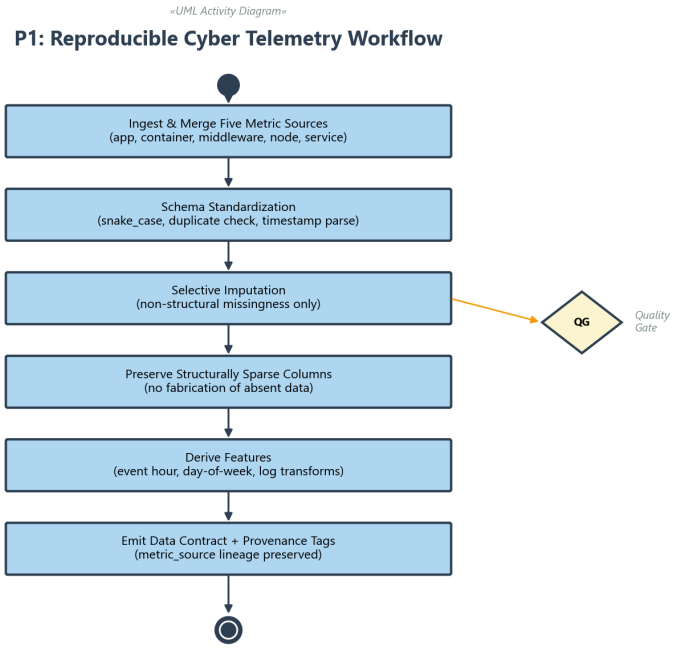

Figure 2.1: UML activity diagram for the reproducible data workflow. The pipeline proceeds through five governed stages: multi-source ingestion with provenance tagging, schema standardization, selective imputation for non-structural missingness only, structural-sparsity preservation, and derived-feature generation. A quality gate checkpoint validates contract compliance before downstream emission.

The most important design choice was treating sparsity as semantic rather than accidental. Several service-specific fields were approximately 99.94% missing because those fields do not apply to most sources. A global imputation rule would have created synthetic values at scale and hidden source-specific semantics.

2.5 Exploratory Analysis and Statistical Characterization

The analysis validated three properties that became foundational for all later projects.

First, structural heterogeneity was dominant. Source counts were highly imbalanced (for example, metric_node was orders of magnitude larger than metric_app). This implied that any later training pipeline needed source-aware diagnostics.

Second, distribution shape was strongly heavy-tailed. The primary value channel showed extreme skewness and kurtosis. A log transform improved shape but did not normalize it, indicating that robust metrics and non-parametric checks would be needed in later stages.

Third, manifold structure was present but not cleanly separable. PCA and t-SNE views showed patterned overlap between sources rather than strict cluster boundaries. This informed later assumptions: separability should be tested, not presumed.

2.6 Key Visual Diagnostics

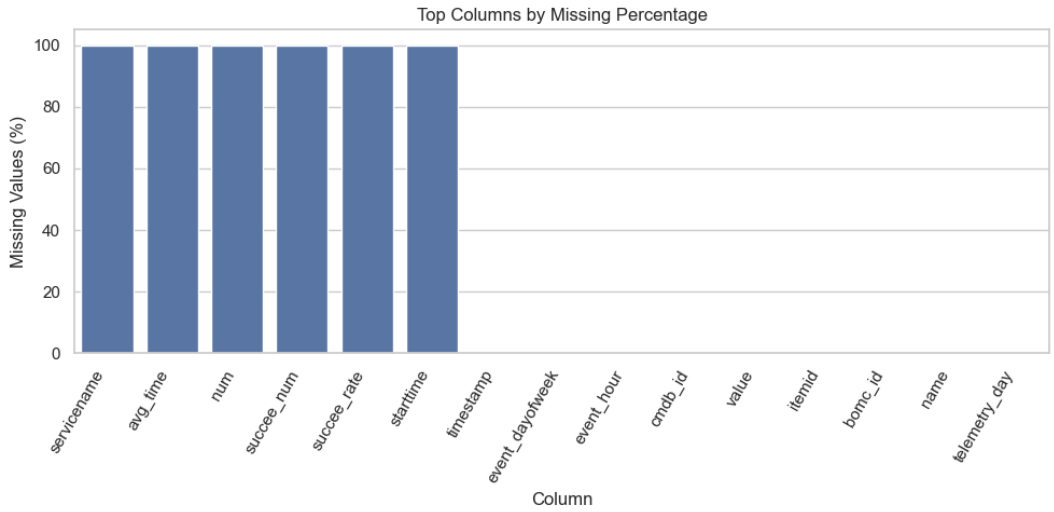

Figure 2.2: Missingness profile across telecom metrics, showing concentrated structural sparsity in source-specific fields.

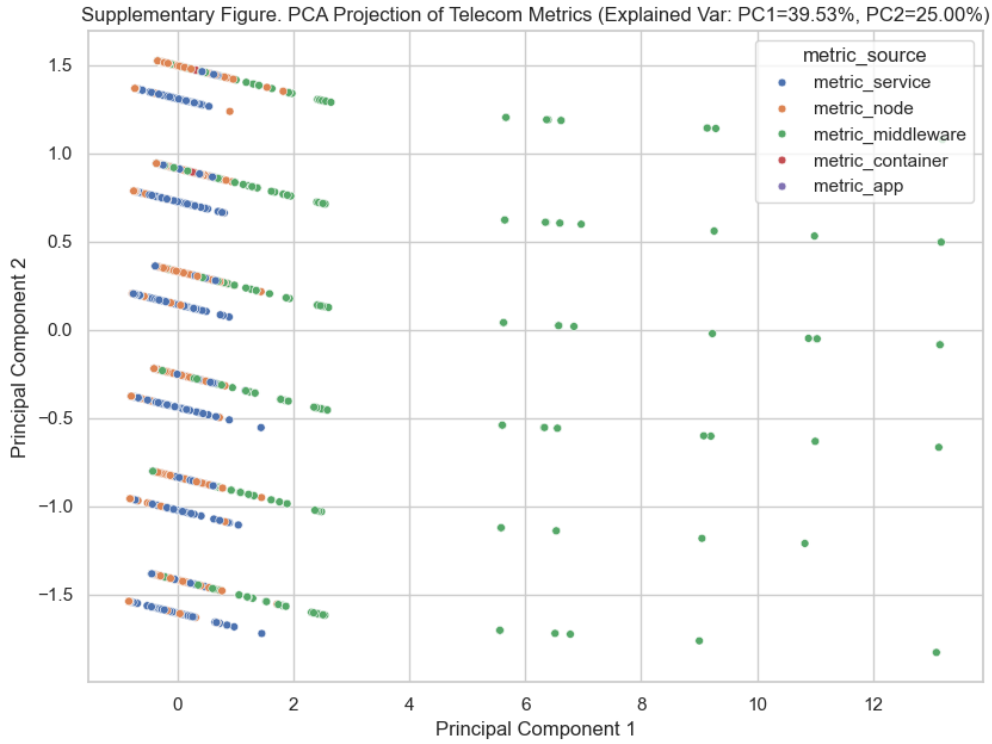

Figure 2.3: PCA projection of telecom metric records, illustrating partial structure with overlapping source clusters.

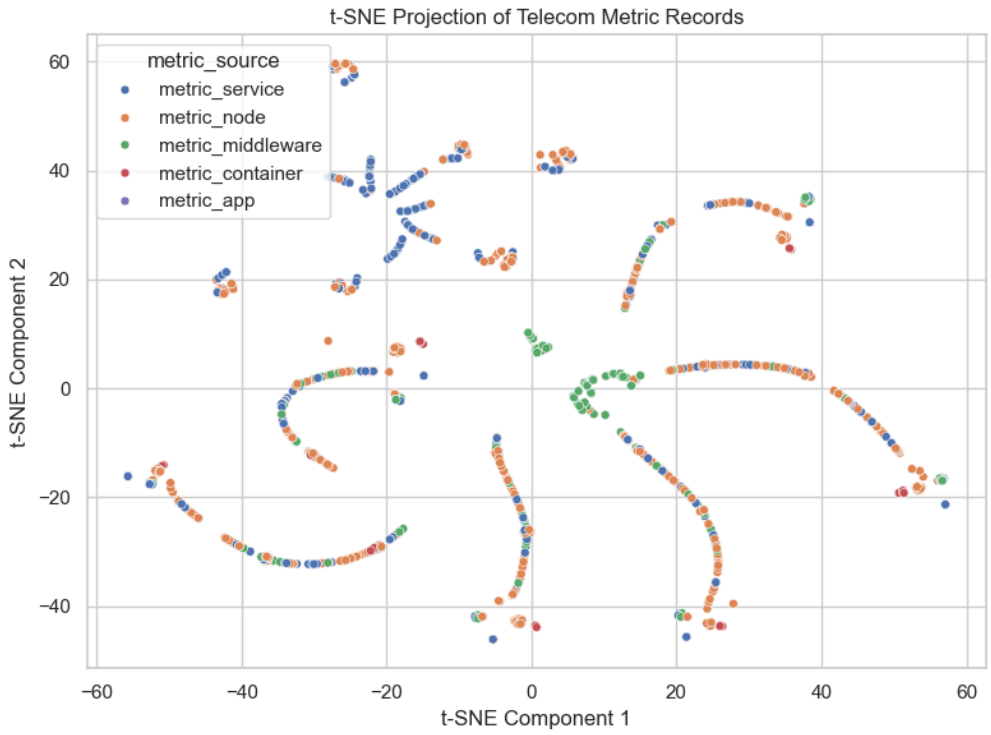

Figure 2.4: t-SNE projection of telecom metric records used to inspect local-neighborhood separability and overlap behavior.

2.7 Responsible AI and Quality Controls

A lightweight AIF360-compatible screening step was included as a risk diagnostic, not as a demographic fairness claim [17, 18]. In this setting, group definitions were technical cohorts tied to telemetry source and schema. The result was treated as an early warning that representation imbalance and structural sparsity can propagate into model behavior if left unchecked.

Reproducibility controls included deterministic subsetting, explicit preprocessing steps, and an execution order that ran end-to-end from a clean notebook kernel. These controls were basic, but they established the discipline used in every later project.

2.8 Production Posture for Continuous Data Governance

For production transition, this workflow should run as a continuous data-contract monitor rather than a one-time preprocessing stage. A practical minimum posture is:

1. schema-drift alarms on new/missing fields and datatype changes,
2. null-pattern drift alerts on structural-sparsity columns,
3. source-composition drift monitoring (for example, sudden metric_source mix change),

4. timestamp integrity checks (clock skew, duplicate windows, missing intervals),
5. automatic quarantine of non-conformant batches with audit ticket generation.

These controls keep P1 aligned with UC-1 and UC-6 under live operations by converting data quality assumptions into continuously verified runtime conditions.

2.9 Data Lineage and Integration Contract

This chapter defines the upstream data contract used by downstream analytics: source provenance tags, schema normalization rules, timestamp hygiene, and structural-missingness preservation. These controls directly support the integrated packet construction described in Chapter 8 by ensuring that later features and decisions remain traceable to governed preprocessing decisions.

In architecture terms, P1 artifacts are not high-level decision outputs; they are integrity constraints that determine whether later evidence is admissible.

2.10 Standards and Assurance Crosswalk

The data workflow controls map to industrial assurance references as follows:

- **NIST CSF 2.0 (GV/ID/DE):** data quality and provenance controls support governed detection and risk analysis [1].
- **NIST SP 800-82r3:** ICS-aware data handling supports reliable operational monitoring inputs [2].
- **IEC 62443:** structured telemetry handling aligns with secure operations and traceable monitoring foundations [11].
- **CMMC 2.0:** reproducible processing and auditable data lineage support assessment-ready evidence generation [12].
- **NIST AI RMF 1.0 (MAP/MEASURE):** data representativeness and subgroup diagnostics are treated as AI risk signals [3].
- **IEEE 1012:** deterministic preprocessing and repeatable execution strengthen verification readiness for downstream models [14].

2.11 Limitations

This stage focused on workflow reliability and data understanding, not predictive performance. Key limitations were:

- the subset was intentionally bounded for runtime,

- fairness checks used proxy cohort definitions,
- exploratory geometry (PCA/t-SNE) remained descriptive, not inferential.

These limitations were acceptable because Project 1 was intentionally positioned as a foundation layer.

2.12 Lessons Learned

Three lessons from this chapter informed the full thesis trajectory.

1. **Data contracts precede model quality.** Stable schema handling and provenance fields are not documentation overhead; they are architectural controls.
2. **Structural missingness must be preserved as information.** Treating all nulls as noise can create large synthetic artifacts.
3. **Shape diagnostics should drive method choice.** Heavy tails and source imbalance justify robust, uncertainty-aware evaluation in all subsequent modeling.

2.13 Bridge to Chapter 3

With data workflow discipline in place, the next project moved from description to inference. Chapter 3 asks whether governance-relevant category structure differs across infrastructure domains and introduces formal hypothesis testing, effect-size interpretation, and assumption diagnostics. That statistical layer became the first explicit evidence framework for system-level cybersecurity decisions.

3 Statistical Inference for Governance Mapping

3.1 Problem Statement

After establishing a reproducible data workflow, the second project evaluated a focused governance question: do NIST CSF function mappings vary across infrastructure domains? The dataset combined AMI, DER, and DGM mappings from public NIST/NESCOR resources into a single analysis table of 502 rows and 15 columns [1].

The objective was not predictive modeling. It was to produce defensible statistical evidence about cross-domain structure so that later system components would be grounded in measurable differences rather than assumptions.

This chapter contributes to the thesis assurance case as the **evidence calibration layer**: it defines how governance structure is measured, bounded, and carried forward with uncertainty rather than assumed as static policy truth.

3.2 GitHub Project and Report

Project repository: https://github.com/Ohara124c41/statistical-analysis-NIST-CFS

Report artifact: https://github.com/Ohara124c41/statistical-analysis-NIST-CFS/blob/main/Statistical_Analysis_Report.pdf

3.3 Related Work and Use-Case Positioning

This chapter is grounded in classical categorical-inference and multiple-testing practice. Agresti's contingency-table treatment provides the methodological basis for association interpretation under sparse categorical structure [19], while modern multiplicity controls (including false-discovery-rate framing) support defensible pairwise claims under repeated testing [20, 21]. These approaches align with security-governance analytics where over-claiming significance can lead to brittle policy decisions.

In use-case terms, this chapter primarily strengthens **UC-2 (triage overload)** by turning policy mapping differences into measurable evidence and **UC-6 (auditability)** by attaching uncertainty, assumption checks, and effect-size bounds to each governance inference.

3.4 Methodological Design

The analysis followed an initial data analysis sequence: schema checks, missingness profiling, categorical frequency structure, then inferential testing. The primary hypothesis test was a chi-square test of independence for domain versus nist_function. Effect size was reported with Cramer's V, and localized behavior was inspected through standardized residuals [19].

To avoid over-reading p-values, the workflow added:

- Holm correction for pairwise comparisons [21],
- bootstrap confidence intervals for Cramer's V [22],
- non-parametric tests (Kruskal-Wallis, Mann-Whitney) for text-length comparisons [23].

This baseline-versus-enhanced comparison design was important: the baseline answered if any association existed; the enhanced layer showed where differences were concentrated and how robust they were under assumption stress.

3.5 Results

The omnibus association test indicated non-independence between domain and NIST function distribution ($\chi^2 = 18.1818$, $df = 6$, $p = 0.005793$). Effect size was modest (Cramer's V $= 0.1434$), with bootstrap CI approximately $[0.1136, 0.1981]$.

Pairwise Holm-adjusted comparisons showed significant differences for AMI vs DER and DER vs DGM, while AMI vs DGM was not significant. Practically, this pointed to DER-related function mix differences as the strongest contributor to the global signal.

A major caveat was sparse expected counts in the contingency table (about one-third of cells below 5). The project did not hide this. Instead, it explicitly qualified interpretation and elevated effect-size plus resampling evidence.

Text-length comparisons of mitigation descriptions were non-significant across domains, suggesting that observed differences were more structural (function mapping emphasis) than stylistic in narrative field length.

Sparse-Cell Mitigation Note (Appendix-Oriented)

To strengthen inferential defensibility in the next revision cycle, sparse-cell handling should be extended with appendix-grade alternatives:

- exact/Monte Carlo p-value estimation for contingency association,
- permutation-based null checks for robustness against asymptotic assumptions,
- category-collapsing sensitivity analysis with documented governance rationale.

This planned extension is intended to keep uncertainty treatment transparent when long-tail category structure is operationally unavoidable.

3.6 Key Visual Diagnostics

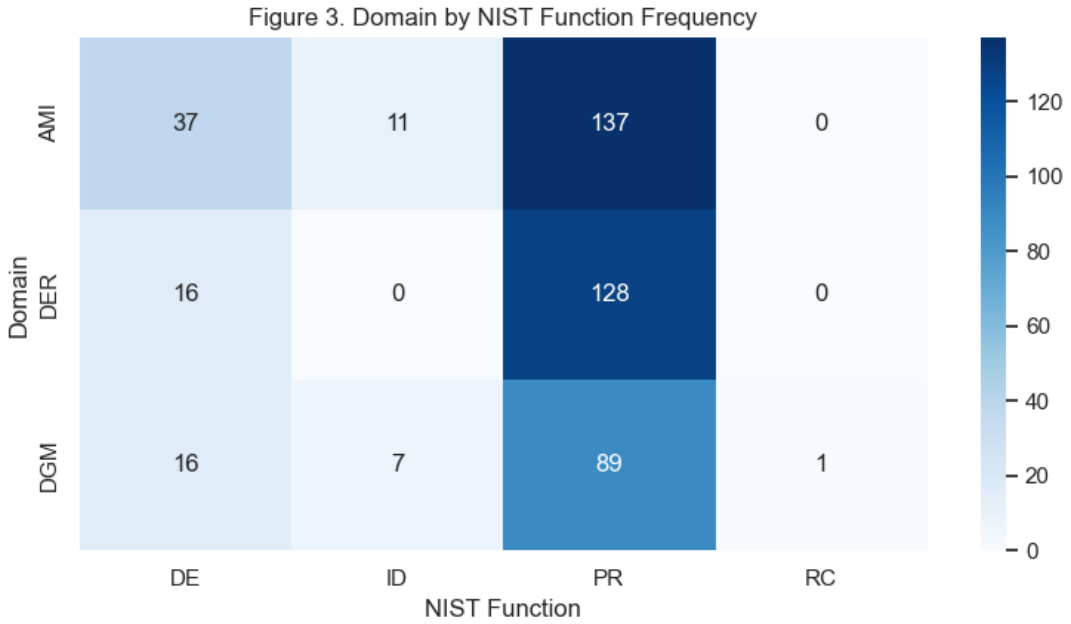

Figure 3.1: Domain-by-NIST function heatmap highlighting cross-domain governance pattern differences.

3.7 Operational Interpretation

The result supports a governance-relevant claim: domain context matters, but the magnitude is moderate rather than extreme. For architecture and policy teams, this means controls should not be copied uniformly across AMI/DER/DGM without domain-aware validation.

The more important contribution was methodological: the project demonstrated a repeatable pattern for handling sparse categorical inference in cybersecurity datasets:

1. run omnibus test,
2. report effect size,
3. check assumptions,
4. localize differences,
5. control multiplicity,
6. communicate uncertainty.

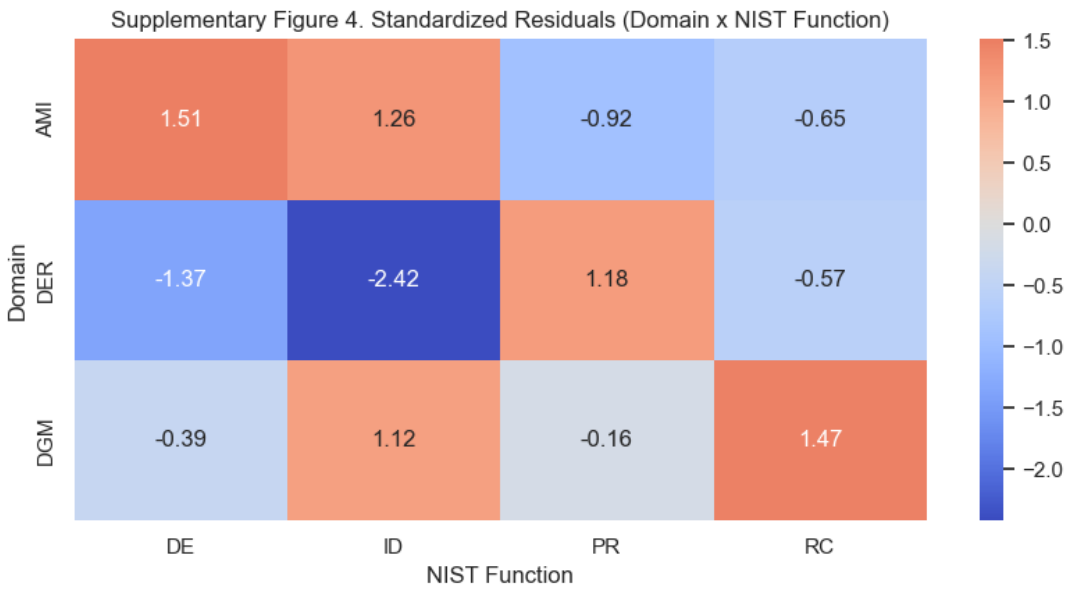

Figure 3.2: Standardized residual heatmap localizing the chi-square association signal.

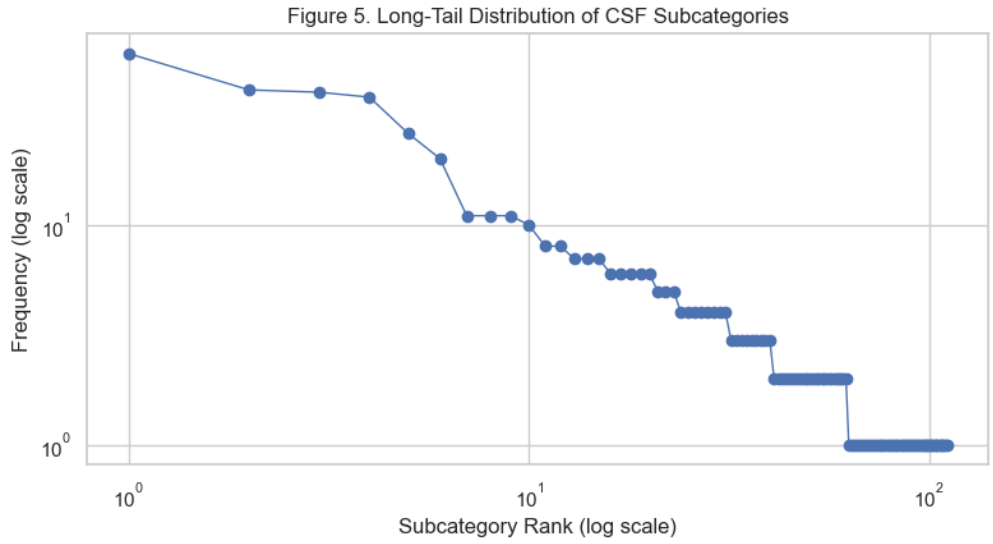

Figure 3.3: Long-tail CSF subcategory distribution demonstrating sparse-category structure in governance mappings.

3.8 Data Lineage and Integration Contract

The statistical outputs from this chapter are treated as governance priors for downstream components, not as standalone policy decisions. In practical terms, domain-function residual structure and effect-size bounds are versioned as chapter artifacts and consumed by later policy validators to contextualize confidence and escalation logic.

Within the integrated architecture (Chapter 8), this chapter therefore contributes to packet interpretation rather than packet scoring: it constrains how P3 risk priors, P4 telemetry signals, and P5 generative RCA summaries are read under domain context. This keeps adjudication behavior traceable to explicit statistical evidence.

3.9 Standards and Assurance Crosswalk

This chapter's outputs are aligned to industrial governance references as traceability evidence, not certification claims.

- **NIST CSF 2.0 (GV/ID/DE/RS):** domain-function association evidence supports governance profiling and response-planning differentiation [1].
- **NIST SP 800-82r3:** domain-aware interpretation aligns with ICS-specific control tailoring and operational context boundaries [2].
- **IEC 62443 (zone/conduit governance intent):** cross-domain differences motivate segmented control posture rather than uniform control assumptions [11].
- **CMMC 2.0 (process maturity and auditable practice):** reproducible statistical workflow provides objective evidence artifacts for review readiness [12].
- **NIST AI RMF 1.0 (MAP/MEASURE/MANAGE):** uncertainty reporting (effect sizes, confidence intervals, assumption diagnostics) supports risk-informed AI use [3].
- **IEEE 7000 and IEEE 1012:** explicit assumptions, uncertainty disclosure, and repeatable inference steps strengthen lifecycle traceability and V&V defensibility [13, 14].

3.10 Limitations

This was a curated mapping dataset, not live incident telemetry. Therefore, findings describe governance mapping structure, not incident prevalence. Category long-tail sparsity also limited fine-grained stability.

3.11 Lessons Learned

1. **Significance is insufficient without magnitude and assumptions.** Statistical claims should include effect size and diagnostic context.

2. **Localization matters for policy action.** Global association is less actionable than domain-pair and residual-level structure.
3. **Reproducible statistics are a systems asset.** A stable inference pipeline can later serve as an assurance control, not just an analysis artifact.

3.12 Bridge to Chapter 4

Chapter 3 established evidence discipline for governance structure. Chapter 4 transitions from statistical association to machine learning prioritization under severe class imbalance. The inferential rigor introduced here (assumption checking, uncertainty framing, and effect interpretation) directly informs the model-evaluation philosophy in the next stage.

4 Machine Learning Foundations for Vulnerability Prioritization

4.1 Industry Problem and Modeling Objective

Security operations teams in industrial and enterprise environments face vulnerability backlogs that exceed analyst capacity. The third project addressed this as a supervised rare-event ranking problem: predict whether a CVE appears in the CISA Known Exploited Vulnerabilities list using features derived from NVD metadata [6, 7].

The modeling objective was explicitly dual:

- improve triage utility under extreme imbalance,
- preserve governance credibility through leakage-aware evaluation.

The working dataset had 50,000 records with 56 positives (prevalence 0.112%).

This chapter contributes to the thesis assurance case as the **risk-prioritization layer**: it converts vulnerability metadata into governed triage priors while explicitly separating operational baseline models from leakage-sensitive experimental upper bounds.

4.2 GitHub Project and Report

Project repository: https://github.com/Ohara124c41/ai-programming-foundations-project

Report artifact: https://github.com/Ohara124c41/ai-programming-foundations-project/blob/main/Machine_Learning_Analysis_Report.pdf

4.3 Related Work and Use-Case Positioning

The chapter's modeling strategy is consistent with rare-event and imbalance-focused literature, where class prevalence distortion requires explicit sampling, thresholding, and metric governance rather than accuracy-centric evaluation [24–26]. It is also aligned with leakage-aware modeling practice, which treats feature provenance and temporal split design as validity controls rather than optional refinements [8].

Within the thesis use cases, this chapter directly targets **UC-2 (triage overload)** by generating governed vulnerability-prior scores and contributes to **UC-5 (authorization integrity)** by separating operationally admissible models from experimental upper-bound models that fail leakage defensibility.

4.4 Sociotechnical Framing of the KEV Target

A key motivation for this chapter is that KEV inclusion should not be treated as a direct substitute for technical severity. CVSS captures intrinsic vulnerability characteristics, while KEV inclusion reflects observed exploitation and institutional prioritization decisions. In practice, that makes the target a joint signal shaped by technical properties, adversary behavior, defender exposure and patch latency, and CISA's operational risk judgment.

This distinction justifies the chapter's machine-learning formulation. If KEV membership were reducible to a simple CVSS threshold, a learned model would add little value. The actual operational problem is harder: analysts need prioritization support that captures interactions across technical metadata, reporting structure, and institutional response patterns. Framing the target this way also clarifies why leakage control is central to validity. Features that encode downstream institutional actions too directly (for example, near-proxy references to CISA artifacts) can collapse the problem into post hoc label reconstruction rather than meaningful prioritization support.

This sociotechnical perspective also changes how interpretability should be read. Feature importance is not only a ranking of predictive variables; it is evidence about which subsystem signals (technical, reporting, institutional) the model is relying on. That is one reason the chapter treats leakage audit and feature provenance as architectural controls rather than optional diagnostics.

4.5 Pipeline Design

A chronological split (70/15/15) replaced random splitting to reduce temporal leakage risk and better approximate deployment conditions. Preprocessing used a ColumnTransformer with median imputation and scaling for numeric fields and most-frequent plus one-hot encoding for categorical fields.

Model families included:

- baseline Dummy classifier,
- Logistic Regression and Decision Tree,
- Random Forest and Extra Trees [27, 28],
- ensemble variants (voting, stacking),
- semi-supervised self-training.

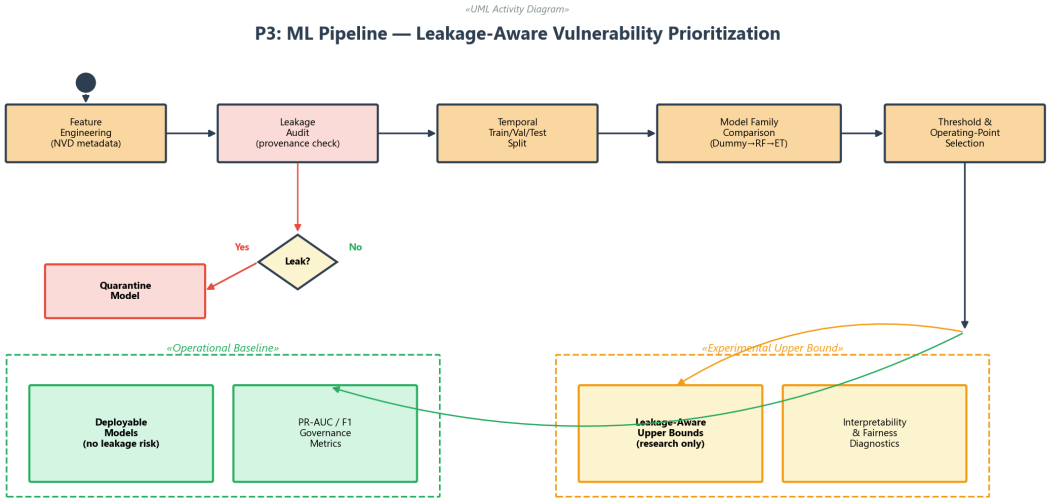

Figure 4.1: UML activity diagram for the leakage-aware vulnerability prioritization pipeline. The workflow progresses from feature engineering through a mandatory leakage audit gate, temporal train/val/test splitting, model family comparison, and governed threshold selection. A decision node separates quarantined models (leakage-compromised) from the operational baseline and experimental upper-bound tracks, each with distinct metric governance.

The pipeline was intentionally comparative rather than model-centric. Selection used F1 as primary, with PR-AUC, recall, and probability-error metrics (Brier, RMSE) as supporting controls [26, 29].

4.6 Results and Decision Logic

The leakage-controlled Random Forest produced the most defensible operational profile on holdout testing, with strong discrimination and useful rare-event recall. A richer experimental configuration (feature enrichment plus threshold tuning) produced substantially better headline metrics.

However, the improvement triggered a governance-critical audit finding: cisa_ref_count behaved as a near-proxy for the target label. In practical terms, the highest scoring model was partially learning reporting artifacts rather than robust exploitation risk [8].

This led to a two-track decision:

1. **Operational baseline:** leakage-controlled Random Forest.
2. **Research upper bound:** enriched Extra Trees retained as experimental only.

That decision pattern is architecturally important. It prioritizes deployment integrity over leaderboard performance.

4.7 Key Visual Diagnostics

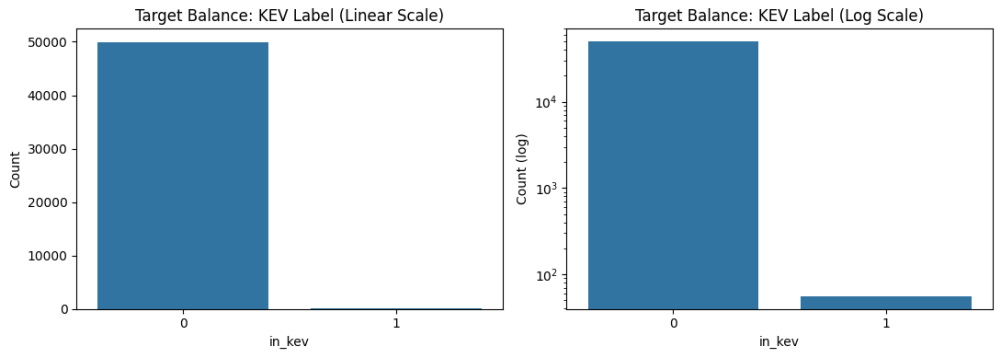

Figure 4.2: Extreme target imbalance in the KEV prediction dataset (rare positive-event regime).

4.8 Uncertainty and Fairness Diagnostics

Bootstrap resampling quantified uncertainty around F1 and PR-AUC. This prevented overconfidence in a low-prevalence regime where small count shifts can move threshold metrics

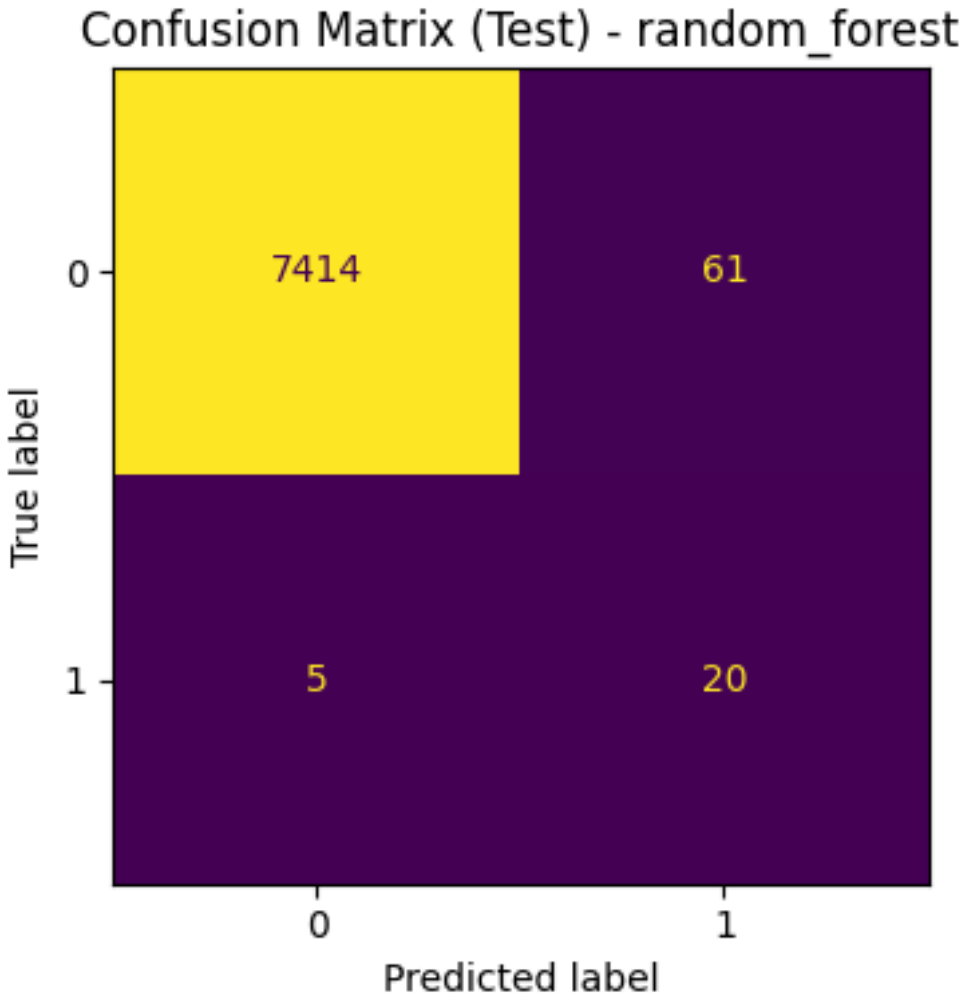

Figure 4.3: Precision-recall curve for holdout KEV prioritization performance under imbalance.

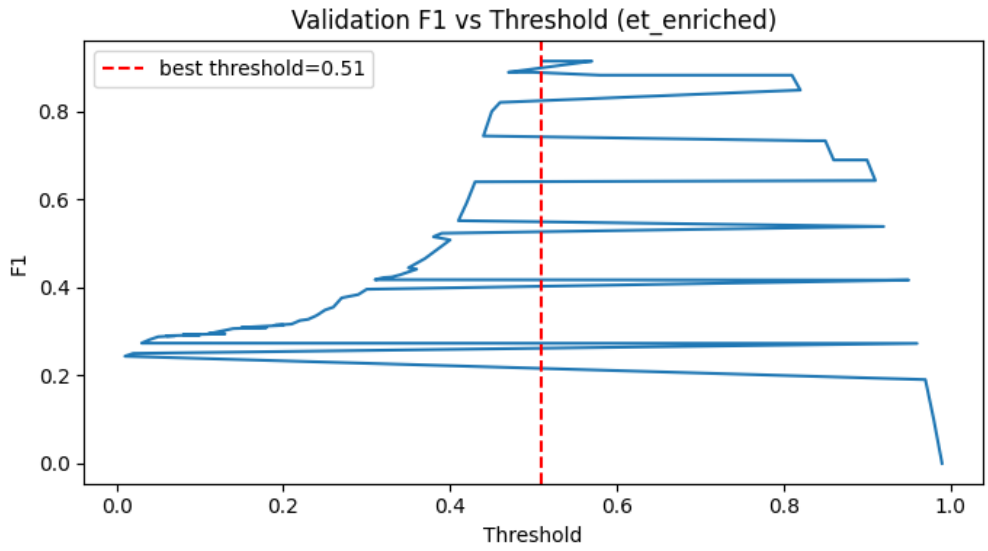

Figure 4.4: Validation F1 versus threshold used to support governed operating-point selection.

materially.

An AIF360-compatible screening view was added across technical cohorts (NETWORK vs NON_NETWORK attack vectors). Disparity metrics were interpreted as operational risk signals (review burden and detection consistency), not demographic fairness claims [17, 18].

4.9 Calibration and Operating-Point Policy

Operational threshold selection should be governed by an explicit cost profile rather than a single metric maximum. A practical policy framing for this chapter is:

- prioritize recall when missed exploited vulnerabilities have high containment cost,
- tighten threshold when analyst queue saturation increases false-positive handling risk,
- require periodic calibration review using Brier/RMSE plus precision-recall stability.

Under this policy, threshold changes become governed configuration updates with documented risk rationale, rather than ad hoc tuning choices.

4.10 What This Chapter Contributes to the Larger System

Chapter 4 introduced four patterns that remained stable through later projects:

1. strict baseline-versus-experimental comparison,

2. leakage audits as mandatory gates,
3. uncertainty reporting as first-class output,
4. governance diagnostics adjacent to model metrics.

In other words, this chapter converted predictive modeling from a score optimization exercise into a risk-aware decision-support component.

4.11 Data Lineage and Integration Contract

This chapter provides the primary vulnerability-prior signal family used in integration. In Chapter 8, these outputs are represented through P3 packet fields such as vuln_kev_rate and vuln_cvss_mean, with accompanying uncertainty and leakage-audit context.

The contract rule is strict: experimental features that introduce post hoc institutional proxies are never promoted to operational packet fields. Only leakage-controlled outputs and their calibration context are eligible for downstream contested orchestration.

4.12 Standards and Assurance Crosswalk

This chapter's ML controls map to industrial frameworks as follows:

- **NIST CSF 2.0 (ID/DE/RS/GV):** rare-event prioritization supports risk identification and detection triage under governance controls [1].
- **NIST SP 800-82r3:** vulnerability triage is constrained by operational-impact awareness rather than CVSS-only sorting [2].
- **CMMC 2.0:** leakage audits and reproducible split logic support repeatable, assessable cybersecurity analytics processes [12].
- **MITRE ATT&CK / ATT&CK for ICS alignment intent:** feature interpretation is framed for operational adversary-context reasoning, not raw score ranking alone [30].
- **NIST AI RMF 1.0 (GOVERN/MAP/MEASURE/MANAGE):** uncertainty, subgroup diagnostics, and leakage controls operationalize AI risk management [3].
- **IEEE 7000 and IEEE 1012:** model-selection decisions are tied to documented risk tradeoffs and verification evidence [13, 14].

4.13 Limitations

Key limitations were label incompleteness (absence from KEV is not proof of non-exploitation), low positive counts, and sensitivity of enrichment features to post hoc information pathways.

4.14 Lessons Learned

1. **Best score is not best system behavior.** High metrics can conceal leakage and produce unsafe confidence.
2. **Chronological evaluation is essential in cyber risk modeling.** Random splits can inflate perceived readiness.
3. **Governance belongs in the training loop.** Fairness and leakage checks should be part of model selection, not post-project appendices.

4.15 Bridge to Chapter 5

With an auditable ML baseline in place, the next step was representation learning over high-volume telemetry. Chapter 5 extends the workflow into deep learning, preserving the same decision philosophy: controlled experimentation, multi-metric evaluation, subgroup diagnostics, and explicit deployment guardrails.

5 Deep Learning Systems for IIoT Intrusion Detection

5.1 Purpose and Scope

The fourth project moved from feature-engineered tabular ML toward deep representation learning for IIoT intrusion detection. The dataset was RT-IoT2022, with a binary attack-versus-benign framing for triage-oriented use [31]. The goal was not novelty in architecture alone, but disciplined evaluation of deep models under imbalance, threshold sensitivity, and operational safeguards.

This chapter contributes to the thesis assurance case as the **telemetry representation layer**: it validates that high-capacity models can improve detection support while remaining bounded by reproducibility, calibration, and governance gates.

5.2 GitHub Project and Report

Project repository: https://github.com/Ohara124c41/deep-learning-iot-intrusion

Report artifact: https://github.com/Ohara124c41/deep-learning-iot-intrusion/blob/main/Deep_Learning_Systems_Analysis_Report_draft.md

5.3 Related Work and Use-Case Positioning

This chapter is positioned relative to tabular/telemetry deep-learning work that adapts attention architectures to mixed categorical-numeric features and compares them against strong non-neural baselines [32–34]. The central lesson from that literature is that representation capacity alone is insufficient; operational value depends on calibration, threshold behavior, and robust error topology under shift.

In thesis use-case terms, the chapter primarily supports **UC-1 (segmentation and posture drift)** and **UC-2 (triage overload)** by improving telemetry-level risk discrimination while retaining governance gates before any downstream decision authority.

5.4 Architecture and Training Strategy

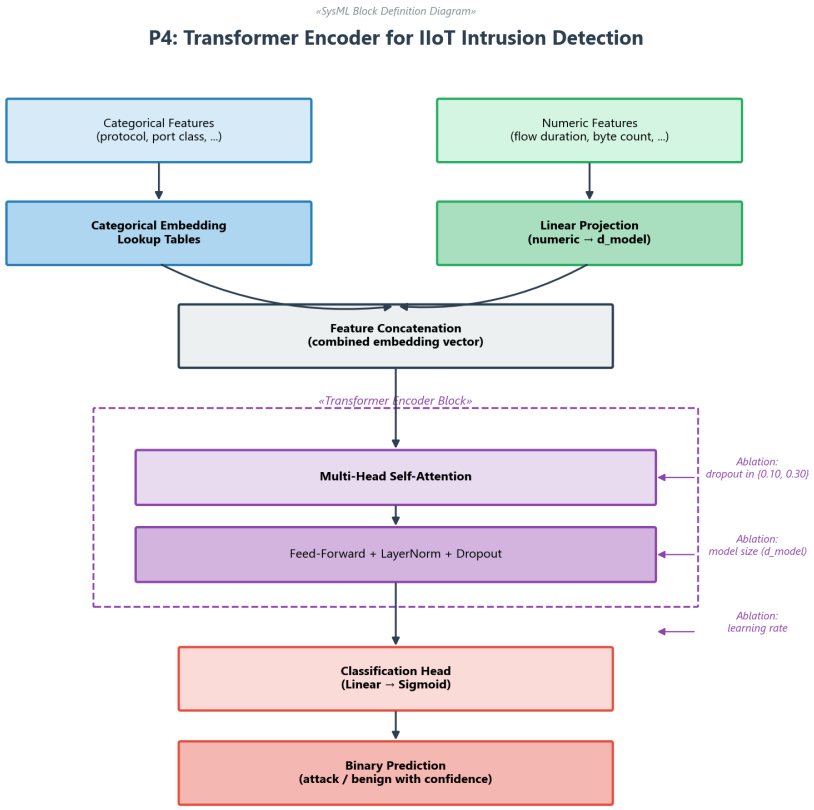

Figure 5.1: SysML block definition diagram of the Transformer encoder adapted to mixed tabular IIoT telemetry. Categorical features pass through embedding lookup tables while numeric features are linearly projected to the model dimension. Concatenated embeddings are processed by a multi-head self-attention encoder block with feed-forward layers, LayerNorm, and dropout, followed by a classification head producing binary attack/benign predictions with confidence scores. Ablation control points for dropout, model size, and learning rate are annotated.

The baseline model was a Transformer encoder adapted to mixed tabular telemetry inputs. Numeric and categorical channels were embedded into a shared representation, aggregated with a learned classification token, and passed to a binary prediction head [32].

Training used class-weighted BCEWithLogits loss, Adam-family optimization, dropout regularization, and early-stopping logic. Reproducibility controls included deterministic seeds, fixed split strategy, and saved processed split artifacts.

5.5 Controlled Experiment Design

The required baseline-versus-experimental comparison was deliberately simple and isolating:

- Baseline dropout: 0.10
- Experimental dropout: 0.30
- All other major factors held constant

This design prevented the common failure mode in deep-learning reports where multiple factors change simultaneously and attribution becomes ambiguous.

5.6 Results

Both configurations performed strongly, but with distinct tradeoffs. The higher-dropout model improved recall, F1, and probability-error metrics (Brier and RMSE), while the lower-dropout baseline retained slightly higher precision and ROC-AUC [26, 29].

Ablations and ensemble variants were then evaluated with an explicit score verifier. The final selected configuration (weighted top-3 ensemble in this run) balanced high discrimination with stronger probability quality and guardrail compliance.

Error analysis showed category-specific failure concentration (for example ARP poisoning in false negatives and ThingSpeak in false positives), demonstrating that aggregate metrics can hide operationally meaningful pockets of risk.

5.7 Key Visual Diagnostics

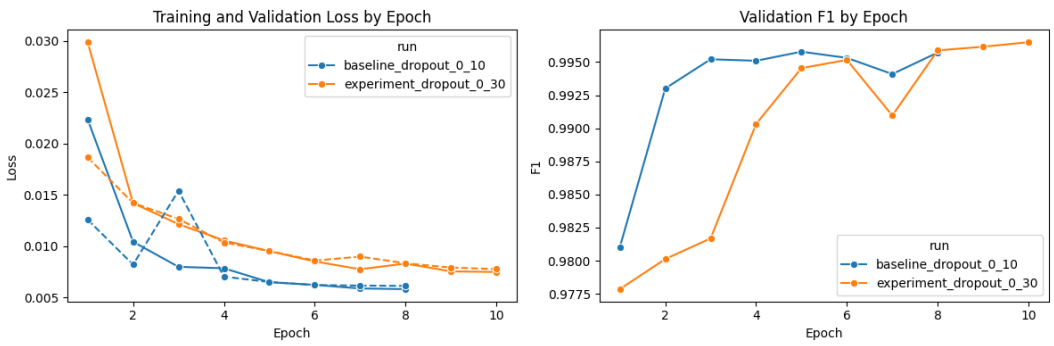

Figure 5.2: Training and validation loss trajectories for baseline and experimental settings.

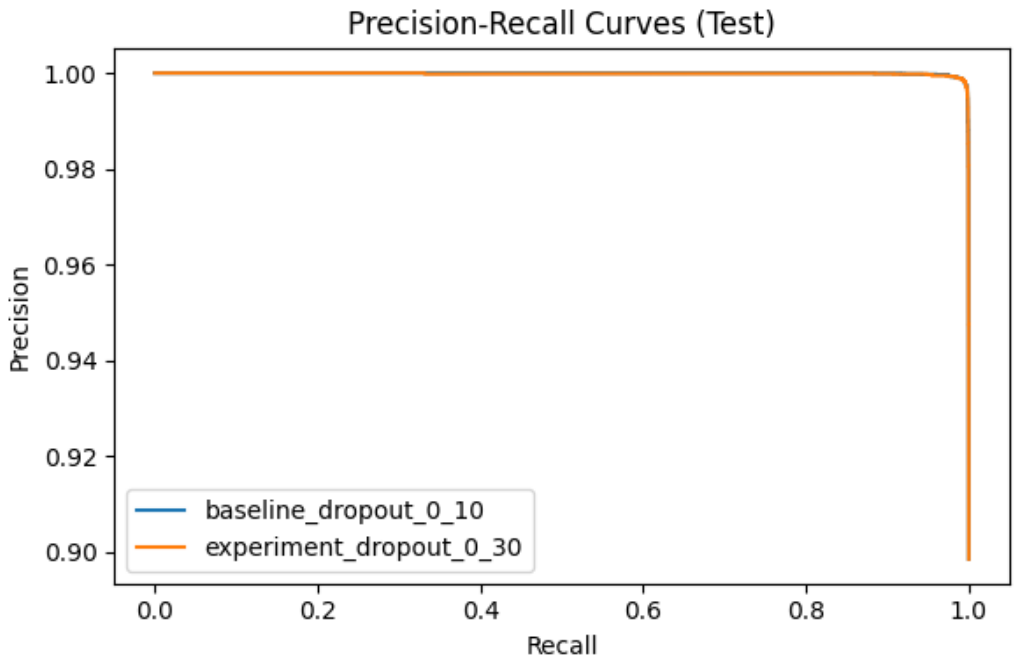

Figure 5.3: Test precision-recall comparison for deep-learning variants under class imbalance.

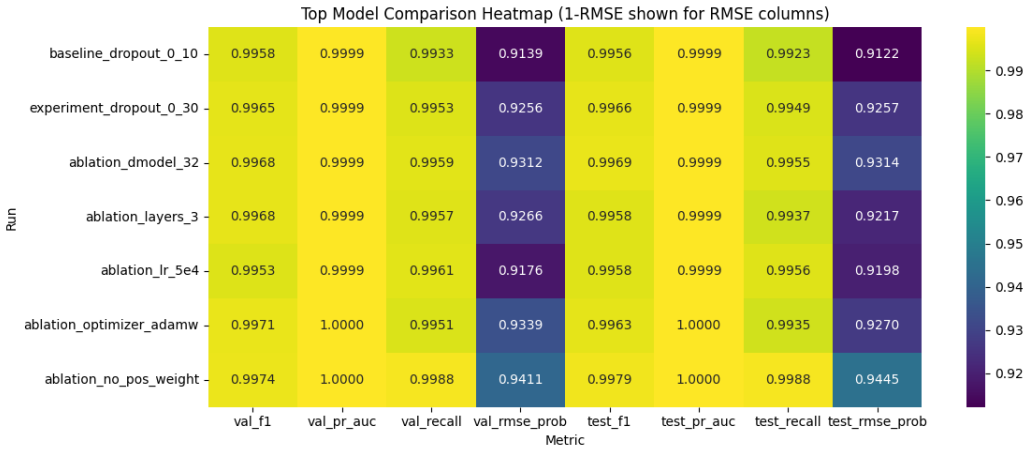

Figure 5.4: Cross-model heatmap summarizing ensemble and ablation performance.

5.8 Governance and Responsible Use

AIF360 subgroup screening on protocol cohorts (for example is_proto_tcp) flagged non-trivial disparity [17, 18]. NIST-aligned readiness checks were used as acceptance gates, including recall/FPR targets, probability quality, reproducibility checks, leakage token scans, and fairness-audit execution [1].

This chapter therefore contributed a deep-learning pattern that is technically and operationally grounded: strong metrics are necessary but not sufficient; deployment claims require subgroup and policy evidence.

5.9 Shift Monitoring and Retraining Triggers

For production deployment, this chapter's model should run under explicit drift governance. Recommended trigger rules are:

- feature-distribution drift alarm when key telemetry channels exceed agreed divergence bounds,
- calibration drift alarm when rolling Brier score degrades beyond policy tolerance,
- class-conditional error alarm for persistent attack-family miss concentration,
- mandatory retraining candidate review when two or more alarms persist across consecutive windows.

This converts distribution-shift risk from a narrative limitation into an actionable operational control loop.

5.10 Data Lineage and Integration Contract

This chapter exports telemetry prevalence and representation-quality signals that are consumed in the integrated packet workflow described in Chapter 8. The integration contract is role-based: deep-learning outputs provide detection-context evidence and uncertainty cues, while final control authority remains in policy-gated orchestration layers.

As a result, this chapter's outputs are used to strengthen branch evidence packets rather than to authorize actions directly.

5.11 Standards and Assurance Crosswalk

This chapter's deep-learning controls are mapped to industrial references as traceability artifacts:

- **NIST CSF 2.0 (DE/RS/GV):** detection quality, threshold governance, and auditable acceptance criteria [1].
- **NIST SP 800-82r3:** operationally safe IDS behavior under industrial constraints and shift-aware monitoring [2].
- **IEC 62443:** supports secure operation intent by emphasizing bounded detection behavior and policy-mediated downstream use [11].
- **CMMC 2.0:** reproducible training/evaluation and documented readiness gates support process maturity evidence [12].
- **NIST AI RMF 1.0:** model-risk measurement includes subgroup disparity and calibration diagnostics [3].
- **IEEE 1012:** controlled ablation and fixed-factor experimentation strengthen verification integrity for model changes [14].

5.12 Limitations

The main limitations were distribution shift risk, binary label simplification (loss of attack-family granularity), and threshold-policy sensitivity. In production, these factors can dominate real-world behavior despite strong in-distribution benchmark scores.

5.13 Lessons Learned

1. **Controlled ablations are mandatory in deep systems.** They provide causal clarity for architecture decisions.
2. **Threshold policy is part of system design.** It should be documented and validated like any other configuration.
3. **Error topology beats headline metrics.** Family-level misses can define risk exposure even when macro metrics are excellent.

5.14 Bridge to Chapter 6

Deep models improved detection power but did not explain incident context in analyst-ready language. Chapter 6 addresses that gap by introducing a generative RCA layer that converts structured telemetry sequences into auditable hypothesis drafts while preserving explicit uncertainty and governance constraints.

6 Generative RCA Layer for Cyber Telemetry Narratives

6.1 Why a Generative Layer Was Needed

The fifth project introduced a generative component to bridge an important gap: predictive models can rank risk, but incident response teams still need structured narrative hypotheses. The goal was to generate sequence-level RCA drafts from LANL-style cybersecurity telemetry while maintaining auditable, bounded behavior [10].

This chapter did not treat generation as a creative writing task. It framed generation as evidence-linked narrative synthesis for operations.

This chapter contributes to the thesis assurance case as the **explanation synthesis layer**: it produces bounded RCA narrative candidates that remain explicitly subordinate to deterministic policy, evidence validation, and human oversight.

6.2 GitHub Project and Report

Project repository: https://github.com/Ohara124c41/generative-modeling-lanl-rca

Report artifact: https://github.com/Ohara124c41/generative-modeling-lanl-rca/blob/main/Generative_AI_Analysis_Report_draft.md

6.3 Related Work and Use-Case Positioning

The chapter aligns with foundation-model literature that treats generative systems as high-capacity but risk-sensitive components requiring explicit boundary controls, uncertainty framing, and downstream validation [35]. It also follows sequence-generation quality work showing that token-level fit and narrative usefulness can diverge under repetition and degeneration pressure [36, 37].

Within the thesis use-case set, this chapter primarily addresses **UC-3 (RCA explainability and evidence traceability)** and **UC-6 (auditability)** by producing confidence-labeled narrative hypotheses that are provenance-bound and never promoted as autonomous causal truth.

6.4 Generation Pipeline Overview

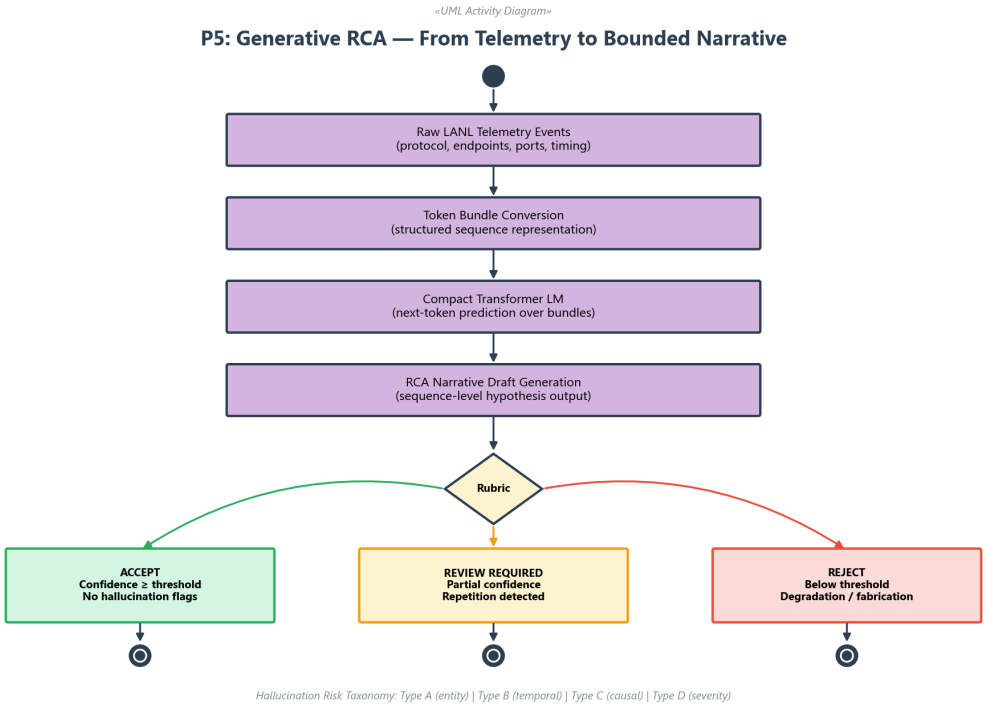

Figure 6.1: UML activity diagram for the generative RCA pipeline. Raw LANL telemetry events are converted to structured token bundles, processed by a compact Transformer language model for next-token prediction, and emitted as RCA narrative drafts. A rubric-based quality gate classifies each output as Accept (evidence-linked, no hallucination flags), Review Required (minor ambiguity), or Reject (fabrication, sequence distortion, or action overreach). The hallucination-risk taxonomy (Types A–D) governs rejection criteria.

6.5 Data Representation and Model Choice

Raw event records were transformed into structured token bundles (protocol, endpoints, ports, traffic magnitudes, and timing attributes). This converted heterogeneous telemetry into a sequence-generation problem.

A compact Transformer language model was selected for next-token prediction over these structured sequences. The model class was chosen for fit with sequential dependency structure and inspectable token-level outputs [32].

6.6 Experimental Design and Ablations

Baseline training used a compact configuration, then single-factor ablations varied dropout, model size, and learning rate while holding preprocessing, splits, seed, and metric logic fixed. A best short-run configuration was then extended to verify whether early gains persisted over longer training.

This two-stage protocol allowed model selection from evidence rather than one-off runs.

6.7 Quality Evaluation

The evaluation stack intentionally combined fit and generation diagnostics:

- validation loss and perplexity behavior,
- distinct-n diversity metrics,
- repetition ratio,
- structured RCA confidence scoring.

The selected model showed clear early-epoch advantages but still exhibited late-epoch degradation, indicating overtraining risk. Generated outputs demonstrated usable structural coherence but notable motif repetition. Diversity metrics reflected moderate novelty with substantial recurrence [36, 37].

The key interpretation was practical: outputs were suitable for analyst hypothesis scaffolding, not autonomous causal truth assignment.

6.8 Key Visual Diagnostics

6.9 Governance and Risk Controls

The chapter included a fairness-risk screening pass, confidence labeling, and export contracts for downstream auditability [17, 18]. Generated artifacts were saved in both CSV and JSONL with fields for sequence text, candidate hypothesis, confidence, and run metadata.

This export discipline is architecturally important: the generative layer became a reusable component for later agentic orchestration rather than a notebook-only result.

6.10 Hallucination-Risk Taxonomy and RCA Acceptance Rubric

To support safer operational use, generative outputs should be screened against a structured risk taxonomy before analyst presentation:

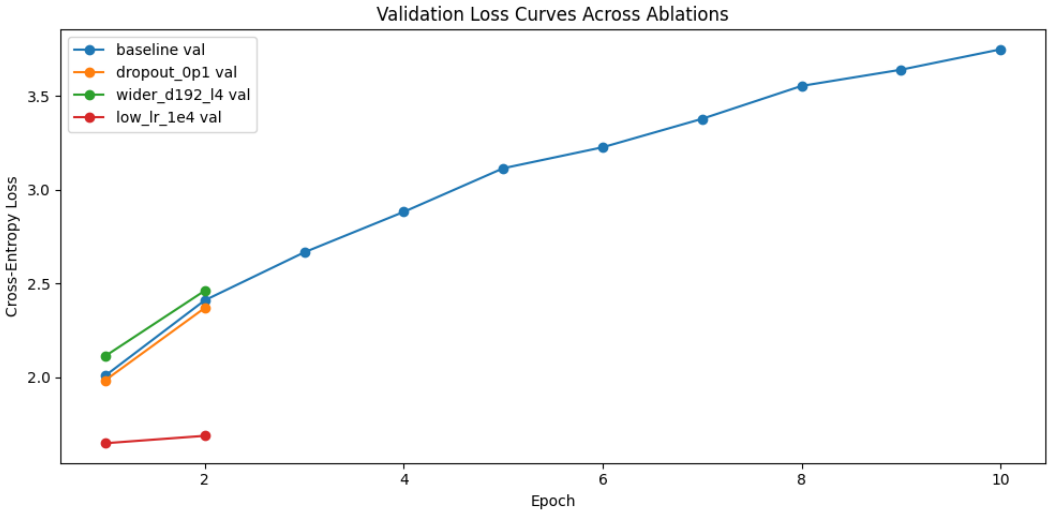

Figure 6.2: Validation loss across generative ablations used for model-selection filtering.

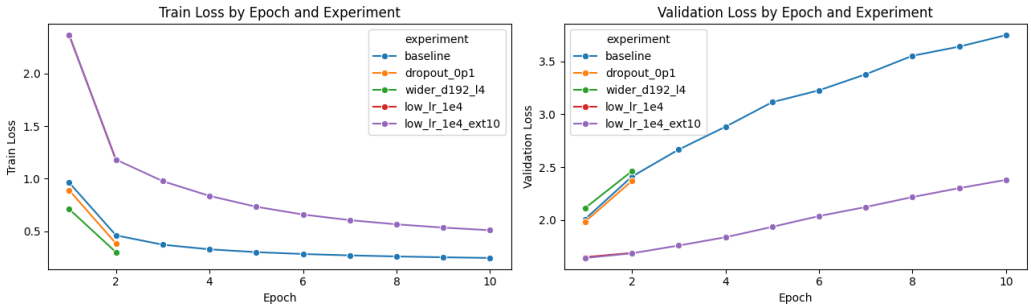

Figure 6.3: Relative validation-loss improvement over baseline configuration during ablation review.

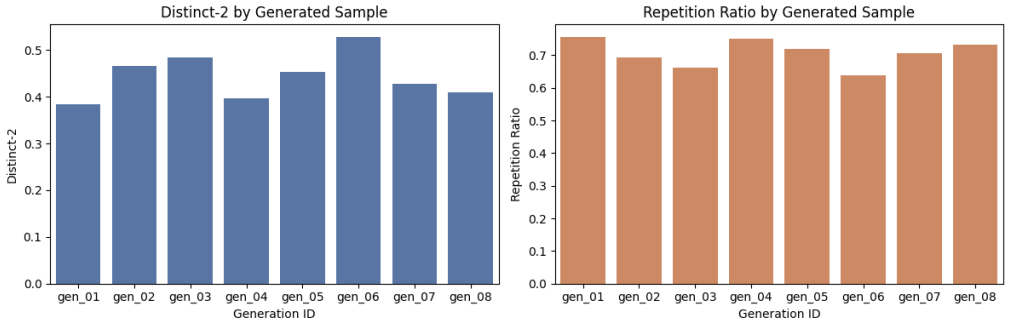

Figure 6.4: Distinct-2 diversity trend by generated sample, indicating repetition pressure in sequence outputs.

- **Type A – Unsupported causal claim**: narrative asserts cause without packet evidence link.
- **Type B – Entity fabrication**: host/user/process references not present in source artifacts.
- **Type C – Sequence distortion**: event ordering inconsistent with timestamped telemetry.
- **Type D – Action overreach**: recommendation exceeds policy or role authority.

Rubric Level	Minimum Condition	Allowed Use	Fallback
Accept	Evidence-linked, no Type A–D findings, confidence above threshold	Analyst draft assist	Pass to orchestration packet
Review Required	Minor ambiguity but no policy overreach	Human verification only	Escalate to evidence agent review
Reject	Any Type D or repeated Type A/B/C pattern	Do not use for RCA support	Refuse output and log audit event

Table 6.1: RCA output acceptance rubric for generative safety control.

6.11 Data Lineage and Integration Contract

The outputs of this chapter are integrated as structured generative RCA summaries and tags in the Chapter 8 packet contract. The intended role is analyst-assistive hypothesis scaffolding: generated narratives enrich branch reasoning packets but never serve as standalone causal proof.

The contract boundary is explicit: generated content is always provenance-labeled and confidence-scored, and it is subject to downstream policy and evidence-gating checks before use in adjudication.

6.12 Standards and Assurance Crosswalk

The generative layer is mapped to industrial governance and AI assurance references as follows:

- **NIST CSF 2.0 (DE/RS/GV):** supports incident response interpretation while preserving governance visibility over generated explanations [1].
- **NIST SP 800-82r3:** aligns with operator-assistive use in ICS environments where unsafe automation is unacceptable [2].
- **IEC 62443:** reinforces least-authority behavior by keeping generated text outside direct control execution pathways [11].

- **CMMC 2.0:** structured, auditable exports strengthen process evidence for incident-analysis workflows [12].
- **NIST AI RMF 1.0 (GOVERN/MAP/MEASURE/MANAGE):** confidence labels, repetition diagnostics, and misuse-boundary disclosure operationalize generative risk control [3].
- **IEEE 7000 and IEEE 1012:** ethical risk articulation plus verification-oriented output contracts support responsible system integration [13, 14].

6.13 Limitations

The main limitations were bounded training scale, persistent repetition under heavy-tail traffic patterns, and heuristic RCA scoring rather than validated causal inference. The project explicitly avoided claiming that generated narratives were sufficient for automated response.

6.14 Lessons Learned

1. **Generative quality is multi-dimensional.** Low validation loss alone does not guarantee operationally useful output diversity.
2. **Sequence token design is a governance decision.** Token schema determines what the model can and cannot express about incidents.
3. **Artifacts must be contract-ready.** Structured exports make generative components composable within larger systems.

6.15 Bridge to Chapter 7

Chapter 6 produced explanation candidates but left open a core systems question: who decides what to do with those explanations? Chapter 7 addresses this by introducing an orchestrated multi-agent workflow with explicit handoffs, policy gates, and auditable decision packets.

7 Agentic Orchestration with Governance-First Controls

7.1 Problem Framing

The sixth project addressed an operational gap left by the prior chapters: analytics and generated hypotheses existed, but decision flow remained implicit. The chapter implemented a multi-agent system for incident triage and RCA support with explicit orchestration, bounded tool use, and policy-enforced output gating.

The system was intentionally semi-autonomous. It supported analysts with structured recommendations and traceable reasoning while preserving human authority for high-impact actions.

This chapter contributes to the thesis assurance case as the **control-orchestration layer**: it operationalizes deterministic governance boundaries around adaptive reasoning components.

7.2 GitHub Project and Report

Project repository: https://github.com/Ohara124c41/agentic-ai-mitre-attack-rca

Report artifact: https://github.com/Ohara124c41/agentic-ai-mitre-attack-rca/blob/main/Agentic_AI_System_Design_Report_draft.md

7.3 Related Work and Use-Case Positioning

This chapter is positioned in relation to tool-using and multi-agent LLM orchestration patterns, where role specialization improves task decomposition but also creates expanded safety and authority surfaces [4, 38]. In operational cybersecurity contexts, incident-handling guidance emphasizes procedural control, escalation discipline, and evidence integrity as primary safety invariants [39].

Accordingly, this chapter most directly targets **UC-4 (prompt/tool safety)**, **UC-5 (change-control integrity)**, and **UC-6 (auditability)** by ensuring that adaptive reasoning remains subordinate to deterministic runtime policy gates and human authority.

7.4 Architecture

The runtime used one orchestrator and five specialized agents:

1. Intake
2. Evidence
3. RCA
4. Response Planner
5. Governance

Shared state was stored in a typed incident object. Each stage consumed and enriched the same state, then emitted logs and snapshots for replay.

Tooling included scenario loaders, ATT&CK mapping utilities, governance validators, audit writers, and an optional Vocareum-compatible LLM refinement client [30]. Deterministic and LLM-enabled modes were both supported.

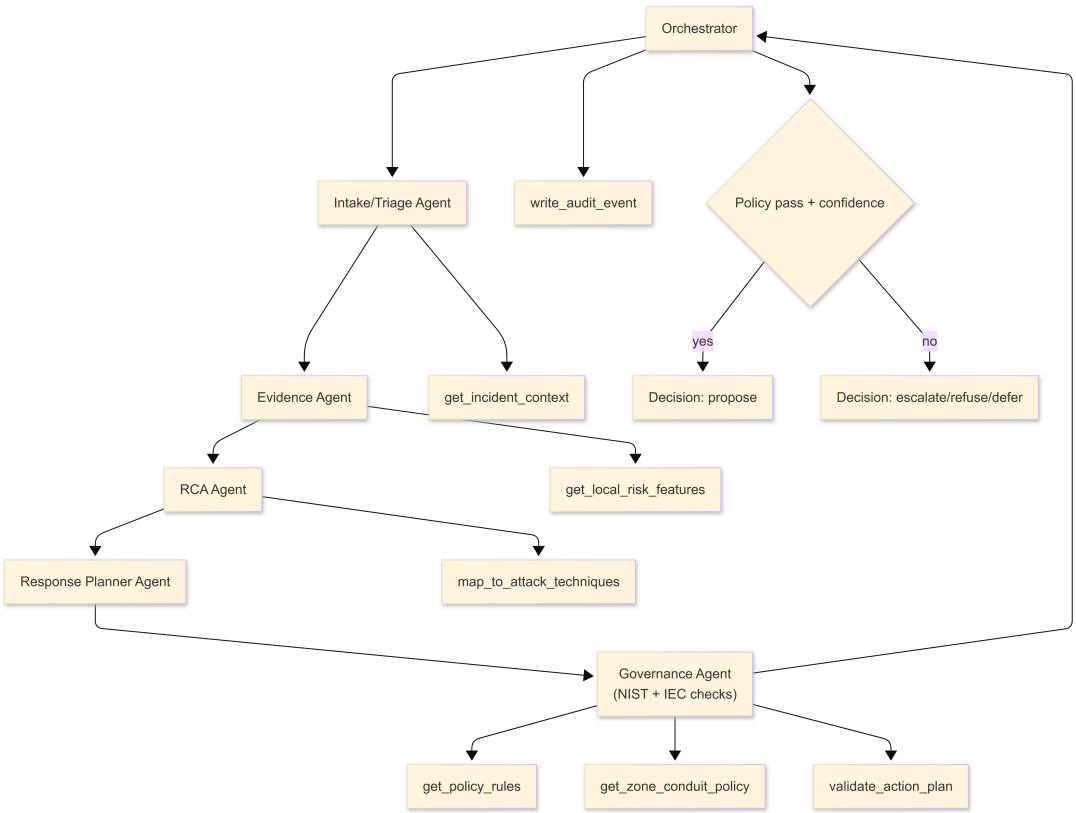

Figure 7.1: P6 multi-agent architecture with orchestrator boundary, specialized agent roles, and governance/tooling layers.

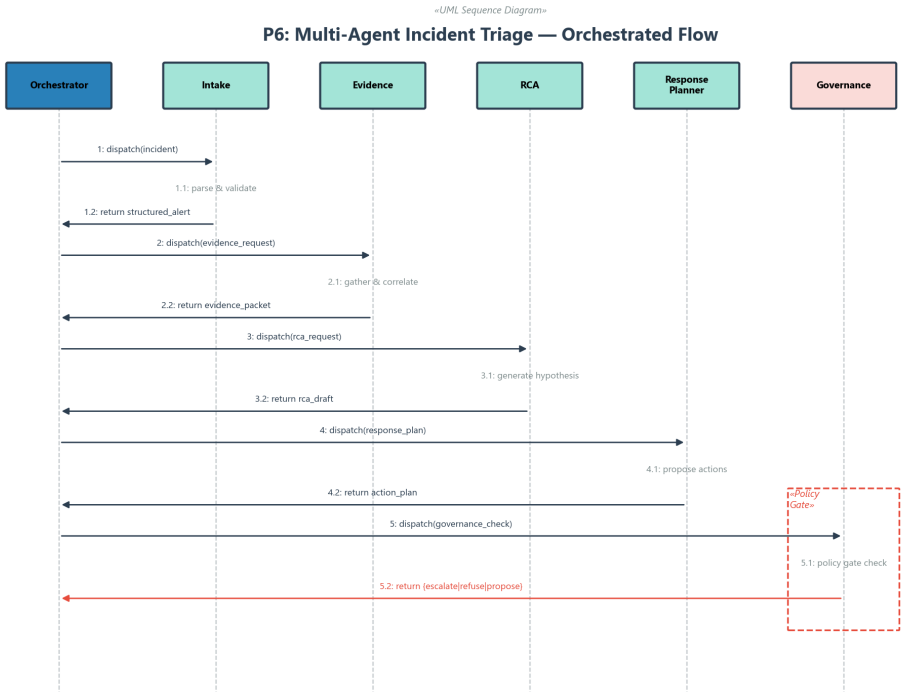

Figure 7.2: UML sequence diagram showing the orchestrated message flow for a single incident through the five specialized agents. The Orchestrator dispatches tasks sequentially to Intake, Evidence, RCA, Response Planner, and Governance agents, each returning typed results. The Governance agent applies policy-gate checks and emits a final decision (escalate, refuse, or propose) with audit linkage.

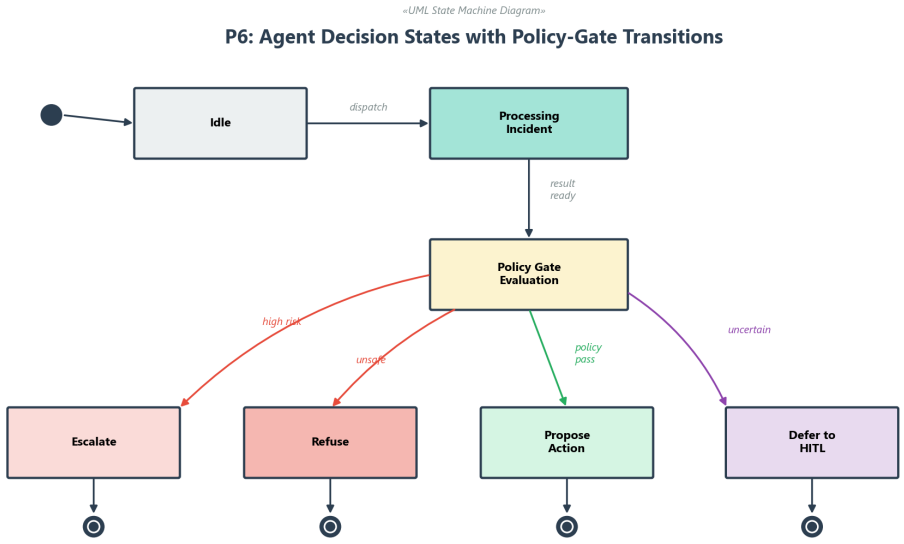

Figure 7.3: UML state machine diagram for agent decision transitions. After dispatching and processing, the policy-gate evaluation determines one of four terminal states: Escalate (high risk), Refuse (unsafe context), Propose Action (policy pass), or Defer to HITL (uncertain conditions). Each transition is governed by deterministic policy rules rather than model confidence alone.

7.5 Safety and Transparency Design

Safety controls included prompt-injection detection, allow-list action checks, IEC zone-conduit constraints, confidence/evidence gating, escalation requirements, and refusal pathways for unsafe instruction patterns [4, 11].

Transparency controls included JSONL audit events, per-incident state snapshots, and LLM trace files with incident linkage. These artifacts made every major decision inspectable post hoc.

7.6 Failure-Mode Matrix and Policy-Gate Fallbacks

7.7 Evaluation

On the reported run (5 scenarios), outcomes were conservative by design:

- decisions: 4 escalate, 1 refuse,
- policy pass rate: 0.4,
- mean evidence quality: 0.74,

Agent Role	Primary Failure Mode	Gate Detection	Fallback Behavior
Intake	malformed scenario payload or missing required fields	schema/integrity validator	reject packet, request re-ingest, log refusal event
Evidence	unsupported ATT&CK mapping or low trace linkage	evidence-quality threshold	escalate to analyst with evidence-gap marker
RCA	hallucinated causal chain or sequence distortion	provenance and confidence checks	suppress narrative, keep deterministic summary only
Response Planner	action proposal outside allow-list/zone constraints	policy allow-list and IEC checks	block action path, emit containment-only recommendation
Governance	inconsistent policy verdict across checks	policy consistency validator	force HITL adjudication and freeze auto-progress

Table 7.1: Failure-mode matrix with policy-gate fallback behavior by agent role.

- mean hypothesis confidence: 0.8187.

Handoff analysis showed full stage coverage. Latency was concentrated in the RCA stage when LLM refinement was enabled.

ATT&CK mapping coverage included six techniques and high technique-hit rates in benchmark checks. Label-level disambiguation accuracy remained a realistic improvement area.

7.8 Key Visual Diagnostics

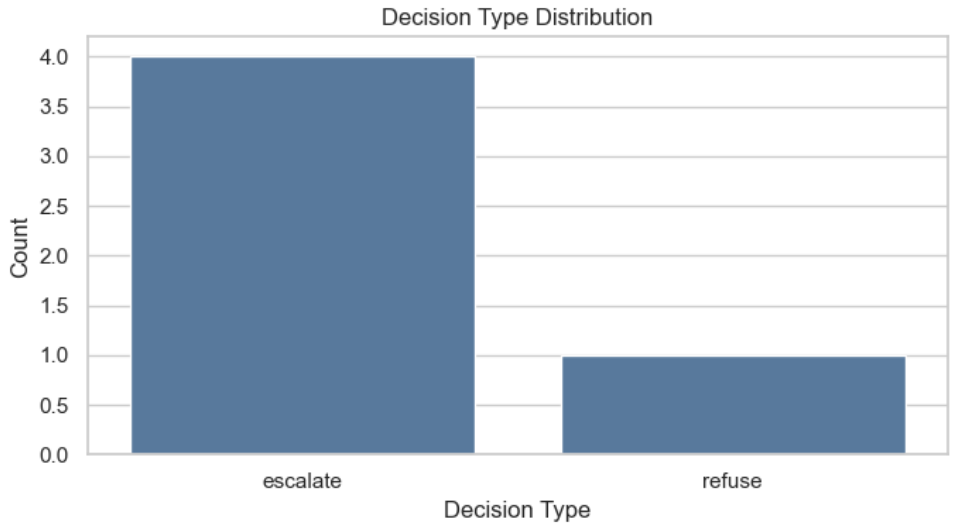

Figure 7.4: Agentic decision distribution across escalate/refuse/propose/defer outcomes.

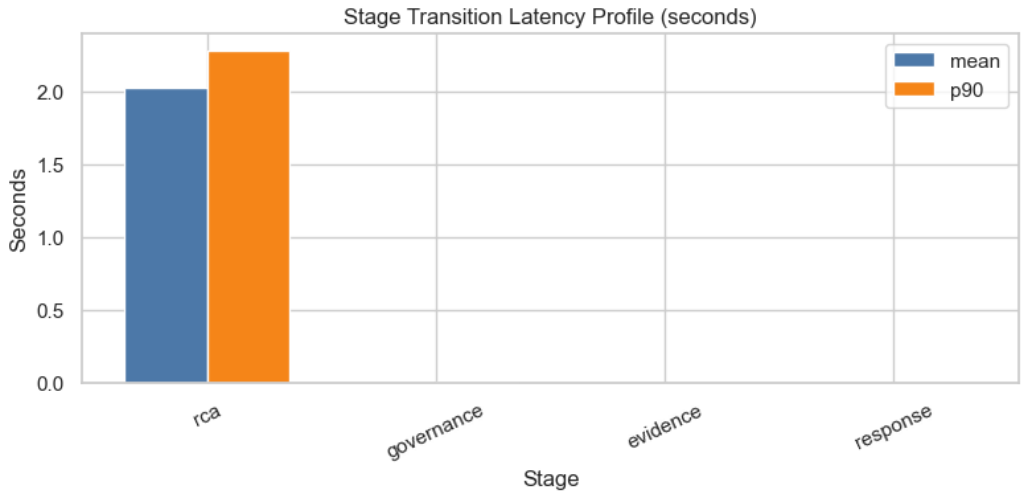

Figure 7.5: Stage transition latency profile across the orchestration pipeline.

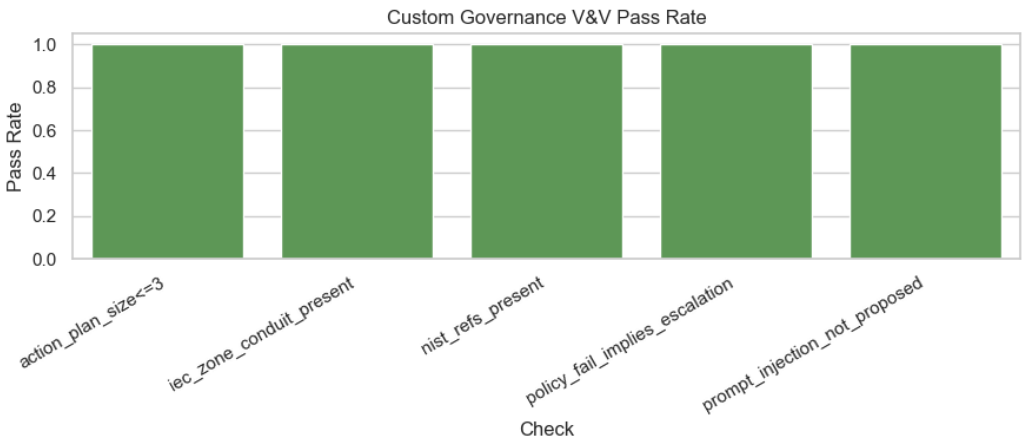

Figure 7.6: Governance V&V pass-rate summary under policy and safety checks.

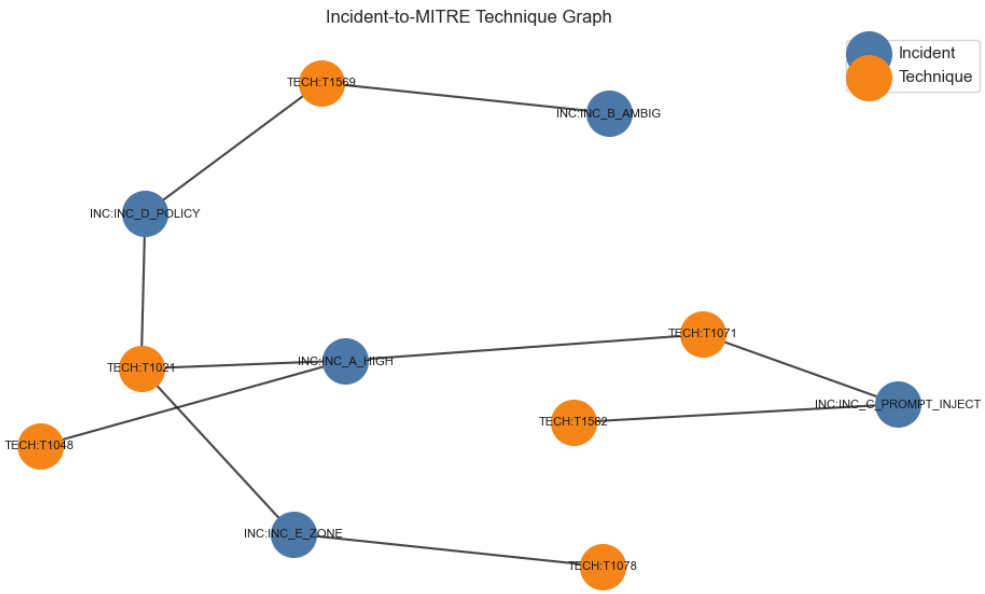

Figure 7.7: Incident-to-MITRE technique graph for evidence-linked threat mapping in the agent workflow.

7.9 Baseline vs Experimental Behavior

A structured deterministic-versus-LLM comparison showed that enabling LLM refinement increased wording diversity while leaving governance-relevant outcomes unchanged on the same scenarios. This result validated the architecture boundary: control authority remained deterministic; language quality was the optional adaptive layer [4].

7.10 Data Lineage and Integration Contract

This chapter consumes P3–P5 outputs as contextual evidence and emits governance-ready decision packet fields used directly in Chapter 8. The contract includes policy status, escalation/refusal markers, confidence/evidence quality signals, and audit linkage identifiers.

Given the pilot sample size (five scenarios), these outputs are positioned as proof-of-architecture evidence, not production-rate performance claims.

7.11 Standards and Assurance Crosswalk

The agentic control design is mapped to industrial references as traceability evidence:

- **NIST CSF 2.0 (GV/DE/RS):** governance-constrained triage and response support with explicit escalation pathways [1].

- **NIST SP 800-82r3:** safety-first control logic for ICS-relevant incident handling contexts [2].
- **IEC 62443:** zone-conduit aware constraints and least-authority action boundaries [11].
- **CMMC 2.0:** auditable workflow execution, policy validation, and trace preservation [12].
- **MITRE ATT&CK / ATT&CK for ICS alignment intent:** adversary-technique mapping for structured analyst interpretation [30].
- **NIST AI RMF 1.0 (GOVERN/MANAGE):** prompt-injection handling, refusal rules, and uncertainty gating as AI risk controls [3].
- **IEEE 7000 and IEEE 1012:** explicit authority boundaries and replayable decision evidence support responsible system validation [13, 14].

7.12 Limitations

The primary limitation was sample size. With five scenarios, stability of fairness and subgroup conclusions is limited. The system also showed conservative decision behavior with few propose outcomes, indicating room for calibrated confidence thresholds.

7.13 Lessons Learned

1. **Agent count is less important than role clarity.** Five focused agents plus one orchestrator delivered traceable behavior without unnecessary complexity.
2. **Policy gates must be first-class runtime components.** Governance implemented as post-processing is too weak for safety-critical contexts.
3. **Deterministic control plus optional LLM enrichment is a robust pattern.** It combines reliability with expressive analyst support.

7.14 Bridge to Chapter 8

Chapter 7 established trustworthy single-orchestrator behavior. The final step was system synthesis: integrate prior analytical artifacts (P3-P6), introduce parallel contested orchestration across competing objectives, and evaluate whether meta-level adjudication plus HITL controls can improve system-level assurance. That full integration is developed in Chapter 8.

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

8.1 Introduction and Industry Framing

Industrial cyber operations rarely fail because teams lack isolated analytics. They fail when high-stakes decisions must be made under uncertainty across competing objectives: security assurance, operational continuity, governance compliance, and human accountability. This chapter addresses that integration problem as a system-design challenge.

The implemented solution is an assurance-centered agentic AIOps workflow for local-cloud IIoT/OT contexts. It does not claim fully autonomous response. Instead, it provides bounded, auditable decision support that integrates statistical priors, ML/deep risk context, generative RCA artifacts, and agentic orchestration under explicit policy gates.

The architectural thesis is straightforward: in safety-critical cyber operations, decision quality is a property of *composed workflows* rather than any single model. Therefore, design emphasis is placed on interface contracts, traceability, disagreement surfacing, and human review triggers.

This chapter contributes to the thesis assurance case as the **system adjudication layer**: it demonstrates how multi-objective branch outputs are reconciled under deterministic controls and human accountability gates.

8.2 GitHub Project and Report

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

Report artifact: https://github.com/Ohara124c41/integrated-industrial-application-acaa/blob/main/Reflective_Synthesis_Paper.pdf

8.3 Related Work and Use-Case Positioning

This chapter extends prior work on orchestrated AI workflows by introducing contested multi-objective adjudication rather than a single optimization path. Its architectural lineage draws from concurrent systems-engineering patterns where subsystem recommendations are preserved and adjudicated under explicit cross-cutting constraints [5, 40]. It is also aligned with

governance-oriented AI risk frameworks that require authority boundaries, traceability, and lifecycle verification evidence in high-impact decision systems [3, 14].

In use-case terms, this chapter integrates all six UC issues and operationalizes their reconciliation: UC-1/UC-2 through integrated evidence packets, UC-3 through RCA-linked packet semantics, UC-4/UC-5 through deterministic gating and HITL override policy, and UC-6 through replayable adjudication traces.

8.4 From Single-Pipeline Orchestration to Parallel Contested Orchestration

A central contribution of this chapter is the transition from a single orchestrator (Chapter 7) to a parallel contested structure with meta-level adjudication. Two orchestrator branches process the same evidence:

- **Security-assurance branch**: optimized for containment confidence, threat coverage, and low false negatives.
- **Operations-continuity branch**: optimized for bounded disruption, service stability, and controlled intervention cost.

A meta-orchestrator compares branch packets under shared constraints and determines one of three outcomes:

1. select assurance branch,
2. select continuity branch,
3. require human-in-the-loop (HITL) adjudication.

This is not a debate mechanism where peer agents argue a single answer. It is a multi-objective adjudication mechanism where different branches are intentionally allowed to disagree. That disagreement is treated as signal, not error.

8.5 System Architecture Views

8.6 Architectural Lineage and Design Rationale

The pattern implemented here follows a lead-architect adjudication model: subsystem recommendations are preserved, but final decisions must satisfy cross-cutting constraints. This logic is directly aligned with the COGENT concurrent engineering lineage, where subsystem teams optimize local objectives and an architecture lead resolves tradeoffs at system level [5, 40].

The mapping to industrial cybersecurity is direct:

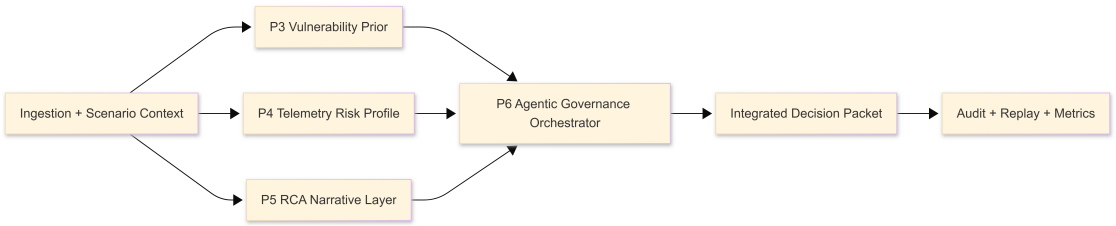

Figure 8.1: End-to-end system pipeline from upstream artifacts to adjudicated outputs in the integrated runtime.

- subsystem recommendations become branch decision packets,
- architecture governance becomes meta-orchestrator policy gates,
- design review boards become HITL override with accountability logs.

This lineage is more than historical context. It explains why the chapter prioritizes explicit interfaces, responsibility traces, and adjudication surfaces rather than monolithic assistant responses.

8.7 Integrated Artifact and Data Contracts

The runtime begins with artifact-availability checks and then constructs unified packets (integrated_packets.js by integrating prior-stage outputs:

- **P3**: vulnerability pressure priors (for example vuln_kev_rate, vuln_cvss_mean),
- **P4**: telemetry prevalence and representation signals,
- **P5**: structured generative RCA summaries and tags,
- **P6**: governance-ready decision packet fields and policy state.

This contract-based synthesis is the operational backbone of the chapter. Fields from prior projects are not simply concatenated for convenience; each is assigned a role in adjudication and audit logic.

The latest integrated run produced 5 packets with 25 fields. Storage outputs were materialized to both SQLite and Parquet to support analytic portability and engineering handoff.

8.8 Cross-Chapter Lineage Closure

This chapter closes the thesis evidence chain by explicitly preserving source-layer semantics in integrated adjudication packets:

- P1 contributes data-governance and provenance assumptions,

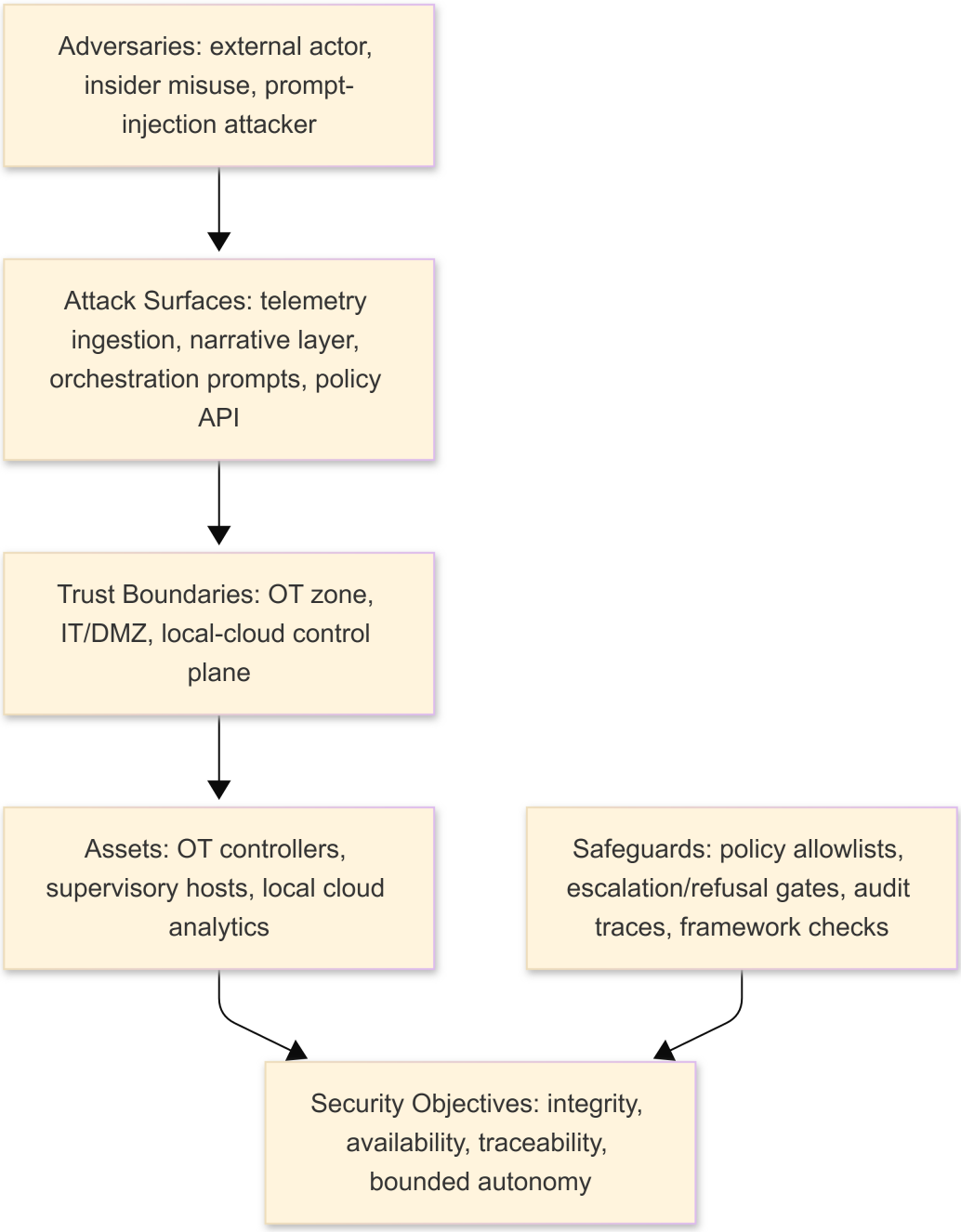

Figure 8.2: Threat model and safety boundary for contested orchestration with deterministic policy gates.

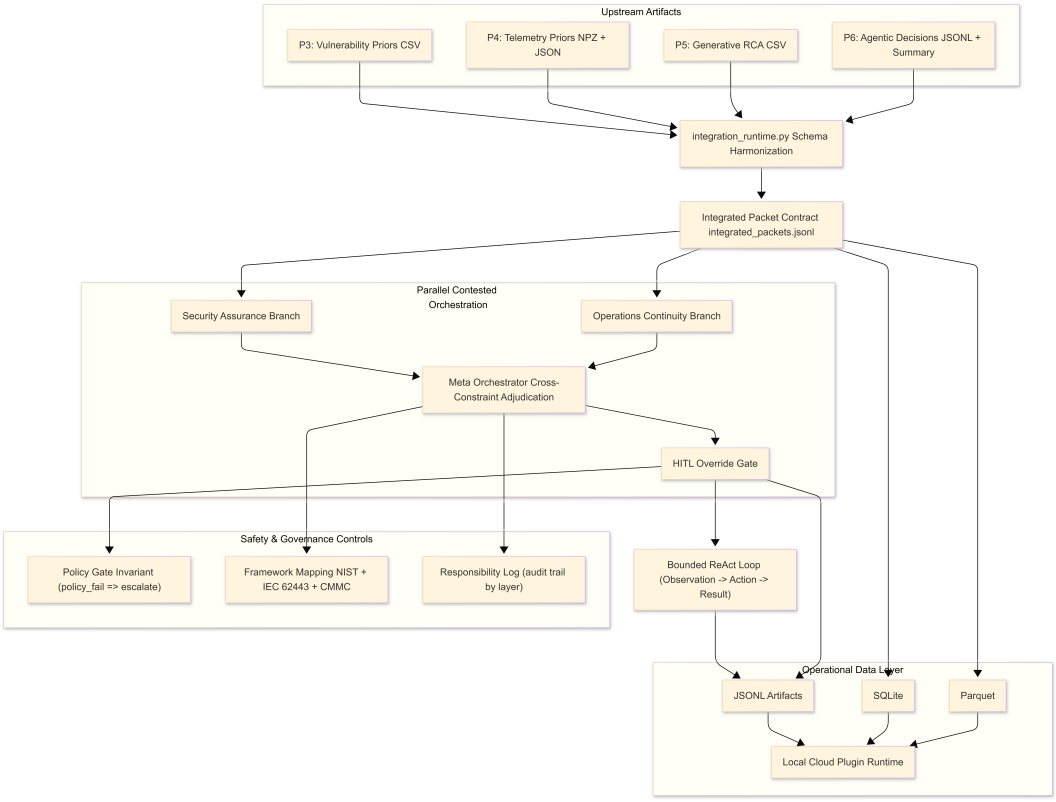

Figure 8.3: Component architecture view used for production-transition narrative and responsibility boundaries.

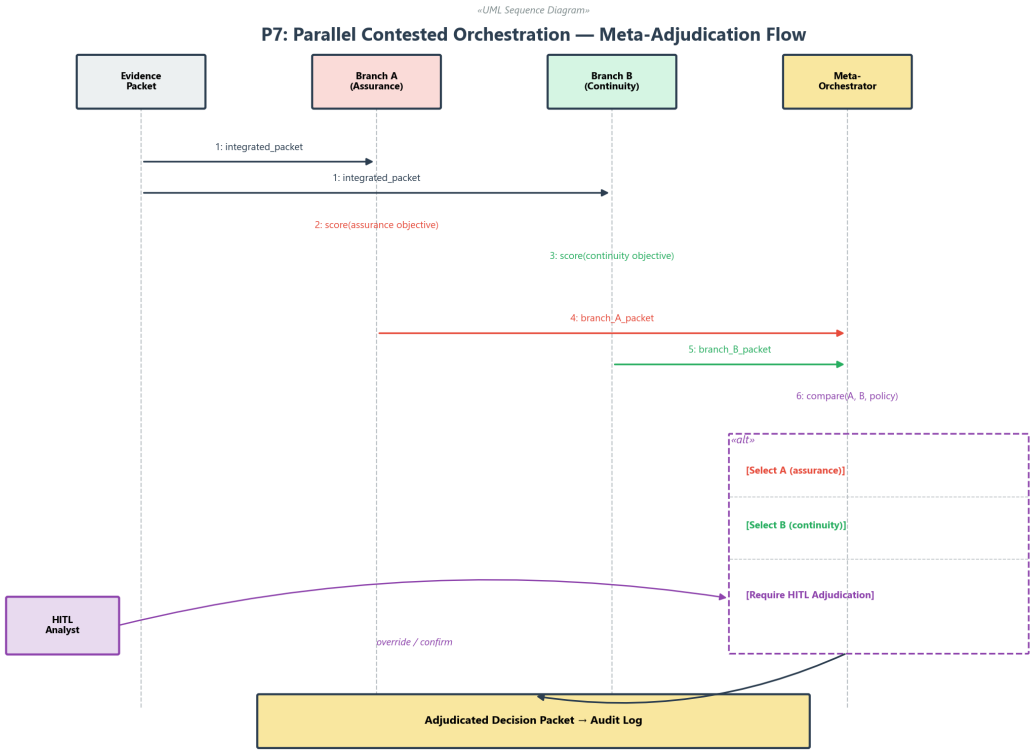

Figure 8.4: UML sequence diagram for parallel contested orchestration and meta-adjudication. An integrated evidence packet is dispatched simultaneously to Branch A (security-assurance objective) and Branch B (operations-continuity objective). Both branches return scored decision packets to the meta-orchestrator, which compares them under shared policy constraints and selects one of three outcomes: select Branch A, select Branch B, or require HITL adjudication. A human analyst can override or confirm contested decisions, and the final adjudicated packet is written to the audit log.

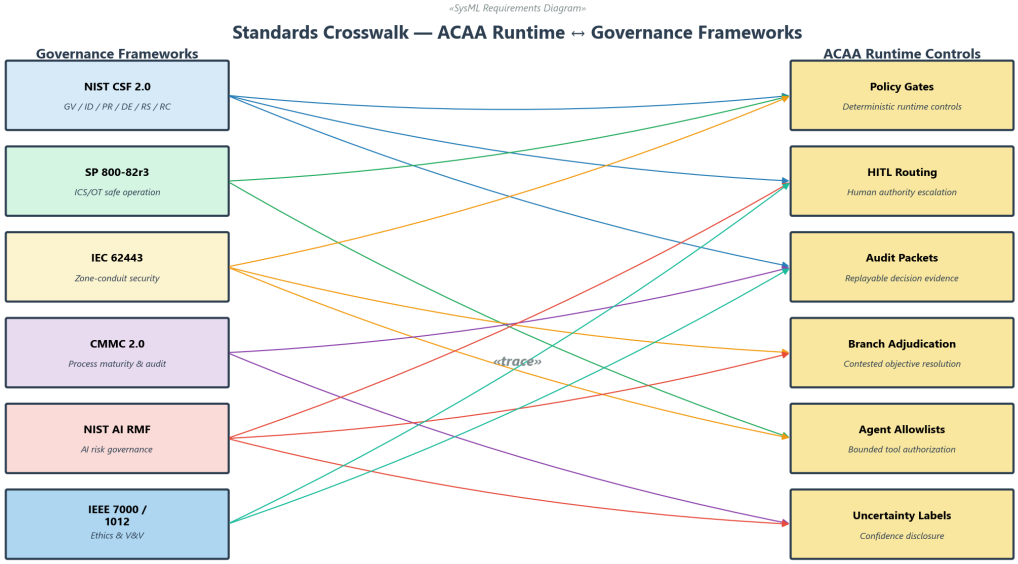

Figure 8.5: SysML requirements diagram mapping six governance frameworks (NIST CSF 2.0, SP 800-82r3, IEC 62443, CMMC 2.0, NIST AI RMF, IEEE 7000/1012) to six ACAA runtime control components (policy gates, HITL routing, audit packets, branch adjudication, agent allowlists, uncertainty labels) via trace relationships. This crosswalk supports architecture-level governance traceability for audit and design review.

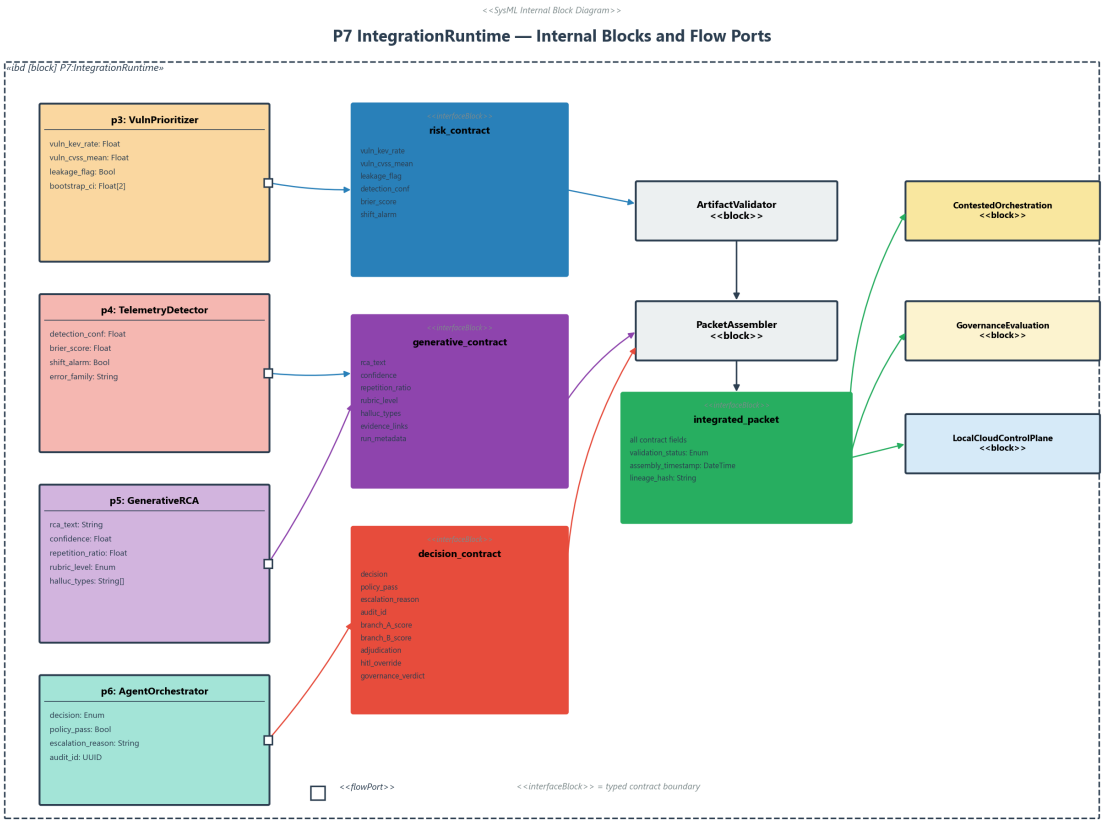

Figure 8.6: SysML internal block diagram of the P7 IntegrationRuntime showing four source parts (P3 VulnPrioritizer, P4 TelemetryDetector, P5 GenerativeRCA, P6 AgentOrchestrator) with typed flow ports connecting through three interface blocks (risk_contract, generative_contract, decision_contract). The ArtifactValidator and PacketAssembler subblocks produce integrated packets consumed by ContestedOrchestration, GovernanceEvaluation, and LocalCloudControlPlane.

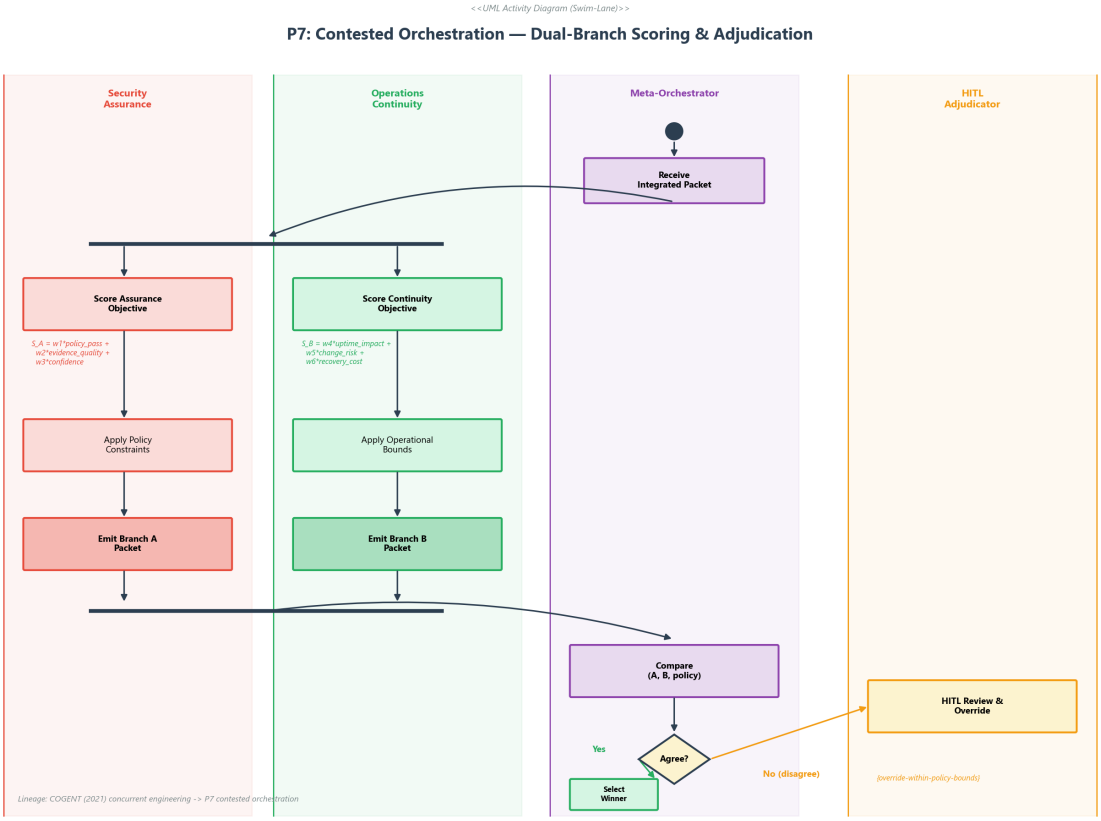

Figure 8.7: UML activity diagram with four swim lanes (Security Assurance, Operations Continuity, Meta-Orchestrator, HITL Adjudicator) showing the parallel dual-branch scoring workflow. After receiving an integrated packet, both branches independently score their objective functions with explicit formulas, apply policy constraints, and emit branch packets. A fork/join bar synchronizes branches before meta-orchestrator comparison, which produces either an automatic branch selection or HITL adjudication under override-within-policy-bounds constraints.

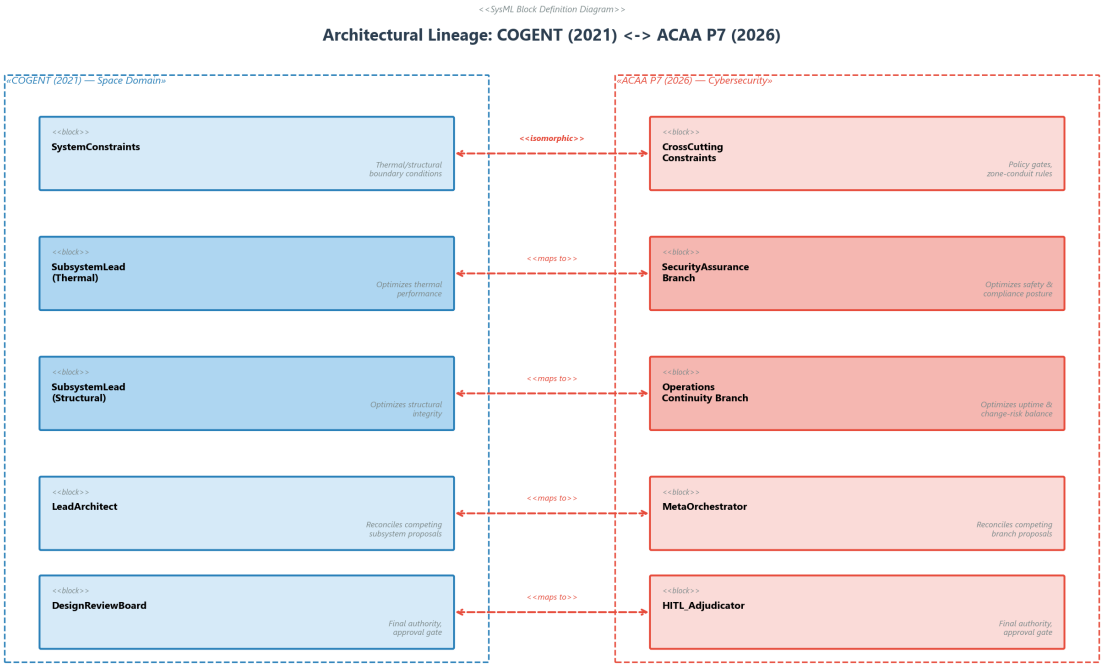

Figure 8.8: SysML block definition diagram showing the isomorphic architectural mapping between COGENT (2021, space domain) and ACAA P7 (2026, cybersecurity domain). Left: concurrent engineering roles (SystemConstraints, SubsystemLead_Thermal, SubsystemLead_Structural, LeadArchitect, DesignReviewBoard). Right: contested orchestration roles (CrossCuttingConstraints, SecurityAssurance, OperationsContinuity, MetaOrchestrator, HITL_Adjudicator). Dashed bidirectional arrows indicate isomorphic role mappings across domains.

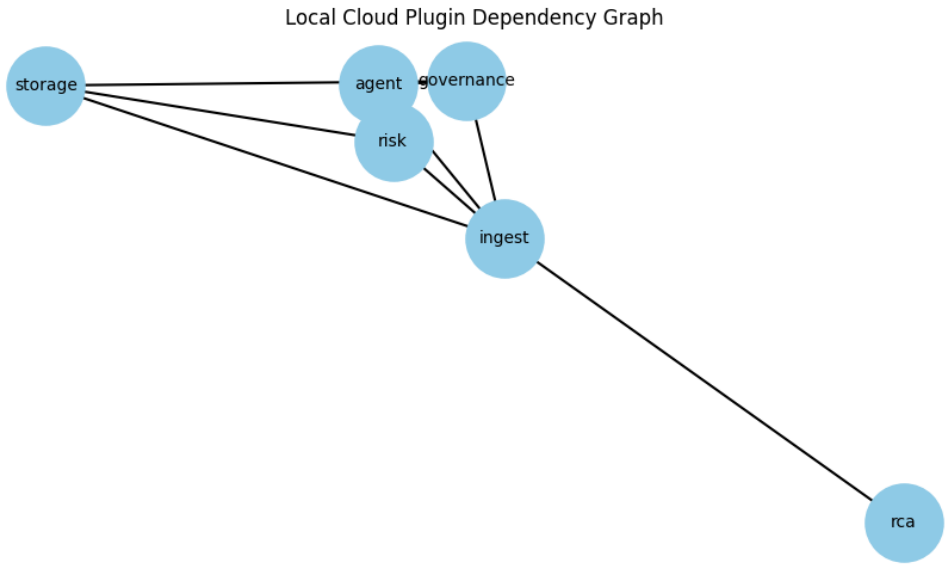

Figure 8.9: Local-cloud plugin dependency graph showing control-plane execution order and plugin coupling.

- P2 contributes governance-structure priors and uncertainty context,
- P3 contributes leakage-controlled vulnerability-prior signals,
- P4 contributes telemetry representation and detection-context evidence,
- P5 contributes confidence-labeled generative hypothesis artifacts,
- P6 contributes policy-state and agentic decision-control traces.

The integration claim is therefore contractual and auditable: each packet field has a defined origin role, and no single model output is treated as unilateral decision authority.

8.9 Threat Model, Safety Case, and Responsibility Boundaries

The system threat model is built around three classes of risk:

1. **Unsafe autonomy risk**: model or agent recommendations bypass policy and overreach authority.
2. **Instruction-channel manipulation**: prompt-injection or adversarial text alters recommendations.
3. **Governance drift**: branch-level optimization silently violates framework constraints.

Mitigations are encoded in runtime controls rather than narrative policy text:

- hard policy gate invariants,

- refusal/escalation patterns for unsafe contexts,
- HITL-required flags under branch disagreement thresholds,
- per-layer responsibility logging (branch A, branch B, meta, human override).

The safety claim is intentionally narrow: the system is suitable as a decision-support layer with explicit safeguards, not as an autonomous responder.

8.10 Evaluation Protocol

Evaluation in this chapter followed a layered protocol.

8.10.1 Layer 1: Runtime Integrity

Checks verified artifact availability, schema consistency, and successful packet generation before any higher-level analysis.

8.10.2 Layer 2: Deterministic vs LLM Comparison

A deterministic-versus-LLM comparison was run to isolate language enrichment effects from governance control effects. The key measured question was whether LLM assistance changed policy outcomes or only explanatory diversity [4].

8.10.3 Layer 3: Contested Branch Behavior

Branch score separation, disagreement frequency, and HITL trigger rates were evaluated to determine whether the parallel architecture produced meaningful differentiation.

8.10.4 Layer 4: Governance and Fairness Screening

Framework alignment views included NIST CSF, IEC 62443, and CMMC reference checks. AIF360-compatible parity screening was used across technical cohorts as an operational-risk indicator [1, 11, 12, 17, 18].

8.10.5 Layer 5: Exploratory Structure Diagnostics

EDA, non-parametric statistical checks, PCA, and t-SNE were retained as architecture diagnostics for packet-level structure. Interpretation was constrained by sample size.

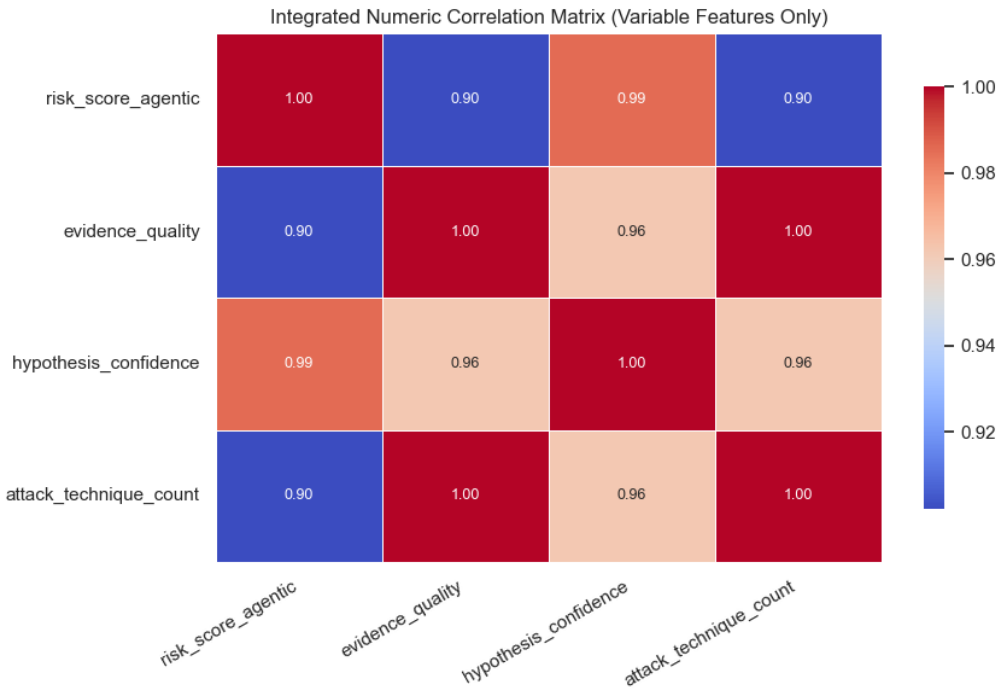

Figure 8.10: Integrated numeric correlation matrix for packet fields used in adjudication context.

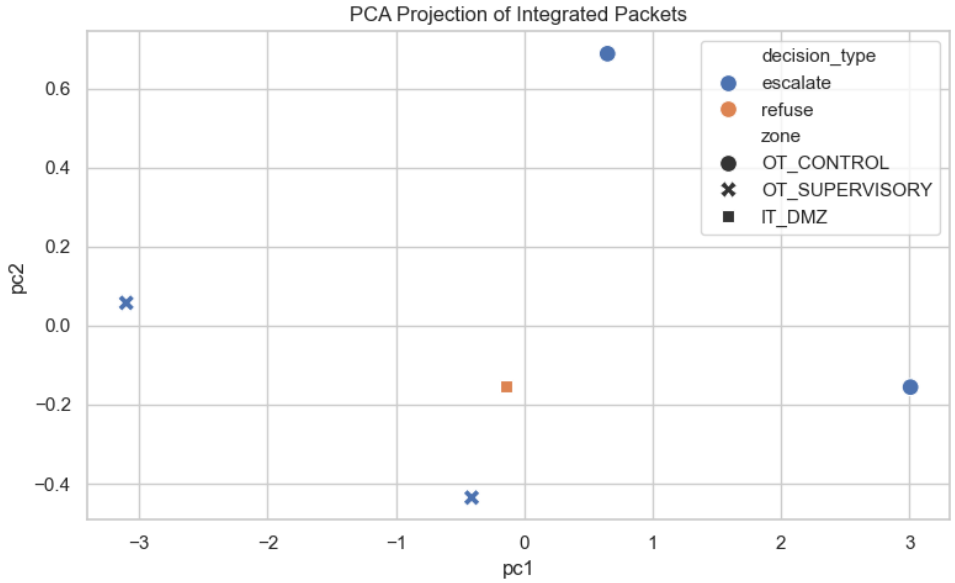

Figure 8.11: PCA projection of integrated packets as a low-n structural diagnostic view.

8.11 Empirical Results

8.11.1 Integrated System Baseline

On the current run:

- integrated packets: 5
- fields per packet: 25
- baseline decisions: 4 escalate, 1 refuse
- policy pass rate: 0.4
- mean hypothesis confidence: 0.8187
- mean ATT&CK technique count: 2.0

These values confirm that the system behaved conservatively, with escalation favored when uncertainty or policy pressure was high.

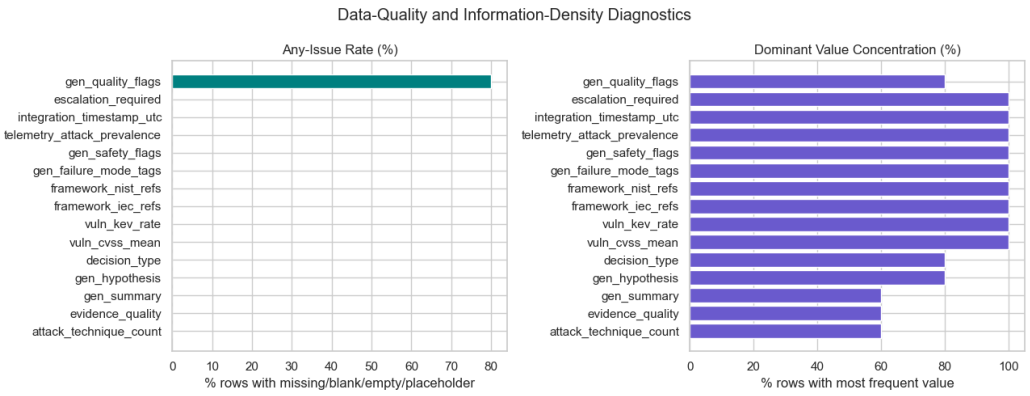

Figure 8.12: Contract-critical field coverage and issue-rate diagnostics for integrated packet quality checks.

8.11.2 Contested Orchestration Outputs

The contested layer produced non-trivial branch differentiation with:

- disagreement rate: 0.2
- HITL-required rate: 0.4
- branch selection split: assurance 3, continuity 2

This is a positive result for the architecture. A disagreement rate of exactly zero would suggest one branch is redundant or both branches encode identical objective surfaces.

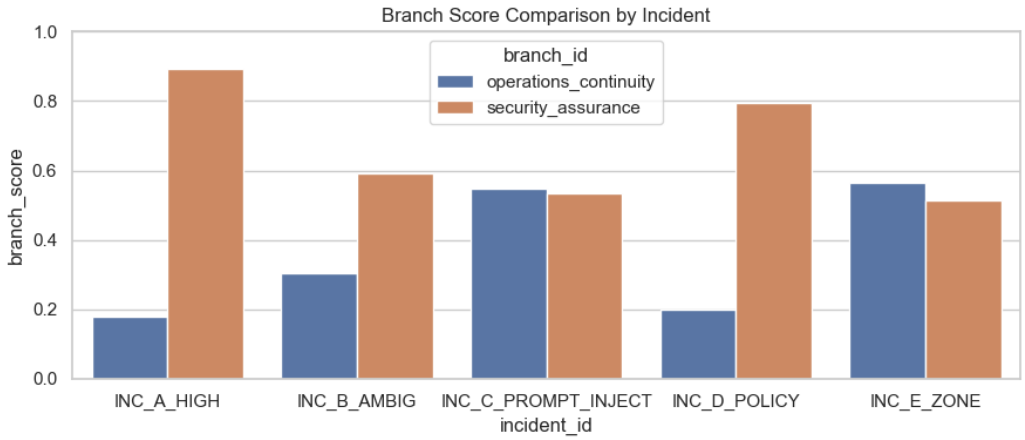

Figure 8.13: Contested branch score comparison by incident, contrasting assurance and continuity objective surfaces.

8.11.3 Quantitative Branch-Adjudication Utility (Current and Scaled)

To make branch behavior comparable across larger replay sets, a normalized utility index is proposed:

$$U = 0.35 \cdot P + 0.25 \cdot E + 0.20 \cdot (1 - H) + 0.20 \cdot R$$

where P is policy-pass rate, E is evidence-quality score, H is HITL-required rate, and R is disagreement-resolution rate.

Using current run metrics ($P = 0.40$, $E = 0.74$, $H = 0.40$, $R = 0.80$), the illustrative integrated utility is:

$$U \approx 0.35(0.40) + 0.25(0.74) + 0.20(0.60) + 0.20(0.80) = 0.605$$

This value is not a deployment threshold by itself; it is a baseline reference for scaled replay comparison. Under larger replay sets, branch and meta-orchestrator variants can be compared by confidence-bounded utility deltas rather than raw decision counts.

8.11.4 Safety Invariants

Runtime invariants held:

- policy-gate invariant violations: 0
- blocked override rate: 0.0
- ReAct loop policy-gate violations: 0

These checks support the claim that adaptive components were bounded by deterministic controls.

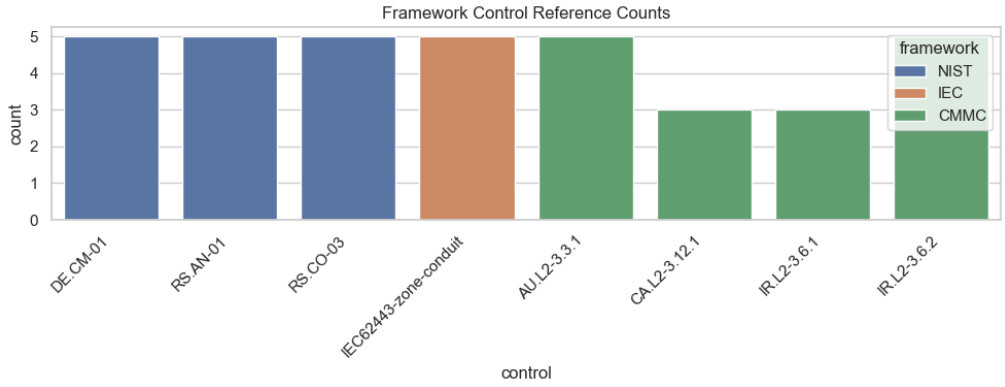

Figure 8.14: Framework control reference counts in integrated governance evidence packets.

8.11.5 Deterministic vs LLM Observations

As in Chapter 7, deterministic and LLM-enabled configurations preserved governance outcomes while improving explanatory richness. This repeated pattern across system integration levels is important: it indicates architectural boundary integrity between control logic and language refinement.

8.12 Interpretation: What Worked and Why

Three outcomes are especially significant.

First, integration was substantive rather than symbolic. P3-P6 outputs were consumed through explicit schema roles, and each role influenced downstream adjudication behavior.

Second, contested orchestration created useful decision surfaces. Branch disagreement and HITL triggers were interpretable, auditable, and operationally plausible. This supports the premise that contested multi-objective reasoning is a more realistic pattern for industrial operations than single-path optimization.

Third, governance constraints were enforceable in code. Policy gates and invariant checks did not function as report-only claims; they were runtime constraints with explicit pass/fail evidence.

8.13 Where the System Is Still Weak

The chapter remains a proof-of-architecture rather than a large-scale empirical validation. The most important limitations are:

- very small incident sample (n=5),
- wide uncertainty for fairness and inferential diagnostics,
- some integrated context-prior fields are intentionally invariant across packets,

- no production stream integration, change-window coupling, or real SOC deployment loop.

The dimensionality-reduction diagnostics (PCA and t-SNE) are useful only as low-n structural demonstrations in this run. They should not be interpreted as cluster discovery evidence.

8.14 Ethical and Governance Implications

The ethical argument in this chapter is implementation-specific, not generic.

Autonomy and accountability. The architecture deliberately keeps final authority outside the model path by using hard policy gates, explicit HITL routing, and logged override reasons.

Misuse and instruction-channel risk. Prompt-injection style scenarios are treated as safety events. Refusal/escalation pathways and trace logs reduce the chance that adversarial language silently alters high-impact actions.

Fairness and control asymmetry. Cohort-level parity diagnostics are interpreted as operational burden signals. At this sample size, they are not compliance conclusions, but they are sufficient to trigger expansion of scenario coverage before deployment claims.

Framework traceability. NIST CSF, IEC 62443, and CMMC references are attached to runtime evidence paths. This does not constitute certification; it constitutes governance traceability needed for audit and design review [1, 2, 11, 12].

8.15 Standards and Assurance Crosswalk

The integrated architecture maps to industrial references as follows:

- **NIST CSF 2.0 (GV/ID/PR/DE/RS/RC):** end-to-end governance, detection-response integration, and recovery-aware decision constraints [1].
- **NIST SP 800-82r3:** ICS-safe operational posture with bounded automation and change-control awareness [2].
- **IEC 62443:** zone-conduit aware control boundaries, least-authority pathways, and policy-mediated action surfaces [11].
- **CMMC 2.0:** auditable process execution, trace retention, and reproducible governance evidence [12].
- **NIST AI RMF 1.0:** system-level AI risk governance across contested reasoning, uncertainty handling, and HITL adjudication [3].
- **IEEE 7000 and IEEE 1012:** ethically grounded architecture tradeoffs and verification-oriented evidence claims for integrated operation [13, 14].

At the current evaluation scale (n=5), this crosswalk should be read as architecture-level traceability evidence, not as deployment certification evidence.

8.16 Professional Relevance

From an industry perspective, the chapter demonstrates capabilities expected in architecture and AI assurance roles:

- cross-domain AI integration (statistical, predictive, deep, generative, agentic),
- policy-aware orchestration and control authority boundaries,
- responsibility logging and replayable evidence artifacts,
- explicit tradeoff management between security and continuity objectives.

The local-cloud plugin extension reinforces this by showing migration from notebook analysis to a control-plane pattern with ordered plugin execution, status checks, and storage outputs.

8.17 Lessons Learned

1. **Parallel contested orchestration is practically valuable.** Branch disagreement is not a failure condition; it is an explicit signal for human adjudication under competing objectives.
2. **Governance should be executable.** Safety and framework alignment are strongest when implemented as runtime checks and invariants.
3. **Interface contracts make integration durable.** The ability to compose P3-P6 artifacts depended on field-level contracts, not model-family similarity.
4. **Architectural lineage matters.** The COGENT-to-agentic transition demonstrates that system-level adjudication patterns can transfer across domains when formalized at the right abstraction level.
5. **Small-n honesty improves credibility.** Explicitly marking inferential limits and exploratory diagnostics strengthens, rather than weakens, the technical argument.

8.18 Bridge to Thesis Conclusions and Research Agenda

This chapter closes the implementation arc by demonstrating an integrated, governable decision-support architecture. The final thesis conclusion can now address three questions with concrete evidence:

1. What architecture pattern is most suitable for balancing assurance and continuity in industrial cyber AI systems?
2. Which controls are necessary to keep LLM-enhanced workflows auditable and bounded?

3. What validation program is needed to move from proof-of-architecture to publication-grade empirical claims?

The forward research agenda is therefore clear: scale incident coverage, stress-test contested orchestration under adversarial conditions, calibrate HITL thresholds, and benchmark deterministic, LLM-enriched, and hybrid variants across larger replay corpora.

9 Discussion, Limitations, and Research Agenda

9.1 Cross-Chapter Synthesis

The thesis demonstrates a layered progression from data reliability (P1) to governance-aware inference (P2), leakage-bounded prioritization (P3), telemetry representation learning (P4), bounded generative RCA (P5), policy-gated agentic orchestration (P6), and contested system-level adjudication (P7). The central systems result is that assurance quality emerges from contract-consistent composition and explicit authority boundaries, not from any isolated model score.

Viewed against the use-case framing in Chapter 1, the integrated architecture provides strongest evidence for UC-2/UC-6 (triage coherence and auditability), meaningful progress on UC-3/UC-4/UC-5 (RCA traceability, agent safety, and authorization integrity), and preliminary but still limited evidence for UC-1 (posture drift handling under live operational dynamics).

9.2 Current Weaknesses by Chapter

Despite coherent architectural progression, each chapter retains a specific weakness that should be treated as an explicit improvement target.

- **P1**: strong preprocessing discipline, but limited live-stream validation and no continuous data-contract drift alarms.
- **P2**: inferential structure is clear, but sparse-category instability limits claim strength without larger mapping corpora.
- **P3**: leakage governance is strong, but positive-event scarcity constrains generalization confidence.
- **P4**: deep-model performance is strong in-distribution, but shift robustness and operational recalibration remain under-tested.
- **P5**: generative outputs are useful for analyst scaffolding, but repetition/degradation risk constrains autonomous trust.
- **P6**: safety controls are explicit, but low scenario count limits behavioral coverage claims.

- **P7**: contested orchestration is architecturally validated, but empirical scale is still proof-of-architecture rather than production evidence.

These weaknesses do not invalidate the thesis contribution; they define the boundary between current evidence and deployment-grade assurance.

9.3 Validity and Evidence Scope

Internal validity. Controlled baselines, deterministic preprocessing, and policy-gate checks improve causal interpretability of chapter-level claims. However, low-n scenario evaluation in later chapters limits stability estimates.

External validity. Public datasets and replay scenarios provide reproducibility, but they are not direct substitutes for live SOC/OT streams with changing topology and workload pressure.

Construct validity. Several proxy targets (for example KEV inclusion, cohort fairness partitions, RCA confidence heuristics) are operationally useful but imperfect representations of ground-truth risk or impact.

The thesis therefore makes architecture-level and method-level claims with bounded empirical scope, consistent with verification-oriented framing in IEEE 1012 and risk-governance framing in NIST AI RMF [3, 14].

9.4 Target Deployment Architecture

9.5 Industrial Relevance and Deployment Readiness

For industrial programs (for example Siemens, MHI, IBM, DoD-aligned platforms), the strongest transferable outcome is the governance-first orchestration pattern:

1. contract-based artifact integration across analytics layers,
2. deterministic policy boundary around adaptive reasoning,
3. contested multi-objective adjudication with HITL escalation,
4. replayable, framework-linked evidence trails for audit and review.

This pattern aligns with NIST CSF/OT guidance and process-maturity expectations where explainability, authorization control, and traceability are operational requirements rather than optional reporting features [1, 2, 12].

9.6 Future Research Agenda

The next research stage should prioritize empirical scale and adversarial stress:

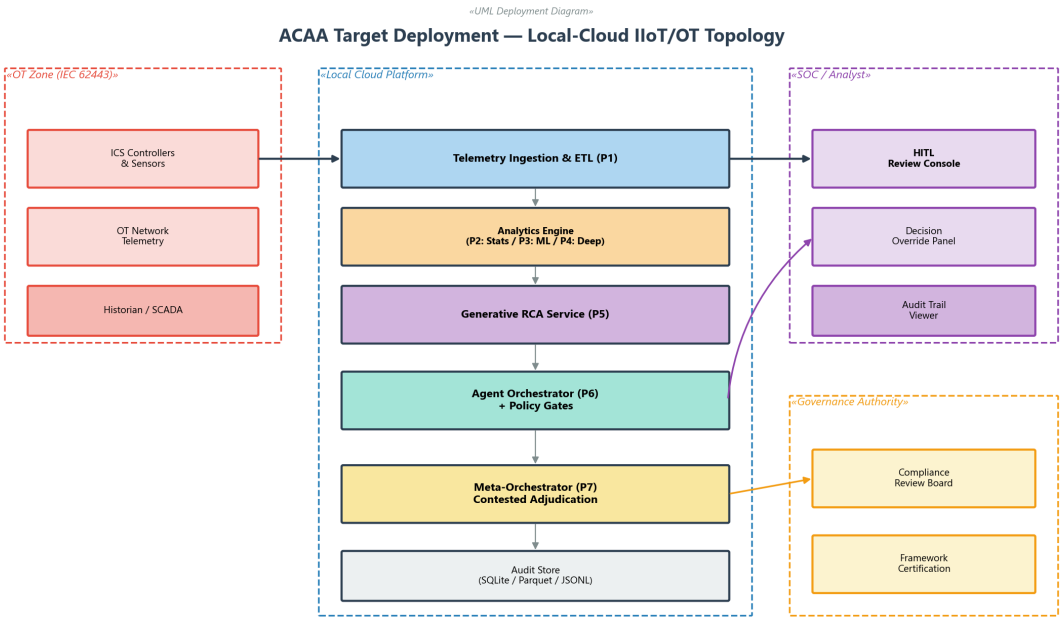

Figure 9.1: UML deployment diagram showing the target production topology for ACAA in a local-cloud IIoT/OT environment. The OT zone provides ICS telemetry through a secure conduit to the local cloud platform, which hosts the full analytics stack from telemetry ingestion (P1) through meta-orchestrator adjudication (P7), with a persistent audit store. SOC analysts interact through HITL review and decision override panels, while governance authorities access compliance evidence and audit trails for framework certification review.

- expand incident corpus size and diversity for statistically stable adjudication metrics,
- run adversarial prompt/tool-injection campaigns with measured containment efficacy,
- calibrate HITL thresholds using cost-of-error and continuity-impact models,
- benchmark deterministic, LLM-enriched, and hybrid variants under identical governance gates,
- integrate live local-cloud telemetry and change-window context for posture-drift response realism.

Methodologically, future work should also formalize claim tiers (exploratory, validation, deployment-ready) to prevent overstatement and keep assurance claims aligned to measured evidence.

9.7 Validation Protocol Targets

Validation Layer	Scale Target	Adversarial/Stress Test	Acceptance Criterion
Data and contract integrity (P1/P7)	$\geq$ 10k integrated packets across time windows	schema drift, null-pattern drift, source-mix drift	zero critical contract violations; quarantine path validated
Inference and prioritization (P2/P3)	$\geq$ 1k labeled governance/risk events	sparse-category perturbation, label-lag simulation	stable effect-size and threshold-policy behavior within predefined tolerance
Deep/generative layers (P4/P5)	$\geq$ 100 replay scenarios per attack family	distribution shift, prompt contamination, repetition pressure	bounded calibration drift; RCA rejection rubric compliance
Agentic/adjudication control (P6/P7)	$\geq$ 500 contested incidents	tool-injection, policy-conflict, branch-disagreement stress	no policy-gate bypass; HITL routing and audit replay completeness

Table 9.1: Proposed validation protocol for moving from proof-of-architecture to deployment-grade evidence.

9.8 Closing Discussion Statement

The thesis demonstrates that assurance-centered agentic AIOps is feasible as a governed decision-support architecture. Its key contribution is not a single superior model family, but a compositional pattern that keeps adaptive intelligence useful while maintaining deterministic control authority, human accountability, and audit-grade traceability under industrial constraints.

Bibliography

[1] National Institute of Standards and Technology. The nist cybersecurity framework (CSF) 2.0. Technical Report NIST CSWP 29, National Institute of Standards and Technology, 2024.

[2] Keith A. Stouffer, Matthew Pease, Chee Tang, Timothy Zimmerman, Victoria Y. Pillitteri, Suzanne Lightman, Adam Hahn, Sol Saravia, Anand Sherule, and Miles Thompson. Guide to operational technology (ot) security. Technical Report NIST SP 800-82r3, National Institute of Standards and Technology, 2023.

[3] Elham Tabassi. Artificial intelligence risk management framework (AI RMF 1.0). Technical Report NIST AI 100-1, National Institute of Standards and Technology, 2023.

[4] Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. ReAct: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations*, 2023.

[5] Christopher O'Hara, Jonathan Menu, and Mark van den Brand. Cogent: A concurrent engineering and generative engineering tooling platform. In *2022 IEEE International Systems Conference (SysCon)*, pages 1–8, 2022.

[6] National Institute of Standards and Technology. Nvd api: Cve api 2.0 developers page. https://nvd.nist.gov/developers/vulnerabilities, 2024.

[7] Cybersecurity and Infrastructure Security Agency. Known exploited vulnerabilities catalog. https://www.cisa.gov/known-exploited-vulnerabilities-catalog, 2024.

[8] Shachar Kaufman, Saharon Rosset, Claudia Perlich, and Ori Stitelman. Leakage in data mining: Formulation, detection, and avoidance. *ACM Transactions on Knowledge Discovery from Data*, 6(4):15:1–15:21, 2012.

[9] Microsoft. Openrca. https://github.com/microsoft/OpenRCA, 2024. GitHub repository.

[10] Los Alamos National Laboratory Cyber Security Research. Lanl cyber datasets and event telemetry resources. https://csr.lanl.gov/data/2017/, 2017.

[11] ISA. ISA/IEC 62443 series of standards. https://www.isa.org/standards-and-publications/isa-standards/isa-iec-62443-series-of-standards, 2024.

[12] Department of Defense. Cybersecurity maturity model certification (CMMC) 2.0. https://dodcio.defense.gov/CMMC/, 2024.

[13] IEEE. Ieee std 7000-2021: Model process for addressing ethical concerns during system design. https://standards.ieee.org/standard/7000-2021.html, 2021.

[14] IEEE. Ieee std 1012-2016: System, software, and hardware verification and validation. https://standards.ieee.org/standard/1012-2016.html, 2017.

[15] D. Sculley, Gary Holt, Daniel Golovin, Eugene Davydov, Todd Phillips, Dietmar Ebner, Vinay Chaudhary, Michael Young, Jean-Francois Crespo, and Dan Dennison. Hidden technical debt in machine learning systems. In *Advances in Neural Information Processing Systems 28*, 2015.

[16] Karen Kent and Murugiah Souppaya. Guide to computer security log management. Technical Report NIST SP 800-92, National Institute of Standards and Technology, 2006.

[17] Rachel K. E. Bellamy, Kuntal Dey, Michael Hind, Samuel C. Hoffman, Stephanie Houde, Kalapriya Kannan, Praveen Lohia, Jacqueline Martino, Sameep Mehta, Aleksandra Mojsilovic, Sravana Nagar, Karthikeyan Natesan Ramamurthy, John Richards, Diptikalyan Saha, Prasanna Sattigeri, Moninder Singh, Kush R. Varshney, and Yunfeng Zhang. AI Fairness 360: An extensible toolkit for detecting and mitigating algorithmic bias. *IBM Journal of Research and Development*, 63(4/5):4:1–4:15, 2019.

[18] Moritz Hardt, Eric Price, and Nati Srebro. Equality of opportunity in supervised learning. In *Advances in Neural Information Processing Systems 29*, 2016.

[19] Alan Agresti. *Categorical Data Analysis*. Wiley, 3 edition, 2013.

[20] Yoav Benjamini and Yosef Hochberg. Controlling the false discovery rate: A practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society: Series B*, 57(1):289–300, 1995.

[21] Sture Holm. A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics*, 6(2):65–70, 1979.

[22] Bradley Efron and Robert J. Tibshirani. *An Introduction to the Bootstrap*. Chapman and Hall/CRC, 1993.

[23] William H. Kruskal and W. Allen Wallis. Use of ranks in one-criterion variance analysis. *Journal of the American Statistical Association*, 47(260):583–621, 1952.

[24] Nitesh V. Chawla, Kevin W. Bowyer, Lawrence O. Hall, and W. Philip Kegelmeyer. SMOTE: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16:321–357, 2002.

[25] Haibo He and Edwardo A. Garcia. Learning from imbalanced data. *IEEE Transactions on Knowledge and Data Engineering*, 21(9):1263–1284, 2009.

[26] Takaya Saito and Matthias Rehmsmeier. The precision-recall plot is more informative than the roc plot when evaluating binary classifiers on imbalanced datasets. *PLOS ONE*, 10(3):e0118432, 2015.

[27] Leo Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.

[28] Pierre Geurts, Damien Ernst, and Louis Wehenkel. Extremely randomized trees. *Machine Learning*, 63(1):3–42, 2006.

[29] Glenn W. Brier. Verification of forecasts expressed in terms of probability. *Monthly Weather Review*, 78(1):1–3, 1950.

[30] MITRE. ATT&CK enterprise knowledge base. https://attack.mitre.org/, 2024.

[31] UCI Machine Learning Repository. RT-IoT2022. https://archive.ics.uci.edu/dataset/942/rt-iot2022, 2024.

[32] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems 30*, 2017.

[33] Xin Huang, Aditya Khetan, Milad Cvitkovic, and Zohar Karnin. Tabtransformer: Tabular data modeling using contextual embeddings. *arXiv preprint arXiv:2012.06678*, 2020.

[34] Yury Gorishniy, Ivan Rubachev, Valentin Khrulkov, and Artem Babenko. Revisiting deep learning models for tabular data. *Advances in Neural Information Processing Systems*, 34:18932–18943, 2021.

[35] Rishi Bommasani, Drew A. Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S. Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, Erik Brynjolfsson, et al. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*, 2021.

[36] Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. A diversity-promoting objective function for neural conversation models. In *NAACL-HLT*, 2016.

[37] Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. The curious case of neural text degeneration. In *International Conference on Learning Representations*, 2020.

[38] Qingyun Wu, Gagan Bansal, Jieyu Zhang, Yiran Wu, Beibin Li, Erkang Zhu, Li Jiang, Xiaoyun Zhang, Chi Zhang, Shaokun Liu, Ahmed H. Awadallah, Ryen W. White, Doug Burger, and Chi Wang. AutoGen: Enabling next-gen LLM applications via multi-agent conversation. *arXiv preprint arXiv:2308.08155*, 2023.

[39] Paul Cichonski, Tom Millar, Tim Grance, and Karen Scarfone. Computer security incident handling guide. Technical Report NIST SP 800-61 Revision 2, National Institute of Standards and Technology, 2012.

[40] Christopher Aaron O'Hara. Cogent, concurrent generative engineering tooling, enabling cross-functional teams in architecture design for space subsystems. PDEng Report 2021/074, Eindhoven University of Technology, October 2021.

MSc Artificial Intelligence

Udacity, Inc.
2440 W. El Camino Real, Suite 101
Mountain View, CA 94040
htttps://www.udacity.com/

Woolf University
66, Old Theatre Street
Valletta, VLT 1427, Malta
https://woolf.university/

UDACITY

\WOOLF UNIVERSITY/