Safety Assurance Objectives for Autonomous Systems Supporting Material

SCSC-190

Safety of Autonomous Systems Working Group (SASWG)

SCSC Publication Number: SCSC-190

The Safety Critical Systems Club (SCSC) is the professional network for sharing knowledge about safety-critical systems. It brings together: engineers and specialists from a range of disciplines working on safety-critical systems in a wide variety of industries; academics researching the arena of safety-critical systems; providers of the tools and services that are needed to develop the systems; and the regulators who oversee safety. Through publications, seminars, workshops, tutorials, a web site and, most importantly, at the annual Safety-critical Systems Symposium (SSS), it provides opportunities for these people to network and benefit from each other's experience in working hard at the accidents that don't happen. It focuses on current and emerging practices in safety engineering, software engineering and product and process safety standards.

This document was written by the Safety of Autonomous Systems Working Group (SASWG), which is convened under the auspices of the SCSC. The goal of the SASWG is to produce clear guidance on how autonomous systems and autonomy technologies should be managed in a safety related context, throughout the lifecycle, in a way that is tightly focused on challenges unique to autonomy.

The material in this document aligns with SCSC-153B which was formally released February 2022. It will not be updated with subsequent issues of SCSC-153X.

Safety Assurance Objectives for Autonomous Systems SCSC-190 - Supporting Material

The Safety of Autonomous Systems Working Group [SASWG]

January 2024

Change History

Version	By	Status	Date
SCSC-190 Supporting Material	The SASWG Team	Initial issue containing retired information from SCSC-153B.	January 2024

Changes Since the Last Version

No changes to report.

Contents

This page is intentionally blank

1 Introduction

The material in this document aligns with SCSC-153B which was released February 2022 [13]. During the development of "Safety Assurance Objectives for Autonomous Systems" [13] a number of comparisons were performed other draft guidance documents produced for safety assurance of Autonomous Systems (AS) and Machine Learning (ML). These comparisons were performed to ensure that either the assurance objectives in SCSC-153B matched elements of the alternative guidelines, or, where there were omissions, these were justified when considering the purpose and scope of SCSC-153B. Each comparison differs, depending on the content, structure, and scope of the alternative document. The individual approaches are described in more detail for each appendix.

The material has been removed from subsequent editions of SCSC-153B as confidence in the content of that document has been established. However, informal activities within the SASWG will still continue to scan for relevant AS and ML assurance issues, and review new standards and guidelines to ensure validity and currency of the SCSC-153X objectives.

The material presented here will not be maintained for any uplifts of the comparison documents or approaches.

This page is intentionally blank

Appendix A Computation-Level Framework: Justification

This appendix summarises the process used to develop the computation-level framework adopted by the Safety of Autonomous Systems Working Group (SASWG). In doing so, it provides some justification for the choice of framework. It also provides some confidence that the framework covers all relevant topic areas.

Initially, a small-scale survey of existing computation-level frameworks was conducted. This identified the items listed in Table 1.

Table 1: Computation-Level Frameworks Considered

Section	Framework
A.1.1	Modified Software Safety Assurance Principles
A.1.2	The "Faria Stack"
A.1.3	Douthwaite and Kelly's "Viewpoints"
A.1.4	Google's Machine Learning Rubric
A.1.5	Ethical and Safety Principles
A.1.6	Burton's "Making the Case" Argument

Each computation-level framework is briefly summarised (subsection A.1) and a preferred framework is selected. A top-level mapping between frameworks is completed, to confirm that the chosen framework incorporates all relevant parts of the other computation-level frameworks (subsection A.2). Similar, top-level mappings from the chosen framework to, firstly, a typical software development approach and, secondly, a generic approach to ML-based development are conducted; these demonstrate the framework provides appropriate coverage of typical development activities (subsection A.3).

A.1 Computation-Level Frameworks

A.1.1 Modified Software Safety Assurance Principles

This computation-level framework is described in a paper presented at the 2017 Safety-critical Systems Symposium (SSS) [1]. The paper considers the "four plus one" software safety assurance principles [9] from the perspective of non-traditional (e.g. ML / Artificial Intelligence (AI)) software. A slightly revised and extended set of six (or "four plus two") principles are proposed:

- Principle One: Software safety requirements shall be defined to address the software contribution to system hazards;
- Principle Two-Primed: The software detailed design shall embody the intent of the software safety requirements;
- Principle Three: Software safety requirements shall be satisfied;
- Principle Four: Hazardous behaviour of the software shall be identified and mitigated;
- Principle Four plus One: The confidence established in addressing the software safety principles shall be commensurate to the contribution of the software to system risk;
- Principle Four plus Two: Software required to produce behaviour not predictable at design time should consider the consequence of behavioural adaptations on its environment.

A.1.2 The "Faria Stack"

This computation-level framework is based on the information presented in a paper at the International Symposium on Software Reliability Engineering (ISSRE) Workshop on Software Certification (WoSoCer) [7]. This framework comprises five projections:

- Experience, which is focused on the data that is available to train a machine learning algorithm;
- Task, which is concerned with the performance of the implemented computation;
- Algorithm, which considers the type of algorithm (e.g. neural network, random forest, etc.);
- Software, which includes considerations such as the language in which the computation is implemented;
- Hardware, which relates to the computational hardware that is used.

When using this framework it may be helpful to consider, at least, the Software and Hardware projections from two perspectives, specifically training and operational use. For example, it is likely that the computational hardware used for training will be different to that used during an operational deployment.

A.1.3 Douthwaite and Kelly's "Viewpoints"

This computation-level framework was presented at the 2018 SSS [6]. Building on the concept of distinct viewpoints used in systems engineering, this paper identifies six viewpoints. Although they were developed from the perspective of Bayesian Networks, the paper suggests the viewpoints are applicable to many types of artificial intelligence software. The viewpoints are:

- Model, which relates to the structure and parametrisation of the model underlying the learnt algorithm;
- Data, which covers all data acquisition, processing and storage concerns (including knowledge engineering and expert elicitation);
- Computational, which includes the properties of all algorithms used for learning and reasoning tasks within the system, their selection process, and the associated assumptions and design decisions;
- Operational, which focuses on the evolution and maintenance of the system after deployment;
- Technology, which covers the necessity, properties, constraints and assumptions of modelling frameworks used in the system;
- Implementation, which addresses all "conventional" software and hardware engineering concerns, including "normal" function allocation, requirements and associated verification and validation activities.

As with the "Faria Stack" considered above, there may be advantages in considering some of the above viewpoints from both training and operational perspectives.

A.1.4 Google's Machine Learning Rubric

This computation-level framework [4] includes a scoring mechanism that is intended to measure how suitable a machine learning approach is for deployment. It is based on computations used in a web-like environment, but may be of relevance to wider autonomous systems.

The framework includes four categories, each of which includes several considerations:

- Tests for Features and Data:
  - Test that the distributions of each feature match your expectations;
  - Test the relationship between each feature and the target, and the pairwise correlations between individual signals;
  - Test the cost of each feature;
  - Test that a model does not contain any features that have been manually determined as unsuitable for use;
  - Test that your system maintains privacy controls across its entire data pipeline;
  - Test the calendar time needed to develop and add a new feature to the production model;
  - Test all code that creates input features, both in training and serving.
- Tests for Model Development:
  - Test that every model specification undergoes a code review and is checked in to a repository;
  - Test the relationship between offline proxy metrics and the actual impact metrics;
  - Test the impact of each tunable hyper-parameter;
  - Test the effect of model staleness;
  - Test against a simpler model as a baseline;
  - Test model quality on important data slices;
  - Test the model for implicit bias.
- Tests for ML Infrastructure:
  - Test the reproducibility of training;
  - Unit test model specification code;
  - Integration test the full ML pipeline;
  - Test model quality before attempting to serve it;
  - Test that a single example or training batch can be sent to the model, and changes to internal state can be observed from training through to prediction;
  - Test models via a canary process before they enter production serving environments;
  - Test how quickly and safely a model can be rolled back to a previous serving version.
- Monitoring Tests for ML:
  - Test for upstream instability in features, both in training and serving;
  - Test that data invariants hold in training and serving inputs;
  - Test that your training and serving features compute the same values;
  - Test for model staleness;
  - Test for Not a Number (NaN) or infinities appearing in your model during training or serving;
  - Test for dramatic or slow-leak regressions in training speed, serving latency, throughput, or Random Access Memory (RAM) usage;
  - Test for regressions in prediction quality on served data.

For each item above, one point is awarded for manual tests (including documenting and distributing the results). A second point is awarded if tests are run automatically and repeatedly. A score is calculated for each of the four categories by adding the scores for each of the listed items. The overall score is then the minimum of these four category scores.

A.1.5 Ethical and Safety Principles

This computation-level framework identifies a perspective on the ethics governing decisions around safety-critical autonomous systems [12]. It aligns with the Modified Software Safety Assurance Principles (discussed above) and is applicable to ethics only so far as these affect safety.

- Principle One: Ethics requirements governing the autonomous system behaviour shall be defined.
- Principle Two: The intent of the ethics requirements shall be maintained throughout decomposition.
- Principle Three: Ethics requirements shall be satisfied.
- Principle Four: The autonomous system shall continue to be safe, and emergent behaviour of the autonomous system which conflicts with the ethics requirements shall be identified and mitigated
- Principle Four plus One: The degree of rigour required to address these ethical principles shall be commensurate with the contribution of the autonomous system to system risk.

A.1.6 Burton's "Making the Case" Argument

This computation-level framework comes from a paper presented at the 2017 International Conference on Computer Safety, Reliability, and Security [5]. The paper outlines an assurance case structure for a highly automated driving system, which could possibly be extended to cover a wide range of autonomous systems. A Goal Structuring Notation (GSN) approach is used; key features include:

- GOAL G1: The residual risk associated with functional insufficiences in the object detection function is acceptable;
- CONTEXT C1: Definition of functional and performance requirements on the object detection function;
- ASSUMPTION A1: Assumptions on the operational profile of the system's environment;
- ASSUMPTION A2: Assumptions on attributes of inputs to the machine learning function;
- ASSUMPTION A3: Assumptions on the performance potential of machine learning;
- STRATEGY S1: Argument over causes of functional insufficiencies in machine learning;
- SUBGOAL G2: The operating context is well defined and reflected in training data;
- SUBGOAL G3: The function is robust against distributional shift in the environment;
- SUBGOAL G4: The function exhibits a uniform behaviour over critical classes of situations;
- SUBGOAL G5: The function is robust against differences between its training and execution platforms;
- SUBGOAL G6: The function is robust against changes in its system context.

A.2 Framework Mappings

Following discussions[1], the SASWG selected the "Faria Stack" as the basis for the computation-level framework. The following paragraphs briefly discuss each projection of the "Faria Stack", taking into account the other frameworks outlined in the preceding subsection. Within these discussions:

[1] SASWG 7, 17 April 2018, York.

- For reasons of brevity, only the top-level of Google's Machine Learning Rubric is considered.
- Due to their similarity to the Modified Software Safety Assurance Principles, the Ethical and Safety principles are not explicitly considered.
- For simplicity, only the goals and subgoals are considered from Burton's "Making the Case" Argument.

The discussions also include a "Not Addressed" pseudo-projection, which captures considerations that do not readily relate to any of the projections. By checking the contents of this pseudo-projection, and confirming that it contains nothing significant, confidence can be gained that the adopted framework covers all relevant topics.

A.2.1 Experience

Consideration of the data used to develop the algorithm directly relates to Douthwaite and Kelly's Data viewpoint, and also to the Tests for Features and Data category from Google's Machine Learning Rubric.

The way the data reflects the operating context directly relates to Subgoal G2 from Burton's "Making the Case" Argument.

A.2.2 Task

Understanding the task should also include understanding the way it contributes to the wider system and, also, any associated computation (or software) safety requirements. This consideration relates to Principle One of the Modified Software Safety Assurance Principles.

Performance measurement against the intended task ought to include explicit measures against requirements (including safety requirements). It also ought to consider whether the computation has introduced any new hazards. These considerations relate to Principles Three and Four of the Modified Software Safety Assurance Principles. They also relate to Goal G1 from Burton's "Making the Case" Argument.

More generally, performance management relates to the Tests for Model Development category from Google's Machine Learning Rubric.

The properties of the operationally-fielded computation relate to Douthwaite and Kelly's Computational viewpoint.

A.2.3 Algorithm

The link between choice of algorithm and intended task mirrors the link between requirements (including safety requirements) and detailed design. This relates to Principle Two-Primed of the Modified Software Safety Assurance Principles.

Part of choosing a specific algorithm also includes choosing hyper-parameters (e.g. number of nodes and layers in a neural network). This relates to Douthwaite and Kelly's Model viewpoint. More general algorithm-related choices relate to Douthwaite and Kelly's Computational viewpoint.

A.2.4 Software

The choice of software (for both development and operational use) is part of detailed design. This relates to Principle Two-Primed of the Modified Software Safety Assurance Principles. It also relates to Douthwaite and

Kelly's Technology and Implementation viewpoints, and also to the Tests for ML Infrastructure category from Google's Machine Learning Rubric.

A.2.5 Hardware

The choice of hardware (for both development and operational use) is part of detailed design. This relates to Principle Two-Primed of the Modified Software Safety Assurance Principles, to Douthwaite and Kelly's Implementation viewpoint, and also to the Tests for ML Infrastructure category from Google's Machine Learning Rubric.

The possibility of different behaviour on development (training) and operational (execution) platforms relates to Subgoal G5 from Burton's "Making the Case" Argument.

A.2.6 Not Addressed

The chosen computation-level framework does not readily address Principle Four plus One of the Modified Software Safety Assurance Principles: "The confidence established in addressing the software safety principles shall be commensurate to the contribution of the software to system risk". This is not a significant concern as this principle is a cross-cutting issue for all assurance, and thus not something that has to be specifically addressed at the computation level.

Likewise, the framework does not readily address Principle Four plus Two: "Software required to produce behaviour not predictable at design time should consider the consequence of behavioural adaptations on its environment.". This is not a significant concern as adaptation is considered at the autonomy architecture-level ([13]).

From the perspective of Douthwaite and Kelly's "Viewpoints" the chosen computation-level framework does not readily address the Operational viewpoint. This is more readily addressed at the autonomy architecture-level and the platform-level ([13]).

Similarly, the Monitoring Tests for ML category from Google's Machine Learning Rubric are addressed at other framework levels, as is Subgoal G3 from Burton's "Making the Case" Argument.

Finally, the chosen computation-level framework does not readily address Subgoal G4 of Burton's "Making the Case" Argument: "The function exhibits a uniform behaviour over critical classes of situations". It is not immediately clear whether this, especially the "uniform behaviour" part, is a generic requirement that should be satisfied by every computation. If it is a requirement for a particular application then it should be addressed by the Task projection (via the relationship to Principle One of the Modified Software Safety Assurance Principles).

A.2.7 Relationship Summary

For ease of reference, the relationships outlined above are summarised in Table 2. Note that this presentation is deliberately simple and top-level.

Table 2: Relationships between Computation-Level Frameworks Considered

Stack Level	Modified Software Safety Assurance Principles	Douthwaite and Kelly's "Viewpoints"	Google's Machine Learning Rubric	Burton's "Making the Case" Argument
Experience	-	Data	Tests for Features and Data	Subgoal G2
Task	Principles One, Three and Four	Computational	Tests for Model Development	Goal G1
Algorithm	Principle Two-Primed	Model and Computational	-	-
Software	Principle Two-Primed	Technology and Implementation	Tests for ML Infrastructure	-
Hardware	Principle Two-Primed	Implementation	Tests for ML Infrastructure	Subgoal G5
Not Addressed	Principle Four plus One, Principle Four plus Two	Operational	Monitoring Tests for ML	Subgoal G4

Overall, the preceding analysis indicates that, based on the selected comparator frameworks, there are no significant omissions from the chosen computation-level framework.

A.3 Software and ML Development Mappings

Table 3 maps the framework's projections to the activities involved in a generic software development [17].

This mapping shows that the chosen computation-level framework is sufficiently complete to address typical software development activities.

Table 3: Mapping Projections to Typical Software Development

	Experience	Task	Algorithm	Software	Hardware
Plan		Y			
Requirements		Y			
Design	Y	Y	Y		
Implement	Y	Y	Y	Y	Y
Test		Y			
Transition				Y	Y

To provide further confidence, Table 4 maps the projects to the steps that are required to produce a useful ML-based computation [16]. This mapping demonstrates the framework fits well with development in an ML context, with most development steps mapping to a single projection.

Table 4: Mapping Projections to Typical ML Development

	Experience	Task	Algorithm	Software	Hardware
Frame the question		Y			
Collect data	Y				
Select features	Y				
Choose algorithm			Y		
Choose metrics		Y			
Conduct experiment				Y	Y
Interpret results		Y			

Appendix B Computation-Level Objectives: Justification

This appendix provides some additional justification for the computation-level objectives listed in the main body. This is achieved by mapping those objectives to separately published material, specifically:

- A suggested list of requirements for a standard to support the use of Neural Networks (NNs) in safety-critical applications [2]. This source dates from 1996. Consequently, it provides a sound theoretical basis, independent from recent trends, against which computation-level objectives can be compared. However, its considerations do not encompass the latest research directions. In addition, whilst many of its requirements are applicable to a number of ML approaches, they have been derived in the specific context of NNs.
- An analysis of gaps in a current automotive standard with regards to the use of ML approaches [14]. This source dates from 2018, so it encapsulates (relatively) recent research. However, the chosen standard, specifically International Organization for Standardization (ISO) 26262 [11] is a functional safety standard; that is, it only addresses unsafe behaviours caused by system malfunctions. For ML approaches, there is also a need to consider the Safety Of The Intended Function (SOTIF).

For the reasons outlined above, the computation-level objectives derived by the SASWG would not be expected to directly match the contents of either reference. Nevertheless, the objectives would be expected to cover all relevant issues raised in the reference material.

It is emphasised that the mappings established below are top-level and approximate. This is considered appropriate as the mappings are intended to justify (or, if necessary, refine) the computation-level objectives. More specifically, the mappings discussed in this appendix were not a key part of the process by which the computation-level objectives were derived.

B.1 Requirements for a NN Standard

Table 5 lists the requirements noted in [2]. Note that these requirements use the term Artificial Neural Network (ANN), rather than NN, which is preferred in the current document. Where appropriate, relevant computation-level objectives are highlighted. If no objectives are relevant, justification for this is provided.

Table 5: Computation-Level Objectives Compared against Requirements for a NN Standard

Standard Requirement	Relevant Objectives
Specify how the high-level goals of, or requirements for, the ANN module are to be obtained	COM2-1, COM2-2
Specify what should be done to ensure that the training data adequately represent the attainment of the high-level goals	COM1-3
Specify what type of networks can be used, and how each type is to be unambiguously designated	COM3-1
Specify how the input-output characteristics are to be unambiguously designated	COM1-1, COM1-2
Specify how the developer must describe the way in which the performance function for the network operates during training	COM2-3

Standard Requirement	Relevant Objectives
Specify what details the ANN developer must provide regarding the way in which the ANN module interfaces with the rest of the system	Out of scope: Autonomy architecture-level
Specify the extent of knowledge, relating to neural networks, required of management and development team personnel	Out of scope: Staffing
Specify what development model is to be used for the ANN module	COM5-1
Specify any outputs which the ANN module is required to produce in addition to its primary functional output	COM2-1
Specify whether formal methods or rigorous argument are to be used to develop the software which implements the neural network	COM4-1, COM4-2
Specify what methods are to be used for quality assurance in the trained network	COM1-1, COM4-1, COM5-1
Specify that the Verification and Validation (V&V) team should use generalisation tests on the trained network to verify that it has learned the principles implicit in the training data	COM2-3, COM2-6, COM2-7
Specify that the V&V team should validate a Safety-Critical Artificial Neural Network (SCANN) by investigating the behaviour of the SCANN over the whole of the input space	COM2-5
Specify how the developers should check that the initial safety assessments made for the system are not affected by the ANN module and how failures in other modules would affect the system, given the intended operation of the ANN	Out of scope: Platform-level
Specify that developers establish possible failure modes of the ANN module itself and the consequences	Out of scope: Autonomy architecture-level (supported by COM1-4, COM2-4, COM3-2, COM5-2, COM5-2)
Specify how Hazard and Operability Study (HAZOP) is to proceed, regarding the operation of network	Out of scope: Platform-level
Specify the brief and form of the HAZOP committee, as well as guide words for their use	Out of scope: Platform-level
Specify that a certification standard should insist that the developers build the network is such a way that the necessary data are available so that it is possible to do Failure Mode and Effects Analysis (FMEA) and HAZOP	COM3-3, COM3-4

It is apparent that all relevant requirements established by [2] are covered by one or more of the computational objectives derived by the SASWG. This provides further confidence in the identified computation-level objectives.

B.2 ML-Related Gaps in an Automotive Standard

The analysis of ISO 26262 identified a number of impacted or new Process Requirements (PRs). The associated phase and description are reproduced (from [14]) in Table 6. Relevant computation-level objectives are then highlighted; if there are no such objectives then justification is provided.

Table 6: Computation-Level Objectives Compared against Impacted or New PRs

Phase	Description	Relevant Objectives
(5) Initiation	Best practices: coding guidelines	COM3-2, COM4-1
(5) Initiation	ML decision gate	Out of scope: Autonomy architecture-level
(6) Software safety requirements	Requirements specification	COM1-3, COM2-1, COM2-2
(6) Software safety requirements	Requirements verification	COM2-3, COM2-5
(7) Architectural design	Fault tolerance	Out of scope: Autonomy architecture-level
(8) Software unit design, implementation	Best practices: notations	COM3-2, COM4-1
(8) Software unit design, implementation	Best practices: design principles	COM3-2
(8) Software unit design, implementation	Best practices: data set collection and verification	COM1-1, COM1-2
(8) Software unit design, implementation	Best practices: model selection	COM3-1
(8) Software unit design, implementation	Best practices: feature selection	COM1-3
(8) Software unit design, implementation	Best practices: training	COM3-2, COM4-1, COM5-1
(8) Software unit design, implementation	Best practices: data set splitting	Out of scope: Approach-specific
(8) Software unit design, implementation	Best practices: validation	COM2-3, COM3-2
(8) Software unit design, implementation	Best practices: testing	COM2-3, COM2-7, COM4-2, COM5-2
(8) Software unit design, implementation	Best practices: testing structural coverage	COM2-5
(8) Software unit design, implementation	Best practices: test vs. operating environment	COM1-4, COM2-6, COM4-1
(8) Software unit design, implementation	Best practices: test result explanation	COM3-3, COM3-4
(8) Software unit design, implementation	Best practices: verification	COM2-3, COM2-4, COM2-5, COM2-6, COM3-4

It is apparent that all impacted or new PRs established by [14] are covered by one or more of the computational objectives derived by the SASWG, or are intentionally outside the scope of this framework. This, again, provides further confidence in the computation-level objectives.

This page is intentionally blank

Appendix C Platform-Level Framework: Justification

This appendix provides a brief, outline survey of platform-level frameworks that could be used to structure thinking about the safety (or assurance) of autonomous systems. For ease of reference, Table 7 lists the frameworks that are considered.

Table 7: Platform-Level Frameworks Considered

Section	Framework
C.1.1	Waymo's System Safety Report
C.1.2	The Twelve Safety Elements from the NHTSA
C.1.3	HORIBA MIRA Framework
C.1.4	Uber Advanced Technologies Group
C.1.5	AAIP BOK
C.1.6	AI Safety Landscape Categories

Each platform-level framework is briefly summarised (subsection C.1) and a preferred framework is developed. A top-level mapping between frameworks is completed, to confirm that the chosen framework incorporates all relevant parts of the other platform-level frameworks (subsection C.2).

C.1 Platform-Level Frameworks

C.1.1 Waymo's System Safety Report

This platform-level framework comes from Waymo's System Safety Report[2]. This report establishes five distinct safety areas:

- Behavioural safety, which is about the behaviour of the vehicle on the road, including the decisions it makes. This is the most novel of the safety areas.
- Functional safety, which considers how the system operates in the presence of faults and failures. This appears to be standard system safety, including the use of redundant sub-systems, for example.
- Crash safety, which is about protecting people in the event of a crash. This appears to be normal automotive crash safety.
- Operational safety, which covers the interaction between Waymo vehicles and their passengers. This seems to be mainly focused on the user interface, which includes helping the passenger understand what the vehicle is perceiving and what it is doing on the road.
- Non-collision safety, which considers how the vehicle could harm those it interacts with in non-crash situations (including passengers, first responders and bystanders).

C.1.2 The Twelve Safety Elements from the NHTSA

This platform-level framework comes from "Automated Driving Systems 2.0: A Vision for Safety", published by the US National Highway Traffic Safety Administration (NHTSA)[3]. This introduces twelve "safety elements":

[2] https://storage.googleapis.com/sdc-prod/v1/safety-report/waymo-safety-report-2017-10.pdf.
[3] https://www.nhtsa.gov/manufacturers/automated-vehicles-manufacturers.

1. System Safety;
2. Operational Design Domain;
3. Object and Event Detection and Response;
4. Fallback (Minimum Risk Condition);
5. Validation Methods;
6. Human Machine Interface;
7. Vehicle Cybersecurity;
8. Crashworthiness;
9. Post-Crash Automated Driving System (ADS) Behaviour;
10. Data Recording;
11. Consumer Education and Training;
12. Federal, State and Local Laws.

C.1.3 HORIBA MIRA Framework

This platform-level framework is described in an HORIBA MIRA presentation [3]. The presentation describes an Autonomous Driver (AD); the following bullets summarise the high-level, generic features of the framework.

- STRATEGY: Argument split according to functionality that is intended, unintended and due to malicious intent.
- CLAIM: Intended Behaviour - The absence of unreasonable risk associated with the intended behaviour of the [autonomous system] has been achieved.
  - STRATEGY: Argument structured by the rationale for, and satisfaction of, specified requirements (REQs).
  - CLAIM: Requirements Rationale - Meeting the REQs yields the absence of unreasonable risk associated with the intended behaviour of the [autonomous system].
  - CLAIM: Requirements Satisfaction - The [autonomous system] behaves according to the REQs. (In this area, the framework also introduces: virtual testing; physical testing; and testing diversity and number.)
- CLAIM: Malfunctioning Behaviour - The absence of unreasonable risk associated with malfunctioning behaviour of the [autonomous system] has been achieved.
- CLAIM: Malicious Intent - The absence of unreasonable risk associated with malicious attack of the [autonomous system] has been achieved.

C.1.4 Uber Advanced Technologies Group

This platform-level framework is the safety case framework developed by Uber Advanced Technologies Group. This framework is intended for use with self-driving vehicles, especially passenger-carrying cars on public roads. This is a narrower scope than the current document, which addresses all autonomous systems. For reasons of brevity only the five top-level goals associated with this framework are listed below:

- G1 - Proficient: The Self-Driving Vehicle is acceptably safe during nominal operation.
- G2 - Fail-Safe: The Self-Driving Vehicle is acceptably safe in presence of faults and failures.
- G3 - Continuously Improving: Any anomaly that could affect the safety of the Self-Driving Vehicle is identified, evaluated, and resolved with appropriate corrective and preventative actions.
- G4 - Resilient: The Self-Driving Vehicle is acceptably safe in case of reasonably foreseeable misuse and unavoidable events.
- G5 - Trustworthy: The Self-Driving Enterprise is trustworthy.

C.1.5 AAIP BOK

This platform-level framework is part of the structure of the Assuring Autonomy International Programme (AAIP) Body Of Knowledge (BOK) [8]. This has a vast scope: it aims to be cross-domain, cross-technology and cross-application, covering all aspects of assurance and regulation of Robotics and Autonomous Systems (RAS). The current document has similar aims with regards to breadth of domains, technologies and applications: however, regulation-specifics are not a focus. For reasons of brevity only the top-level items are listed below:

- Defining required behaviour.
- Implementation of an RAS to provide the required behaviour.
- Understanding and controlling deviations from required behaviour.
- Gaining approval for operation of RAS.

Note that a more detailed, objective-level comparison between this document and the AAIP BOK is provided in Appendix D.

C.1.6 AI Safety Landscape Categories

This platform-level framework is based on work associated with the AI Safety 2019 conference. This presents a series of seven categories[4], one of which is underpinning and one of which is overarching. These are illustrated in Table 8.

[4] https://www.aisafetyw.org/ai-safety-landscape.

Table 8: AI Safety Landscape Categories

Safety-related Ethics, Security and Privacy				
Specification and Modelling	Verification and Validation	Runtime Monitoring and Enforcement	Human-Machine Interaction	Process Assurance and Certification
AI Safety Foundations				

Most of the categories are self-explanatory; the exception is AI Safety Foundations. This includes concepts such as uncertainty and generality, as well as characteristics like levels of autonomy and safety criticality. More generally, this category collects concerns in AI safety that span multiple other categories.

C.1.7 Choice of Framework

Whilst they provide a useful structure against which a chosen framework can be benchmarked, none of the preceding frameworks are suitable for use by the SASWG: they are either too focused on a specific type of platform, often a self-driving car, whereas the SASWG's work aims to cover all types of autonomous system; or they adopt a balanced view of system safety, whereas the SASWG's work deliberately targets aspects related to autonomy.

Consequently, having been informed by the frameworks listed above (and related items) a four-projection framework has been developed. These projections are described detail in [13]. For ease of reference, a summary is provided below:

- Behavioural Specification, which is about what the platform is required to do (and not do).
- Interacting Items, which is about things intended or required to interact with the platform, not directly part of the platform under consideration.
- People, which is about how the platform interacts with people, from a whole lifecycle perspective.
- Environment, which is about things in the operational environment outside the control of the platform developer or operator.

C.2 Framework Mappings

The following paragraphs discuss the relationship between the projections in the adopted framework and the aspects of the other frameworks outlined in the preceding subsection. Given the complexity of the items in the various frameworks, only top-level relationships are described.

The discussions also include a "Not Addressed" pseudo-projection, which captures considerations that do not readily relate to any of the projections in the adopted framework. By checking the contents of this pseudo-projection, and confirming that it contains nothing significant (from the perspective of the current document), confidence can be gained that the adopted framework covers all relevant topics.

C.2.1 Behavioural Specification

This projection relates to the Behavioural Safety, Functional Safety and Crash Safety elements from Waymo's System Safety Report.

It also relates to five of the NHTSA Safety Elements, specifically: Object and Event Detection and Response; Fallback (Minimum Risk Condition); Crashworthiness; Post Crash Automated Driving System Behaviour; and Federal, State and Local Laws.

Two of the top-level elements from the HORIBA MIRA framework are also relevant, specifically: Intended Behavior and Malfunctioning Behaviour.

The majority of the Uber Advanced Technologies Group framework is relevant to this projection. In particular, Proficient; Fail-Safe; Continuously Improving; and Resilient are all relevant.

Likewise, the majority of the top-level items from the AAIP BOK are relevant, specifically: Defining Required Behaviour; Implementing Required Behaviour; and Understanding and Controlling Deviations.

Finally, two of the AI Safety Landscape Categories are relevant: Specification and Modelling; and Verification and Validation.

C.2.2 Interacting Items

No elements from any of the frameworks are directly relevant to this projection. This is because none of the frameworks explicitly highlight the off-platform elements of the wider system. Instead, considerations of this type are implicitly included within discussions relating to the platform. However, as indicated by the objectives in [13], interacting items can have significant safety implications. Consequently, they are deemed worthy of separate identification.

C.2.3 People

Two elements from Waymo's System Safety Report are relevant: Operational Safety; and Non-Collision Safety.

There are three NHTSA Safety Elements that are relevant: Human Machine Interface; Vehicle Cybersecurity; and Consumer Education and Training.

A single top-level element from the HORIBA MIRA framework is relevant, namely, Malicious Intent.

None of the top-level items in the Uber Advanced Technologies Group framework are relevant to this projection.

Likewise, none of the AAIP BOK top-level items are relevant either.

Two of the AI Safety Landscape Categories are relevant, specifically: Safety-related Ethics, Security and Privacy; and Human-Machine Interface.

C.2.4 Environment

The Operational Design Domain safety element from the NHTSA directly maps to this projection, as does the Runtime Monitoring and Enforcement AI Safety Landscape category.

None of the other frameworks have items that directly map to this projection. This does not mean that these frameworks ignore the environment; it just means that these types of consideration appear at a lower-level, having been brigaded in a different fashion to the framework adopted by the SASWG.

C.2.5 Not Addressed

All elements from Waymo's System Safety Report are directly addressed by the collection of projections used in the framework adopted by the SASWG.

There are three NHTSA Safety Elements that are not directly addressed: System Safety; Validation Methods; Data Recording. The first of these is addressed by all three SASWG frameworks (i.e. by the entirety of the current document); the other two are addressed by the Computation-Level framework (see [13]).

All elements from the HORIBA MIRA framework are directly addressed.

There is a single top-level element from the Uber Advanced Technologies Group framework that is not directly addressed: Trustworthy. This element relates to the trustworthiness of the self-driving enterprise, which is a much wider consideration than the safety assurance objectives of the current document.

There is also a single top-level element from the AAIP BOK that is not directly addressed: Gaining Approval. This relates to liaison with certification authorities, which is outside the scope of the current document.

There are two AI Safety Landscape Categories that are not directly addressed, specifically: Process Assurance and Certification; and AI Safety Foundations. The former of these is outside the scope of the current document; the latter is a broad category that spans much of the content of the current document.

C.2.6 Relationship Summary

For ease of reference, the relationships outlined above are summarised in Table 9. Note that this presentation is deliberately simple and top-level.

Table 9: Relationships between Platform-Level Frameworks Considered

Framework	Behavioural Specification	Interacting Items	People	Environment	Not Addressed
Waymo's System Safety Report	Behavioural Safety, Functional Safety, Crash Safety	-	Operational Safety, Non-Collision Safety	-	-
NHTSA Safety Elements	Object and Event Detection and Response, Fallback, Crashworthiness, Post Crash Automated Driving System Behaviour, Federal State and Local Laws	-	Human Machine Interface, Vehicle Cybersecurity, and Consumer Education and Training	Operational Design Domain	System Safety, (Validation Methods, Data Recording are computation-level)
HORIBA MIRA	Intended Behaviour, Malfunctioning Behaviour	-	Malicious Intent	-	-

Framework	Behavioural Specification	Interacting Items	People	Environment	Not Addressed
Uber Advanced Technologies Group	Proficient, Fail-Safe, Continuously Improving, Resilient	-	-	-	Trustworthy
AAIP BOK	Defining Required Behaviour, Implementing Required Behaviour, Understanding and Controlling Deviations	-	-	-	Gaining Approval
AI Safety Landscape Categories	Specification and Modelling, Verification and Validation	-	Safety-related Ethics Security and Privacy, Human-Machine Interface.	-	Process Assurance and Certification, AI Safety Foundations

Overall, the preceding analysis indicates that, based on the selected comparator frameworks, there are no significant omissions from the chosen platform-level framework.

This page is intentionally blank

Appendix D Comparison with AAIP Body of Knowledge

This appendix provides a high-level comparison between the objectives established in this document and the Assuring Autonomy International Programme (AAIP) Body Of Knowledge (BOK) [8] structure.

To simplify presentation, each main section of the BOK argument structure is considered separately in the following subsections, with relevant objectives being highlighted. Brief explanations are provided for cases where there are no related objectives, for example, because of a difference in scope between the BOK and this document. For example, the BOK is concerned with an argument that covers the entire system (or platform); conversely, this document is intentionally focussed on aspects related to autonomy.

Note that the comparison reported in this appendix is deliberately high-level, with the aim of identifying whether there are any notable omissions from the collection of objectives discussed in this document. In particular, matching an objective to a BOK element does not necessarily mean that satisfying the objective will provide sufficient evidence to fully address the BOK element.

D.1 Defining Required Behaviour

Table 10 shows relevant objectives for BOK elements associated with the "defining required behaviour" section of the BOK argument structure.

Table 10: Objectives Comparison: Defining Required Behaviour

BOK Element	Relevant Objective
1.1 Identifying hazards	PLT1-2, PLT1-4, PLT2-2
1.1.1 Defining system scope	PLT1-1, PLT1-2
1.1.2 Defining the operating environment	PLT1-2, PLT2-1, PLT3-2, PLT4-1
1.1.3 Defining operating scenarios	PLT1-2, PLT2-1, PLT3-2, PLT4-1
1.2 Identifying hazardous system behaviour	PLT1-6, PLT2-3, PLT4-2
1.2.1 Considering human/ machine interactions	PLT3-1, PLT3-2, PLT3-3
1.3 Defining safety requirements	PLT1-2, PLT2-2
1.2.1 Validation of safety requirements	PLT1-5
1.4 Impact of security on safety	PLT1-4, PLT1-6, PLT3-4

This table prompts several observations. Firstly, the related objectives all come from the platform-level framework: since this framework is primarily concerned with requirements, this is reassuring. Secondly, most of the BOK elements have multiple related objectives: this is a consequence of the different structural approaches that have been used; this also emphasises the point that, whilst they were a useful aid when deriving objectives, the projections (and frameworks) need not be slavishly followed. Thirdly, all of the platform-level objectives appear at least once in the table. Fourthly, considering a more detailed point, the BOK differentiates between the operating environment (1.1.2) and operating scenarios (1.1.3); conversely, this document distinguishes between the platform (i.e. behavioural specification), interacting items and the (wider) environment.

More generally, this discussion indicates that the objectives listed in this document provide an appropriate level of coverage of the "defining required behaviour" section of the BOK argument structure. This observation provides some confidence in these objectives.

D.2 Implementation to Provide the Required Behaviour

Table 11 shows relevant objectives for BOK elements associated with the "implementation to provide the required behaviour" section of the BOK argument structure.

Table 11: Objectives Comparison: Implementation to Provide the Required Behaviour

BOK Element	Relevant Objective
2.1 System-level verification	PLT1-5
2.2 Implementation of Sense, Understand, Decide, Act (SUDA) elements	All computation-level and autonomy architecture-level objectives
2.2.1 Defining requirements for SUDA elements	COM1-3, COM2-1, COM2-2
2.2.1.1 Validation of requirements for SUDA elements	COM2-1, COM2-2
2.2.2 Defining requirements on components	COM1-3, COM2-1, COM2-2
2.2.2.1 Validation of requirements on components	COM2-1, COM2-2
2.2.3 Controlling interactions between components	PLT1-2, ARC1-1, ARC1-5
2.2.4 Verification of requirements for SUDA elements	PLT1-5, ARC1-5
2.3 Implementing requirements using ML	All computation-level objectives
2.3.1 Sufficiency of training data	COM1-1, COM1-2, COM1-3
2.3.2 Effective learning	COM2-3, COM2-4, COM2-5, COM3-1, COM3-2
2.3.3 Verification of the learned model	COM2-5, COM2-6, COM2-7, COM3-3
2.4 Controlling interactions with other systems	PLT2-1, PLT2-2, PLT2-3, PLT4-1
2.5 Controlling interactions at the system-level	PLT1-1, PLT1-2
2.6 Handling change during operation	ARC2-2, ARC3-1, ARC3-2, PLT2-3
2.6.1 Monitoring RAS operation	PLT1-6, ARC1-1, ARC1-2, ARC1-3, ARC1-4
2.7 Using simulation	COM2-6, ARC3-1, PLT1-5
2.8 Explainability	COM3-3, ARC2-1, PLT3-2

Consideration of this table highlights a number of points. For example, the BOK breaks the system down into SUDA elements and further down into components, whereas this document focuses on autonomy-enabling technologies that may be used within, or to deliver, elements and components. This means the BOK provides a more balanced, system-wide perspective; conversely, by design, this document focuses on aspects related to autonomy.

It is also apparent that the BOK explicitly highlights simulation (BOK Element 2.7). Given the importance of this topic, this explicit highlighting is advantageous. Within the current document, simulation is considered at each framework level. Whilst this potentially dilutes the importance of the topic, it does allow specific aspects to be addressed in greater detail: for example, considerations associated with platform-level simulation are somewhat different to those associated with computation-level activities.

Although mappings can be, and have been, made, this document's consideration of issues related fleets of autonomous systems (e.g. Objective PLT2-3) is not readily apparent in the BOK.

Overall, the preceding discussions do not suggest there are any significant omissions in the objectives listed in this document from the perspective of the "implementation to provide the required behaviour" section of the BOK.

D.3 Understanding and Controlling Deviations from Required Behaviour

Table 12 shows relevant objectives for BOK elements associated with the "understanding and controlling deviations from required behaviour" section of the BOK argument structure.

Table 12: Objectives Comparison: Understanding and Controlling Deviations from Required Behaviour

BOK Element	Relevant Objective
3.1 Identification of potential deviation from required behaviour	ARC1-1, ARC1-2, ARC1-3, ARC1-4, ARC1-5, PLT1-4, PLT3-2, PLT3-4
3.1.1 Identifying 'Sensing' deviations	
3.1.2 Identifying 'Understanding' deviations	
3.1.3 Identifying 'Deciding' deviations	
3.1.4 Identifying 'Acting' deviations	
3.1.5 Identifying infrastructure deviations	
3.1.6 Identifying ML deviations	
3.1.7 Identifying interaction deviations	
3.1.8 Identifying human / machine interaction deviations	
3.2 Mitigating potential deviations	PLT1-4
3.2.1 Failure mitigation	ARC1-3
3.2.2 Managing assurance deficits	Out of scope

This table clearly illustrates the different philosophies adopted by the BOK and this document. As noted earlier, the former adopts a balanced, system-wide approach that addresses all aspects of safety; it also separately highlights each of the SUDA elements. Conversely, this document is deliberately focused on autonomy-related items and is largely agnostic of where these are used within a system (or platform) architecture.

The table also shows the BOK's explicit focus on assurance, something that is more implicit within the current document. In particular, the lack of objectives that directly relate to the management of assurance deficits (BOK Element 3.2.2) is not considered to be a significant omission. This should occur naturally through appropriate consideration of the various objectives in this document.

More generally, there do not appear to be any significant omissions in the objectives listed in this document from the perspective of the "understanding and controlling deviations from required behaviour" section of the BOK.

D.4 Gaining Approval for Operation

Table 13 shows relevant objectives for BOK elements associated with the "gaining approval for operation" section of the BOK argument structure.

Table 13: Objectives Comparison: Gaining Approval for Operation

BOK Element	Relevant Objective
4.1 Conforming to rules and regulations	Out of scope
4.1.1 Identifying applicable rules and regulations	

BOK Element	Relevant Objective
4.1.2 Understanding the requirements rules and regulations	
4.2 Risk acceptance	Out of scope
4.2.1 Evaluating risks and benefits of RAS operation	
4.2.2 Consideration of ethical issues	
4.3 Provision of sufficient confidence in the required behaviour	COM2-5, PLT1-2, PLT1-5
4.4 Provision for investigation of incidents and accidents	COM3-4, ARC2-3

This table illustrates the different scopes of the BOK and the current document. In particular, the BOK includes laws and regulations, which are intentionally out of scope for this document, as they are expected to be addressed by standard system engineering processes. The BOK also explicitly considers ethics and risks of deployment. The former, whilst important, is out of scope for this document. The latter is not identified as a separate item in this document, but satisfying the associated objectives should provide a considerable body of evidence to inform risk evaluations.

Given these considerations, and taking into account the intended scope of this document, this table does not suggest any significant omissions in this document's objectives from the perspective of the "gaining approval for operation" section of the BOK.

D.5 Non-Related Objectives

Collectively, the preceding four tables include all but four of the objectives listed in this document. Collectively, these four objectives make up the software and hardware projections, both of which relate to the computation-level. These considerations were motivated by the autonomy-focused, projection-based way that objectives were derived. In contrast, this low-level, cross-cutting concern does not readily appear from the approach adopted within the BOK. Despite this, the objectives remain important.

The SASWG view the different approaches adopted by the BOK and this document as strengths rather than weaknesses. Taking different approaches to largely the same question (accepting there is some difference in the respective scopes) helps ensure nothing is overlooked. To that end, the top-level mappings established in this appendix provide some confidence that the collection of objectives listed in this document are appropriate to their intended use.

Appendix E Comparison with AMLAS

Version 1 of Assurance of Machine Learning in Autonomous Systems (AMLAS) [10] was released in February 2021. It was authored by the Assuring Autonomy International Programme (AAIP).

Assurance of Machine Learning in Autonomous Systems (AMLAS) considers six lifecycle stages, specifically: ML safety assurance scoping; safety requirements elicitation; data management; model learning; model verification; and model deployment. For each stage, a safety argument pattern is defined. This is intended to be used to explain how, and the extent to which, the generated evidence supports the relevant ML safety claims. For each stage, a small number of objectives are also described. These form the basis of the comparison outlined in this appendix. In addition, AMLAS details inputs to, and outputs from, each stage.

Within AMLAS there is a focus on offline, supervised learning. Additionally, AMLAS is not intended to be used in isolation. In particular, it is intended to be used alongside other standards and guidelines, for example, those that specify best-practices in safety-critical systems. These attributes make it a good comparator item for this document.

The following subsections consider each of the AMLAS stages in turn. Each objective associated with the relevant stage is listed, along with a brief discussion of related objectives from this document. Despite both documents using the same term (i.e. "objective"), it is interpreted slightly differently. In this document, objectives relate to intended outcomes. Conversely, in AMLAS, objectives also relate to activities (e.g. "Integrate the machine learnt component into the target system"). Whilst this discrepancy would make it difficult to conduct a detailed comparison, it has only a minor effect on the top-level analysis considered in this appendix.

For clarity, it should be noted that AMLAS refers to a "system", whereas this document uses the term "platform". For the purposes of the comparison discussed in this appendix, these terms may be considered interchangeable.

It should also be noted that AMLAS makes repeated reference to assurance argument patterns. Conversely, this document adopts a more flexible approach, describing elements of evidence that should support an argument, but not dictating the precise form of that argument. Given this difference in approach, the following comparison does not consider the assurance argument patterns within AMLAS in any detail.

E.1 Stage 1: ML Safety Assurance Scoping

- **Objective 1: Define the scope of the safety assurance process for the ML component.**

  From the perspective of AMLAS, the scope of the safety assurance process for the ML component is based on a combination of: system safety requirements; the system's operating environment; a description of the system; and a description of the ML component.

  Within this document, platform-level safety considerations are, not surprisingly, addressed within the platform-level framework. For example: Objective PLT1-2 establishes what safe means in the context of the platform; Objective PLT1-4 extends these considerations to include faults, failures and foreseeable misuse; and Objective PLT4-1 considers the effects of the environment.

  In addition, the tolerance projection of the autonomy architecture-level framework provides a link between the platform and the ML component (in AMLAS terminology) or the computation (using the terminology of this document). For example, Objective ARC1-1 considers failures of sub-systems that provide computation inputs and ARC1-4 considers potential adversarial actions.

  Finally, the task projection of the computation-level framework describes what is required of the ML component. For example: Objective COM2-1 considers functional requirements; Objective COM2-2 considers non-functional requirements; and Objective COM2-3 considers how performance is measured.

- **Objective 2: Define the scope of the safety case for the ML component.**

  The "scoping" nature of this objective, combined with the close relationship between the safety assurance process (previous objective) and the safety case (this objective), means that the previous discussion is also applicable here.

- **Objective 3: Create the top-level safety assurance claim and specify the relevant contextual information for the ML safety argument.**

  This objective directly relates to argument structures and, as such, it is not considered in this comparison. It is included in this appendix for completeness.

E.2 Stage 2: ML Safety Requirements Assurance

- **Objective 1: Develop the machine learning safety requirements from the allocated system safety requirements.**

  This document does not specifically distinguish between "Safety Assurance" (AMLAS Stage 1) and "Safety Requirements" (AMLAS Stage 2). Hence, much of the discussion in the preceding subsection is also relevant here.

- **Objective 2: Validate the machine learning safety requirements against the allocated safety requirements, the system and software architecture and operational environment.**

  This AMLAS objective introduces two new points, specifically, the system architecture and the software architecture: the safety requirements have been discussed previously (e.g. Objective PLT1-2), as has the operational environment (e.g. Objective PLT4-1).

  In terms of system architecture, the autonomy architecture-level framework specifically considers the link between the ML component and the system. For example: Objective ARC1-1 considers the sub-systems that provide inputs to the ML component; Objective ARC1-5 considers how the system tolerates incorrect outputs from the ML component; and Objective ARC2-1 manages passage of information from the ML component to the wider system.

  Questions relating to software architecture are covered within the computation-level framework. For example: Objective COM3-1 relates to the choice of algorithm type; Objective COM2-6 considers the test environment; Objective COM4-1 talks about software standards; and Objective COM5-1 talks about hardware standards.

- **Objective 3: Create an assurance argument, based on the evidence generated by meeting the first two objectives, that provides a clear justification for the ML safety requirements. This should explicitly explain the tradeoffs, assumptions and uncertainties concerning both the safety requirements and the process by which they are developed and validated.**

  This objective directly relates to argument structures and, as such, it is not considered in this comparison. It is included in this appendix for completeness.

E.3 Stage 3: Data Management

- **Objective 1: Develop data requirements which are sufficient to allow for the ML safety requirements to be encoded as features against which the data sets to be produced in this stage may be assessed.**

  Although this document does not specifically focus on data requirements, Objective COM1-3 ensures that the required behaviour is suitably captured in data.

- **Objective 2: Generate data sets in accordance with the data requirements for use in the development and verification stages, providing a rationale for those activities undertaken with respect to the ML safety requirements.**

  This AMLAS objective uses the term "generate", whereas Objective COM1-1 adopts "acquired". The latter term includes data sets that are generated, for example, by some synthetic process, as well as data sets that are collected, for example, by observing some external (possibly, difficult to control) process. This slight difference in terminology is not considered to be of significance: in particular, AMLAS is equally-applicable to data sets created by observation. Both AMLAS and this document are also applicable to the use of pre-existing data sets, including those from external sources.

  From the perspective of this document, Objective COM1-1 covers data acquisition (in whatever manner) and Objective COM1-2 addresses pre-processing.

- **Objective 3: Analyse the data sets obtained by Objective 2 to determine their sufficiency in meeting the data requirements.**

  The sufficiency of data is considered in Objective COM1-3. Additionally, considerations specifically relating to distribution shift are addressed by Objective COM1-4.

- **Objective 4: Create an assurance argument, based on the evidence generated by meeting the first three objectives, that provides a clear justification of the ML Data requirements. This should explicitly state the assumptions and tradeoffs made and any uncertainties concerning the data requirements and the processes by which they were developed and validated.**

  This objective directly relates to argument structures and, as such, it is not considered in this comparison. It is included in this appendix for completeness.

E.4 Stage 4: Model Learning

- **Objective 1: Develop the machine learnt model using the development data obtained in the previous stage such that the allocated ML safety requirements are satisfied.**

  Several of this document's objectives relate to model development. For example: Objective COM3-1 considers the type of machine learning algorithm employed; Objective COM2-1 addresses functional requirements; and Objective COM2-2 addresses non-functional requirements.

- **Objective 2: Use internal test data to assess the extent to which the machine learnt model is able to meet the ML safety requirements when presented with data not used for development.**

  In addition to the two objectives relating to functional and non-functional requirements (noted in relation to the previous AMLAS objective), Objective COM2-3 considers how the algorithm's performance is measured and Objective COM2-5 considers verification coverage.

- **Objective 3: Create an assurance argument, based on the evidence generated by meeting the first two objectives, which provides a clear justification that the ML model meets the ML safety requirements. This should explicitly explain the tradeoffs, assumptions and uncertainties concerning both the ML model and the process by which it is developed and validated.**

This objective directly relates to argument structures and, as such, it is not considered in this comparison. It is included in this appendix for completeness.

E.5 Stage 5: Model Verification

- **Objective 1: Demonstrate that the model will meet the ML safety requirements when exposed to inputs not present during the development of the model.**

  This document does not specifically distinguish between "Model Development" (AMLAS Stage 4) and "Model Verification" (AMLAS Stage 5). Consequently, the discussion in the previous subsection is also relevant here.

  In addition, this document explicitly considers the issue of distribution shift, which specifically relates to data different to that observed during development, via Objective COM1-4.

- **Objective 2: Create an assurance argument, based on the evidence generated by the first objective. The argument should clearly demonstrate the relationship between the verification evidence and the ML safety requirements. It should explicitly explain the tradeoffs, assumptions and uncertainties concerning the verification results and the process by which they were generated.**

  This objective directly relates to argument structures and, as such, it is not considered in this comparison. It is included in this appendix for completeness.

E.6 Stage 6: Model Deployment

- **Objective 1: Integrate the machine learnt component into the target system in such a manner that the system satisfies the allocated system safety requirements. The component should be integrated in the pipeline linking its inputs and outputs to other system components.**

  Within this document, integration is addressed via the autonomy architecture-level framework. For example: Objective ARC1-1 considers system components that provide inputs; Objective ARC1-5 considers the wider system's response to an invalid computation output; and Objective ARC2-1 considers wider information provision.

- **Objective 2: Demonstrate that the allocated system safety requirements are still satisfied during operation of the target system and environment.**

  This AMLAS objective is, in essence, related to system safety. From the perspective of this document, it as addressed by a combination of Objective PLT1-2, which provides a description of "safe" in the context of the target system; Objective PLT1-4, which explicitly considers faults and failures; Objective PLT1-5 which verifies behaviour; and Objective PLT1-6, which concerns operational monitoring (e.g. to help identify new hazards).

- **Objective 3: Create an assurance argument to demonstrate that the ML model will continue to meet the ML safety requirements once integrated into the target system.**

  This objective directly relates to argument structures and, as such, it is not considered in this comparison. It is included in this appendix for completeness.

E.7 Summary

This document and AMLAS adopt different perspectives: for example, the AMLAS is based on an explicitly-defined lifecycle and explicitly creates an assurance argument, whereas this document supports a more flexible approach.

Nevertheless, the top-level mapping outlined above demonstrates that the topics considered in AMLAS are suitably covered within this document.

This page is intentionally blank

Appendix F Comparison with UL4600

This appendix provides a high-level comparison between the objectives established in this document and a draft version (dated 2 October 2019) of UL4600 [15] which has been developed by Underwriters Laboratories (UL) and Edge Case Research (ECR).

The comparison is focused on identifying objectives or projections from the current document that map to the various section headings in UL4600. This provides some confidence that relevant topics have been addressed: it does not indicate that compliance with this document guarantees compliance with UL4600, or vice versa.

Each section of UL4600 that contains objectives is considered in turn below. This is followed by a very brief summary of the conclusions from this comparison exercise.

F.1 UL4600 Sections

F.1.1 Safety Case and Arguments

This section of UL4600 is mainly concerned with the structure, content and presentation of the platform-level safety case. These considerations are outside the scope of this document. However, it is noted that a platform-level safety argument would be expected to be supported by the sort of evidence produced via compliance with the objectives in this document.

F.1.2 Risk Assessment

This section of UL4600 considers fault and hazard identification, risk evaluation and risk mitigation. These topics are considered in various places in this document. Examples include Objectives PLT1-2 and PLT1-4, which relate to the behavioural specification projection within the platform-level framework. Other projections in that framework are also relevant, including: Objective PLT2-3, which considers interacting items; Objective PLT3-1, which considers people; and Objective PLT4-1, which considers the environment.

F.1.3 Interaction with Humans and Road Users

The title of this UL4600 section illustrates how its scope (or at least its genesis) differs from this document. In particular, UL4600 has an implicit focus on autonomous road vehicles, whereas this document is intended to cover a much wider variety of autonomous systems including, for example, medical diagnosis systems.

The general contents of this UL4600 section are addressed by projections in the platform-level framework, for example: Objective PLT2-2 in the interacting items projection; Objective PLT3-1 in the people projection; and Objective PLT4-1 in the environment projection. There are, inevitably, some differences in the detail: for example, UL4600 explicitly identifies animals, whereas this document is less prescriptive in terms of possible elements of the environment.

F.1.4 Autonomy Functions and Support

This section of UL4600 considers (using the terminology of this document) autonomy-enabling techniques. It also considers definition of the operational design domain and specific platform-level functions that may be autonomy-related (e.g. sensing, perception, planning, prediction). Considerations related to timing are also

included.

At the platform-level, use of autonomy-enabling techniques is addressed by the by the behavioural specification projection, for example, Objective PLT1-1. With regards to platform-level functions, this document is less descriptive than UL4600. However, similar notions are represented, for example: sensing and perception are related to Objective PLT4-2 in the platform-level framework; understanding the performance of ML algorithms is addressed by Objectives COM2-3 and COM2-4, from the task projection in the computation-level framework; likewise, timing requirements are addressed by Objective COM2-2.

F.1.5 Software and System Engineering Processes

The contents of this section of UL4600 are clearly described by its title. From the perspective of this document, software development processes are covered by Objective COM4-1 from the software projection in the computation-level framework; likewise, (computational) hardware development processes are covered by Objective COM5-1 from the hardware projection in the computation-level framework. Software and hardware processes are explicitly included as there are autonomy-specific considerations relevant to these areas. Conversely, within the current document there is no corresponding objective for system-level engineering processes. This reflects a deliberate focus on aspects directly related to autonomy and the associated desire to avoid duplicating existing guidance on general topics. More specifically, the autonomy architecture-level framework should allow autonomy-enabling technologies to be incorporated within standard system engineering processes.

F.1.6 Dependability

This section of UL4600 considers maintaining safety in the presence of faults, including fault detection and recovery. The use of redundancy and isolation are also covered, as are incident response and cyber security.

Within the current document, dependability (including fault prevention and fault tolerance) is covered at all three framework levels. Relevant examples include: Objectives COM4-2 and COM5-2 at the computation-level; all of the objectives related to the tolerance projection in the autonomy architecture-level; and Objectives PLT1-4 and PLT1-6 at the platform-level. Likewise, incident response is covered across a number of levels, for example, via Objectives COM3-4, ARC2-3 and PLT3-2. The same is also true for cyber security; relevant items include Objectives ARC1-4, ARC2-4, ARC3-1, PLT1-6 and PLT3-4.

F.1.7 Data and Networking

This section of UL4600 considers data communications and networks (essentially, data in motion), data storage (essentially, data at rest) and associated infrastructure. The section appears to focus on data communications to, from and within a platform, that is, "platform data": data associated with training algorithms using ML is covered in the "Autonomy Functions and Support" section. The concept of "platform data" is less explicit in this document than in UL4600. Nevertheless, relevant concepts are covered. For example, data-related faults and failures should be managed by Objectives ARC1-1 and ARC1-3, regardless of whether the data is in motion or at rest.

F.1.8 Verification, Validation and Test

In addition to the three topics listed in the section title, this section of UL4600 also includes run-time monitoring and updates to the safety case. Within this document, the concepts of verification, validation and test are considered

at both the computation-level (e.g. Objectives COM2-3 and COM2-5) and the platform-level (e.g. Objective PLT1-5). Run-time monitoring is addressed through Objective PLT1-6. The specific nature and construct of a safety case are outside the scope of this document; likewise, updates to the safety case are also out of scope.

F.1.9 Tool Qualification, COTS and Legacy Components

The contents of this UL4600 section are as indicated by its title. From the perspective of this document, qualification of software and hardware engineering tools is covered by Objectives COM4-1 and COM5-1, respectively. Commercial Off-The-Shelf (COTS) items are not explicitly addressed by this document, mainly because some readers can interpret the term too narrowly, for example, excluding open source frameworks and pre-trained ML models. However, some relevant concepts are covered, for example, by Objectives ARC1-4 and PLT3-4.

F.1.10 Lifecycle Concerns

This section of UL4600 steps through typical lifecycle phases, including requirements, design, manufacturing, operation and disposal. It also includes field modifications and updates.

This document does not include such an explicit listing of lifecycle phases. However, aspects of these are addressed, for example, in Objectives ARC2-2, PLT1-6 and PLT3-3. Updates are covered by the adaptation projection at the autonomy architecture-level. Allowable field modifications are considered in the same way; unauthorised field modifications are covered, for example, by Objectives ARC1-4, PLT1-4 and PLT3-4.

F.1.11 Maintenance

This section of UL4600 includes maintenance and other aspects of non-operational safety. In this document, maintenance is addressed via Objectives PLT3-2 and PLT3-3. Other aspects of non-operational safety are less explicit in this document than in UL4600: they are at least partially addressed by, for example, Objectives PLT1-2, PLT1-4, PLT2-2 and PLT4-1.

F.1.12 Metrics and Safety Performance Indicators

This section of UL4600 is mainly concerned with creating and monitoring platform-level Safety Performance Indicators (SPIs). The need to continually-demonstrate safety is not as explicit in this document as it is in UL4600. However, the same notion is considered by Objective PLT1-6, in the platform-level framework. This is supported by Objective ARC2-2, in the autonomy architecture-level framework, and Objectives COM2-3 and COM2-4, in the computation-level framework.

F.1.13 Assessment

This section of UL4600 is concerned with assessing conformance to UL4600, including the use of independence and monitoring. This type of conformance is outside the scope of this document.

F.2 Summary

The preceding discussions have provided a high-level comparison between a draft version of UL4600 (dated 2 October 2019) [15] and this document. Whilst there are some intentional differences in scope, this UL4600-based analysis has not identified any significant omissions from this document.

Appendix G Comparison with OECD Principles on AI

In May 2019, member countries of the Organisation for Economic Co-operation and Development (OECD) adopted a number of principles[5] on AI. This appendix provides a high-level indication of how the contents of this document may support these principles.

In general, AI only has an effect when it is embodied within a wider system (or platform). Consequently, all three frameworks used in this document are potentially relevant to the OECD principles. The following paragraphs indicate how the projections and, where relevant, objectives associated with these frameworks relate to each of the principles.

G.1 Principles

G.1.1 AI should benefit people and the planet by driving inclusive growth, sustainable development and well-being.

This principle is focused on the effects of the AI. This relates most directly to the requirements that are placed on the behaviour of the associated platform: this is addressed in the behavioural specification projection within the platform-level framework.

G.1.2 AI systems should be designed in a way that respects the rule of law, human rights, democratic values and diversity, and they should include appropriate safeguards – for example, enabling human intervention where necessary – to ensure a fair and just society.

From the perspective of this guidance document, the "rule of law" part of this principle is expected to be covered by standard systems engineering process. Hence, it is not directly related to any projection (or objective).

The "human rights" and "democratic values" pieces are, arguably, about platform-level requirements: these are covered by the behavioural specification projection within the platform-level framework. Including appropriate interfaces (e.g. to support explanation of an AI-based decision) may also be relevant; this relates to Objective PLT3-2.

In order for an AI to respect diversity, this must be included in the data used to support the AI's development: considerations associated with the experience projection, in the computation-level framework, are relevant here.

The current document's focus on safety assurance means that safeguards are considered from multiple viewpoints. For example: Objectives COM4-2 and COM5-2 provide safeguards against software and hardware misbehaviour; the tolerance projection, in the autonomy architecture-level framework, provides safeguards against faults, failures and adversarial attempts to disrupt a computation; Objective ARC3-2 provides safeguards against inappropriate adaptation; and Objective PLT1-4 provides safeguards against foreseeable misuse and abuse.

Finally, the people projection, within the platform-level framework, supports the need for human intervention, where necessary.

[5] https://www.oecd.org/going-digital/ai/principles/.

G.1.3 There should be transparency and responsible disclosure around AI systems to ensure that people understand AI-based outcomes and can challenge them.

In general, publishing information against the objectives listed in this document will support transparency and responsible disclosure. This information should also allow for reasonable challenge. This could arise, for example, as part of a formal certification process; alternatively, it could come from less formal interactions with the general public.

In addition, the objectives associated with the information provision projection, within the autonomy architecture-level framework, should ensure that people are provided with accurate information. Furthermore, the objectives associated with the people projection, in the platform-level framework, should ensure that appropriate information is provided in an intelligible manner.

G.1.4 AI systems must function in a robust, secure and safe way throughout their life cycles and potential risks should be continually assessed and managed.

Arguably, all of the objectives in this document are relevant to this principle. Picking out some specific examples: Objective PLT1-2 leads to a definition of "safe operation"; Objective PLT1-4 maintains safety in the presence of faults and failures, as well as foreseeable misuse and abuse; Objective PLT1-6 provides monitoring during operational use (e.g. to identify new hazards, as part of continual assessment and management); Objective COM1-4 protects against distribution shift; Objective ARC1-2 ensures the platform is tolerant to "out of support" operational inputs; Objective ARC1-4 protects against adversarial attempts to disrupt a computation; Objective ARC3-1 protects against inappropriate or unauthorised adaptations; and the people projection, within the platform-level framework, explicitly covers the whole system lifecycle.

G.1.5 Organisations and individuals developing, deploying or operating AI systems should be held accountable for their proper functioning in line with the above principles.

This principle is mainly concerned with the legal and regulatory environment within which AI systems are used. These considerations are deliberately outside the scope of the current document. Nevertheless, requiring compliance with the objectives in this document may be one way of holding to account those responsible for developing, deploying or operating such systems.

Appendix H Comparison with ALTAI

In June 2018, the European Union (EU) established a High-Level Expert Group (HLEG) on AI. In July 2020, the EU AI HLEG released the Assessment List for Trustworthy Artificial Intelligence (ALTAI)[6]. This list is intended to be used as part of a self-assessment process. It has been informed by a piloting process, which included fifty in-depth interviews with selected companies and two publicly accessible questionnaires for technical and non-technical stakeholders.

Assessment List for Trustworthy Artificial Intelligence (ALTAI) is structured around seven requirements. Within each requirement, there are a series of topic groups. A number of questions are associated with each of the topic groups. Those questions are further refined into sub-questions. The following subsections provide a mapping between these sub-questions and the objectives established in this document.

H.1 ALTAI Requirement 1: Human Agency and Oversight

This requirement has the following topic groups: human agency; oversight.

Table 14 maps questions within the human agency topic area to this document's objectives.

Table 14: Objectives Comparison: Human Agency

ALTAI Question	Relevant Objective
Is the AI system designed to interact, guide or take decisions by human end-users that affect humans or society?	PLT1-2, PLT1-5
Could the AI system generate confusion for some or all end-users or subjects on whether they are interacting with a human or AI system?	PLT3-2
Could the AI system affect human autonomy by generating over-reliance by end-users?	PLT1-4, PLT3-3 (Noting that, for PLT1-4, over-reliance is a form of misuse.)
Could the AI system affect human autonomy by interfering with the end-user's decision-making process in any other unintended and undesirable way?	PLT3-1, PLT3-2, PLT3-3
Does the AI system simulate social interaction with or between end-users or subjects?	PLT1-2
Does the AI system risk creating human attachment, stimulating addictive behaviour, or manipulating user behaviour?	PLT1-2

[6] https://futurium.ec.europa.eu/en/european-ai-alliance/pages/altai-assessment-list-trustworthy-artificial-intelligence.

Table 15 maps questions within the oversight topic area to this document's objectives.

Table 15: Objectives Comparison: Oversight

ALTAI Question	Relevant Objective
Have the humans (human-in-the-loop, human-on-the-loop, human-in-command) been given specific training on how to exercise oversight?	PLT3-3
Did you establish any detection and response mechanisms for undesirable adverse effects of the AI system for the end-user or subject?	PLT1-4, PLT1-6
Did you ensure a 'stop button' or procedure to safely abort an operation when needed?	PLT1-2
Did you take any specific oversight and control measures to reflect the self-learning or autonomous nature of the AI system?	PLT1-6

H.2 ALTAI Requirement 2: Technical Robustness and Safety

This requirement has the following topic groups: resilience to attack and security; general safety; accuracy; reliability, fall-back plans and reproducibility.

Table 16 maps questions within the resilience to attack and security topic area to this document's objectives.

Table 16: Objectives Comparison: Resilience to Attack and Security

ALTAI Question	Relevant Objective
Could the AI system have adversarial, critical or damaging effects (e.g. to human or societal safety) in case of risks or threats such as design or technical faults, defects, outages, attacks, misuse, inappropriate or malicious use?	PLT1-4, ARC1-4
Is the AI system certified for cybersecurity (e.g. the certification scheme created by the Cybersecurity Act in Europe) or is it compliant with specific security standards?	Domain-specific certification is intentionally outside the scope of this document; cyber security is considered, though.
How exposed is the AI system to cyber-attacks?	PLT1-4, ARC1-4
Did you put measures in place to ensure the integrity, robustness and overall security of the AI system against potential attacks over its lifecycle?	COM4-1, COM5-1, ARC1-4, ARC3-1, PLT1-6
Did you red-team/pentest the system?	PLT1-4
Did you inform end-users of the duration of security coverage and updates?	This is a system / platform-level deployment issue, rather than an autonomy-enabling one.

Table 17 maps questions within the general safety topic area to this document's objectives.

Table 17: Objectives Comparison: General Safety

ALTAI Question	Relevant Objective
Did you define risks, risk metrics and risk levels of the AI system in each specific use case?	PLT1-2
Did you identify the possible threats to the AI system (design faults, technical faults, environmental threats) and the possible consequences?	PLT1-2, PLT1-4
Did you assess the dependency of a critical AI system's decisions on its stable and reliable behaviour?	ARC1-5
Did you plan fault tolerance via, e.g. a duplicated system or another parallel system (AI-based or 'conventional')?	ARC1-3, ARC1-4, ARC1-5
Did you develop a mechanism to evaluate when the AI system has been changed to merit a new review of its technical robustness and safety?	ARC3-1

Table 18 maps questions within the accuracy topic area to this document's objectives.

Table 18: Objectives Comparison: Accuracy

ALTAI Question	Relevant Objective
Could a low level of accuracy of the AI system result in critical, adversarial or damaging consequences?	COM2-3, ARC1-5
Did you put in place measures to ensure that the data (including training data) used to develop the AI system is up-to-date, of high quality, complete and representative of the environment the system will be deployed in?	COM1-1, COM1-2, COM1-3, COM1-4
Did you put in place a series of steps to monitor, and document the AI system's accuracy?	COM2-3, COM3-4, ARC2-2, ARC2-3, PLT1-6
Did you consider whether the AI system's operation can invalidate the data or assumptions it was trained on, and how this might lead to adversarial effects?	COM1-4, ARC1-2, ARC1-4
Did you put processes in place to ensure that the level of accuracy of the AI system to be expected by end-users and/or subjects is properly communicated?	PLT1-2

Table 19 maps questions within the reliability, fall-back plans and reproducibility topic area to this document's objectives.

Table 19: Objectives Comparison: Reliability, Fall-Back Plans and Reproducibility

ALTAI Question	Relevant Objective
Could the AI system cause critical, adversarial, or damaging consequences (e.g. pertaining to human safety) in case of low reliability and/or reproducibility?	ARC1-3, ARC1-4, ARC1-5
Did you put in place verification and validation methods and documentation (e.g. logging) to evaluate and ensure different aspects of the AI system's reliability and reproducibility?	COM2-3, COM2-5, COM2-6, COM3-4, ARC2-2, ARC2-3, PLT1-5

ALTAI Question	Relevant Objective
Did you define tested failsafe fallback plans to address AI system errors of whatever origin and put governance procedures in place to trigger them?	PLT1-4, ARC1-1, ARC1-3, ARC1-5
Did you put in place a proper procedure for handling the cases where the AI system yields results with a low confidence score?	PLT1-2, PLT1-6
Is your AI system using (online) continual learning? [Did you consider potential negative consequences from the AI system learning novel or unusual methods to score well on its objective function?]	COM1-1, ARC3-1

H.3 ALTAI Requirement 3: Privacy and Data Governance

This requirement has the following topic groups: privacy; data governance.

Table 20 maps questions within the privacy topic area to this document's objectives.

Table 20: Objectives Comparison: Privacy

ALTAI Question	Relevant Objective
Did you consider the impact of the AI system on the right to privacy, the right to physical, mental and/or moral integrity and the right to data protection?	ARC2-4 (Noting this is more about security than privacy.)
Depending on the use case, did you establish mechanisms that allow flagging issues related to privacy concerning the AI system?	PLT1-6 (Noting this is more about operational monitoring than a mechanism for users to identify issues.)

Table 21 maps questions within the data governance topic area to this document's objectives.

Table 21: Objectives Comparison: Data Governance

ALTAI Question	Relevant Objective
Is your AI system being trained, or was it developed, by using or processing personal data (including special categories of personal data)?	Legal considerations are intentionally outside the scope of this document.
Did you put in place any specific measures some of which are mandatory under the General Data Protection Regulation (GDPR), or a non-European equivalent?	
Did you consider the privacy and data protection implications of the AI system's non-personal training-data or other processed non-personal data?	
Did you align the AI system with relevant standards (e.g. ISO, IEEE) or widely adopted protocols for (daily) data management and governance?	Domain-specific standards are intentionally outside the scope of this document.

H.4 ALTAI Requirement 4: Transparency

This requirement has the following topic groups: traceability; explainability; communication.

Table 22 maps questions within the traceability topic area to this document's objectives.

Table 22: Objectives Comparison: Traceability

ALTAI Question	Relevant Objective
Did you put in place measures that address the traceability of the AI system during its entire lifecycle?	COM1-1, COM1-2, COM2-1, COM2-2, COM4-1, COM5-1, ARC2-2, PLT1-6

Table 23 maps questions within the explainability topic area to this document's objectives.

Table 23: Objectives Comparison: Explainability

ALTAI Question	Relevant Objective
Did you explain the decision(s) of the AI system to the users?	COM3-3, ARC2-1, PLT3-2, PLT3-3
Do you continuously survey the users if they understand the decision(s) of the AI system?	PLT1-6 (Noting this is more about operational monitoring than user surveys.)

Table 24 maps questions within the communication topic area to this document's objectives.

Table 24: Objectives Comparison: Communication

ALTAI Question	Relevant Objective
In cases of interactive AI systems (e.g. chatbots, robo-lawyers), do you communicate to users that they are interacting with an AI system instead of a human?	PLT3-2, PLT3-3
Did you establish mechanisms to inform users about the purpose, criteria and limitations of the decision(s) generated by the AI system?	PLT3-2, PLT3-3

H.5 ALTAI Requirement 5: Diversity, Non-Discrimination and Fairness

This requirement has the following topic groups: avoidance of unfair bias; accessibility and universal design; stakeholder participation.

Table 25 maps questions within the avoidance of unfair bias topic area to this document's objectives.

Table 25: Objectives Comparison: Avoidance of Unfair Bias

ALTAI Question	Relevant Objective
Did you establish a strategy or a set of procedures to avoid creating or reinforcing unfair bias in the AI system, both regarding the use of input data as well as for the algorithm design?	COM1-1, COM1-3, COM2-3
Did you consider diversity and representativeness of end-users and/or subjects in the data?	COM1-3, COM2-3
Did you put in place educational and awareness initiatives to help AI designers and AI developers be more aware of the possible bias they can inject in designing and developing the AI system?	Staff competency is intentionally outside the scope of this document.

ALTAI Question	Relevant Objective
Did you ensure a mechanism that allows for the flagging of issues related to bias, discrimination or poor performance of the AI system?	PLT1-6 (Noting this is more about operational monitoring than a mechanism for users to identify issues.)
Is your definition of fairness commonly used and implemented in any phase of the process of setting up the AI system?	No directly associated objectives.

Table 26 maps questions within the accessibility and universal design topic area to this document's objectives.

Table 26: Objectives Comparison: Accessibility and Universal Design

ALTAI Question	Relevant Objective
Did you ensure that the AI system corresponds to the variety of preferences and abilities in society?	PLT3-1, PLT3-2
Did you assess whether the AI system's user interface is usable by those with special needs or disabilities or those at risk of exclusion?	PLT3-1, PLT3-2
Did you ensure that Universal Design principles are taken into account during every step of the planning and development process, if applicable?	This is a general systems engineering question, rather than an AI-enabling technology one.
Did you take the impact of the AI system on the potential end-users and/or subjects into account?	PLT3-1, PLT3-2

Table 27 maps questions within the stakeholder participation topic area to this document's objectives.

Table 27: Objectives Comparison: Stakeholder Participation

ALTAI Question	Relevant Objective
Did you consider a mechanism to include the participation of the widest range of possible stakeholders in the AI system's design and development?	This is a general systems engineering question, rather than an AI-enabling technology one.

H.6 ALTAI Requirement 6: Societal and Environmental Well-Being

This requirement has the following topic groups: environmental well-being; impact on work and skills; impact on society at large and democracy.

Table 28 maps questions within the environmental well-being topic area to this document's objectives.

Table 28: Objectives Comparison: Environmental Well-Being

ALTAI Question	Relevant Objective
Are there potential negative impacts of the AI system on the environment?	PLT1-2 (Assuming that "safe" includes environmental harm.)
Where possible, did you establish mechanisms to evaluate the environmental impact of the AI system's development, deployment and/or use (for example, the amount of energy used and carbon emissions)?	PLT1-2 (Assuming that "safe" includes environmental harm.)

Table 29 maps questions within the impact on work and skills topic area to this document's objectives.

Table 29: Objectives Comparison: Impact on Work and Skills

ALTAI Question	Relevant Objective
Does the AI system impact human work and work arrangements?	These questions focus on the societal (specifically, non-safety) impact of the system, or platform. This is intentionally outside the scope of this document.
Did you pave the way for the introduction of the AI system in your organisation by informing and consulting with impacted workers and their representatives (trade unions, (European) work councils) in advance?	
Did you adopt measures to ensure that the impacts of the AI system on human work are well understood?	
Could the AI system create the risk of de-skilling of the workforce?	
Does the system promote or require new (digital) skills?	

Table 30 maps questions within the impact on society at large and democracy topic area to this document's objectives.

Table 30: Objectives Comparison: Impact on Society at Large and Democracy

ALTAI Question	Relevant Objective
Could the AI system have a negative impact on society at large or democracy?	This question focuses on the societal (specifically, non-safety) impact of the system, or platform; this is intentionally outside the scope of this document.

H.7 ALTAI Requirement 7: Accountability

This requirement has the following topic groups: auditability; risk management.

Table 31 maps questions within the auditability topic area to this document's objectives.

Table 31: Objectives Comparison: Auditability

ALTAI Question	Relevant Objective
Did you establish mechanisms that facilitate the AI system's auditability (e.g. traceability of the development process, the sourcing of training data and the logging of the AI system's processes, outcomes, positive and negative impact)?	COM1-1, COM2-3, COM4-1, COM5-1, ARC2-2, ARC2-3, PLT1-6
Did you ensure that the AI system can be audited by independent third parties?	This document generates the necessary information; it says nothing about who can access that information.

Table 32 maps questions within the risk management topic area to this document's objectives.

Table 32: Objectives Comparison: Risk Management

ALTAI Question	Relevant Objective
Did you foresee any kind of external guidance or third-party auditing processes to oversee ethical concerns and accountability measures?	COM4-1, COM5-1 (Noting these point to external guidance; this document does not mention third-party auditing.)
Did you organise risk training and, if so, does this also inform about the potential legal framework applicable to the AI system?	Training for staff is intentionally outside the scope of this document, as are legal frameworks.
Did you consider establishing an AI ethics review board or a similar mechanism to discuss the overall accountability and ethics practices, including potential unclear grey areas?	Explicit consideration of ethics is intentionally outside the scope of this document.
Did you establish a process to discuss and continuously monitor and assess the AI system's adherence to this Assessment List for Trustworthy AI (ALTAI)?	This is ALTAI-specific; corresponding objectives would not be expected.
Did you establish a process for third parties (e.g. suppliers, end-users, subjects, distributors/vendors or workers) to report potential vulnerabilities, risks or biases in the AI system?	PLT1-6 (Noting that is about operational monitoring; this document does not mention reporting from third parties.)
For applications that can adversely affect individuals, have redress by design mechanisms been put in place?	This is a question about system employment, rather than AI-enabling technology.

H.8 Summary

The preceding tables indicate a different approach between ALTAI and this document. Typically, the former is more prescriptive, for example, highlighting red-teams, penetration testing and "stop buttons". Conversely, the latter, adopts a goal-based approach, highlighting general topics, for example, foreseeable misuse and abuse.

The preceding tables also indicate differences in scope. For example, ALTAI includes ethics and domain-specific certification, both of which are intentionally outside the scope of this document. Similarly, ALTAI has a focus on societal issues associated with the deployment of AI; these issues are intentionally not considered in this document.

Despite the differences in approach and scope, it is apparent that there is a reasonably complete mapping from this document's objectives to the ALTAI questions. In particular, this analysis has not highlighted any significant shortcomings in this document.

Appendix I References

[1] R. Ashmore and E. Lennon, Progress towards the assurance of non-traditional software, in Developments in System Safety Engineering, Proceedings of the Twenty-fifth Safety-Critical Systems Symposium, Safety-Citical Systems Club, 2017. ISBN 978-1540796288.

[2] D. Bedford, G. Morgan, and J. Austin, Requirements for a standard certifying the use of artificial neural networks in safety critical applications, in Proceedings of the international conference on artificial neural networks, 1996.

[3] J. Birch, Safety argument framework and considerations for highly automated vehicles. Personal Communication, September 2017.

[4] E. Breck, S. Cai, E. Nielsen, M. Salib, and D. Sculley, What's your ML test score? A rubric for ML production systems, in NIPS Workshop on Reliable Machine Learning in the Wild, 2016.

[5] S. Burton, L. Gauerhof, and C. Heinzemann, Making the case for safety of machine learning in highly automated driving, in International Conference on Computer Safety, Reliability, and Security, Springer, 2017, pp. 5–16.

[6] M. Douthwaite and T. Kelly, Safety-critical software and safety-critical artificial intelligence: Integrating new practices and new safety concerns for AI systems, in Evolution of System Safety, Proceedings of the Twenty-sixth Safety-Critical Systems Symposium, Safety-Critical Systems Club, 2018. ISBN 978-1979733618.

[7] J. M. Faria, Non-determinism and failure modes in machine learning, in 2017 IEEE International Symposium on Software Reliability Engineering Workshops (ISSREW), IEEE, 2017, pp. 310–316.

[8] R. Hawkins, Body of knowledge - structure and scope, tech. rep., Assuring Autonomy International Programme, April 2019.

[9] R. Hawkins, I. Habli, and T. Kelly, The principles of software safety assurance, 31st International System Safety Conference, Boston, Massachusetts USA, (2013).

[10] R. Hawkins, C. Paterson, C. Picardi, Y. Jia, R. Calinescu, and I. Habli, Guidance on the assurance of machine learning in autonomous systems (AMLAS), Tech. Rep. Version 1, Assuring Autonomy International Programme, February 2021.

[11] ISO, Road vehicles - functional safety, Tech. Rep. ISO 26262, ISO, 2011.

[12] C. Menon and R. Alexander, A safety-case approach to ethical considerations for autonomous vehicles, in Proceedings of the 12 IET International Conference on System Safety and Cyber Security, IET, 2017.

[13] Safety of Autonomous Systems Working Group, Safety Assurance Objectives for Autonomous Systems, Safety-Critical Systems Club, 2020. SCSC-153B.

[14] R. Salay and K. Czarnecki, Using machine learning safely in automotive software: An assessment and adaption of software process requirements in iso 26262, arXiv, 1808.01614 (2018).

[15] Underwriters Laboratories, The standard for safety for the evaluation of autonomous products, tech. rep., Underwriters Laboratories, Edge Case Research, October 2019.

[16] K. Wagstaff, Machine learning that matters, in Proceedings of the 29th International Conference on Machine Learning, 2012, 2012, pp. 529–536.

[17] Y. Yang, M. He, M. Li, Q. Wang, and B. Boehm, Phase distribution of software development effort, in Proceedings of the Second ACM-IEEE International Symposium on Empirical Software Engineering and Measurement, ESEM '08, New York, NY, USA, 2008, ACM, pp. 61–69.

This page is intentionally blank

Appendix J Contributors

SCSC-153B has had the benefit of contributions from a large number of people, who work for a variety of organisations, which collectively span a range of different sectors. This includes contributions to the comparisons. Note that contributions have been made on an individual basis and, in particular, the inclusion of an individual or organisation in the following list does not necessarily mean that individual or organisation agrees with the entire contents of the document.

Contributors to SCSC-153B include:

- Rasmus Adler, Fraunhof Institute (Kaiserslautern)
- Rob Alexander, University of York
- Hamid Asgari, Thales UK
- Rob Ashmore, Dstl
- Andrew Banks, LDRA
- Victor Bolbot, University of Strathclyde
- Rajiv Bongirwar, Hemraj Consultants
- Ben Bradshaw, ZF
- John Bragg, MBDA UK Ltd.
- John Clegg, Independent (ex-QinetiQ)
- Mike Ellims, Ricardo
- Jane Fenn, BAE Systems
- Hector Figueiredo, QinetiQ
- Gavin Gunny, University of York
- Chris Harper, University of the West of England
- David Harvey, Thales UK
- Rob Harwood-Smith, TP Group
- Richard Hawkins, University of York
- Nikita Johnson, University of York
- Shakir Laher, NHS Digital
- Catherine Menon, University of Hertfordshire
- Mike Parsons, CGI
- Davy Pissoort, KU Leuven
- Lavinia Pollock, AECOM
- Stewart Radcliffe, Thales UK
- Roger Rivett, Independent (ex-Jaguar Land Rover)

- Philippa Ryan, Adelard LLP
- Alan Simpson, EBENI
- Mark Sujan, Human Reliability
- Nick Tudor, D-RisQ
- Stuart Tushingham, Altran
- Bernard Twomey, Kongsberg
- Michael Wong, Codeplay

Contributors to the previous versions include:

- Rob Alexander, University of York
- Hamid Asgari, Thales UK
- Rob Ashmore, Dstl
- Andrew Banks, LDRA
- John Birch, Horiba-MIRA
- Rajiv Bongirwar, Hemraj Consultants
- Ben Bradshaw, ZF
- John Bragg, MBDA UK Ltd.
- Lavinia Burski, AECOM
- John Clegg, Independent (ex-QinetiQ)
- Timothy Coley, XPI Simulation
- Jane Fenn, BAE Systems
- Chris Harper, Atkins
- David Harvey, Thales UK
- Nikita Johnson, University of York
- Neil Lewis, Dyson
- Catherine Menon, University of Hertfordshire
- Ken Neal, Ebeni
- Ashley Price, Raytheon
- Stuart Reid, STA Consulting
- Roger Rivett, Jaguar Land Rover
- Philippa Ryan, Adelard LLP
- Alan Simpson, Ebeni

- Rod Steel, Thales
- Mark Sujan, Human Reliability
- Nick Tudor, D-RisQ
- Stuart Tushingham, Altran

This page is intentionally blank