Table of Contents

Appendix L Safety Case Construction

Note: We often refer to the artefacts in Appendices H and I because they contain detailed tables and figures. However, each section in Appendices H and I is also associated with sections in Chapters 6, 7 and Appendix M.

L.1 SECoT Validation Report Template

This template provides a structured approach for validating Systems Engineering Chains of Thought (SECoTs) in a systematic and repeatable manner. It ensures that each SECoT follows a rigorous predictive thinking process, aligning with general systems principles and enabling a thorough exploration of emergent behaviours. The template can be adapted for any SECoT by defining specific elements and applying the methodology to various systems engineering contexts. For a full description of SECoTs, see section D.8. for full definition of SECoT methods. The following are the main sections of the report:

L.1.1 Purpose of the Report

- Clearly define the purpose of the validation report.
- Explain why the SECoT needs to be validated.

Example: This report validates the "AIC Perspective Shift SECoT," ensuring that its predictive thinking steps systematically uncover unforeseen scenarios that impact autonomous safety assurance.

L.1.2 Scope of Validation

- Define the boundaries of what is being validated.
- Specify whether the validation applies to a single SECoT, a set of SECoTs, or a broader system application.
- You may also include “Out of Scope” to summarise what would be outside the scope of validation.

Example: The scope of this validation includes examining predictive thinking pipelines, general systems rules, and structured outputs to ensure alignment with the SECoT_1 principle.

L.1.3 SECoT Output Validation

L.1.3.1 Validation Criteria

- Define measurable criteria (pass/fail) to determine whether SECoT predictions are valid and valuable.
- Include qualitative and quantitative measures.
- Examine if the step completion criteria in the SECoT are preserved.

Example criteria:

- The predictions align with general systems axioms.
- The SECoT process reveals previously unrecognised emergent behaviours.
- The thinking pipelines yield multiple alternative scenarios.
- Step completion criteria are preserved.

L.1.3.2 Validation Results Table

Define the appropriate textual description of the validation test that needs to be checked for a given SECoT.

Table **Error! No text of specified style in document.**.1 SECoT validation results table

Validation Step	Evaluation Criteria	Pass/Fail	Notes
Step 1	Alignment with general systems rules.	Pass	Predictions adhere to known system behaviours.
Step 2	Generation of unforeseen scenarios.	Pass	Identified new emergent behaviours.
Step 3	Applicability to real-world cases.	Pass	Scenario aligns with real-world system failures.
Step 4	Step completion criteria are preserved.	Pass	The output predictions preserve step completion criteria.

L.1.4 Summary and Next Steps

L.1.4.1 Summary of Findings

- Summarize the key takeaways from the SECoT validation.
- Highlight whether the SECoT successfully identified emergent scenarios and preserved general systems rules.

Example: The SECoT validation confirmed that predictive thinking pipelines effectively identify unforeseen scenarios in autonomous drone operations, revealing three unaccounted failure cases.

L.1.4.2 Recommended Actions

Outline the next steps for refining the SECoT or integrating it into system design.

Example:

- Incorporate SECoT predictions into risk mitigation planning.
- Expand SECoT pipelines to cover additional interaction types.
- Validate SECoT predictions through simulation or real-world testing.

L.1.5 Appendices (Optional)

Include supporting materials such as:

- Diagrams of the SECoT predictive steps.
- Simulation results validating SECoT predictions.
- Additional examples of general systems rules in use.

L.1.6 SECoT Validation Review Process

- Validation reports must follow the peer review process.
- The process starts with the naming of the architect who is responsible for generating the report.
- It must also include signatures of at least one more reviewer to affirm the validity of the process.

L.2 SECoT Validation Report for AVOIDDS case study[1]

Below is a sample SECoT Validation Report drafted by the L.1 SECoT Validation Report Template. This example applies to the "Identification and Analysis of Unsafe Problematic Behaviours in Mid-Air Collision" SECoT, Table I.1, validating the correctness and usefulness of "Architect Assertion 1.1.4" about confusing and unsafe behaviours that may lead to collisions.

L.2.1 Purpose of the Report

This validation report evaluates the "Identification and Analysis of Unsafe Problematic Behaviours in Mid-Air Collision" SECoT, ensuring that its predictive thinking steps systematically uncover unforeseen or unsafe behaviours in the problem domain. It also confirms whether the SECoT product meets the expected completeness, consistency, and alignment with general systems rules and organisational needs.

- **SECoT Under Review:** "Unsafe Problematic Behaviours List Identification"
- **Architect Assertion 1.1.4:** Outlines two major problematic behaviours in mid-air collision contexts:
    1. Aircraft deviating unexpectedly from assigned flight paths
    2. By-passing aircraft moving erratically at high altitudes near crowded airspace.

L.2.2 Scope of Validation

This validation covers:

- Examination of how the SECoT identifies and describes confusing or unsafe behaviours in the domain of mid-air collision avoidance.
- Authentication that the **Predictive Thinking** steps are properly followed—i.e., that they adhere to "General rule A" and "General rule B" from your general systems rules.
- Assess alignment with the Step Completion Criteria (1.1.3) to confirm that the SECoT product sufficiently describes the problem context and indicates undesirable impacts without prematurely jumping to design solutions.

[1] Associated with section 7.9 and table I.1

Out of Scope:

- Detailed assessment of proposed solutions or mitigations for any of the identified unsafe behaviours.
- Authentication of other SECoTs beyond the "Identification and Analysis of Unsafe Behaviours" focus.
- Full end-to-end system safety or certification processes outside the immediate SECoT's boundaries.

L.2.3 SECoT Output Validation

L.2.3.1 Validation Criteria

We define the following **pass/fail** criteria to determine whether the "Identification and Analysis of Unsafe Problematic Behaviours in Mid-Air Collision" SECoT output is valid and valuable:

1. **Alignment with General Systems Rules**
   - The SECoT must illustrate that **General rule A** (unsafe behaviour is a type of confusing behaviour) and **General rule B** (confusing behaviours cause undesirable emergent outcomes) are correctly applied.
2. **Relevance to the Stakeholder/Architect Sphere of Concern**
   - The problem statements must show that the described behaviours (e.g., erratic or unauthorised flight path deviations) fall within the domain's sphere of concern (e.g., collision risk in congested airspace).
3. **Identification of Realistic Undesirable Outcomes**
   - The SECoT must show how each unsafe or confusing behaviour leads to specific, undesirable consequences (e.g., near misses or actual collisions).
4. **Exclusion of Solutions**
   - The deliverable should focus on analysing behaviours and not present solutions or mitigations (in line with the step's instructions).
5. **Step Completion Criteria (1.1.3) Preserved**
   - The SECoT must sufficiently describe the problem (the "Confusing Complex") and the unsafe impact on some element B.

L.2.3.2 Validation Results Table

Below is an example table summarising the evaluation of each step or key aspect in the SECoT.

Validation Step	Evaluation Criteria	Pass/Fail	Notes
Step 1: Identify Problem Domain	Does the SECoT describe a mid-air collision scenario with enough clarity?	Pass	The problem domain is stated clearly: mid-air collisions. The scenario references "aircraft deviate from assigned path."
Step 2: Apply General Rule A, B	Are unsafe/ confusing behaviours recognised as contradictory? Do they cause emergent risk?	Pass	Identified two primary behaviours: unexpected flight path deviation and erratic flight near crowded airspace, leading to collision risk.
Step 3: Problem-Focused, No Solutions	Does the text avoid introducing solutions?	Pass	The SECoT only defines problems: "Aircraft deviating" and "Moving erratically..." It does not mention new solutions or systems.
Step 4: Step Completion Criteria	Does an "undesirable outcome accompany each behaviour"?	Pass	The undesirable outcomes are clearly described as "Increased collision risk," "ATC confusion," and "reactive manoeuvres."
Overall	Does the SECoT produce an acceptable description of the Confusing Complex?	Pass	The problem context is consistent, relevant, and no solutions overshadow the problem articulation.

L.2.4 Summary and Next Steps

L.2.4.1 Summary of Findings

- The SECoT effectively identifies critical unsafe behaviours (such as "unexpected flight path deviations" or "unauthorised, erratic flight in crowded space").
- Each behaviour is framed as "confusing" and includes specific undesirable outcomes (e.g., collision risk, unpredictability for other pilots).
- The content adheres to General Rule A and General Rule B, explaining how each contradictory behaviour escalates collision likelihood and thus is within the "sphere of concern."

- No solutions appear in the text, thus fulfilling the requirement to remain problem-focused.

All these observations indicate that the SECoT output meets the objectives set out in the instructions for “Identification and Analysis of Unsafe Problematic Behaviours.” Hence, the SECoT is validated as meeting the acceptance criteria for this stage.

L.2.4.2 Recommended Actions

To refine future SECoT steps or expansions, the following actions are suggested:

1. **Feed into Next Phase**
   - Use these validated unsafe behaviours to inform the following process steps:
     - Step 1.2) Generate a descriptive image that visualises the unsafe behaviour

L.2.5 Appendices (Optional)

If needed, include:

- Minutes of problem articulation reviews with the customer that support this SECoT's finding.
- **Illustrative diagrams** or references to concept sketches of the flight paths, relevant logs of near-misses, or any additional data compiled during the problem articulation.

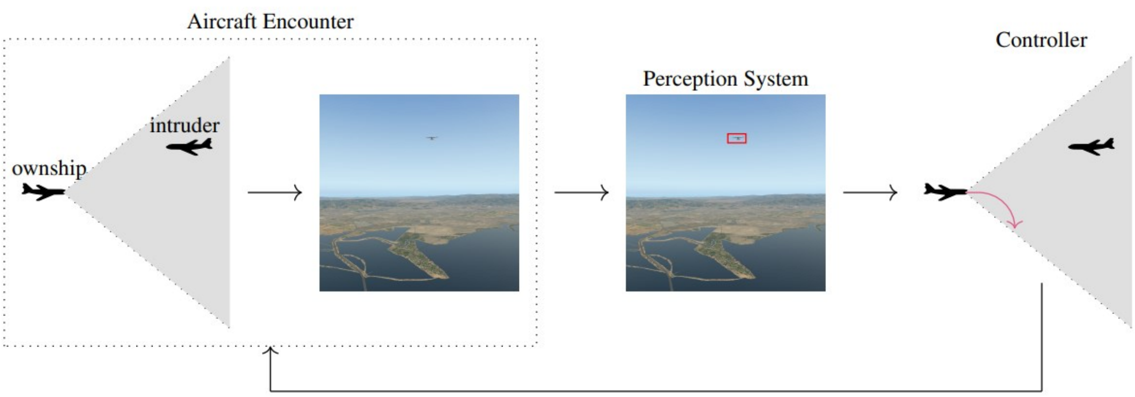


Figure **Error! No text of specified style in document.**.1 Diagram taken from [1]

L.2.6 SECoT Validation Review Process

- **Report Author/Architect**: Haider Al-Shareefy [University of Southampton], responsible for generating the SECoT validation content.
- **Reviewer 1**: Prof Michael Butler [University of Southampton], validating alignment with General Rules D, E, F.
- **Reviewer 2**: Dr Son Hoang [University of Southampton], verifying coverage of elements and clarity of PrimePs.
- **Reviewer 3**: Prof Hamid Asgari [Thales], performs customer acceptance of validation report output.
- **Sign-off**: The above signatories affirm that the **SECoT validation** process was carried out thoroughly, and the results are valid as of this report date.

By signing below, the reviewers confirm that **I.4.1.1 Step 1.1) Identify a list of unsafe or confusing behaviours** of the SECoT is validated as meeting the defined criteria.

Signatures:

Signatures:

- Architect: ______________________________ (Date: ____________)
- Reviewer 1: ___________________________ (Date: ____________)
- Reviewer 2: ___________________________ (Date: ____________)
- Reviewer 3: ___________________________ (Date: ____________)

End of SECoT Validation Report

L.3 SECoT Validation Report for Unsafe Train Tracks case study[2]

L.3.1 Purpose of the Report

This report aims to validate the SECoT step "H.4.1.4 Step 1.4) Define the Supra-Complexes and Their PrimePs" ensuring that its predictive thinking processes and outputs adhere to the established general systems rules and follow the method's intended objective.

The SECoT step under validation identifies two supra-complexes within the train tracks problem domain—Train Network and Adversarial Scheme—each associated with a distinct Prime Purpose (PrimeP). Validating this SECoT output is crucial to confirm that the methodology correctly organises the system elements into supra-complexes with consistent, clearly stated PrimePs, thereby guiding subsequent design and solution development.

L.3.2 Scope of Validation

- **Focus**: The validation is limited to the single SECoT step "Define the supra-complexes and their PrimePs" as documented in Table H.4 and summarised in Architect Assertion 1.4.6.
- **Boundaries**: The assessment excludes detailed downstream solutions, mitigations, or expansions of these supra-complexes beyond verifying that the SECoT step (1) correctly applies the relevant general system rules (D, E, F) and (2) satisfies the completion criteria (1.4.5).
- **Related References**: This validation specifically examines integrating with the "General Systems Rules" (especially D, E, F) and aligning with the AIC-based method for clarifying purpose and system scope.

L.3.3 SECoT Output Validation

L.3.3.1 Validation Criteria

The following criteria (Pass/Fail) are applied to judge whether the SECoT step output is acceptable:

1. **Alignment with General System Rules**
    - **General Rule D**: Has the architect defined supra-complexes as "larger collections of complexes"?

---

[2] Associated with section 6.10 and table H.4

- **General Rule E**: Are the purpose statements consistent, stable, and clear across scenarios?
- **General Rule F**: Does each supra-complex have a well-defined Prime Purpose in the sense of a "Primary Purpose" that influences design?

2. **Clarity of Supra-Complex Scope**
   - Each supra-complex must logically group relevant elements to achieve its chosen PrimeP.
3. **Correctness and Completeness**
   - The final list of supra-complexes must encompass all visible elements.
   - The step must not omit significant elements or add extraneous ones that do not fit the scenario.
4. **Step Completion Criteria (1.4.5) Preservation**
   - The step's deliverable is considered complete when at least one supra-complex is identified to represent each group of coexisting elements, and each is assigned a PrimeP.
5. **Absence of Unwarranted Solutions**
   - This phase must remain descriptive about system structure and purpose rather than specifying a design or solution.

Validation Step	**Evaluation Criteria**	**Pass/Fail**	**Notes**
Step 1: Application of Gen. Rules D, E, F	Are supra-complexes formed properly (Rule D)? Are the defined PrimePs consistent (Rules E, F)?	**Pass**	Two supra-complexes—Train Network and Adversarial Scheme—are logically separated, each with a stable, domain-consistent PrimeP.
Step 2: Clarity of Supra-Complex Scope	Do these supra-complexes collectively include all identified elements?	**Pass**	The Train Network includes the train, fence, vegetation, and power lines. The Adversarial Scheme consists of the adversarial drone. Comprehensive coverage.
Step 3: Correctness and Completeness	Are the key elements (train, fence, drone, etc.) correctly allocated?	**Pass**	All visible elements are accounted for; each belongs to a supra-complex that logically groups them by purpose.

Step 4: Step Completion Criteria 1.4.5 Preserved	Is each supra-complex assigned a PrimeP?	**Pass**	Train Network → "transport people/goods safely" Adversarial Scheme → "disrupt train operations" Aligned with the method's instructions.
Step 5: No Solutions Included	Does it avoid design solutions?	**Pass**	The final output remains at the level of problem domain structuring—no solutions or mitigations overshadow the definition process.

L.3.4 Summary and Next Steps

L.3.4.1 Summary of Findings

1. **General Systems Rule Application**
   - The step properly operationalises Rule D by defining supra-complexes that unite sets of systems with shared overarching objectives.
   - Rules E and F are upheld: each supra-complex's PrimeP is well defined and consistent across potential scenarios.
2. **Thoroughness in Coverage**
   - All major observable elements (train, tracks zone, vegetation, fence, power lines, drone) are included and correctly assigned to either the Train Network or Adversarial Scheme supra-complex.
3. **Clarity for Future Stages**
   - The logic behind choosing these two supra-complexes clarifies how future design or risk analysis can proceed (e.g., focusing on how adversarial elements threaten the train network's primary purpose).

Therefore, the SECoT step "Define the supra-complexes and their PrimePs" meets the validation criteria, with no unmet requirements or contradictions discovered.

L.3.5 Recommended Actions

1. Incorporate into Next Steps
   - Use these supra-complex definitions to guide scenario-based hazard analyses or system design choices, especially in subsequent steps focusing on emergent interactions.
2. Validate Worst-Case Assumptions

- Validate or refine the assumption that the “Adversarial Scheme” might be more widespread than a single drone (i.e., a multi-drone orchestrated threat).

L.3.6 Appendices

- (If desired) Diagrams showing the two supra-complexes and their sub-elements.
- Additional references or notes on re-scope definitions, if the “Adversarial Scheme” requires broadening.

L.3.7 SECoT Validation Review Process

- **Report Author/Architect**: Haider Al-Shareefy [University of Southampton], responsible for generating the SECoT validation content.
- **Reviewer 1**: Prof Michael Butler [University of Southampton], validating alignment with General Rules D, E, F.
- **Reviewer 2**: Dr Son Hoang [University of Southampton], verifying coverage of elements and clarity of PrimePs.
- **Reviewer 3**: Prof Hamid Asgari [Thales], performs customer acceptance of validation report output.
- **Sign-off**: The above signatories affirm that the **SECoT validation** process was carried out thoroughly, and the results are valid as of this report date.

By signing below, the reviewers confirm that **H.4.1.4 Step 1.4) Define the supra-complexes and their PrimePs** of the SECoT is validated as meeting the defined criteria.

Signatures:

- Architect: ______________________________ (Date: ____________)
- Reviewer 1: ____________________________ (Date: ____________)
- Reviewer 2: ____________________________ (Date: ____________)
- Reviewer 3: ____________________________ (Date: ____________)

End of SECoT Validation Report

L.4 SACE Safety Case Argumentation Patterns

L.4.1 Stage 1: Operating Context Assurance

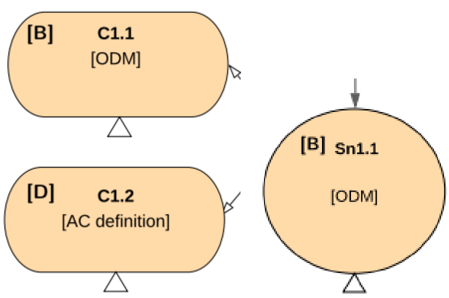
The first stage of SACE requires defining the AS's capabilities, validating its Operational Domain Model (ODM), and defining operating scenarios. The primary outcomes of this stage are:

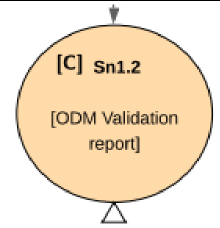
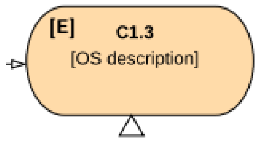
1. Autonomous Capabilities are specified.
2. AS's Operational Domain Model (ODM) are established and confirmed.
3. Operating Scenarios within the established ODM are identified and validated.
4. Development of Operating Context Assurance Argument.

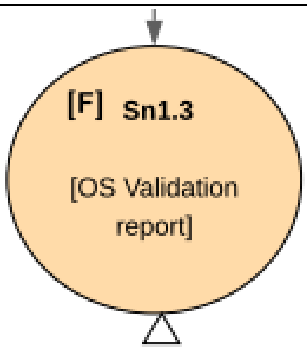
AIC Systems Approach Processes involved to satisfy the objectives:

- Stage 1: Uncertainty Problem Articulation and Operational Environment Modelling
- Stage 2: Architect Intent and Autonomous Solution Needs Definition
- Stage 3A: HazTOPS and Ordered AIC-driven Autonomous System Requirements Development
- Stage 3B: Comprehensive Operational Environment Definition

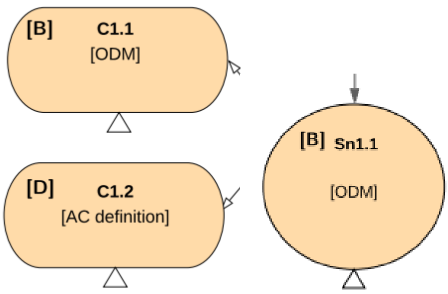
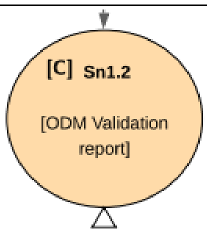
L.4.1.1 Argument Pattern for Eagle Robot Operating Context Assurance

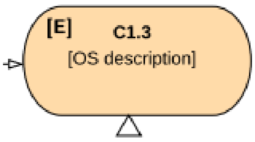
Assurance Argument Pattern	**AIC Systems Approach supportive methods and artefacts** The following process outputs satisfy the artefact's demonstration requirements:
	Table H.15 Architect High-Level Solution Prescription. • Sections 6.2, 7.2.
	Table H.35 Operational Design Definition for Eagle Robot Deployment in Train Track Zone

	• Sections L.1 for SECoT validation template, sections H.12, L.2 for implementations.
	H.4.4.3 Step 4.3) Define the assumptions made about factors. Table J.8 unsafe train tracks problem domain assumptions.
	H.4.3.1 Step 3.1) Model detailed AIC interactions scenarios for the problem domain. Figure H.3 Modelling a complicated interaction n6. Table J.6 AIC problem domain scenarios definition
	H.4.3.2 Step 3.2) Predict the extended list of emergent AIC interactions scenarios. Table H.10 Complexity Field for n6 Interaction SECoT definition. Table J.7 AIC Extended Scenarios.
	H.6.1 Predictive Thinking Pipeline 1: Introducing Autonomous systems into Feed-forward complexity. Table H.18 Implementing Architect Intent and Forward-Feed AIC Interaction Framework for addressing train derailment caused by adversarial drones. Table H.19 Mapping AIC interactions of the Eagle Drone and adversarial drone behaviours in mitigating train derailment risks.

	H.7 Stage 3B: Comprehensive Operational Environment Definition. Table H.35 Operational Design Definition for Eagle Robot Deployment in Train Track Zone.
	Validation is done by documenting expert reviews of architect assertions and predictions and the appropriate application of SECoT. A validation report template has been generated, which can be found No validation report had been generated as part of PhD scope.

L.4.1.2 Argument Pattern for AVOID System Operating Context Assurance

Assurance Argument Pattern	**AIC Systems Approach supportive methods and artefacts;** The following process outputs satisfy the artefact's demonstration requirements:
	Table I.13 Architect High-Level Solution Prescription.
	Table I.27 Operational Design Definition for AVP
	L.2 SECoT Validation Report for AVOIDDS case study. (I.4.1.1 Step 1.1) Identify a list of unsafe or confusing behaviours]
	I.4.4.3 Step 4.3) Define the assumptions and hazards made about factors.

	Table K.2 Extended assumptions, plausibility, concern and hazards analysis. In no particular order.
	I.4.3.1 Step 3.1) Model detailed AIC interactions scenarios for the problem domain. Figure I.2 Modelling AIC scenario from interaction n1.
	I.4.3.2 Step 3.2) Predict the extended list of emergent AIC interactions scenarios. Table I.10 AIC extended scenario for n1 Interaction SECoT definition. Table K.2 Extended assumptions, plausibility, concern and hazards analysis. In no particular order.
	I.6.1 Predictive Thinking Pipeline 1: Introducing Autonomous systems into Feed-forward complexity. Table I.14 Implementing Architect Intent and Forward-Feed AIC Interaction Framework for addressing AVP reliability for by-passing aircraft. Table I.15 Mapping AIC interactions of AVP with ownship aircraft and the environment
	I.7 Stage 3B: Comprehensive Operational Environment Definition. Table I.27 Operational Design Definition for AVP

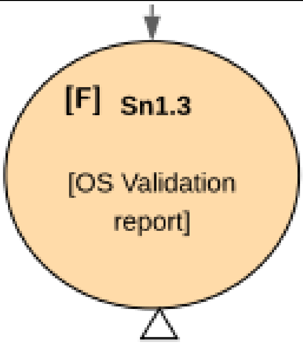
	Validation is done by documenting expert reviews of architect assertions and predictions and the appropriate application of SECoT. A validation report template has been generated, which can be found No validation report had been generated as part of PhD scope.

Table L.2 captures the primary artefacts that need to be presented:

Table **Error! No text of specified style in document.**.2 SACE Stage 1 artefacts and AIC approach mapping

SACE Artefact	**Explanation**	**The substantiating AIC methods and artefacts**
[A]: AS Concept Definition	entails producing a high-level document outlining the intended functions, objectives, and constraints. The systems approach artefact should demonstrate that the intended system functionality, scope of autonomy, and interactions with humans are clearly defined. It should include a use-case description and stakeholder involvement agreement.	• Sections 6.3, M.3. • H,I.5 Stage 2: Architect Intent and Autonomous Solution Needs Definition method satisfies artefacts [A]. This is because the method for defining the architect's intent specifically delineates how the architect plans to address the anticipated problematic situations identified in Stage 1 and how autonomous functionalities may provide solutions for those issues. This stage encompasses establishing a contractual agreement between the architect and the stakeholder regarding the requirements to be fulfilled by the architect. For

		instance, the following may be observed: • Table H.15 - Architect High-Level Solution Prescription • Table I.13 - Architect High-Level Solution Prescription
[B]: Operational Domain Model (ODM)	entails modelling the scope of operations for AS, including the assumptions made about the environment and operating conditions. The systems approach artefact should demonstrate: • All relevant environmental and operational conditions are included. • All scenarios the AS may encounter are covered. • It should explicitly list assumptions, constraints, and non-mission interactions.	• Sections 6.2, M.2. • H,I.4 Stage 1: Uncertainty Problem Articulation and Operational Environment Modelling. In Stage 1, the architect methodically formulates a comprehensive Operational Domain Model (ODM) by clearly delineating uncertainties related to environmental conditions and contextual constraints that may be deemed critical to the safety of autonomous systems. This stage employs structured activities, such as the Predictive Thinking Pipeline (H.4.4), which assesses pivotal assumptions and factors related to the problem domain, ensuring a thorough operational environment representation. For instance, the Operational Design Definitions presented in Table H.35 (Eagle Robot

		Deployment in Train Track Zone) and Table I.27 (Automated Valet Parking, AVP) illustrate the sources of uncertainty the architect perceives. Furthermore, the complexity inherent in the operational domain is acknowledged. Consequently, Stage 1 asserts that the resultant ODM is adequately detailed and resilient, thereby helping minimise unforeseen occurrences upon the deployment of the autonomous system. It distinctly identifies the assumptions and variances necessary for subsequent safety analysis and assurance while explicitly addressing potential risks introduced by emergent complexity and operational nuances uncertainty.
[C]: ODM Validation Report	entails documenting the validation of plausibility and correctness of the ODM and that it sufficiently defines the operating scope. The architect must provide clear evidence that the ODM is comprehensive and correct. Since we utilise SECoT to model the ODM, validating it becomes more objective and	• Sections L.1 for SECoT validation process, The structured SECoT Validation Report Template (Artefact L.1), along with its specific instances about the AVOIDDS system (L.2) and Unsafe Train Tracks (L.3), systematically fulfils the SACE demonstration requirements.

	straightforward. For such end, we developed a structured SECoT validation report template (section L.1).	Specifically, the AIC method distinctly delineates: • **Validation objective of Completeness:** The validation report examines whether the set of predictions is complete in implementing the thinking method and those general systems rules enforced; thus, the architect predicts all they can or need to predict. • **Validation objective of Correctness:** The validation report requires a formal review process to ensure the predictions regarding implementing the thinking method and general systems rules (axioms) are correct. • **Validation objective of plausibility:** The validation report examines whether SECoT predictions align with possible real-world complexity. Consequently, the structured AIC substantiating method meets the SACE artefact requirements by ensuring that the Operating Domain Model is thoroughly validated for

		relevance, accuracy, completeness, and suitable granularity, with documented expert review outlined in the validation review process (L.1.6). The validation report template can be used to validate any AIC model schema.
[D]: Autonomous Capabilities Definition	entails specifying AS functionalities. The systems approach artefact is expected to clearly define the scope of autonomy and the tasks that may require human intervention.	• Architect High-Level Solution Prescription (Table H.15, I.13). • H,I.6 Stage 3A: HazTOPs and Ordered AIC-driven Autonomous System Requirements Development Stage 2 of the AIC approach develops the high-level solution for identified problematic situations and assumptions derived from stage 1. At this stage, the architect and stakeholders decide what autonomous capabilities need to be used. Stage 3A further refines the system level needs into more granular definitions of autonomous capabilities.
[E]: Operating Scenarios Definition	entail a detailed description of all scenarios the AS may encounter within the anticipated ODM. The systems	• H,I.6 Stage 3A: HazTOPs and Ordered AIC-driven Autonomous System

	approach artefact must define actions, events and environmental assumptions that could affect AS performance.	Requirements Development • H,I.7 Stage 3B: Comprehensive Operational Environment Definition. • H,I.8 Stage 4: Disordered AIC-Driven Black Swan Scenarios Prediction The AIC approach predicts scenarios at every stage of the design process and translates those scenarios into design requirements. One important activity is predicting Black Swan scenarios, which are part of the long-tailed probability distribution of possible events that could happen in the operational domain. The development stages define the actions, events and operational environment assumptions constituting the operational scenarios.
[F]: Operating Scenarios Validation Report	assesses whether the defined operating scenarios comprehensively capture all relevant AS interactions. The substantiating artefact must provide evidence of expert review, simulation-based verification, and real-world validation data to confirm the	• Sections L.1 for SECoT validation template, sections H.12, L.2 for implementations. Validation is done by documenting expert reviews of architect assertions and predictions and the appropriate application of

	completeness of the operating scenarios.	SECoT. A validation report template has been generated, which can be found. No validation report was generated as part of PhD scope.
[G]: AS Operating Context Assurance Argument Pattern & [H]: AS Operating Context Assurance Argument:	A structured assurance argument framework is explicitly formulated to demonstrate that the Autonomous System (AS) can safely operate within its specifically defined operational context.	The assurance argument framework presented in artefact L.4.1.1 (Argument Pattern for Eagle Robot Deployment) explicitly addresses the requirement by structuring the safety justification around systematically established operational domain assumptions and scenarios.

L.4.2 Stage 2: AS Hazardous Scenario Identification

SACE Stage 2 and AIC Stage 3A emphasise identifying and validating hazardous scenarios. The main outcomes of this stage:

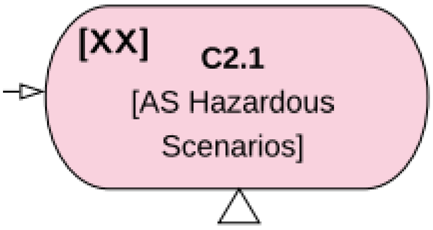
1. Potential hazardous scenarios for the AS are identified and outlined.
2. Hazardous Scenarios of the AS are Validated.
3. Development of the Assurance Argument for the AS Hazardous Scenarios.

The AIC approach introduces HazTOPs (Hazards, Threats, and Opportunities Scenarios) to refine the scope of risk mitigation strategies.

AIC Systems Approach: Processes involved to satisfy the objectives:

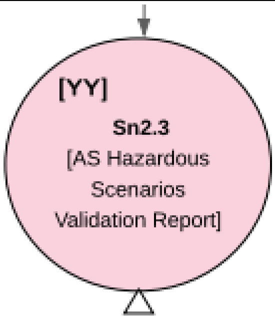
- Stage 3A: HazTOPS and Ordered AIC-driven Autonomous System Requirements Development
- L.1 SECoT Validation Report Template

L.4.2.1 Argument Pattern for Eagle Robot Hazardous Scenarios

Assurance Argument Pattern	**AIC Systems Approach supportive methods and artefacts;** The following process outputs satisfy the artefact's demonstration requirements:
	H.6.3 Predictive Thinking Pipeline 3: Hazards, Threats and Opportunities Scenarios (HazTOPs) Analysis. Figure H.16 Hazards Complexity Field Scope: graphically scoping the hazards within the complexity field by placing hazard icons on target interaction. Figure H.17 Threats Complexity Field Scope Figure H.18 Opportunities Complexity Feels Scope H.6.3.2 Step 2) Characterise the scoped interactions. Figure H.19 Hazards associated with Eagle Drone preventing derailed train complexity field Table H.24 The table describes the AIC interaction dynamics between Eagle Drones and adversarial drones H.6.3.3 Step 3) Apply predictive potential complications guide words. Table H.25 HazTOPs Analysis of "3 Drones Attack" Scenario with Risk and Surprise Assessment

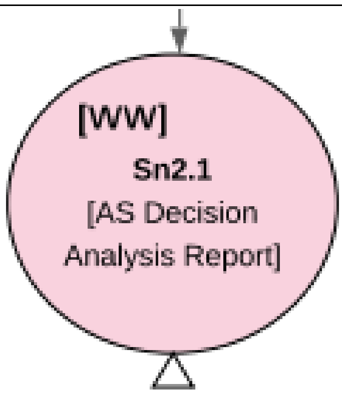
<table>
<tr><td></td><td>Table H.26 HazTOPs Analysis of adversarial drone using smart lasers scenario

Table H.27 HazTOPs Analysis of adversarial drone hiding behind fence scenario

Figure Error! No text of specified style in document..2 Soft Hazard Complexity Field Model</td></tr>
<tr><td>
</td><td>Validation is done by documenting expert reviews of architect assertions and predictions and the appropriate application of SECoT.

A validation report template has been generated, which can be found in

L.1 SECoT Validation Report Template

Examples: L.3 SECoT Validation Report for Unsafe Train Tracks case study</td></tr>
<tr><td>
</td><td>H.6.1 Predictive Thinking Pipeline 1: Introducing Autonomous systems into Feed-forward complexity

H.6.2 Predictive Thinking Pipeline 2: Designing the affecting Backward-Feed complexity field

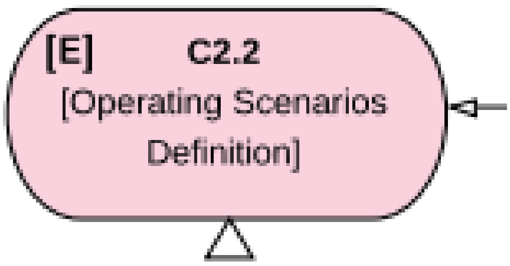
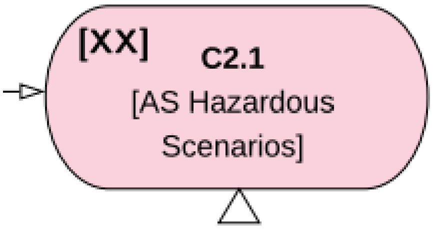
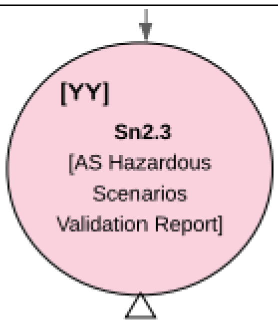
H.9 Stage 5: CuneiForm-based Syllabus for Safety-Driven ML Epistemic Intelligence Development</td></tr>
<tr><td>
</td><td>H.6.1 Predictive Thinking Pipeline 1: Introducing Autonomous systems into Feed-forward complexity.

Table H.18 Implementing Architect Intent and Forward-Feed AIC Interaction Framework for addressing train derailment caused by adversarial drones.</td></tr>
</table>

	Table H.19 Mapping AIC interactions of the Eagle Drone and adversarial drone behaviours in mitigating train derailment risks.

L.4.2.2 Argument Pattern for AVOID System Hazardous Scenarios

Assurance Argument Pattern	**AIC Systems Approach supportive methods and artefacts;** The following process outputs satisfy the artefact's demonstration requirements:
	I.6.3 Predictive Thinking Pipeline 3: Hazards, Threats and Opportunities Scenarios (HazTOPs) Analysis. Figure I.11Sources of Hazards AIC Complexity Field H.6.3.2 Step 2) Characterise the scoped interactions. Table I.22 Considering Hazards related to I1 interaction H.6.3.3 Step 3) Apply predictive potential complications guide words. Table I.23 Example "More" guide word complication
	Validation is done by documenting expert reviews of architect assertions and predictions and the appropriate application of SECoT. A validation report template has been generated, which can be found in L.1 SECoT Validation Report Template Examples:

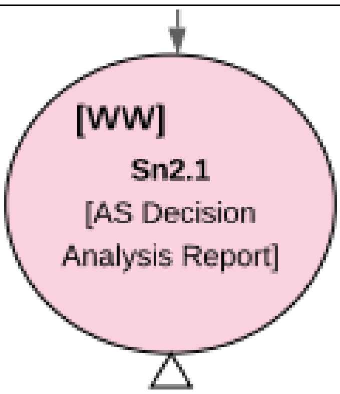
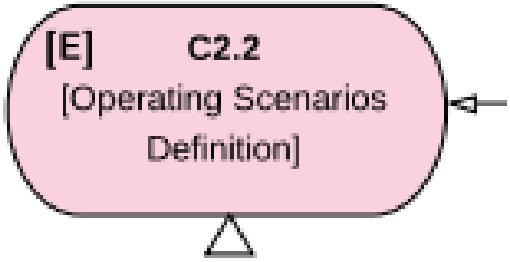
	L.2 SECoT Validation Report for AVOIDDS case study
	I.6.1 Predictive Thinking Pipeline 1: Introducing I.6.2 Predictive Thinking Pipeline 2: Designing the affecting Backward-Feed complexity field I.9 Stage 5: CuneiForm-based Syllabus for Safety-Driven ML Epistemic Intelligence Development
	I.6 Stage 3A: HazTOPs and Ordered AIC-driven Autonomous System Requirements Development. Table I.14 Implementing Architect Intentand Forward-Feed AIC Interaction Framework for addressing AVP reliability for by-passing aircraft. Table H.19 Mapping AIC interactions of the Eagle Drone and adversarial drone behaviours in mitigating train derailment risks.

Table L.3 captures the primary artefacts that need to be presented in the safety case:

Table **Error! No text of specified style in document.**.3 SACE Stage 2 artefacts and AIC approach mapping

SACE Artefact	**Explanation**	**The substantiating AIC methods and artefacts**
[WW]: AS Decision Analysis Report:	A report analysing the decisions made by the AS at key decision points within different operating scenarios.	• I, H.6.1 Predictive Thinking Pipeline 1: Introducing Autonomous systems

	The systems approach artefact must demonstrate that all AS decisions and judgments have been identified and analysed for potential hazards and belief states of AS.	into Forward-Feed complexity • I,H.6.2 Predictive Thinking Pipeline 2: Designing the affecting Backward-Feed complexity field In this process, the architect determines an AS's decisions when involved in a particular problematic scenario. For example, • Table H.18 Implementing Architect Intent and Forward-Feed AIC Interaction Framework for addressing train derailment caused by adversarial drones The Eagle Drone should decide to inhibit the adversarial drone and prevent derailment of the train incident. However, to do so it must perform the following activities: • Recognises the visibility of vegetation appearance. • Avoids crashing into vegetation structures. • Physically inhibit the by-passing drone. • It cannot effectively be influenced or

		controlled by the adversarial drone. Also, I,H.9 Stage 5: Safety-Driven ML-based Perception Training, Testing and Validation Process (CuneiForm Strategy Development) In this stage, the architect determines what decision AS should make when face a particular scenario, for example, in a situation such as (Table H.39 CuneiForm Pictorial situation articulation): A drone landing on the ground {1}. The drone has a camouflaged skin {2} landing at various distances from the track fence or train tracks or both {3}, trees {4}, local birds surrounding the landed drone{5} bushes {6}, gravel{7}, soil{8}, pavement{9} trash {10}. Then, the Eagle Drone should be able to recognise an adversarial drone and act accordingly. We do not specifically define a category of knowledge as "decision"; however, any action required and derived through the process from the ML

		component to perform is a decision made.
[XX]: AS Hazardous Scenarios Definition:	A comprehensive specification of all identified hazardous scenarios, including the interactions, environment states, and decisions leading to unsafe outcomes. The systems approach artefact should clearly show how the architect predicted hazardous scenarios and what mitigation requirements were devised to reduce their impact	- H.6 Stage 3A: HazTOPs and Ordered AIC-driven Autonomous System Requirements Development - H.8 Stage 4: Disordered AIC-Driven Black Swan Scenarios Prediction Stages 3A and 4 enable the architect to assess the problematic scenarios identified in Stage 1 thoroughly and discover additional scenarios. HazTops adopts a holistic approach that pinpoints hazards while identifying threats and opportunities. For instance, in Figure H.16, we modelled the operational scenario where a desired autonomous capability was defined, in which a hazard was predicted due to tree cover around the tracks, as it would obstruct the Eagle Drone's perception to identify. However, it can also be viewed as an opportunity because trees can complicate adversarial drone missions and can be utilised to provide an advantage for the Eagle Drone to ambush adversarial

		drones. Meanwhile, we identified a Soft Hazard concerning the relationship between agitated locals and the presence of surveillance robots, using the destruction of nature as a pretext to obstruct the continuation of the Eagle Drone's operations
[YY]: AS Hazardous Scenarios Validation Report:	This validation document confirms the completeness and correctness of the identified hazardous scenarios. The systems approach artefact should demonstrate that potential hazards are comprehensively covered and that all have been accounted for in the design.	The outputs of all predictive thinking pipelines and design steps provide comprehensive justification and design traceability for hazardous scenarios, including Black Swan scenarios. Validation is done by documenting expert reviews of architect assertions and predictions and the appropriate application of SECoT. A validation report template has been generated, which can be found in: • L.1 SECoT Validation Report Template • Examples: • L.2 SECoT Validation Report for AVOIDDS case study • L.3 SECoT Validation Report for Unsafe Train Tracks case study

[J]: AS Hazardous Scenarios Assurance Argument & [I]: AS Hazardous Scenarios Assurance Argument Pattern:	A structured assurance argument demonstrating that all hazardous scenarios have been sufficiently identified and validated.	• L.4.2.1 Argument Pattern for Eagle Robot Hazardous Scenarios. • L.4.2.2 Argument Pattern for AVOID System Hazardous Scenarios.

L.4.3 Stage 3: Safe Operating Concept Assurance

This phase involves actions aimed at defining and validating the safe operating concept for an AS. The following are the primary outcomes of this stage:

1. The Safe Operating Concept for the AS is clearly defined.

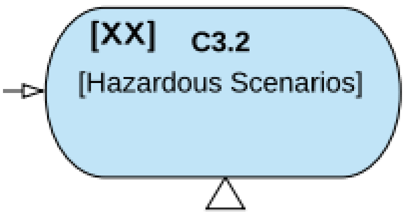
2. The Safe Operating Concept is validated.

3. Development of the Safe Operating Concept Assurance Argument.

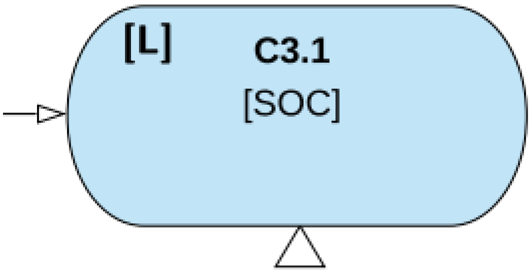
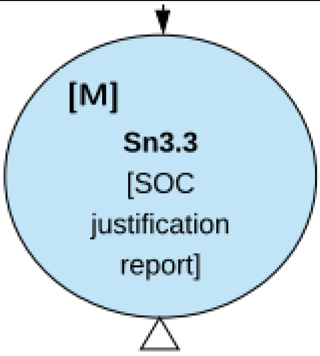
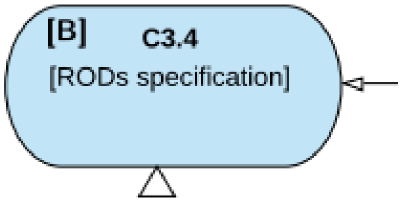
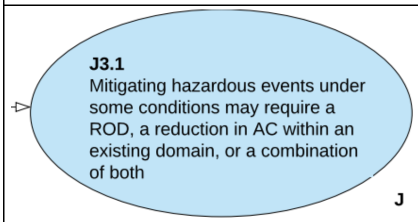
AIC Systems Approach Processes involved to satisfy the objectives:

- Stage 3A: HazTOPS and Ordered AIC-driven Autonomous System Requirements Development
- Stage 3B: Comprehensive Operational Environment Definition
- Stage 4: Disordered AIC-Driven Black Swan Scenarios Prediction

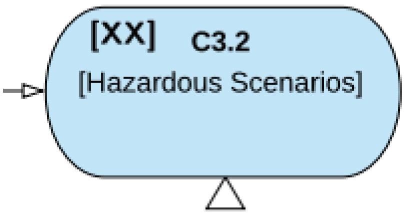
L.4.3.1 Argument Pattern for Eagle Robot SOC Assurance

Assurance Argument Pattern	**AIC Systems Approach supportive methods and artefacts;** The following process outputs satisfy the artefact's demonstration requirements:
	H.6.3 Predictive Thinking Pipeline 3: Hazards, Threats and Opportunities Scenarios (HazTOPs) Analysis. Figure H.16 Hazards Complexity Field Scope: graphically scoping the hazards within the

	complexity field by placing hazard icons on target interaction. Figure H.17 Threats Complexity Field Scope Figure H.18 Opportunities Complexity Feels Scope H.6.3.2 Step 2) Characterise the scoped interactions. Figure H.19 Hazards associated with Eagle Drone preventing derailed train complexity field Table H.24 The table describes the AIC interaction dynamics between Eagle Drones and adversarial drones H.6.3.3 Step 3) Apply predictive potential complications guide words. Table H.25 HazTOPs Analysis of "3 Drones Attack" Scenario with Risk and Surprise Assessment Table H.26 HazTOPs Analysis of adversarial drone using smart lasers scenario Table H.27 HazTOPs Analysis of adversarial drone hiding behind fence scenario Figure **Error! No text of specified style in document.**.3 Soft Hazard Complexity Field Model

	H.6.4.1,2 (c) Step 3) Ordered-AIC-based Mitigating System or Safety Requirements Derivation (Safety Concept) H.8.5 Step 5) Define mitigating ML Development and Safety Requirements.
	H.6.4 Predictive Thinking Pipeline 4:Elicitate AIC System-Level Requirements and Training Requirements
	To be completed [outside PhD scope]
	To be completed [outside PhD scope]

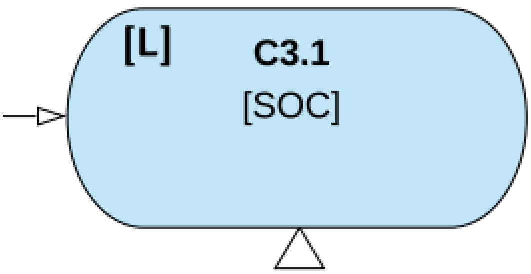
L.4.3.2 Argument Pattern for AVOID System SOC Assurance

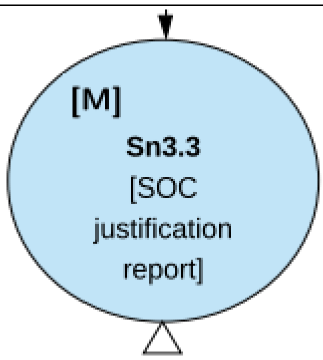
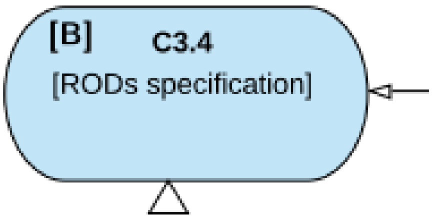
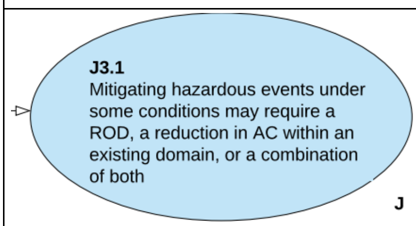
Assurance Argument Pattern	**AIC Systems Approach supportive methods and artefacts;** The following process outputs satisfy the artefact's demonstration requirements:
	I.6.3 Predictive Thinking Pipeline 3: Hazards, Threats and Opportunities Scenarios (HazTOPs) Analysis. Figure I.11Sources of Hazards AIC Complexity Field

<table>
<tr>
<td></td>
<td>Figure H.17 Threats Complexity Field Scope

Figure I.12 Sources of Threats AIC Complexity Field

Table I.22 Considering Hazards related to I1 interaction

Table I.23 Example “More” guide word complication</td>
</tr>
<tr>
<td>
</td>
<td>I.6.4 Predictive Thinking Pipeline 4:Elicitate Ordered AIC System-Level Requirements and training requirements

I.6.4.1,2 (c) Step 3) Ordered-AIC-based Mitigating System or Safety Requirements Derivation (Safety Concept)</td>
</tr>
<tr>
<td>
</td>
<td>I.6.4 Predictive Thinking Pipeline 4:Elicitate Ordered AIC System-Level Requirements and training requirements</td>
</tr>
<tr>
<td>
</td>
<td>To be completed [outside PhD scope]</td>
</tr>
<tr>
<td>
</td>
<td>To be completed [outside PhD scope]</td>
</tr>
</table>

The following are the primary artefacts that need to be presented:

Table **Error! No text of specified style in document.**.4 SACE Stage 3 artefacts and AIC approach mapping

SACE Artefact	Explanation	The substantiating AIC methods and artefacts
[K]: Definition of Sufficiently Safe	Defines what constitutes an acceptable level of safety for the AS, considering legal, ethical, and stakeholder risk tolerance factors. The systems approach artefact must justify why the defined safety criteria are sufficient, referencing legal and regulatory guidelines, ethical considerations, and risk acceptance criteria.	The structure of SECoTs can be used as evidence to justify how the architect arrived at predicting hazards and Black Swan scenarios in the context of providing a comprehensive justification. However, justifying based on specific ethical standards or comparisons to human performance is outside the scope of the PhD. Nonetheless, SECoT is ethics-based on the application of universal systems rules about complexity. Ethical regulations can substitute those rules and thus provide systematic evidence of ethical design considerations for assurance purposes.
[L]: Safe Operating Concept Definition	A formal specification of how the AS must operate within its defined environment to ensure safety, incorporating necessary constraints and system safety requirements. The systems approach artefact must define specific	The safety concept comprises a set of safety requirements at the system level, not solely for the ML component. In our approach, we combine the system-level requirements with the training requirements (training concept), which do

	system-level safety requirements, ensuring they are clear, unambiguous, and sufficient to mitigate hazardous scenarios.	not precisely specify how to implement the training. S6age 4 provides further safety requirements based on mitigating Black Swan scenarios. The following is examples of how we captured safety concept and training concept: • H.6.4.1,2 (c) Step 3) Ordered-AIC-Based Mitigating System or Safety Requirements Derivation (Safety Concept) • H.6.4 Predictive Thinking Pipeline 4:Elicitate AIC System-Level Requirements and Training Requirements • H.8.5 Step 5) Define mitigating ML Development and Safety Requirements.
[M]: Safety Concept (SOC) Justification Report	A structured report validating that the SOC sufficiently mitigates the identified hazardous scenarios. The systems approach artefact must systematically justify how each safety requirement and operational constraint contributes to mitigating specific hazardous scenarios.	Stages 3A and 4 provide the comprehensive justification of how the architect arrives at their predictions and what mitigation plans should be considered. For example, • I,H.6.3 Predictive Thinking Pipeline 3: Hazards, Threats and Opportunities

		Scenarios (HazTOPs) Analysis • I, H.6.4 Predictive Thinking Pipeline 4:Elicitate AIC System-Level Requirements and Training Requirements
[N]: SOC Assurance Argument Pattern & [O]: SOC Assurance Argument:	A structured framework to argue that the Safe Operating Concept sufficiently mitigates all hazardous scenarios identified in previous stages. The systems approach artefact must systematically justify how each safety requirement and operational constraint contributes to mitigating specific hazardous scenarios.	We constructed the GSN argument in the following sections: • L.4.3.1 Argument Pattern for Eagle Robot SOC Assurance. • L.4.3.2 Argument Pattern for AVOID System SOC Assurance

L.4.4 Stage 4: Safety Requirements Assurance

SACE stage 4 revolves around demonstrating that the safety requirements are comprehensively and correctly captured. The primary outcome of this stage is:

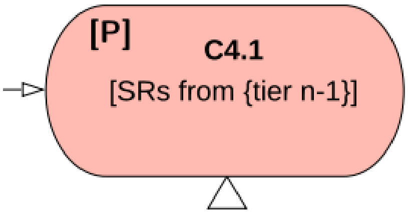
1. Safety requirements for each tier of the requirements development is clearly defined.
2. The defined safety requirements are validated.
3. Development of Safety Requirements Argument.

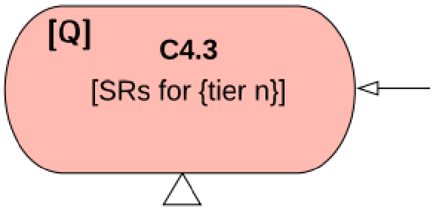
AIC Systems Approach Processes involved to satisfy the objectives:

The pattern is mandated for each tier of the system approach. Therefore, the following structure would be:

- **Tier n-3:** Stage 2: Architect Intent and Autonomous Solution Needs Definition. Sections: I.5, H.5.
- **Tier n-2:** Stage 3A: HazTOPS and Ordered AIC-driven Autonomous System Requirements Development. Sections: I.6, H.6.
- **Tier n-1**: Stage 4: Disordered AIC-Driven Black Swan Scenarios Prediction. Sections: I.8, H.8.
- **Tier n:** Stage 5: CuneiForm-based Syllabus for Safety-Driven ML Epistemic Intelligence Development. Sections: I.9, H.9.

L.4.4.1 Argument Pattern for Eagle Robot Safety Requirements Assurance

Assurance Argument Pattern	**AIC Systems Approach supportive methods and artefacts;** The following process outputs satisfy the artefact's demonstration requirements:
 Given: Tier n: Safety Driven CuneiForm Characterisation Then Tier n-1 : Safety ML Development Requirements (Training Concept)	Stage 3A, (c) Step 3) Ordered-AIC-based Mitigating System or Safety Requirements Derivation (Safety Concept). Table H.37 Eagle Drone Safety Training Requirements for Black Swan Scenarios Table H.38 ML component training dataset requirements

<table>
<tr>
<td>

Tier n: Safety Driven CuneiForm Characterisation</td>
<td>Table H.39 CuneiForm Pictorial situation articulation

Table H.40 Characteristic Training Classes definitions for a CuneiForm abstract image

Figure H.36 Example output CuneiForm with appropriate instantiation using a simple CuneiForm canvas template example. We defined a Black Swan Scenarios Validation Dataset.

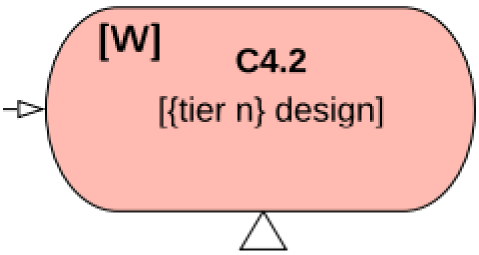
</td>
</tr>
<tr>
<td>
</td>
<td>Table H.37 Eagle Drone Safety Training Requirements for Black Swan Scenarios

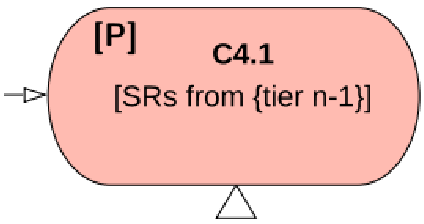
Figure H.36 Example output CuneiForm with appropriate instantiation using the CuneiForm canvas template example. We defined a Black Swan Scenarios Validation Dataset.

Table H.35 Operational Design Definition for Eagle Robot Deployment in Train Track Zone

Table H.31 4HnWs method for Eagle Drone adjusting patrol functionality</td>
</tr>
</table>

	Table H.15 Architect High-Level Solution Prescription Table H.16 Architect High-Level Solution Prescription related to the impact of roaming adversarial drones Table H.17 Architect High-Level Solution Prescription related to the police incapability to capture adversarial drone
	All stages, from stage 1 to stage 5,

L.4.4.2 Argument Pattern for AVOID Safety Requirements Assurance

Assurance Argument Pattern	**AIC Systems Approach supportive methods and artefacts;** The following process outputs satisfy the artefact's demonstration requirements:
 Tier n-1 : Safety ML Development Requirements (Training Concept)	Table I.31a AVP Training Requirements for Black Swan Scenarios Table I.31b ML Safety Training Requirements and Perception Dataset Specifications for AVP

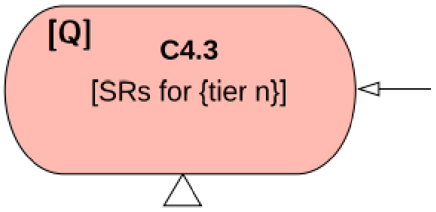
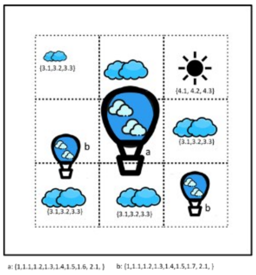
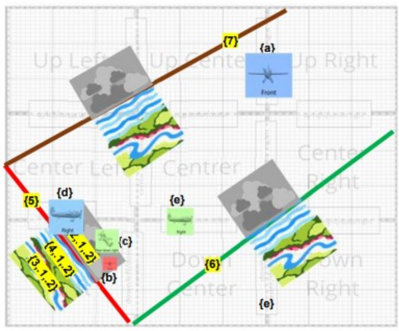
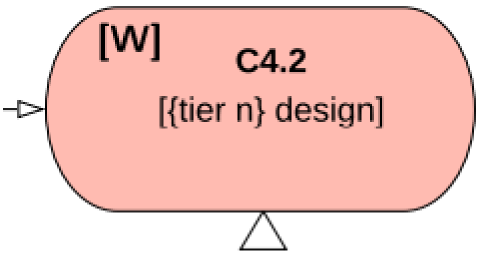
 Tier n: Safety Driven CuneiForm Characterisation	Table I.32 CuneiForm Pictorial situation articulation Table I.33 Characteristic Training Classes definitions for a CuneiForm abstract image Figure I.20 Example CuneiForm and instantiated image   Appendix D Safety Validation report for AVOIDDS dataset. Pages: 15, 19, 23, 27, 31, 35. For example, the following is safety-driven CuneiForm characterisation of a training dataset. 
 	Table I.13 Architect High-Level Solution Prescription Table I.26 Safety requirements derivations to mitigate the concealed drone problem. Table I.27 Operational Design Definition for AVP

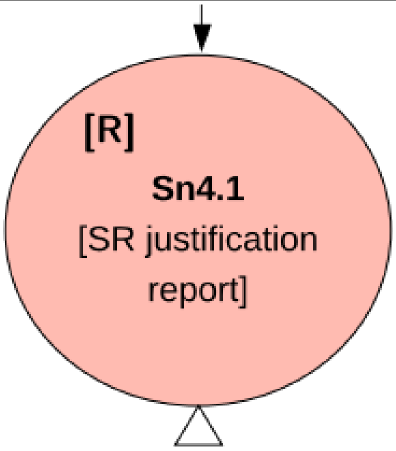
	Table I.31a AVP Training Requirements for Black Swan Scenarios Figure I.20 Example CuneiForm and instantiated image
	All stages, from stage 1 to stage 5.

Table L.5 captures primary artefacts that need to be presented:

Table **Error! No text of specified style in document.**.5 SACE Stage 4 artefacts and AIC approach mapping

SACE Artefact	**Explanation**	**The substantiating AIC methods and artefacts**
[P]: Safety Requirements from tier n-1 & [Q]: Safety Requirements for tier n:	The safety requirements are defined at the higher level tier of decomposition, which must be correctly allocated and interpreted at the current tier. The systems approach artefact must demonstrate that higher-level safety requirements are adequately decomposed, ensuring consistency and completeness in allocating to system components	in the AIC systems approach, the tier-n-1 is relative; for example, the Stage 3A, (c) Step 3) Ordered-AIC-based Mitigating System or Safety Requirements Derivation (Safety Concept). Would be the tier n-1 for (d) Step 4) Extended Concrete Safety Concept and ML Safety Training Concept, output (tier n)

		Also, Given: Tier n: Safety Driven CuneiForm Characterisation Then, Tier n-1 : Safety ML Development Requirements (Training Concept) Therefore, the following would be evidence to be provided:
[R]: Safety Requirements Justification Report	A structured report validating that the decomposed safety requirements adequately maintain the intent of the original requirements. The systems approach artefact must provide traceability and justification for each safety requirement, ensuring they correctly address the identified hazards and are appropriately assigned to system components.	All stages, from stage 1 to stage 5, operate based on tight traceability. They also operate on the premise of exposing the architectural thought process that informs any decision, thus providing a very clear insight into the justification behind design choices and the principles used to make such engineering decisions. For example: H, I.6 Stage 3A: HazTOPs and Ordered AIC-driven Autonomous System Requirements Development Since we are including CuneiForms as the implementation of SOC, then the following tables will provide the justification required (examples):

		Table M.21 Characteristic Training Classes definitions for a CuneiForm abstract image (AVOIDDS) Table M.20 CuneiForm Pictorial situation articulation (AVOIDDS) Table M.19 ML Safety Training Requirements and Perception Dataset Specifications for AVP Table M.18 AVP Training Requirements for Black Swan Scenarios (AVOIDDS)
[W]: tier n Design:	The design specification at the current tier of decomposition defines system components and their interactions. The systems approach artefact must explicitly define design requirements which meet the higher-level safety requirements.	These are the more concrete requirements or specifications directly relating to higher-level safety requirements. For example, • Table H.31 4HnWs method for Eagle Drone adjusting patrol functionality. • Figure I.20 Example CuneiForm and instantiated image. We would also include the following tables as evidence: Table 7.10 Black Swan Scenarios Batch A and B CuneiForms Table 7.11 Typical operations CuneiForms

		Figure 7.6 H.54 Out-of-context CuneiForm of drones. Figure M.20 Example CuneiForm and instantiated image for AVOIDDS case study They represent the concrete level implementation of safety concept at dataset level.
[S]: Safety Requirements Argument Pattern & [T]: Safety Requirements Argument:	A structured framework for demonstrating that the safety requirements at each tier adequately capture the intent of the previous tier's requirements.	We captured the patterns for this stage in the following sections: • L.4.5.1 Argument Pattern for Eagle Robot Design Assurance • L.4.5.2 Argument Pattern for AVOID system Design Assurance

L.4.5 Stage 5: AS Design Assurance

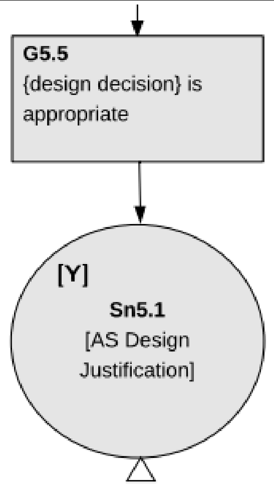
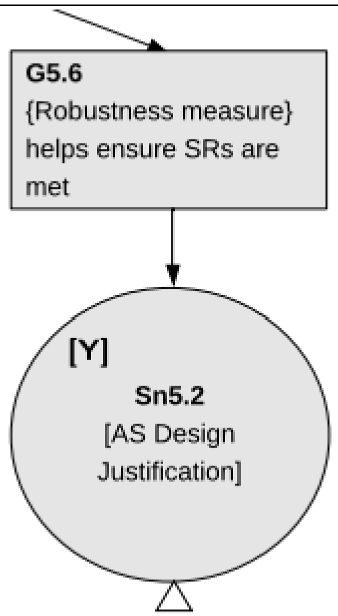
This stage focuses on assuring the concrete design. It is iterative in nature, accounting for the AS's design assurance at various design decomposition levels. The following are the main outcomes:

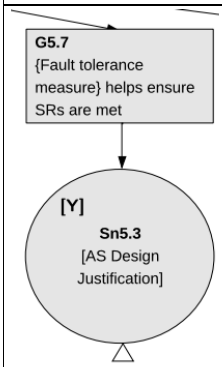
1. Design at tier n, which satisfies safety requirements, is specified.
2. The sufficiency of the design at tier n is justified.
3. The claim of sufficiency is validated.
4. Development of As design assurance argument.

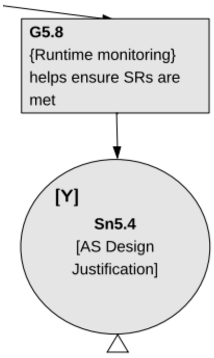
AIC Systems Approach Processes involved to satisfy the objectives:

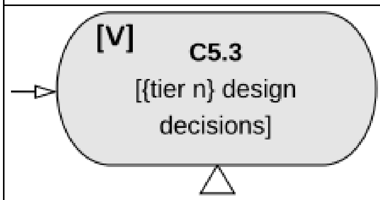
- All AIC Systems Approach stages.

L.4.5.1 Argument Pattern for Eagle Robot Design Assurance

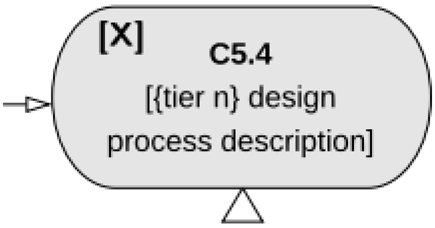
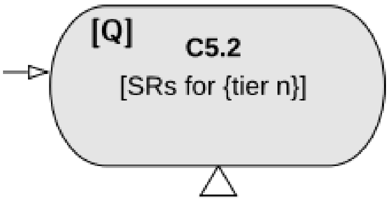
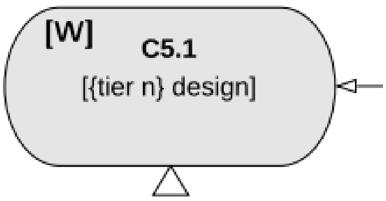
Assurance Argument Pattern	**AIC Systems Approach supportive methods and artefacts;** The following process outputs satisfy the artefact's demonstration requirements:
 Justification that key design decisions taken at tier n are appropriate to meet the defined safety requirements.	All Predictive Thinking Pipelines and design steps' outputs
 Demonstrating that the system has been designed to be robust against potential unexpected environmental changes.	Table H.37 Eagle Drone Safety Training Requirements for Black Swan Scenarios Figure H.36 Example output CuneiForm with appropriate instantiation using a simple CuneiForm canvas template example. We defined a Black Swan Scenarios Validation Dataset. Table H. 42 Comparison of Black Swan Lacked vs. Black Swan Enhanced Object Detection Experiments H.10.5 Experiment 1: Limited Live video-based experimentation

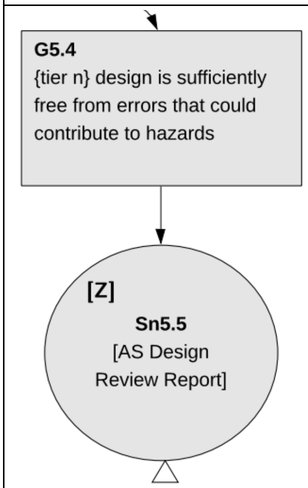
<table>
<tr><td></td><td>H.10.6 Experiment 2: ML development environment-based validation</td></tr>
<tr><td>

Justification of fault tolerance mechanisms ensuring continued operation despite failures.</td><td>Outside PhD scope</td></tr>
<tr><td>

Demonstrating that the system has an active monitoring mechanism to track and adapt its behavior based on operational feedback.</td><td>Outside PhD scope</td></tr>
<tr><td>

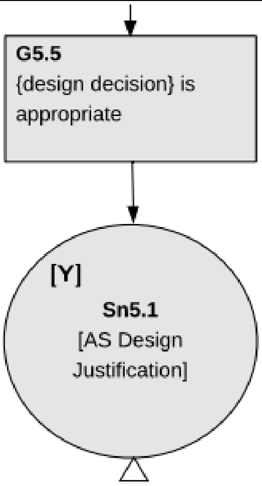
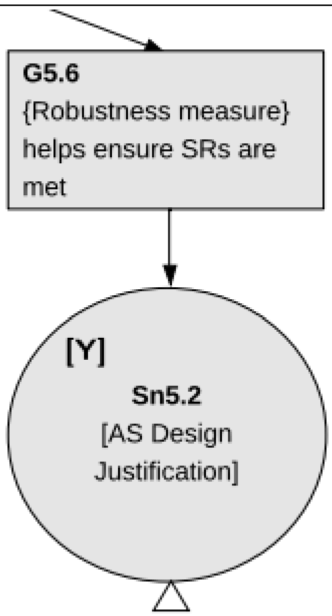
Presenting an argument over the appropriateness of each design decision to ensure system safety.</td><td>All SECoTs present a clear argument on the thought processes that went into</td></tr>
</table>

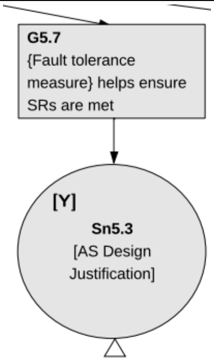
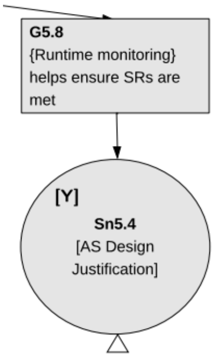
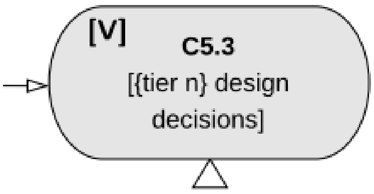
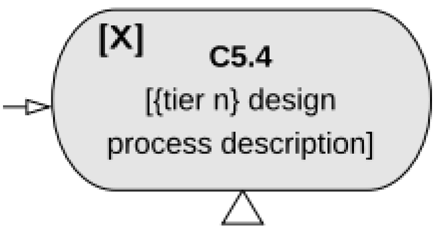
 Ensuring that the design process itself includes sufficient validation steps to prevent hazardous failures.	L.1 SECoT Validation Report Template L.3 SECoT Validation Report for Unsafe Train Tracks case study Stage (1-6) in AIC Systems Approach H.10.5 Experiment 1: Limited Live video-based experimentation H.10.6 Experiment 2: ML development environment-based validation
 The context must demonstrate that the defined safety requirements at tier n comprehensively address operational risks, failure modes, and system constraints, ensuring the AS design meets safety assurance objectives and regulatory compliance.	Table H.33 Safety requirements derivations to mitigate the concealed drone problem. Table H.34 ML Safety Requirements Derivation Table H.37 Eagle Drone Safety Training Requirements for Black Swan Scenarios Table H.38 ML component training dataset requirements Figure H.36 Example output CuneiForm with appropriate instantiation using a simple CuneiForm canvas template example. We defined a Black Swan Scenarios Validation Dataset. 
	Table H.37 Eagle Drone Safety Training Requirements for Black Swan Scenarios Figure H.36 Example output CuneiForm with appropriate instantiation using the CuneiForm canvas template

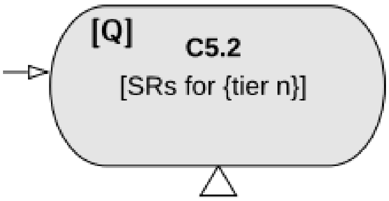
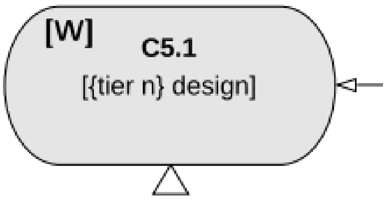
	example. We defined a Black Swan Scenarios Validation Dataset. Table H.35 Operational Design Definition for Eagle Robot Deployment in Train Track Zone Table H.31 4HnWs method for Eagle Drone adjusting patrol functionality Table H.15 Architect High-Level Solution Prescription Table H.16 Architect High-Level Solution Prescription related to the impact of roaming adversarial drones Table H.17 Architect High-Level Solution Prescription related to the police incapability to capture adversarial drone
 	Validation is done by documenting expert reviews of architect assertions and predictions and the appropriate application of SECoT. A validation report template has been generated, which can be found in L.1 SECoT Validation Report Template L.3 SECoT Validation Report for Unsafe Train Tracks case study

L.4.5.2 Argument Pattern for AVOID System Design Assurance

Assurance Argument Pattern	**AIC Systems Approach supportive methods and artefacts;** The following process outputs satisfy the artefact’s demonstration requirements:

 Justification that key design decisions taken at tier n are appropriate to meet the defined safety requirements.	All Predictive Thinking Pipelines and design steps' outputs
 Demonstrating that the system has been designed to be robust against potential unexpected environmental changes.	Table I.31a AVP Training Requirements for Black Swan Scenarios Table I.31b ML Safety Training Requirements and Perception Dataset Specifications for AVP I.9 Stage 5: Safety-Driven ML-based Perception Training, Testing and Validation Design (CuneiForm Strategy Development) I.10 Stage 6: Black Swan-driven ML Development and Testing

 Justification of fault tolerance mechanisms ensuring continued operation despite failures.	Outside PhD scope
 Demonstrating that the system has an active monitoring mechanism to track and adapt its behavior based on operational feedback.	Outside PhD scope
 Presenting an argument over the appropriateness of each design decision to ensure system safety.	All SECoTs present a clear argument on the thought processes that went into
	L.1 SECoT Validation Report Template L.2 SECoT Validation Report for AVOIDDS case study Stage n in AIC Systems Approach

Ensuring that the design process itself includes sufficient validation steps to prevent hazardous failures.	
 The context must demonstrate that the defined safety requirements at tier n comprehensively address operational risks, failure modes, and system constraints, ensuring the AS design meets safety assurance objectives and regulatory compliance.	Given Tier n is the dataset: Table I.26 Safety requirements derivations to mitigate the concealed drone problem. Table I.31a AVP Training Requirements for Black Swan Scenarios Table I.31b ML Safety Training Requirements and Perception Dataset Specifications for AVP Figure I.18 We used DALL-E to generate this Black Swan Scenario for validation. Figure I.20 Example CuneiForm and instantiated image
	Table I.13 Architect High-Level Solution Prescription Table I.26 Safety requirements derivations to mitigate the concealed drone problem. Stage 3A, Step 4) Extended Concrete Safety Concept and ML Safety Training Concept Table I.27 Operational Design Definition for AVP Table I.31a AVP Training Requirements for Black Swan Scenarios Table I.31b ML Safety Training Requirements and Perception Dataset Specifications for AVP Figure I.20 Example CuneiForm and instantiated image

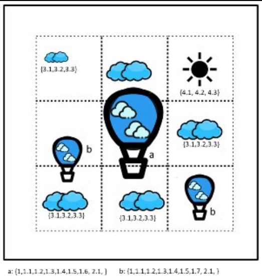
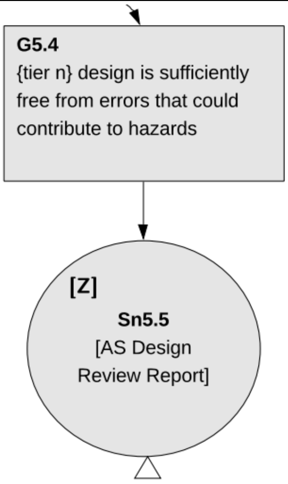
	 
	Validation is done by documenting expert reviews of architect assertions and predictions and the appropriate application of SECoT. A validation report template has been generated, which can be found in L.1 SECoT Validation Report Template L.2 SECoT Validation Report for AVOIDDS case study

Table 8.5 captures primary artefacts that need to be presented:

Table **Error! No text of specified style in document.**.6 SACE Stage 5 artefacts and AIC approach mapping

SACE Artefact	**Explanation**	**The substantiating AIC methods and artefacts**
[V]: AS Development Log, [X]: Design Process for tier n, [Z]: AS Design Review	A comprehensive record of the design evolution, decisions, and iterations taken throughout development. The systems approach artefact must track all design changes, safety considerations, and iterations, ensuring full traceability of design decisions.	All SECoTs present a clear argument on the thought processes that went into all engineering judgements. This can be organised similarly as Appendix H and I. Also, we can add in the validation reporting that captures the design reviews outcomes.
[AA]: AS Design Assurance Argument & [U]: AS Design	we captured the GSN patterns in:	

Assurance Argument Pattern:	• L.4.5.1 Argument Pattern for Eagle Robot Design Assurance • L.4.5.2 Argument Pattern for AVOID system Design Assurance.

L.4.6 Stage 6: Hazardous Failures Management

SACE Stage 6 provides evidence demonstrating that the systems approach had thoroughly considered the hazards analysis. The primary outcomes of stage 6 are:

1. AS Hazardous Failures are identified.
2. The identified AS Hazardous Failures are mitigated.
3. Developing Hazardous Failures Assurance Argument Pattern

AIC Systems Approach Processes involved to satisfy the objectives:

- Stage 3A: HazTOPS and Ordered AIC-driven Autonomous System Requirements Development.
- Stage 3B: Comprehensive Operational Environment Definition.
- Stage 4: Disordered AIC-Driven Black Swan Scenarios Prediction.
- Stage 5: CuneiForm-based Syllabus for Safety-Driven ML Epistemic Intelligence Development.

L.4.6.1 Argument Pattern for Eagle Robot Hazardous Failures

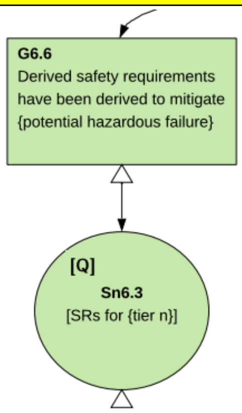
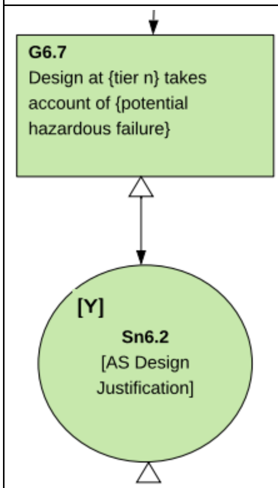
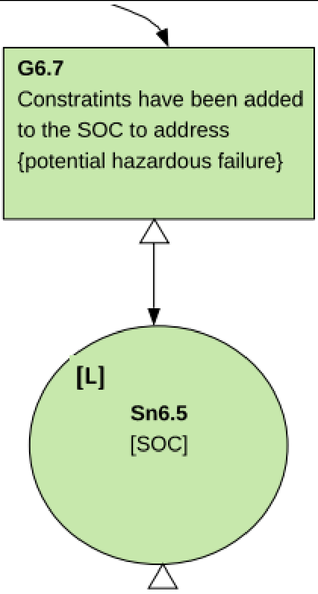
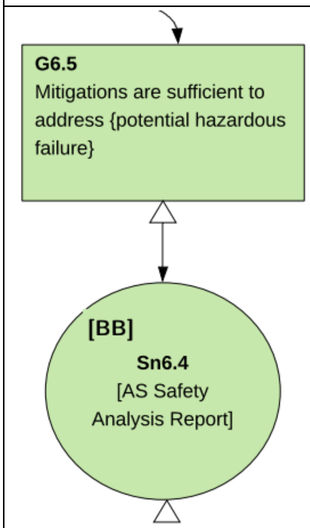
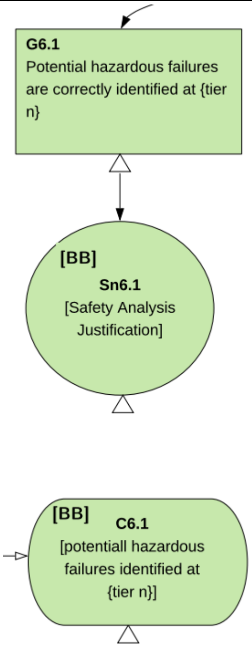
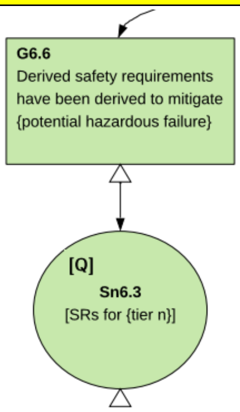
Assurance Argument Pattern	**AIC Systems Approach supportive methods and artefacts;** The following process outputs satisfy the artefact's demonstration requirements:
	Table H.33 Safety requirements derivations to mitigate the concealed drone problem. Table H.34 ML Safety Requirements Derivation Table H.37 Eagle Drone Safety Training Requirements for Black Swan Scenarios

	Table H.38 ML component training dataset requirements Figure H.36 Example output CuneiForm with appropriate instantiation using a simple CuneiForm canvas template example. We defined a Black Swan Scenarios Validation Dataset. 
	Table H.33 Safety requirements derivations to mitigate the concealed drone problem. Table H.34 ML Safety Requirements Derivation Table H.37 Eagle Drone Safety Training Requirements for Black Swan Scenarios Table H.39 CuneiForm Pictorial situation articulation Table H.40 Characteristic Training Classes definitions for a CuneiForm abstract image Figure H.36 Example output CuneiForm with appropriate instantiation using the CuneiForm canvas template example. We defined a Black Swan Scenarios Validation Dataset. 

 Within the context of maitaining Reduced Operating Domain (ROD)	We did not consider ROD in our approach for nw, hence it is Outside PhD scope
	H.6 Stage 3A: HazTOPs and Ordered AIC-driven Autonomous System Requirements Development H.7 Stage 3B: Comprehensive Operational Environment Definition H.8 Stage 4: Disordered AIC-Driven Black Swan Scenarios Prediction

L.4.6.2 Argument Pattern for AVOID system Hazardous Failures

Assurance Argument Pattern	**AIC Systems Approach supportive methods and artefacts;** The following process outputs satisfy the artefact's demonstration requirements:
	Given Tier n is the dataset: Table I.26 Safety requirements derivations to mitigate the concealed drone problem. Table I.31a AVP Training Requirements for Black Swan Scenarios Table I.31b ML Safety Training Requirements and Perception Dataset Specifications for AVP Figure I.18 We used DALL-E to generate this Black Swan Scenario for validation. Figure I.20 Example CuneiForm and instantiated image

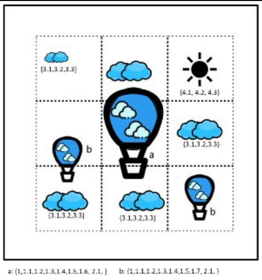
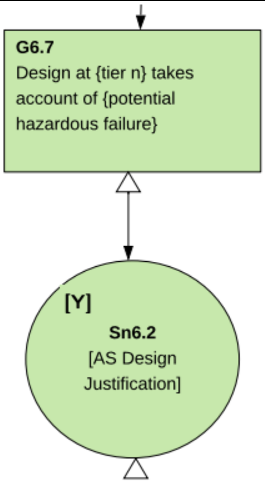
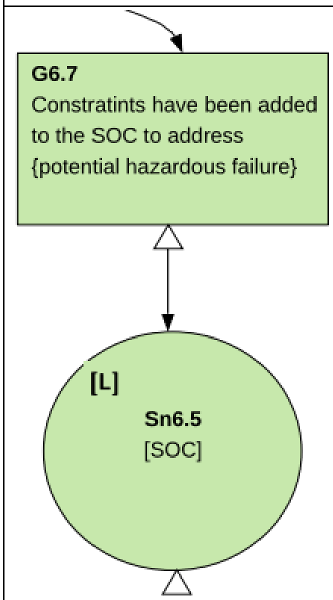
<table>
<tr><td></td><td> </td></tr>
<tr><td>
</td><td>Table I.31a AVP Training Requirements for Black Swan Scenarios

Table I.31b ML Safety Training Requirements and Perception Dataset Specifications for AVP

I.9 Stage 5: Safety-Driven ML-based Perception Training, Testing and Validation Design (CuneiForm Strategy Development)

I.10 Stage 6: Black Swan-driven ML Development and Testing</td></tr>
<tr><td>

Within the context of maitaining Reduced Operating Domain (ROD)</td><td>Outside PhD scope</td></tr>
</table>

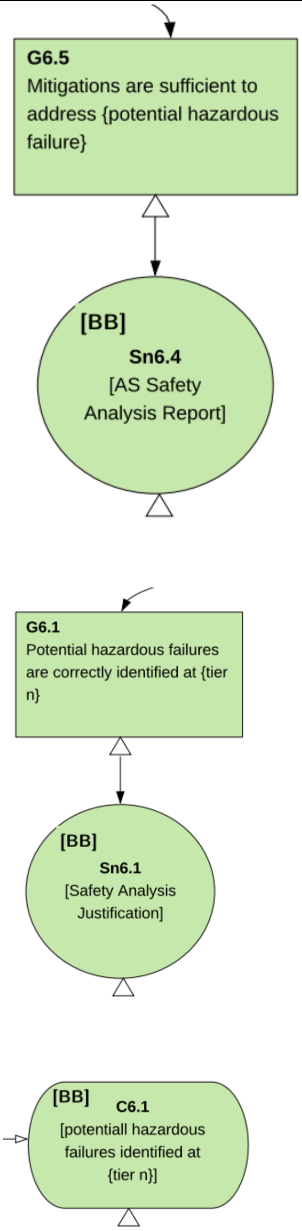
	I.6 Stage 3A: HazTOPs and Ordered AIC-driven Autonomous System Requirements Development I.7 Stage 3B: Comprehensive Operational Environment Definition I.8 Stage 4: Disordered AIC-Driven Black Swan Scenario Predictions and Validation

Table 8.6 captures primary artefacts that need to be presented:

Table **Error! No text of specified style in document.**.7 SACE Stage 6 artefacts and AIC approach mapping

SACE Artefact	**Explanation**	**The substantiating AIC methods and artefacts**
[BB]: AS Safety Analysis Report:	A comprehensive report detailing the identified hazardous failures and the justification for the analysis approach used. The systems	AIC systems approach includes the following techniques dedicated to discovering Hard and Soft hazards and then translating

	approach must document all identified hazardous failures, the rationale for their identification, failure modes, potential consequences, and their impact on AS safety.	them into safety concepts and training concepts: • H.6 Stage 3A: HazTOPs and Ordered AIC-driven Autonomous System Requirements Development • H.7 Stage 3B: Comprehensive Operational Environment Definition • H.8 Stage 4: Disordered AIC-Driven Black Swan Scenario Predictions and Validation
[DD]: Hazardous Failures Argument Pattern & [EE]: Hazardous Failures Argument:	We captured the AIC outputs in the following sections: • L.4.6.1 Argument Pattern for Eagle Robot Hazardous Failures • L.4.6.2 Argument Pattern for AVOID System Hazardous Failures	

L.4.7 Stage 7: Out-of-Context Operation Assurance

Outside PhD scope

L.4.8 Stage 8: AS Verification Assurance

Outside PhD scope

L.4.9 Summary of SACE artefacts and AIC Systems Approach implementation

SACE Artefact	SACE Stage	Definition	General Demonstration Requirements	**Substantiating AIC methods/Example:** The following process outputs satisfy the artefact's demonstration requirements:
[A]: AS Concept Definition	1	High-level document outlining the system's intended functions, objectives, and constraints.	The systems approach artefact must define intended system functionality, scope of autonomy, and interaction with humans. Should include use-case descriptions and stakeholder agreements.	H,I.5 Stage 2: Architect Intent and Autonomous Solution Needs Definition Table H.15 Architect High-Level Solution Prescription Table I.13 Architect High-Level Solution Prescription
[AA]: AS Design Assurance Argument [U] : AS Design Assurance Argument Pattern	5	A structured assurance case demonstrates that tier n's design sufficiently satisfies the defined safety requirements.	The systems approach artefact must logically argue that design decisions at each tier ensure safety, referencing artefacts [Y], [Z], and [V] to provide supporting evidence.	L.4.5.1 Argument Pattern for Eagle Robot Design Assurance L.4.5.2 Argument Pattern for AVOID system Design Assurance
[B] : Operational Domain Model (ODM)	1	A model defining the scope of operation for the AS, including assumptions about the environment and operating conditions.	The systems approach artefact must include all relevant environmental and operational conditions, ensuring that all scenarios the AS may encounter are covered. It should explicitly list	H,I.4 Stage 1: Uncertainty Problem Articulation and Operational Environment Modelling Table H.35 Operational Design Definition for Eagle

			assumptions, constraints, and non-mission interactions.	Robot Deployment in Train Track Zone Table I.27 Operational Design Definition for AVP
[BB] : AS Safety Analysis Report	6	A comprehensive report detailing the identified hazardous failures and the justification for the analysis approach used.	The systems approach artefact must document all identified hazardous failures, the rationale for their identification, failure modes, potential consequences, and their impact on AS safety.	H.6 Stage 3A: HazTOPs and Ordered AIC-driven Autonomous System Requirements Development H.7 Stage 3B: Comprehensive Operational Environment Definition H.8 Stage 4: Disordered AIC-Driven Black Swan Scenarios Prediction I.6 Stage 3A: HazTOPs and Ordered AIC-driven Autonomous System Requirements Development I.7 Stage 3B: Comprehensive Operational Environment Definition I.8 Stage 4: Disordered AIC-Driven Black Swan Scenario Predictions and Validation

[C] : ODM Validation Report	1	A documented validation of the completeness and correctness of the ODM, ensuring it sufficiently defines the operating scope.	Must provide evidence of review, simulation testing, and field validation to verify that all necessary operational elements have been captured. It should also justify the granularity level of the ODM.	L.1 SECoT Validation Report Template L.2 SECoT Validation Report for AVOIDDS case study L.3 SECoT Validation Report for Unsafe Train Tracks case study
[D] : Autonomous Capabilities Definition	1	A document specifying the AS's autonomous functionalities, limitations, and human-AS interaction boundaries.	The systems approach artefact must clearly define the scope of autonomy, specifying tasks the AS can perform independently and those requiring human intervention. Should also outline conditions under which autonomy is constrained.	Table H.15 Architect High-Level Solution Prescription. Table I.13 Architect High-Level Solution Prescription H,I.6 Stage 3A: HazTOPs and Ordered AIC-driven Autonomous System Requirements Development
[DD] : Hazardous Failures Argument Pattern [EE]: Hazardous	6	A structured framework for reasoning about identifying and mitigating hazardous failures at each tier of AS design.	The systems approach artefact must demonstrate that hazardous failures have been systematically identified and addressed, ensuring that the AS design	L.4.6.1 Argument Pattern for Eagle Robot Hazardous Failures L.4.6.2 Argument Pattern for AVOID system Hazardous Failures

Failures Argument			mitigates safety risks effectively.	
[E] : Operating Scenarios Definition	1	A detailed description of all scenarios the AS may encounter within the defined ODM, including static and dynamic interactions.	The systems approach artefact must define actions, events, and environmental conditions affecting the AS. Should specify scenes (static conditions) and scenarios (temporal developments over time).	H.4.4.3 Step 4.3) Define the assumptions made about factors. Table J.8 unsafe train tracks problem domain assumptions. H.4.3.1 Step 3.1) Model detailed AIC interactions scenarios for the problem domain. Figure H.3 Modelling a complicated interaction n6. Table J.6 AIC problem domain scenarios definition H.4.3.2 Step 3.2) Predict the extended list of emergent AIC interactions scenarios. Table H.10 Complexity Field for n6 Interaction SECoT definition. Table J.7 AIC Extended Scenarios. H.6.1 Predictive Thinking Pipeline 1: Introducing Autonomous systems into Feed-forward complexity. Table H.18 Implementing Architect Intent and Forward-Feed AIC Interaction Framework for addressing train derailment

				caused by adversarial drones. Table H.19 Mapping AIC interactions of the Eagle Drone and adversarial drone behaviours in mitigating train derailment risks. H.7 Stage 3B: Comprehensive Operational Environment Definition. Table H.35 Operational Design Definition for Eagle Robot Deployment in Train Track Zone. I.4.4.3 Step 4.3) Define the assumptions and hazards made about factors. Table K.2 Extended assumptions, plausibility, concern and hazards analysis. In no particular order. I.4.3.1 Step 3.1) Model detailed AIC interactions scenarios for the problem domain. Figure I.2 Modelling AIC scenario from interaction n1. I.4.3.2 Step 3.2) Predict the extended list of emergent AIC interactions scenarios.

				Table I.10 AIC extended scenario for n1 Interaction SECoT definition. Table K.2 Extended assumptions, plausibility, concern and hazards analysis. In no particular order. I.6.1 Predictive Thinking Pipeline 1: Introducing Autonomous systems into Feed-forward complexity. Table I.14 Implementing Architect Intentand Forward-Feed AIC Interaction Framework for addressing AVP reliability for by-passing aircraft. Table I.15 Mapping AIC interactions of AVP with ownship aircraft and the environment I.7 Stage 3B: Comprehensive Operational Environment Definition. Table I.27 Operational Design Definition for AVP
[F] : Operating Scenarios Validation Report	1	A validation document assessing whether the defined operating scenarios comprehensively capture all	The systems approach artefact must provide evidence of expert review, simulation-based verification, and real-world validation data to confirm the completeness of the operating scenarios.	Validation is done by documenting expert reviews of architect assertions and predictions

		relevant AS interactions.		and the appropriate application of SECoT. A validation report template has been generated, which can be found No validation report had been generated as part of PhD scope.
[G] : AS Operating Context Assurance Argument Pattern [H] : AS Operating Context Assurance Argument	1	A structured assurance argument framework for ensuring the AS operates safely within its defined context.	The systems approach artefact must logically argue that the defined Operational Domain Model (ODM) and operating scenarios are sufficient, comprehensive, and correctly validated to support safe operations.	L.4.1.1 Argument Pattern for Eagle Robot Operating Context Assurance L.4.1.1 Argument Pattern for AVOID System Operating Context Assurance
[I] : AS Hazardous Scenarios Assurance Argument Pattern [J] : AS Hazardous Scenarios Assurance Argument	2	A structured framework ensures that all hazardous scenarios have been correctly identified and analysed.	The systems approach artefact must logically argue and provide evidence that all hazardous scenarios have been identified, relevant decisions analysed, and interactions between the AS and its environment adequately considered.	L.4.2.1 Argument Pattern for Eagle Robot Hazardous Scenarios L.4.2.2 Argument Pattern for AVOID System Hazardous Scenarios
[K] : Definition of Sufficiently Safe	3	Defines what constitutes an acceptable level	The systems approach artefact must justify why the defined safety	Outside PhD scope

		of safety for the AS, considering legal, ethical, and stakeholder risk tolerance factors.	criteria are sufficient, referencing legal and regulatory guidelines, ethical considerations, and risk acceptance criteria. It may include comparisons to human operators or specific scenario-based safety assessments.	
[L] : Safe Operating Concept Definition	3	A formal specification of how the AS must operate within its defined environment to ensure safety, incorporating necessary constraints and system safety requirements.	The systems approach artefact must define specific system-level safety requirements, ensuring they are clear, unambiguous, and sufficient to mitigate hazardous scenarios. Should reference the Operational Domain Model (ODM) and the AS's autonomous capabilities.	H.6.4.1,2 (c) Step 3) Ordered-AIC-based Mitigating System or Safety Requirements Derivation (Safety Concept) H.6.4 Predictive Thinking Pipeline 4:Elicitate AIC System-Level Requirements and Training Requirements
[M] : SOC Justification Report	3	A structured report validating that the SOC sufficiently mitigates the identified hazardous scenarios.	The systems approach artefact must systematically justify how each safety requirement and operational constraint contributes to mitigating specific hazardous scenarios. Stakeholder reviews, scenario simulations, and expert evaluations should be included as validation methods.	I,H.6.3 Predictive Thinking Pipeline 3: Hazards, Threats and Opportunities Scenarios (HazTOPs) Analysis I, H.6.4 Predictive Thinking Pipeline 4:Elicitate AIC System-Level Requirements and Training Requirements

[N] : SOC Assurance Argument Pattern [O] : SOC Assurance Argument	3	A structured framework to argue that the Safe Operating Concept sufficiently mitigates all hazardous scenarios identified in previous stages.	The systems approach artefact must logically demonstrate that the SOC addresses all identified hazardous scenarios, ensuring that system safety requirements, reduced operating domains (RODs), and constraints sufficiently mitigate risks.	L.4.3.1 Argument Pattern for Eagle Robot SOC Assurance L.4.3.2 Argument Pattern for AVOID System SOC Assurance
[P] : Safety Requirements from tier n-1	4	The safety requirements are defined at the previous tier of decomposition, which must be correctly allocated and interpreted at the current tier.	The systems approach artefact must demonstrate that higher-level safety requirements are adequately decomposed, ensuring consistency and completeness in allocating system components.	(c) Step 3) Ordered-AIC-based Mitigating System or Safety Requirements Derivation (Safety Concept) Table H.37 Eagle Drone Safety Training Requirements for Black Swan Scenarios Table H.38 ML component training dataset requirements Table I.31a AVP Training Requirements for Black Swan Scenarios Table I.31b ML Safety Training Requirements and Perception Dataset Specifications for AVP
[Q] : Safety Requirements for tier n	4	The newly defined safety requirements at the current tier,	The systems approach artefact must prove that these safety requirements align with	Table H.39 CuneiForm Pictorial situation articulation

		allocated to relevant system components.	the previous tier's intent, effectively mitigating risks and guiding system design.	Table H.40 Characteristic Training Classes definitions for a CuneiForm abstract image Figure H.36 Example output CuneiForm with appropriate instantiation using a simple CuneiForm canvas template example. We defined a Black Swan Scenarios Validation Dataset. Table I.32 CuneiForm Pictorial situation articulation Table I.33 Characteristic Training Classes definitions for a CuneiForm abstract image Figure I.20 Example CuneiForm and instantiated image
[R] : Safety Requirements Justification Report	4	A structured report validating that the decomposed safety requirements adequately maintain the intent of the original requirements.	The systems approach artefact must provide traceability and justification for each safety requirement, ensuring they correctly address the identified hazards and are properly assigned to system components.	H, I.6 Stage 3A: HazTOPs and Ordered AIC-driven Autonomous System Requirements Development H,I.8 Stage 4: Disordered AIC-driven Black Swan ScenariosPredictions and Validation H,I.10 Stage 6: Black Swan-driven ML Development and Testing H,I.9 Stage 5: CuneiForm-based Syllabus for Safety-

				Driven ML Epistemic Intelligence Development
[S] : Safety Requirements Argument Pattern [T] : Safety Requirements Argument	4	A structured framework for demonstrating that the safety requirements at each tier adequately capture the intent of the previous tier's requirements.	The systems approach artefact must logically show that the defined safety requirements at each tier align with those from the previous tier, maintaining intent and ensuring completeness.	L.4.5.1 Argument Pattern for Eagle Robot Design Assurance L.4.5.2 Argument Pattern for AVOID system Design Assurance
[V] : AS Development Log	5	A comprehensive record of the design evolution, decisions, and iterations taken throughout development.	The systems approach artefact must track all design changes, safety considerations, and iterations, ensuring full traceability of design decisions.	All SECoTs present a clear argument on the thought processes that went into all engineering judgement.
[W]: tier n Design	4	The design specification at the current tier of decomposition defines system components and their interactions.	The systems approach artefact must explicitly define design requirements which meet the higher-level safety requirements, ensuring that architectural decisions address identified risks and failure modes.	Table H.37 Eagle Drone Safety Training Requirements for Black Swan Scenarios Figure H.36 Example output CuneiForm with appropriate instantiation using the CuneiForm canvas template example. We defined a Black Swan Scenarios Validation Dataset. Table H.31 4HnWs method for Eagle Drone adjusting patrol functionality

				Stage 3A, Step 4) Extended Concrete Safety Concept and ML Safety Training Concept Table I.31a AVP Training Requirements for Black Swan Scenarios Table I.31b ML Safety Training Requirements and Perception Dataset Specifications for AVP Figure I.20 Example CuneiForm and instantiated image
[WW] : AS Decision Analysis Report	2	A report analysing the decisions made by the AS at key decision points within different operating scenarios.	The systems approach artefact must demonstrate all decision points have been identified and analysed for potential hazards, considering different environmental conditions and belief states of the AS. Should include examples of decision failures and their potential hazardous outcomes.	H.6.1Predictive Thinking Pipeline 1: Introducing Autonomous systems into Feed-forward complexity H.6.2 Predictive Thinking Pipeline 2: Designing the affecting Backward-Feed complexity field H.9 Stage 5: CuneiForm-based Syllabus for Safety-Driven ML Epistemic Intelligence Development I.6.1 Predictive Thinking Pipeline 1: Introducing Autonomous systems into Feed-forward complexity I.6.2 Predictive Thinking Pipeline 2: Designing the affecting Backward-Feed complexity field

				I.9 Stage 5: CuneiForm-based Syllabus for Safety-Driven ML Epistemic Intelligence Development
[X] : Design Process for tier n	5	A structured process outlining the methodology for developing and validating the AS design at tier n.	The systems approach artefact must document and justify the design approach, ensuring that potential hazards are considered and that design decisions align with safety requirements.	L.1 SECoT Validation Report Template L.3 SECoT Validation Report for Unsafe Train Tracks case study Stage (1-6) in AIC Systems Approach H.10.5 Experiment 1: Limited Live video-based experimentation H.10.6 Experiment 2: ML development environment-based validation L.1 SECoT Validation Report Template L.2 SECoT Validation Report for AVOIDDS case study Stage n in AIC Systems Approach
[XX]: AS Hazardous Scenarios Definition	2	A comprehensive specification of all identified hazardous scenarios, including the interactions, environment states, and decisions	The systems approach artefact must document hazardous scenarios using the structure: <AS Operating Scenario><Relevant Environment State(s)> AND <Decision>. Should include examples of failure	[The research did not include the imposed structure] H.6.3 Predictive Thinking Pipeline 3: Hazards, Threats and Opportunities Scenarios (HazTOPs) Analysis. Figure H.16 Hazards Complexity Field Scope: graphically scoping the

		leading to unsafe outcomes.	modes, decision errors, and unsafe interactions.	hazards within the complexity field by placing hazard icons on target interaction. Figure H.17 Threats Complexity Field Scope Figure H.18 Opportunities Complexity Feels Scope H.6.3.2 Step 2) Characterise the scoped interactions. Figure H.19 Hazards associated with Eagle Drone preventing derailed train complexity field Table H.24 The table describes the AIC interaction dynamics between Eagle Drones and adversarial drones H.6.3.3 Step 3) Apply predictive potential complications guide words. Table H.25 HazTOPs Analysis of "3 Drones Attack" Scenario with Risk and Surprise Assessment Table H.26 HazTOPs Analysis of adversarial drone using smart lasers scenario Table H.27 HazTOPs Analysis of adversarial drone hiding behind fence scenario Figure H.20 Soft Hazard Complexity Field Model

				I.6.3 Predictive Thinking Pipeline 3: Hazards, Threats and Opportunities Scenarios (HazTOPs) Analysis. Figure I.11Sources of Hazards AIC Complexity Field I.6.3.2 Step 2) Characterise the scoped interactions. Table I.22 Considering Hazards related to I1 interaction I.6.3.3 Step 3) Apply predictive potential complications guide words. Table I.23 Example “More” guide word complication
[Y] : AS Design Justification	5	A structured report providing justification for each design decision made at tier n, ensuring alignment with safety requirements.	The systems approach artefact must explain how design choices ensure safety, robustness, fault tolerance, and runtime monitoring while addressing potential risks.	All Predictive Thinking Pipelines and design steps’ outputs
[YY]: AS Hazardous Scenarios Validation Report	2	A validation document confirming the completeness and correctness of the identified hazardous scenarios.	The systems approach artefact must provide evidence of review, expert validation, simulation-based verification, or real-world testing. It should justify that no significant hazardous	Validation is done by documenting expert reviews of architect assertions and predictions and the appropriate application of SECoT. A validation report template has been generated, which can be found in

			scenario has been overlooked.	L.1 SECoT Validation Report Template Examples: L.2 SECoT Validation Report for AVOIDDS case study L.3 SECoT Validation Report for Unsafe Train Tracks case study
[Z] : AS Design Review	5	Independent reviewers formally assess the AS design to ensure compliance with safety requirements.	The systems approach artefact must verify that design choices do not introduce new hazards, confirm adherence to the design process, and evaluate robustness measures.	Validation is done by documenting expert reviews of architect assertions and predictions and the appropriate application of SECoT. A validation report template has been generated, which can be found in L.1 SECoT Validation Report Template L.2 SECoT Validation Report for AVOIDDS case study L.3 SECoT Validation Report for Unsafe Train Tracks case study

L.5 Eagle Robot AMLAS Safety Case Argumentation Patterns

L.5.1 Stage 1. ML Safety Assurance Scoping

This stage defines the argument for the ML component's boundaries and safety assurance objectives. The AIC Systems Approach Processes involved to satisfy the objectives:

- Stage 1: Uncertainty Problem Articulation and Operational Environment Modelling
- Stage 2: Architect Intent and Autonomous Solution Needs Definition.
- Stage 3A: HazTOPS and Ordered AIC-driven Autonomous System Requirements Development.
- Stage 4: Disordered AIC-Driven Black Swan Scenarios Prediction.

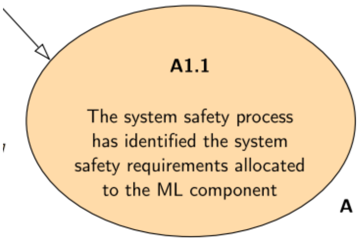
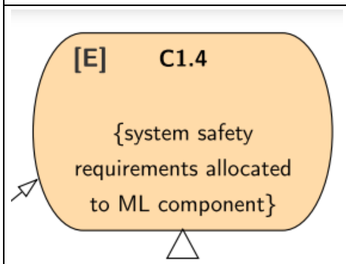
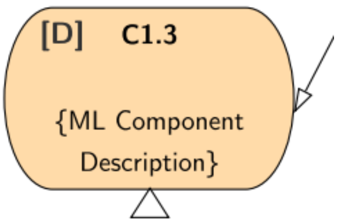
Assurance Argument Pattern	**AIC Systems Approach supportive methods and artefacts** The following process outputs satisfy the artefact's demonstration requirements:
	H.6 Stage 3A: HazTOPs and Ordered AIC-driven Autonomous System Requirements Development H.8 Stage 4: Disordered AIC-Driven Black Swan Scenario Predictions and Validation
	Table H.34 ML Safety Requirements Derivation Table H.37 Eagle Drone Safety Training Requirements for Black Swan Scenarios
	Table H.15 Architect High-Level Solution Prescription Table H.16 Architect High-Level Solution Prescription related to the impact of roaming adversarial drones

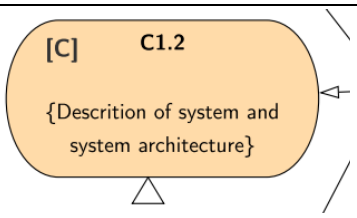
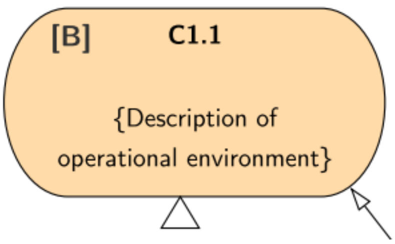
	Table H.17 Architect High-Level Solution Prescription related to the police incapability to capture adversarial drone
	Table H.15 Architect High-Level Solution Prescription Figure H.28 AIC hierarchical modelling schema for 4HnWs analysis Table H.23 The table outlines the interactions and behavioural influences of Eagle Drone Figure H.15 Corrected Eagle Drone complexity field Table H.33 Safety requirements derivations to mitigate the concealed drone problem. Table H.34 ML Safety Requirements Derivation Table H.37 Eagle Drone Safety Training Requirements for Black Swan Scenarios For model architecture: ML Model Type: Roboflow 3.0 Object Detection (Fast)
	H.4.1.5 Architect Prediction 1.5 Table H.12 Police Force Response to Adversarial Drones in Train Track Zones Table H.13 Predicted Factors Output Table H.14 Problem domain factors definitions

	Table H.35 Operational Design Definition for Eagle Robot Deployment in Train Track Zone

Table **Error! No text of specified style in document.**.8 AMLAS Stage 1 artefacts and AIC approach mapping

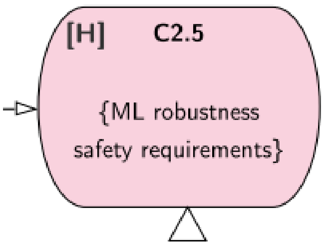
AMLAS Artefact	**Explanation**	**The substantiating AIC methods and artefacts**
[A]: System Safety Requirements	The safety requirements define the acceptable risk level and the system's necessary controls.	Stages 3A and 4 clearly capture and analyse the problem domain to derive safety requirements. For example, H.6 Stage 3A: HazTOPs and Ordered AIC-driven Autonomous System Requirements Development
[B]: Description of Operating Environment of System	A formal description of the operational conditions, environmental factors, and constraints in which the ML component will function.	Stage 1 clearly and comprehensively articulates the problem domain defining the operational environment. For example, H.4.1.5 Architect Prediction 1.5.
[C]: System Description	A structured description of the overall system architecture, including interactions between components and the role of ML.	Stage 2 of the AIC process defines the architect's and stakeholders' needs for what the systems should do in response to problematic situations in the problem domain. For example, Table H.15 Architect High-Level Solution Prescription.
[D]: ML Component Description	Defines the function, scope, and interactions of the ML component within the system.	

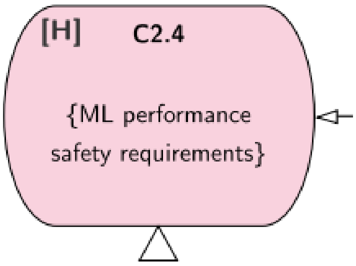
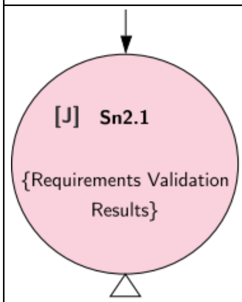
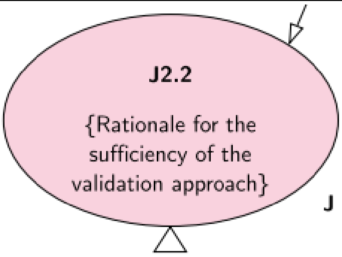
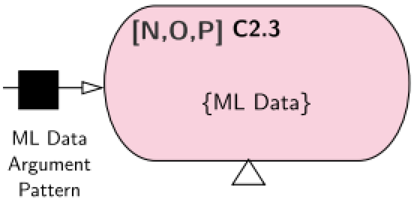
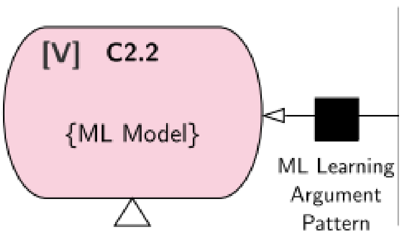
[E]: Safety Requirements Allocated to ML Component	The ML component must meet specific safety requirements to ensure system safety.	The process involved the derivation of ML safety training requirements from system-level safety concepts. For example, Table H.34 ML Safety Requirements Derivation.
[F]: ML Assurance Scoping Argument Pattern **[G]: ML Safety Assurance Scoping Argument**	A structured argument pattern that ensures the ML component is correctly scoped within the broader safety case.	We captured the argument in the following section: L.5.1 Stage 1. ML Safety Assurance Scoping

L.5.2 Stage 2. ML Safety Requirements Assurance

This stage develops the argument that demonstrates the safety requirements specific to the ML system are defined and validated. The AIC Systems Approach Processes involved to satisfy the objectives:

- SECoT Validation Report for Unsafe Train Tracks case study
- Stage 3A: HazTOPS and Ordered AIC-driven Autonomous System Requirements Development.
- Stage 4: Disordered AIC-Driven Black Swan Scenarios Prediction
- Stage 6: Black Swan-driven ML Development and Testing

Assurance Argument Pattern	**AIC Systems Approach: supportive methods and artefacts** The following process outputs satisfy the artefact's demonstration requirements:
	Table H.34 ML Safety Requirements Derivation Table H.37 Eagle Drone Safety Training Requirements for Black Swan Scenarios 7.2.2 Step 1: Design CuneiForms

	Table H.40 Characteristic Training Classes definitions for a CuneiForm abstract image 7.8 Chosen training and testing strategies with required performance The table includes the type of robustness (CuneiForms) used for validation and testing. It also includes the minimum performance requirement.
	L.3 SECoT Validation Report for Unsafe Train Tracks case study 7.2.3 Step 2: SOC Compliance 7.2.5 Step 4: CuneiForm Compliance Validation
	L.3 SECoT Validation Report for Unsafe Train Tracks case study Table H.37 Eagle Drone Safety-Training Requirements for Black Swan Scenarios Table H.34A ML Safety Requirements Derivation
	[N] Development Data, [O] Internal Test Data, [P] Verification Data : Dataset Config7 (augmented)
	**ML Config7: All in dataset** See **ML Config7 (augmented)**

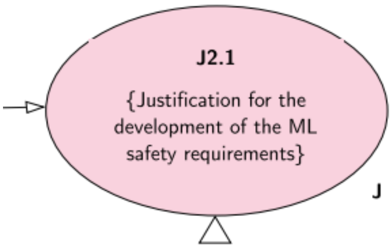
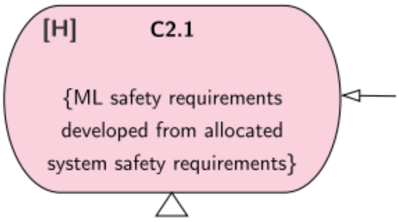
	H.6 Stage 3A: HazTOPs and Ordered AIC-driven Autonomous System Requirements Development H.8 Stage 4: Disordered AIC-Driven Black Swan Scenarios Predictions and Validation
	Table H.34 ML Safety Requirements Derivation Table H.37 Eagle Drone Safety Training Requirements for Black Swan Scenarios Table H.38 ML component training dataset requirements

Table **Error! No text of specified style in document.**.9 AMLAS Stage 2 artefacts and AIC approach mapping

AMLAS Artefact	**Explanation**	**The substantiating AIC methods and artefacts**
[H]: ML Safety Requirements	The ML model must satisfy explicit safety requirements, including performance and robustness constraints. Must demonstrate clear, measurable ML safety constraints (e.g., accuracy, robustness to environmental variations) and how they mitigate system hazards.	The process involved deriving ML safety training requirements from system-level safety concepts. For example, see Table H.34 ML Safety Requirements Derivation.
[I]: ML Safety Requirements Argument Pattern **[K]: ML Safety Requirements Argument**	Must justify the translation of system-level safety requirements into ML-specific requirements, ensuring relevance, completeness, and correctness.	We captured the pattern in section: L.5.2 Stage 2. ML Safety Requirements Assurance

[J]: ML Safety Requirements Validation Results	The documented results of validation activities performed on ML safety requirements. Must demonstrate that ML safety requirements correctly reflect system safety needs and can be realistically implemented and verified.	In stage 6, for the unsafe training tracks problem domain, we performed the following validation activities: 7.2.3 Step 2: SOC Compliance To validate that CuneiForm correctly captures safety-training requirements. See the following sections in this chapter, which capture the way we would validate CuneiForms tractability to safety requirements: • 7.1.3, 7.1.5 7.2.5 Step 4: CuneiForm Compliance Validation In this step, we performed a validation process to determine whether an image is compliant with a CuneiForm, which is compliant with ML safety training requirements and, in turn, with SOC requirements. Appendix D captures the CuneiForm validation report for AVOID dataset. In addition to direct compliance of CuneiForms, we also provide an audit trail demonstrating how the training syllabus is achieving

		compliance towards safety requirements preservation during unforeseen (black swan) scenarios. See Table 7.12 Black Swan-driven incremental ML development process to assure model performance under Black Swan operations. We also developed a SECoT validation report template: L.1 SECoT Validation Report Template Which we instantiated to: L.3 SECoT Validation Report for Unsafe Train Tracks case study

L.5.3 Stage 3. Data Management

This stage develops the argument that data used in training and validation meets the necessary quality and safety standards. The AIC Systems Approach Processes involved to satisfy the objectives:

- Stage 5: CuneiForm-based Syllabus for Safety-Driven ML Epistemic Intelligence Development
- CuneiForm Training syllabus as a Validation Process for Datasets
- Stage 6: Black Swan-driven ML Development and Testing

The following dataset validation artefacts can be used as evidence to demonstrate how the dataset captures a CuneiForm, which in turn shows how a safety requirement is captured in a dataset (example taken from AVOIDDS case study):

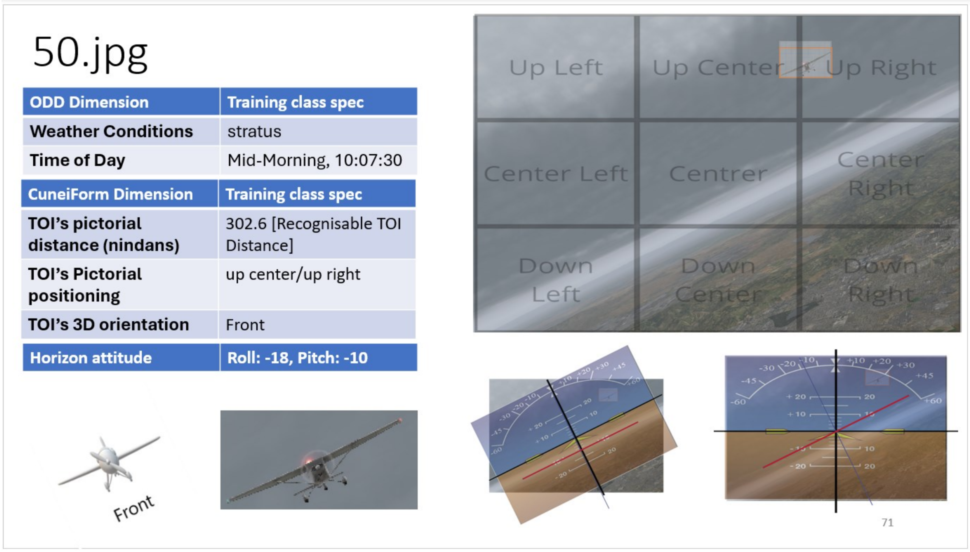


Figure L.4 Example validation report of how a sample training image validates CuneiForm 5

CuneiForm 5 has been retrospectively generated:

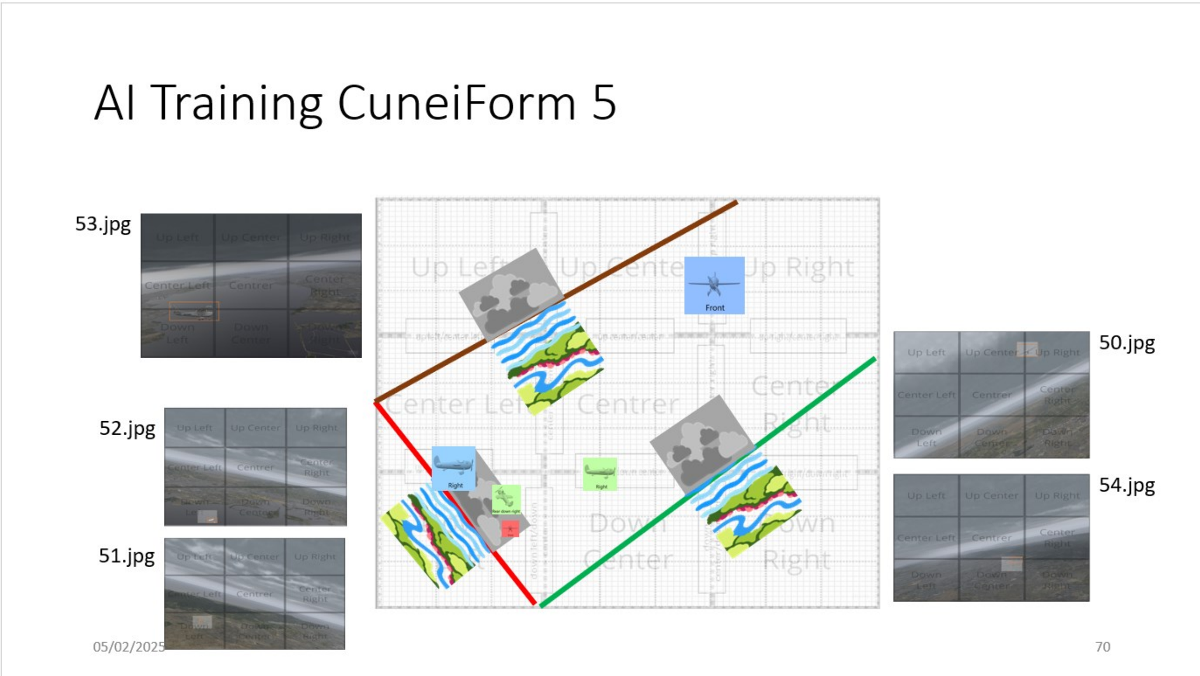


Figure L.5 AVOIDDS training CuneiForm 5 and examples training images, which instantiate different aspects of the abstract image. Appendix D is the actual validation report from the AVOID dataset.

Table L.10 captures primary artefacts that need to be presented:

Table **Error! No text of specified style in document.**.10 AMLAS Stage 3 artefacts and AIC approach mapping

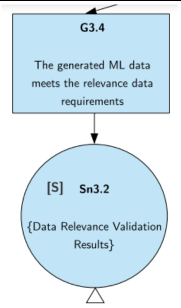
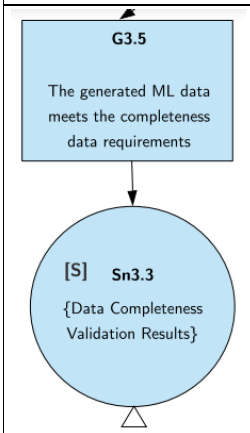
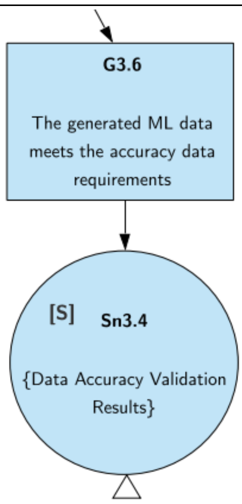
AMLAS Artefact	Explanation	The substantiating AIC methods and artefacts

[L]: Data Requirements	Defines the ML data's characteristics to ensure the model meets its safety requirements. It must include data relevance, completeness, accuracy, and balance requirements, ensuring sufficient operational domain representation.	The AIC systems approach included a process for systematically identifying data requirements derived from the training concept. For example, Table H.38 ML component training dataset requirements Table 6.35 Black Swan Scenarios Batch A and B CuneiForms Table 6.36 Typical operations CuneiForms Figure 6.36 H.54 Out-of-context CuneiForm of drones Figure 6.27 represents the Cuneiform (H.36) characterisation for the Black Swan Scenario Validation. Figure 7.20 Example CuneiForm and instantiated image for AVOIDDS case study
[M]: Data Requirements Justification Report	Justifies that the data requirements are sufficient to develop a safe ML model. It must demonstrate how the data requirements were derived, validated, and justified to capture the necessary variations in the operational environment.	The following describes the systematic process that demonstrates how the data requirements were derived: I, H.9 Stage 5: CuneiForm-based Safety-Driven ML Training, Testing Process

		I.10, Section 7.2 Stage 6: Dataset generation and validation process
[N]: Development Data	Data used for training and validating the ML model during development.	We developed a dataset using the requirements derived from stage 4 and 5. We also used a pre-existing dataset and injected the specially derived dataset and then validated its performance against a Black Swan scenario. For more details, see section L.5 and the following link to the output model. [N] Development Data, [O] Internal Test Data, [P] Verification Data : Dataset Config7 (augmented)
[O]: Internal Test Data	Data used for testing the ML model internally before verification. This is equivalent to what we define as a validation dataset.	
[P]: Verification Data	A separate dataset, equivalent to what we define as a test dataset, is used for final model evaluation to assess performance under unseen conditions.	
[Q]: Data Generation Log	Documents the process of data collection, pre-processing, and augmentation. It must capture decisions made during data collection, processing, and augmentation, providing a rationale for data sufficiency.	The CuneiForm method in stage 5 defines the rationale of how data requirements are translated into the dataset. Also, 7.2.2 Step 1: Design CuneiForms, and 7.2.4 Step 3: Instantiates CuneiForms
[S]: ML Data Validation Results	Documents the validation outcomes, ensuring the generated data meets ML data requirements. It must demonstrate that data relevance, completeness, balance, and accuracy were	In the unsafe train tracks case study, we did not consider the validation report over the produced dataset. However, in the AVOIDDS case study, we did. Therefore, we will include

	verified and any discrepancies justified.	the following from the second case study: I.10.1 CuneiForm Training syllabus as a Validation Process for Datasets I.10.3 CuneiForm Validation Artifact Safety Validation Report for AVOID Dataset using the CuneiForm method [can be retrieved from here] As for unsafe train tracks, an example CuneiForm validation can be found in section 7.2. Specifically see section 7.2.5 Step 4: CuneiForm Compliance Validation
[R]: ML Data Argument Pattern **[T]: ML Data Argument**	A structured assurance argument justifying the adequacy of the ML data for safety assurance.	We captured the artefacts in the following sections: L.5.3 Stage 3. Data Management

Assurance Argument Pattern	**AIC Systems Approach supportive methods and artefacts** The following process outputs satisfy the artefact's demonstration requirements:

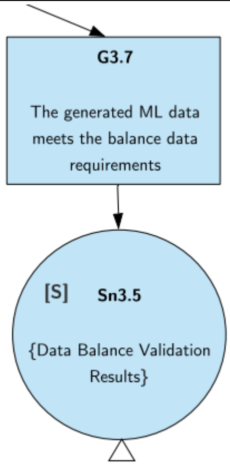
	H.9 Stage 5: CuneiForm-based Syllabus for Safety-Driven ML Epistemic Intelligence Development An example CuneiForm validation can be found in section 7.2. Specifically see section 7.2.5 Step 4: CuneiForm Compliance Validation
	H.9 Stage 5: CuneiForm-based Syllabus for Safety-Driven ML Epistemic Intelligence Development An example CuneiForm validation can be found in section 7.2. Specifically see section 7.2.5 Step 4: CuneiForm Compliance Validation
 Data Accuracy[3]	7.2.3 Step 2: SOC Compliance Validate that CuneiForm correctly captures safety-training requirements. See the following sections in this chapter, which capture the way we would validate CuneiForms tractability to safety requirements: sections 6.7.3, 6.7.5. Also, 7.2.5 Step 4: CuneiForm Compliance Validation where we validate that data accurately captures the CuneiForms.

[3] Data accuracy refers to how reliably a dataset reflects real-world conditions, ensuring labels and metadata truthfully represent the objects or scenarios they describe. For drone detection systems, this means training data (images, sensor readings, labels) must precisely capture drones in diverse, realistic settings to avoid biases or errors that could compromise the model's performance.

<table>
<tr><td></td><td>In the AVOIDDS case study, we did. Therefore, we will include the following from the second case study:

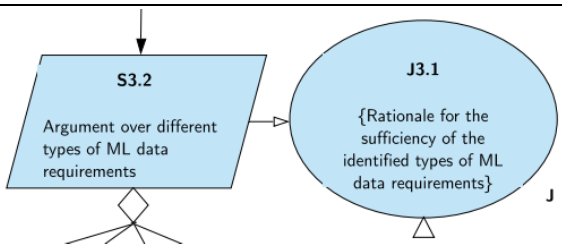
I.10.1 CuneiForm Training Strategy as a Validation Process for Datasets

I.10.3 CuneiForm Validation Artifact

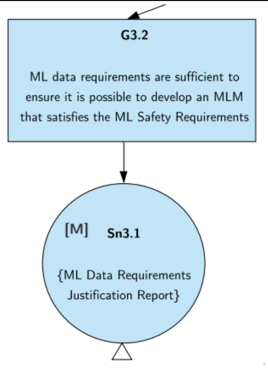
Safety Validation Report for AVOID Dataset using the CuneiForm method [can be retrieved from here]</td></tr>
<tr><td>
</td><td>In the unsafe train tracks case study, we did not consider the validation report over the produced dataset. However, in the AVOIDDS case study, we did. Therefore, we will include the following from the second case study:

I.10.1 CuneiForm Training Strategy as a Validation Process for Datasets

I.10.3 CuneiForm Validation Artifact

Safety Validation Report for AVOID Dataset using the CuneiForm method [can be retrieved from here]</td></tr>
<tr><td>
</td><td>H.9 Stage 5: CuneiForm-based Syllabus for Safety-Driven ML Epistemic Intelligence Development

H.10 Stage 6: Black Swan-driven ML Development and Testing</td></tr>
</table>

<table>
<tr>
<td>
</td>
<td>H.9 Stage 5: CuneiForm-based Syllabus for Safety-Driven ML Epistemic Intelligence Development

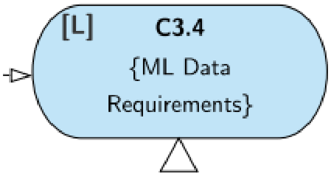
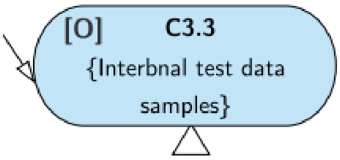
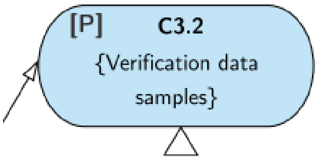
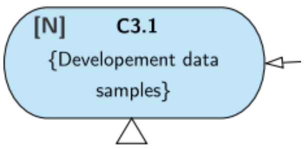
H.10 Stage 6: Black Swan-driven ML Development and Testing</td>
</tr>
<tr>
<td>
</td>
<td>Table H.38 ML component training dataset requirements

Table 6.35 Black Swan Scenarios Batch A and B CuneiForms

Table 6.36 Typical operations CuneiForms

Figure 6.36 H.54 Out-of-context CuneiForm of drones</td>
</tr>
<tr>
<td>
</td>
<td rowspan="3">[N] Development Data, [O] Internal Test Data, [P] Verification Data : <u>Dataset Config7 (augmented)</u></td>
</tr>
<tr>
<td>
</td>
</tr>
<tr>
<td>
</td>
</tr>
</table>

L.5.4 Stage 4. Model Learning

This stage develops the argument for the creation and evaluation of the ML model correctly and comprehensively to ensure safety objectives are achieved. The AIC Systems Approach Processes involved to satisfy the objectives:

- Stage 5: CuneiForm-based Syllabus for Safety-Driven ML Epistemic Intelligence Development
- Stage 6: Black Swan-driven ML Development and Testing

The model training process we produced is characterised below[4]:

Table **Error! No text of specified style in document.**.11 Final training syllabus and trained output model

<table>
<tr><td rowspan="6">ML Config7: All in dataset

See ML Config7 (augmented)

Total: 23533 images (without augmentation)

With augmentation:

40321 images</td><td>Total Training: 16788 (71%) (no augm.)</td><td>Total Valid: 4426 (19%) (no augm.)</td><td>Total Test: 2319 (10%) (no augm.)</td></tr>
<tr><td>Out-of-context images: 10954 (65%)</td><td>Out-of-context images: 1969 (44%)</td><td>Out-of-context images: 268 (11%)</td></tr>
<tr><td>Black Swans A CuneiForm Scenarios: H.36, 37, 38, 39, 40, 41: 1866 (11%)</td><td>Black Swans A CuneiForm Scenarios: H.36, 37, 38, 39, 40, 41: 622 (14%)</td><td>Black Swans A CuneiForm Scenarios: H.36, 37, 38, 39, 40, 41: 622 (26%)</td></tr>
<tr><td>Black Swans B CuneiForm Scenarios: H.51,52,53: 695 (4%)</td><td>Black Swans B CuneiForm Scenarios: H.51,52,53: 231 (5%)</td><td>Black Swans B CuneiForm Scenarios: H.51,52,53: 232 (10%)</td></tr>
<tr><td>Typical Operations CuneiForm Scenarios: H.42, 43,44: 3273 (19%)</td><td>Typical Operations CuneiForm Scenarios: H.45,46,47: 1604 (36%)</td><td>Typical Operations CuneiForm Scenarios: 48,49,50: 1197 (51%)</td></tr>
<tr><td colspan="3">Applied Pre-processing: Grayscale: Applied

Applied Pre-processing:</td></tr>
</table>

[4] See section H.9.5 for more details on CuneiForms.

	Outputs per training example: 2 Noise: Up to 2.39% of pixels **Performance:** mAP50/test 99%, mAP50/validation 98%,

Table L.12 captures the primary artefacts that need to be presented:

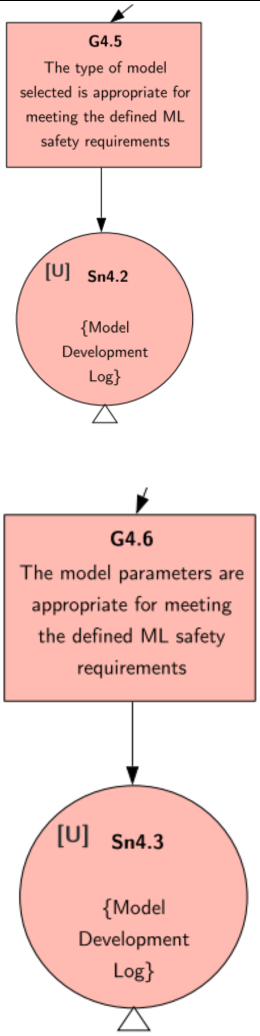
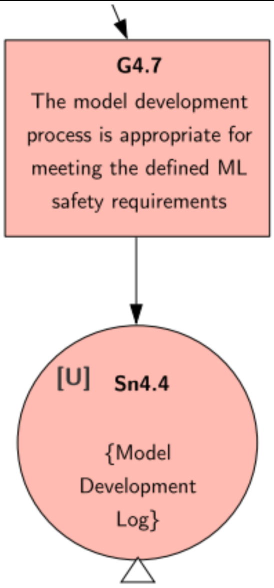
Table L.12 AMLAS Stage 4 artefacts and AIC approach mapping

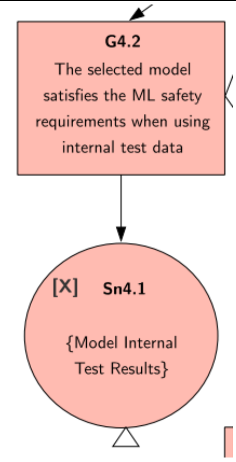
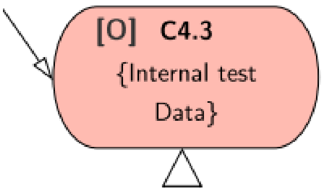
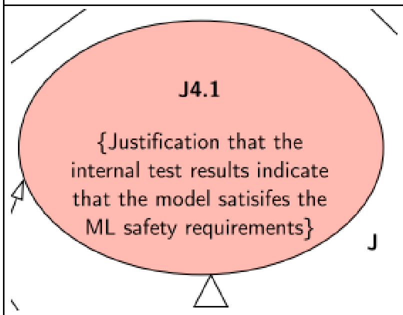
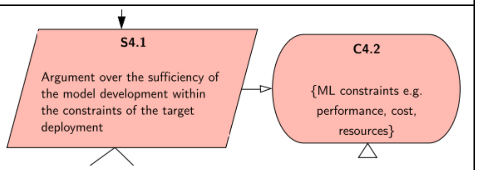
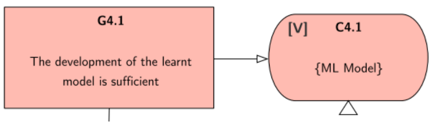
AMLAS Artefact	Explanation	The substantiating AIC methods and artefacts
[U]: Model Development Log	A record of all decisions, configurations, and justifications during model training and development. It must capture all design choices, hyperparameter settings, and rationale for selecting the final model—document methods to prevent overfitting and ensure robustness.	For ML development audit trail see Table 7.12 Black Swan-driven incremental ML development process to assure model performance under Black Swan operations, Section 7.2.8. We used the RoboFlow platform to capture the model development log activities. The following model is the output of experiment 9.6 where we injected the black swan training subset and validated it against the Black Swan Validation subset: **ML Config7 (augmented)** Trained based on: Experiment 9.6 full dataset including black swans, typical operations and random out-of-context: [N] Development Data, [O] Internal Test Data, [P] Verification

		Data : [Dataset Config7 (augmented)] Furthermore, the following stages also capture activities that were directly involved development of the model: H.9 Stage 5: CuneiForm-based Safety-Driven ML Training, Testing Process H.10.6 Captures a comprehensive experimentation to justify the training syllabus and model.
[V]: ML Model	The final ML model was developed based on the training process. It must demonstrate that the trained model satisfies all ML safety requirements, including performance, robustness, and failure handling.	We used the RoboFlow platform to capture the model development log activities. The following model is the output of experiment 9.6 where we injected the black swan training subset and validated it against the Black Swan Validation subset: **ML Config7 (augmented)**
[X]: Internal Test Results	Documented results of evaluating the ML model on internal test data.	The test dataset can be found here: [N] Development Data, [O] Internal Test Data, [P] Verification Data : Dataset Config7 (augmented) The validation and test results can be found in **ML Config7 (augmented)** Also, Black Swan (only) testing can be found in the following ML

		configuration 6 which demonstrates the model's ability to handle out-of-distribution, data shift.
[Y]: ML Learning Argument **[W]: ML Learning Argument Pattern**	The instantiated ML learning assurance argument is based on development and testing evidence.	The instantiated argument is captured in the following section: L.5.4 Stage 4. Model Learning

Assurance Argument Pattern	**AIC Systems Approach supportive methods and artefacts** The following process outputs satisfy the artefact's demonstration requirements:

<table>
<tr>
<td>
</td>
<td>For ML development audit trail see Table 7.12 Black Swan-driven incremental ML development process to assure model performance under Black Swan operations, Section 7.2.8.</td>
</tr>
<tr>
<td>
</td>
<td></td>
</tr>
</table>

	The validation and test results can be found in **ML Config7 (augmented)** Also, Black Swan (only) testing can be found in the following ML configuration 6 which demonstrates the model's ability to handle out-of-distribution, data shift.
	The validation and test dataset can be found here: Dataset Config7 (augmented)
	H.9 Stage 5: CuneiForm-based Syllabus for Safety-Driven ML Epistemic Intelligence Development 7.8 Chosen training and testing strategies with required performance 7.2.5 Step 4: CuneiForm Compliance Validation 7.2 Stage 6: Black Swan-driven ML Development and Testing
	
	**ML Config7 (augmented)**

L.5.5 Stage 5. Model Verification

Outside the PhD scope

L.5.6 Stage 6. Model Deployment

Outside the PhD scope