UL 4600

STANDARD FOR SAFETY

Evaluation of Autonomous Products

UL Standard for Safety for Evaluation of Autonomous Products, UL 4600

Third Edition, Dated March 17, 2023

Summary of Topics

This New Edition of ANSI/UL 4600 dated March 17, 2023 includes the following changes in requirements:

- ***Revised safety case framework to support Autonomous Trucking***
- ***Revised and added examples specific to Autonomous Trucking***
- ***Added more precise definition for one preferred form of Safety Performance Indicator (SPI)***
- ***Added requirements concerning post-incident behaviors in Sections 10.6.6 and 10.6.7***

The new and revised requirements are substantially in accordance with Proposal(s) on this subject dated July 22, 2022 and January 20, 2023.

No Text on This Page

UL 4600

Standard for Evaluation of Autonomous Products

First Edition – April, 2020
Second Edition – March, 2022

Third Edition

March 17, 2023

This ANSI/UL Standard for Safety consists of the Third Edition.

The most recent designation of ANSI/UL 4600 as an American National Standard (ANSI) occurred on March 17, 2023. ANSI approval for a standard does not include the Cover Page, Transmittal Pages, and Title Page.

Comments or proposals for revisions on any part of the Standard may be submitted to ULSE at any time. Proposals should be submitted via a Proposal Request in ULSE's Collaborative Standards Development System (CSDS) at https://csds.ul.com.

No Text on This Page

CONTENTS

1 Preface (Informative)

1.1 Goal

1.1.1 This standard is intended to help ensure that an acceptably thorough consideration of safety for an AUTONOMOUS product has been performed during the design process and will continue to be done throughout the system lifecycle. It does so by emphasizing repeatable ASSESSMENT of the thoroughness of a SAFETY CASE.

1.1.2 CONFORMANCE with this standard is not a guarantee of a SAFE automated vehicle. However, CONFORMANCE with this standard promotes more rigorous engineering in support of a SAFE automated vehicle. It is also recognized that a SAFETY CASE is just one of many important parts to a complete safety assurance framework for automated vehicles, and it is expected that this standard will be used in conjunction with other standards and test methodologies defined by standards organizations and regulators.

1.2 Scope

1.2.1 The scope of this standard is a generalized AUTONOMOUS system standard framework for AUTONOMOUS road vehicles. A safety case is to be written around the AUTONOMOUS ITEM.

NOTE: Many of the prompts will apply to other AUTONOMOUS ground vehicles and even other types of AUTONOMOUS systems, but no specific attempt has been made to include extensive prompts for other applications.

1.2.2 The approach taken in this standard (UL 4600) is to require a CLAIM-based SAFETY CASE that encompasses essentially the entirety of the material necessary for safety assurance. The SAFETY CASE includes a structured set of CLAIMS, argument, and evidence supporting the proposition that an ITEM (a vehicle plus all other support contributing to safety) is acceptably SAFE for deployment. In support of that goal, UL 4600 ASSESSMENTS emphasize ensuring that the SAFETY CASE is reasonably complete and well formed. In particular, UL 4600 provides guidance to improve consistency and completeness of the SAFETY CASE. To this end, some best-practice process activities and granular work products are specifically required (e.g., creation of a hazard log). However, no specific overall design process is mandated, nor are there mandates for specific methods used to create the majority of work products (e.g., a V-style development process is not required; any reasonable approach used to create a list of hazards can be ACCEPTABLE).

1.2.3 This standard does NOT define a process, but rather puts forth ASSESSMENT criteria to determine the acceptability of a SAFETY CASE. As such, the ordering of sections, clauses, and PROMPT ELEMENTS does NOT imply temporal ordering or other process path dependencies.

1.3 Use of this standard with other standards

1.3.1 This standard is intended to work with existing standards to provide the additional elements necessary to assure that safety aspects of fully AUTONOMOUS ITEM operation have been considered in a comprehensive manner when creating a SAFETY CASE.

1.3.2 To the maximum extent practicable, it is intended that developers can take advantage of effort expended and ASSESSMENT credit gained for CONFORMANCE to other existing standards. Developers may incorporate materials into their SAFETY CASE generated as a result of executing processes and generating work products required by other standards.

1.3.3 It is the intent of this standard to be compatible with existing relevant safety standards to the maximum extent practicable, and in particular avoid prohibiting any activity or approach that is required by those standards. In particular, compatibility with ISO 26262:2018 and ISO 21448:2022 has been

considered. Annex A discusses a mapping of some clauses of this standard onto ISO 26262:2018 and ISO 21448:2022. Other safety standards such as IEC 61508 are relevant and expected to be generally compatible, but detailed analysis of IEC 61508 and other functional safety standards is out of scope for this version of UL 4600.

1.3.4 Two areas out of scope for this standard are setting ACCEPTABLE RISK levels and setting forth requirements for ethical product release decisions and any ethical aspects of product behavior. For both topics the developer records what decisions have been made, but this standard does not establish acceptance criteria beyond that they have been recorded. Other standards such as the IEEE P7000 series provide guidance on those topics.

2 Scope

2.1 Scope summary

2.1.1 This standard covers the safety principles, RISK mitigation, tools, techniques, and lifecycle processes for building and evaluating a SAFETY ARGUMENT for vehicles that can operate in an AUTONOMOUS mode, whether the ITEM is individual or part of a team such as a PLATOON.

2.1.2 Operation is assumed to occur without human supervision and without expectation of human intervention in performing and supervising the dynamic driving task and other normal system operations based upon the current ITEM state and ability to sense and otherwise interpret the operating environment. Human contributions to safety in other than normal operation are considered (e.g., maintenance), as are interactions with humans who are not operating the ITEM (e.g., pedestrians).

2.1.3 This standard generally uses the term “ITEM” rather than “system” or “product” when referring to the scope of the SAFETY CASE as well as the operation of the ITEM. This approach is in recognition of the possibility that the safety of the ITEM might rely upon infrastructure, services, support processes, and other factors that might not normally be considered part of a system such as a vehicle per se, but which materially affect its safety and therefore are all considered within the scope of the ITEM being assessed for CONFORMANCE. For a team of vehicles the ITEM might be scoped as an individual vehicle that is a member of a team or instead the ITEM might be designed to function as a team as a whole without reducing the safety of any one vehicle.

2.1.4 This standard assumes that the ITEM autonomously operates starting at some well-defined initial state to some other well-defined end state without human intervention. Human input might influence the selection of desirable states (e.g., via an occupant requesting a destination). However, the extent to which human operators MITIGATE or introduce RISK by performing or supervising a dynamic control task (e.g., by driving or taking responsibility for monitoring system operation) is outside the scope of the standard. Similarly, the extent to which human operator performance or non-performance is involved in RISKS related to transferring human driver control to or from the ITEM is also outside the scope of the standard. However, ensuring that the ITEM itself properly performs any change of control functions if and when it is supposed to is generally within the scope of the standard since it can adversely affect operation in fully AUTONOMOUS mode as well. Thus, while portions of this standard might be helpful for addressing less than fully AUTONOMOUS vehicles, issues involving human driver responsibilities, vigilance, and ability to properly accept responsibility for vehicle control are out of scope for this standard.

2.1.5 While information security is an essential topic, the details of that area are out of scope for this standard beyond a general requirement for a Security Plan and PROMPT ELEMENTS that are possibly unique to AUTONOMOUS vehicle operation in comparison to other vehicular security requirements. Reasonably foreseeable misuse and abuse as well as physical attacks (e.g., physical sensor damage) are in scope.

2.1.6 The requirements of this standard are considered to be at a necessary, but possibly not sufficient, level of completeness and rigor to create an acceptably well-formed and acceptably complete ITEM SAFETY

CASE. In particular, PROMPT ELEMENT lists are considered non-exhaustive, with an expectation that design teams will include additional ITEMS as relevant to the ITEM and its OPERATIONAL DESIGN DOMAIN.

2.2 Elements in scope

2.2.1 Specific aspects of ITEM operation and SAFETY RELATED issues which are explicitly intended to be in scope for this standard include:

a) Operation of AUTONOMOUS ITEMS in potentially unstructured environments

EXAMPLE: A vehicle is the first vehicle directed into an open farm field containing a mixture of viable and non-viable areas for traversal and parking as part of an ad hoc overflow event parking process. There are no lane markings and no positioning beacons. Moreover, there are cows and hay bales randomly placed in the field. There are no humans assisting with organizing vehicle parking positions. This situation, if within the ITEM ODD, is in scope for the standard.

EXAMPLE: An AUTONOMOUS truck operating on a paved median to navigate around a road blockage. A truck in this situation may find itself in an opposing direction traffic lane or an unstructured off-road environment while passing the obstacle and attempting to return back to a normal travel lane.

EXAMPLE: A crowd has spilled into the street at a fire scene. Emergency response equipment, response personnel, victims, and casual observers are moving without regard to normal road use patterns. Fire hoses, falling pieces of burning debris, small explosions, traffic signal power outages, damaged pavement, and other disruptions to normal infrastructure expectations exist. Multiple injured people at building exits are calling for pickup by AUTONOMOUS vehicle ride hail services to be transported to urgent care medical facilities. This situation is in scope for the standard.

NOTE: A particular ITEM's SAFETY CASE might require a structured environment for SAFE operation as specified by an OPERATIONAL DESIGN DOMAIN (ODD) description. However, structure is not assumed to be present by default. Therefore, operation in unstructured environments must be specifically disclaimed by the SAFETY CASE if applicable.

b) Operation with potentially inaccurate, incorrect, incomplete, or misleading data provided by sensors

c) The effects of potentially inaccurate, incorrect, incomplete, or biased data, including test data, field report data, other validation data and machine learning training data.

d) The effects of potentially imprecise, inaccurate, or incomplete simulation models.

e) Potential defects and failures of hardware and/or software in the ITEM, data collection functions, data processing functions, communications, engineering support systems, tools, and infrastructure support.

f) Human contributions to potential RISK, including occupants, pedestrians, other road users, non-road users, cargo handlers, maintainers and inspectors. This includes acts of omission and commission; accidental and malicious physical acts; and human roles in creating as well as mitigating RISK.

g) Lifecycle considerations, including design data collection, engineering data management, tool qualification, design, implementation, testing, other validation, field data collection, operations, maintenance, updates, upgrades, and retirement. Lifecycle considerations also encompass potential changes to the environment which may affect ODDs, changes in object types, changes in behaviors, etc.

h) Inclusion of RISK mitigation and other aspects of contributions to the SAFETY CASE made by CONFORMANCE with other standards, and in particular both ISO 26262 and ISO 21448 standards for products within scope for those standards.

i) Ability to use a heterogeneous approach to arguments, including use of diverse standards to support safety (e.g., use of different but ACCEPTABLE functional safety standards for different ITEM subsystems).

2.2.2 None of the described in-scope topics is intended to require that the ITEM successfully delivers full service in all situations described. Rather, the requirement is to consider all PROMPT ELEMENTS and ARGUE that RISK is ACCEPTABLE despite these factors. In many cases that will involve crafting an ODD that excludes problematic PROMPT ELEMENTS. However, excluding a PROMPT ELEMENT from the ODD (or similar approach) creates an obligation to ARGUE that the exclusion does not itself result in unacceptable RISK.

EXAMPLE: Unpaved roads without lane markings are excluded from the ODD. The SAFETY CASE generally ARGUES that geo-fencing and map creation will exclude all unpaved roads. It is further ARGUED that this exclusion encompasses quickly identifying roads undergoing repaving projects that are temporarily unmarked but still carrying traffic.

EXAMPLE: Snow is excluded from the ODD. Snow is still part of the SAFETY CASE to cover un-forecast snow that occurs during an operational mission. The SAFETY CASE generally ARGUES that it can successfully terminate a mission via in-lane stop despite snow. It further ARGUES (with evidence) that snow will happen so infrequently in the deployment location that the elevated product RISK presented by occasional in-lane stops is ACCEPTABLE.

2.3 Scope limitations

2.3.1 A significant scope limitation of this standard is that it does not cover the detailed topics relevant to ensuring that humans are able to provide effective safety supervision for an AUTONOMOUS ITEM. Rather, coverage is limited to fully AUTONOMOUS operation with no human supervision as well as aspects of the ITEM during human supervised operation that do not relate to arguing human supervision effectiveness. Similarly, aspects of the ability of a human operator to safely control the ITEM are out of scope. More specifically, the following are explicitly intended to be out of scope for this standard:

a) Aspects of ITEMS as well as ITEM-level safety of operational modes for which the locus of control is outside the ITEM itself.

EXAMPLES: End-to-end system-level safety (including the human supervisor or driver) of teleoperated modes of operation is excluded to the degree it relies upon a human teleoperator to control, supervise, or otherwise ensure system safety.

NOTE: A product-level "ITEM" is intended to include offboard functions that participate in control of a vehicle, such as a cloud-based route planning system.

NOTE: The proper response to teleoperated commands and proper transmission of teleoperation data out of the vehicle is in scope. However, whether teleoperation is actually SAFE at a system level is out of scope due to human involvement in the driving task.

b) Human factors related to safety during or after a handoff or mode switch that makes a human responsible for the dynamic control task safety.

EXAMPLE: The details of ensuring that a human supervisor is available and able to safely take over ITEM operation upon request and continue to operate the vehicle safely when some or all autonomy functions are disabled are out of scope.

c) RISK mitigation or other SAFETY ARGUMENT credit taken for the contribution of humans to the dynamic driving task (e.g., human driver, human safety supervisor, human teleoperator, issues of human alertness, issues of human situational awareness).

EXAMPLE: The details of how SAFE and effective human/machine interfaces should be provided to teleoperators are out of scope.

d) Road testing of prototype vehicles to the degree that SAFETY ARGUMENT takes credit for a human performing and/or supervising the dynamic driving task.

e) Specifics regarding how to ensure that humans acceptably meet expectations for non-driving roles in ITEM safety. To be clear, identifying the human contribution to the SAFETY CASE (e.g., via performing inspections, or a human's ability to correctly perceive and interpret signals provided by the ITEM) is in scope. However, specifying the details of how to actually ensure that the contribution is being done in an ACCEPTABLE manner in terms of human behaviors, psychology, limitations, and so on is out of scope. While competency frameworks, staff skill lists, and experience requirements are potentially helpful topics to cover a SAFETY CASE, specifics regarding these topics are out of scope for this standard.

f) Specifics regarding how to evaluate the suitability and effectiveness of human interface devices. To be clear, the need to IDENTIFY such devices and the need to ensure suitability and effectiveness is in scope, but specifying requirements for how to meet that need is out of scope.

EXAMPLE: A vehicle intended to, among other things, transport occupants in AUTONOMOUS operation is not supposed to transfer vehicle control to an occupant under any circumstance during a mission. It does in fact attempt to transfer vehicle control to an occupant, providing three seconds of warning.

In Scope for UL 4600: Vehicle attempting a handoff to a driver when it was not supposed to.

In Scope for UL 4600: Incorrectly attempting to transfer control in violation of fully AUTONOMOUS operational mode.

Out of Scope for UL 4600: Whether three seconds is enough warning for effective handoff and whether the occupant is a qualified driver.

EXAMPLE: An AUTONOMOUS vehicle is designed to transfer control to a human driver under some circumstances with a 10 second warning when the driver is qualified, competent, and aware of this handoff mission parameter.

In Scope for UL 4600: Transferring control without the full 10 seconds of warning; A defectively designed brake control mechanism that under some circumstances prevents the human driver from actually regaining control at the designated time (e.g., human driver's brake pedal disabled despite having attempted to perform a handoff); Whether brake pedal actuation by a human is intended to initiate a handoff under specified conditions.

Out of Scope for UL 4600: Whether ten seconds is enough warning for effective handoff in any particular handoff situation; Whether depression of a brake pedal by a human driver is a SAFE (from human factors point of view) handoff initiation mechanism; Checking whether occupant is a qualified driver (including having appropriate class of driver's license); Checking whether the driver has sufficient cognitive ability for SAFE operation; Checking whether human driver is in correct seating position to assume operational control; Safety of vehicle once control has been transferred to the human driver; Effectiveness of Advanced Driver Assistance (ADAS) functions in mitigating RISK while under human driver control.

2.3.2 There are a number of additional topics out of scope. Reference to these topics should be made where relevant to the SAFETY CASE, but specifics such as PROMPT ELEMENTS to provide technical depth are not included in this standard:

a) The specific intended function (e.g. surface cleaning, fragile cargo delivery)

NOTE: This topic might be covered by an end product standard.

b) End product requirements

NOTE: It is intended that end product standards can reference or require this standard.

c) Legal and policy issues

EXAMPLES: Determining liability, what records retention policies are appropriate, what level or product RISK is actually ACCEPTABLE to society.

d) Ethical issues

EXAMPLES: Resolving questions of ACCEPTABLE RISK, evaluating comparative severity of different LOSS event scenarios

e) Electric Vehicle safety

EXAMPLES: SAFE battery design, SAFE battery management algorithms, battery thermal management

f) General vehicle safety

EXAMPLES: Crash mitigation, occupant restraints, refueling/recharging safety

g) Non-SAFETY RELATED quality aspects of performing the intended function

EXAMPLES: Ride quality, fuel economy

h) Effectiveness of crash and injury mitigation mechanisms

EXAMPLES: Seat belts, air bags, child seats

2.3.3 Also out of scope for the standard is any implied redefinition of existing standards and accepted practices to the degree that they are ACCEPTABLE to support the SAFETY CASE being made. Reference to these topics should be made where relevant to the SAFETY CASE, but specifics are not included in this standard. These topics include:

a) Government regulations

EXAMPLES: FMVSS, Federal Communications Commission radio frequency interference emission certification

b) Mechanical aspects of the ITEM

EXAMPLES: Sharp edges, pinch points, window lift motor closing pressure limits

c) Legacy operational procedures

EXAMPLES: Car door operation, child locks on car doors, cargo loading and loading

d) Specifics of vehicle support for other than normal operations (except to the degree that the autonomy is expected to provide such support and that such provision is SAFETY RELATED)

EXAMPLES: Autonomy support for non-SAFETY RELATED routine maintenance procedures in situations which do not present a hazard

e) Sufficiency and performance of controls and ITEM response when a human is operating or supervising the dynamic driving task

EXAMPLE: As required by FMVSS

f) Licensing, training, qualification and other aspects of ensuring human competence to operate or participate in the vehicle lifecycle

NOTE: Credit can be taken for these in the SAFETY CASE only to the degree that any standard or procedure conferring or documenting qualification provides objective evidence of abilities

3 Referenced Publications

3.1 Normative references

3.1.1 The following referenced documents are indispensable for the application of this document:

N/A

3.2 Informative references

3.2.1 The following references included in this standard are for information only. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

California Consumer Privacy Act, AB-375 (CCPA)

Competence Criteria for Safety-Related System Practitioners; published by the Institution of Engineering and Technology

Def Stan 00-56, *Safety Management Requirements for Defence Systems*

EN 50129:2018, *Railway applications – Communication, signaling and processing systems – Safety related electronic systems for signaling*

FAA AC 25.1309-1A, *System Design and Analysis*

FAA FO 8040.4B, *Safety Risk Management Policy*

FAA FO 8110.49A, *Software Approval Guidelines*

IEC 61508-4, *Functional safety of electrical/electronic/programmable electronic safety-related systems – Part 4: Definitions and abbreviations*

IEEE 1012-2012, *System, Software, and Hardware Verification and Validation*

IEEE 24765, *Systems and Software Engineering – Vocabulary*

IEEE P7000, *Standard for Model Process for Addressing Ethical Concerns During System Design*

ISO/IEC 17024, *Conformity Assessment – General Requirements for Bodies Operating Certification of Persons*

ISO/IEC 17065, *Conformity Assessment – Requirements for Bodies Certifying Products, Processes, and Services*

ISO 26262:2018, *Road vehicles – Functional safety*

ISO 21448:2022, *Road vehicles – Safety of the intended functionality*

MIL-STD-882E, *Standard Practice System Safety*

Regulation (EU) 2016/679 (General Data Protection Regulation)

Reinventing Safety: A Joint Approach to Automated Driving Systems (Mercedes Benz & Bosch): https://www.daimler.com/documents/innovation/other/vssa-mercedes-benz-and-bosch.pdf

SAE J3016:2018, *Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles*

Safety First for Automated Driving; Published by eleven automotive and mobility industry leaders – a first-of-its-kind framework for SAFE automated passenger vehicles: https://newsroom.intel.com/news/intel-auto-industry-leaders-publish-new-automated-driving-safety-framework/#gs.drv3al

SCSC-153A, *Safety Assurance Objectives for Autonomous Systems* – Safety Critical Systems Club – UK;

Note: this document was written by the Safety of Autonomous Systems Working Group (SASWG), which is convened under the auspices of the SCSC, UK. 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 document intends to address safety due to Artificial Intelligence (AI) from Machine Learning only and is scheduled to be formally released at SSS'20 11-13 February 2020. Link to document: https://scsc.uk/SCSC-153A.

4 Terms, Definitions, and Document Usage

4.1 How to interpret normative elements (Normative)

4.1.1 Commonly used constructions of this standard affect the SAFETY CASE as follows. All elements are normative except "EXAMPLE," and "REFERENCE" statements as well as any other content that is explicitly stated to be informative. (See Table 4.1 below for a summary of key SAFETY CASE DEVIATION explanations.)

a) **Numbered clauses** (starting at 5.1.1) are generally stated as "shall" CONFORMANCE obligations. These are intended to be general statements, with supporting normative PROMPT ELEMENTS providing further detail. Each clause is specifically addressed in the SAFETY CASE with the exception of CONFORMANCE ASSESSMENT process clauses in Section 17 that deal with activities performed upon the SAFETY CASE itself. An important part of navigability of the SAFETY CASE is a capability to IDENTIFY the portion(s) of the SAFETY CASE that support fulfillment of each clause. The scope of all clauses is the SAFETY RELATED portion of the ITEM unless otherwise stated.

b) **MANDATORY PROMPT ELEMENTS:** Addressed by the SAFETY CASE. SAFETY CASE DEVIATIONS not permitted. Any SAFETY CASE DEVIATION results in a non-CONFORMANCE.

EXAMPLE: "IDENTIFY hazards" is mandatory – it must be done.

EXAMPLE: A team attempts to ARGUE that MANDATORY PROMPT ELEMENT X does not apply to their ITEM. This is an invalid attempt at a SAFETY CASE DEVIATION.

NOTE: In some cases, a MANDATORY PROMPT ELEMENT refers to consideration of a different clause in a hierarchical manner. That should be interpreted as a mandatory inclusion of the associated higher-level CLAIM in a SAFETY ARGUMENT, but not mandatory inclusion of all the non-mandatory PROMPT ELEMENTS of the clause being referred to. In particular, such hierarchical references are not intended to override the SAFETY CASE DEVIATION rules.

EXAMPLE: MANDATORY PROMPT ELEMENT X states that section Y is addressed by the SAFETY CASE. Section Y has a HIGHLY RECOMMENDED PROMPT ELEMENT Z. The net requirement is that satisfaction of all clauses in Section Y must be addressed by the SAFETY CASE, but a SAFETY CASE DEVIATION of PROMPT ELEMENT Z is still permitted in accordance with its HIGHLY RECOMMENDED categorization.

c) **REQUIRED PROMPT ELEMENTS:** Addressed by the SAFETY CASE. SAFETY CASE DEVIATION is permitted only if documented by argument that the PROMPT ELEMENT is intrinsically incompatible with the ITEM and/or its SAFETY CASE. Support for each SAFETY CASE DEVIATION is explicitly noted in the SAFETY CASE. End product standards can enumerate REQUIRED elements that can be omitted from the SAFETY CASE (i.e., blanket default SAFETY CASE DEVIATIONS specified by an end product standard). The SAFETY CASE is non-conformant if SAFETY CASE DEVIATIONS are not acceptably documented for REQUIRED elements. SAFETY CASE DEVIATION for a reason other than intrinsic inapplicability results in a non-CONFORMANCE. FIELD ENGINEERING FEEDBACK and change impact analysis are used to detect the possibility of a CLAIM of intrinsic incompatibility becoming invalid. Examples of ACCEPTABLE SAFETY CASE DEVIATIONS include:

EXAMPLE: SAFETY CASE DEVIATION for requirements on machine learning if ITEM does not use machine learning-based techniques in any manner, including design, operation, and field operational data analysis.

EXAMPLE: SAFETY CASE DEVIATION from recognizing road signs if ITEM does not rely upon road signs in any SAFETY RELATED way. Arguments supporting this might be that the ODD specifically excludes road signs, or that the ITEM exclusively uses a means other than road signs to gather equivalent information.

EXAMPLE: SAFETY CASE DEVIATION from requirements specific to subdivision of an ODD into multiple ODD SUBSETS if the ITEM does not define ODD SUBSETS (i.e., if an ITEM uses a single monolithic ODD, then any requirement related to ODD SUBSETS is not applicable).

EXAMPLE: Records of correction of defects discovered during ITEM operations if no ITEMS have yet been deployed and data does not yet exist because a potential event or condition has never occurred in the life of the ITEM. However, SAFETY CASE DEVIATION would not be permitted for the mechanisms and procedures to collect and process such potential events.

d) **HIGHLY RECOMMENDED PROMPT ELEMENTS:** These are best practices that should be followed, but may be omitted, especially for low RISK ITEMS. Omissions are explicitly noted in the SAFETY CASE with reasonable supporting argument to provide a hook for tracing root cause analysis back to those omissions. The SAFETY CASE is considered ACCEPTABLE so long as these omissions are noted with an ACCEPTABLE rationale. In cases in which a generic PROMPT ELEMENT of “others” is included, an omission rationale of “no others” is ACCEPTABLE. A primary purpose of FIELD ENGINEERING FEEDBACK is to ensure that a SAFETY CASE DEVIATION of any PROMPT ELEMENT that contributes substantively to SAFETY RELATED issues is identified and the deviation revoked.

EXAMPLE: The use of a specific analysis technique is HIGHLY RECOMMENDED. The SAFETY CASE notes that the technique was not used with a rationale of “other analysis techniques being used provide comparable information.” The SAFETY CASE includes an argument that there is no history of INCIDENTS with an unknown root cause that could plausibly have been prevented via addition of this technique to the design approach. This argument is backed up by root cause

analysis logs showing all root causes have been resolved, leaving no unresolved candidates that might in fact trace to the PROMPT ELEMENT deviation in the absence of other adverse information.

EXAMPLE: The rationale for a HIGHLY RECOMMENDED PROMPT ELEMENT deviation is "SAFETY CASE review meeting of 9/23/2019 determined this was inapplicable." While light on technical content, this is specific documentation that a deliberate process was said to be used to decide not to address a PROMPT ELEMENT. However, if root cause analysis was not performed to determine the applicability that might invalidate this rationale.

e) **RECOMMENDED PROMPT ELEMENTS:** These are optional PROMPT ELEMENTS documenting good practices and/or suggestions for helpful techniques. If adopted, they can be included in the SAFETY CASE. However, the SAFETY CASE is considered well-formed whether they are included or not. Omissions need not be noted in the SAFETY CASE.

f) **PITFALL PROMPT ELEMENTS:** These are anti-patterns, typically of the general form: "If X is true, then ITEM is prone to increased RISK of Y." The intended interpretation is that if X is true (e.g., use of some design pattern X or engineering technique X appears in the SAFETY CASE), then the SAFETY CASE is considered invalid unless epistemic defeater Y has been explicitly ARGUED to be false. In other words, Y presents ITEM RISK that is presumed to have been activated by X unless the SAFETY CASE presents reasonable argument and evidence of mitigation of RISK Y. More formally, the term "Pitfall" tags a conditional epistemic defeater prompt regarding defeasibility of a CLAIM that the parent requirement has been met. Responding to a Pitfall PROMPT ELEMENT might be accomplished in two ways. (1) Arguing that the precondition X is not true. This amounts to an ACCEPTABLE SAFETY CASE DEVIATION for the PROMPT ELEMENT. (2) Arguing that RISK due to post condition Y is mitigated when or if precondition X is or might be true. SAFETY CASE DEVIATION rules for Pitfalls apply according to the categorization (MANDATORY, REQUIRED, HIGHLY RECOMMENDED, or RECOMMENDED). A SAFETY CASE DEVIATION has occurred when there is no argument that the Pitfall has been avoided.

NOTE: For ASSESSMENT purposes each Pitfall only applies to the scope of the specific clause in which it is listed. However, it is a good practice (but optional) to consider while creating the SAFETY ARGUMENT that Pitfalls might have a larger impact.

g) **CONFORMANCE statements.** CONFORMANCE with each clause is evaluated via both self-assessment and independent ASSESSMENT according to Section 17. Each clause has a CONFORMANCE statement that provides guidance identifying portions of the SAFETY CASE and other information sources that are especially relevant to assessing CONFORMANCE to that clause. ASSESSORS are permitted to consider objective evidence beyond the CONFORMANCE statement when the ASSESSOR determines that the situation warrants, but are limited in CONFORMANCE determination by the written scope of the clause. (Both self-ASSESSORS and independent ASSESSORS can and should consider whether PROMPT ELEMENTS beyond those included in this standard need to be added for a particular ITEM's SAFETY CASE.) CONFORMANCE checks are performed by someone other than the developer of the ITEM being checked for CONFORMANCE (See Section 17). (There are limited exceptions for SAFETY CASE artifacts being self-assessed; see Section 17.2.2.)

h) **NOTE statements.** These are notes on how to interpret the normative elements of a section and in some cases provide rationale material.

i) **EXAMPLE lists (non-normative).** In some cases examples are provided. Examples are non-normative and, in many cases, will not apply to all ITEMS. Their primary purpose is to define by example to reduce potential ambiguity. It is expected and required that construction and ASSESSMENT of the SAFETY CASE go beyond the bounds of any examples. Exclusion of examples in a SAFETY CASE does not by itself result in a finding of non-CONFORMANCE.

j) **REFERENCE (non-normative).** References provide citations to materials such as other standards. They are non-normative by default.

4.1.2 Summary of SAFETY CASE DEVIATION approaches for elements of different types is shown in Table 4.1:

Table 4.1
Safety Case Deviation Approach

MANDATORY	No SAFETY CASE DEVIATIONS permitted.
REQUIRED	SAFETY CASE DEVIATIONS only for requirements that are intrinsically inapplicable due to the fundamental nature of the ITEM and/or the current deployment state of the ITEM. All SAFETY CASE DEVIATIONS recorded in SAFETY CASE with justification. Impact analysis and lifecycle tracking monitor the possibility of a change of applicability status.
HIGHLY RECOMMENDED	SAFETY CASE DEVIATIONS permitted with an ACCEPTABLE rationale. Impact analysis and lifecycle tracking monitor the possibility of a change of applicability status. All SAFETY CASE DEVIATIONS recorded in SAFETY CASE with justification.
RECOMMENDED	Optional ITEMS. Need not be mentioned by SAFETY CASE. No argument support required for SAFETY CASE DEVIATION.

4.1.3 Lists that support a particular clause, such as a list of MANDATORY or REQUIRED ITEMS, are interpreted in the following manner:

a) The context of each ITEM in a list is the overarching clause, even if not explicitly stated. This means lists generally provide a set of prompts for brevity and usability rather than fully stated "shall" statements.

b) There is no implied ranking, preference, or priority implied by list order unless explicitly stated.

c) **All PROMPT ELEMENT lists are presumed to be potentially incomplete unless specifically stated otherwise. A well-formed and complete SAFETY CASE extends lists as needed to achieve an ACCEPTABLE safety outcome.** Unless otherwise stated, PROMPT ELEMENT lists are to be considered as a minimum set of points to be considered in the SAFETY CASE, and not an exhaustive set of all possible points.

d) If different PROMPT ELEMENT lists appear to overlap, it is ACCEPTABLE to trace a single argument or evidence branch to multiple PROMPT ELEMENTS if desired. A common situation is that a MANDATORY PROMPT ELEMENT will give a generalized PROMPT ELEMENT, whereas one or more REQUIRED PROMPT ELEMENTS will give specific examples of that generalized element to ensure that they are considered if applicable.

e) If a single PROMPT ELEMENT list contains apparently overlapping ITEMS, it is sufficient to trace to whichever PROMPT ELEMENT seems most applicable to the developers. Such overlapped lists are often used to address issues of potentially different interpretations and terminology across domains and sub-domains.

f) Each PROMPT ELEMENT in the list is accounted for at the stated level of SAFETY CASE DEVIATION rigor on a per-element basis. Therefore, if SAFETY CASE DEVIATION is desired for a ten-element list for a single REQUIRED section, the SAFETY CASE documents SAFETY CASE DEVIATION in a way that is individually traceable to each of the ten PROMPT ELEMENTS in the list. (A single justification statement for SAFETY CASE DEVIATION might trace to some or all of the ten elements, but that traceability must be documented.)

g) "At least one of" phrasing contains a sub-list of alternatives. Only one alternative within the list need be addressed in the SAFETY CASE, although others might additionally be addressed if desired. This is true even if the "at least one of" occurs in a MANDATORY or REQUIRED list. Note that in some cases two or more PROMPT ELEMENTS might be required in practice due to Pitfall statements. Alternatives not listed can be considered to meet the "at least one of" criterion with suitable

argument support as to acceptability. (This is according to the principle that PROMPT ELEMENT lists are not considered to be exhaustive, and therefore SAFETY CASES can add PROMPT ELEMENTS as appropriate to the local version of those PROMPT ELEMENT lists.)

h) Tailoring of list ITEMS must adhere to SAFETY CASE DEVIATION rules summarized in Table 4.1 above.

4.1.4 For simplicity and uniformity, references to sections, clauses, and PROMPT ELEMENTS are made using a decimalized notation, notwithstanding the typographical conventions used in numbering the sections themselves.

EXAMPLE: 9.2.3.3.b.1.i is a correctly formatted reference to a hypothetical PROMPT ELEMENT sub-sub-list in the HIGHLY RECOMMENDED category of Section 9.2.3.

4.1.5 See also 2.1.6 regarding scope of prompt lists.

4.2 Terms and definitions (Normative)

4.2.1 **ACCEPTABLE**
sufficient to achieve the overall ITEM RISK as determined in the SAFETY CASE

Example: "ACCEPTABLE test coverage" means that the amount of test coverage is sufficient to support the SAFETY CASE's CLAIM of overall ITEM RISK after taking claimed RISK mitigation credit.

NOTE: This is an objective term related to the validity and completeness of the SAFETY CASE, and not a subjective term related to any particular ASSESSOR's personal point of view.

NOTE: This definition differs from MIL-STD-882E in that UL 4600 does not define an acceptance authority.

See: Section 6.4.3

4.2.2 **ACTIVATION (of faults or hazards)**
input or situation that causes the system to potentially fail due to a fault or hazard

EXAMPLE: A bit in memory corrupted by a single event upset is a fault. That fault is activated when the memory location is read and results in a computational error that causes a failure in the form of an incorrect or unsafe ITEM behavior.

NOTE: Fault mitigation such as error detection coding, among other mitigations, can prevent an activated fault from causing a failure.

4.2.3 **AI TECHNIQUES**
computational algorithms and other techniques that generally include inductive learning, intentionally non-deterministic behavior, rule-based systems, computer vision, heuristic searches, and other techniques that are often referred to as "Artificial Intelligence" techniques

NOTE: This term is meant to be a broadly interpreted descriptive term to encompass software that is not generally amenable to software safety integrity approaches applicable to deterministic software implemented with an imperative programming language approach. Whether or not actual "intelligence" is involved is a matter beyond the scope of this standard.

4.2.4 **ARGUE**
reference a subset of a SAFETY CASE

NOTE: This term does not constrain notation used to construct the SAFETY CASE.

4.2.5 **ASSESSMENT**
review and evaluation of the SAFETY CASE to determine whether the argument that RISK will be ACCEPTABLE is both valid and sound

NOTE: A valid argument fully supports its conclusion. A sound argument is one in which the premises of the argument are actually true.

NOTE: This definition differs in scope from the definition in ISO 26262:2018.

4.2.6 **ASSESSOR**
one or more people who perform and are responsible for the outcome of an ASSESSMENT

4.2.7 **AUTONOMOUS**
operates without human oversight or intervention

NOTE: Usage of the terms “automated” and “AUTONOMOUS” are not uniformly established across domains. In some domains and standards, such as SAE J3016, the terms “automated” or “highly automated” are preferred to the term “AUTONOMOUS.” The use of the term “AUTONOMOUS” is not intended to preclude use of the term “automated” in a SAFETY CASE.

4.2.8 **CHAOTIC (of a system)**
sensitive to initial conditions to the point of exhibiting apparently nondeterministic behavior

EXAMPLE: A robot that is pointed exactly at the center of mass of an obstacle using a completely deterministic algorithmic approach might decide on an apparently random basis to go around the obstacle to the left or right depending on minute variations in conditions beyond the capability of a tester to accurately control, making it impossible in practice to predetermine which action it will take.

4.2.9 **CLAIM**
falsifiable statement intended to contribute to establishing that post-mitigation RISK is ACCEPTABLE

NOTE: This term does not constrain notation used to construct the SAFETY CASE. For example, in GSN this would be a “goal.”

4.2.10 **CONFORMANCE**
independent ASSESSMENT result indicating that the ITEM SAFETY CASE meets the requirements of UL 4600 (See Section 17.1)

4.2.11 **CRITICALITY LEVEL**
level categorizing the RISK associated with an unmitigated hazard

NOTE: CRITICALITY LEVEL is a generic term that encompasses integrity level approaches and assurance level approaches such as SIL, DAL, and ASIL.

4.2.12 **DEMONSTRATION**
operation of the ITEM, functions, or elements to show that a specific property, functionality, or other aspect of the ITEM matches the SAFETY CASE

4.2.13 **DOER/CHECKER**
heterogeneous architectural design pattern in which a Doer FAULT CONTAINMENT REGION (FCR) performs a function while a second Checker FCR performs acceptance tests on the Doer

4.2.14 **ELEMENT OUT OF CONTEXT (EOOC)**
stand-alone element that is the subject of a SAFETY CASE fragment and CONFORMANCE ASSESSMENT

See Section 5.7.3.

NOTE: This is analogous to the term “SEooC” defined in ISO 26262:2018.

4.2.15 **FAULT CONTAINMENT REGION (FCR)**
collection of one or more elements that operates correctly regardless of any fault outside the region considering the relevant fault model

NOTE: Two FCRS are required to ensure fault detection and/or fault mitigation in the presence of an arbitrary FCR failure.

4.2.16 **FIELD ENGINEERING FEEDBACK**
acquisition of data from system operation for the purpose of supporting the SAFETY CASE and identifying potential SAFETY CASE issues

NOTE: This term can apply to validation data before deployment, e.g., from road testing.

4.2.17 **IDENTIFY**
create an enumerated list responsive to a specified category, property, or other aspect of a clause with sufficient specificity to enable ASSESSMENT
(Alternate word form: IDENTIFICATION)

4.2.18 **INCIDENT**
occurrence of a safety-related failure that could result in a LOSS event

NOTE: An INCIDENT does not necessarily result in a LOSS event. It is sufficient that in other circumstances an INCIDENT might possibly have resulted in a LOSS event.

EXAMPLE: A car fails to stop at a stop sign. There is no cross traffic, so no collision results. If cross traffic had been present, a collision could have occurred. This is an INCIDENT even though no LOSS event occurred.

NOTE: All LOSS events are also INCIDENTS. Therefore, the phrase “INCIDENTS” is equivalent to “INCIDENTS and LOSS events.”

4.2.19 **INDEPENDENCE**
absence of dependent failure between two or more elements that could lead to the violation of a safety requirement or an organizational separation of the parties performing an action (Source: ISO 26262:2018)

NOTE: Two or more FAULT CONTAINMENT REGIONS (FCRS) fail independently if they have no correlated, common cause or common mode fault conditions.

NOTE: See also 17.3.2 for additional organizational INDEPENDENCE considerations.

4.2.20 **ITEM**
product, element, system of systems, or other product-related scope for which CONFORMANCE to UL 4600 is assessed

NOTE: The "ITEM" may need to include infrastructure, offboard computing, offboard data storage, development processes, lifecycle support processes, supply chain quality assurance measures, and other aspects of ensuring safety beyond the boundaries of a deployed product itself. The "ITEM" can include an entire product, or only portions of a product, but in any event will include all safety-related product aspects required for CONFORMANCE.

4.2.21 **LIFE CRITICAL**
capable of initiating or preventing LOSS events that could include LOSS of human life

NOTE: This is strictly a severity concept. A hazard can be LIFE CRITICAL even if developers ARGUE that the associated RISK is low due to an improbability of occurrence in operation.

4.2.22 **LOSS**
a substantive adverse outcome, from damage to property or the environment, to animal injury or death, to human injury or death

NOTE: Which losses are considered adverse is system specific. However, human death and significant human injury are always to be considered to be substantive adverse losses.

NOTE: SAFETY CASES might elect to consider a financial cost caused by an ITEM failure to also be a substantive adverse outcome.

NOTE: "LOSS event" generally corresponds to the term "accident" in FAA Order 8040.4B and DefStan 00-56. However, the term "LOSS event" is used to avoid any preconceptions regarding liability and foreseeability.

4.2.23 **MINIMUM EQUIPMENT LIST (MEL)**
minimum ACCEPTABLE set of equipment, functionality, capabilities and other relevant requisites for a specified operational mode

NOTE: An MEL is associated with each operational mode, including both normal and degraded operational modes.

NOTE: Requisites can be associated with operation, field modifications, software updates, inspections, maintenance, configuration management, or other relevant aspects of the ITEM's status.

4.2.24 **MITIGATE**
reduce from an unacceptable level of RISK to an ACCEPTABLE level of RISK, or to judge that a recognized level of RISK is ACCEPTABLE without additional RISK reduction activity

NOTE: The level of ACCEPTABLE RISK depends upon the criticality of the specific RISK being mitigated (e.g., life-critical vs. not LIFE CRITICAL) and its contribution to ITEM-level RISK as established in the SAFETY CASE. A hazard that presents acceptably low RISK without overt mitigation action is still considered "mitigated" so long as it is ARGUED that the RISK presented is acceptably low. The word "mitigated" is equivalent to the concept of "acceptably mitigated." (See definition of "ACCEPTABLE.") A RISK is not considered mitigated until the mitigation approach, if any, is implemented and associated portions of the SAFETY CASE regarding mitigation have been updated ("tracked to closure"). An "accepted RISK" is one for which partial (or no) mitigation has been performed (See Section 5.5.1)

4.2.25 **NONDETERMINISM**
phenomena that are not repeatable

NOTE: NONDETERMINISM might be caused by real time scheduling perturbations, use of pseudo-random algorithms lacking deterministic seeding, or other factors that result in a lack of practicable repeatability

4.2.26 **NON-DEVELOPMENTAL ITEM (NDI)**
component or function not developed as part of the current design effort

NOTE: This is a generic descriptive term inclusive of commercial off-the-shelf (COTS) components, legacy components, commercial data services, and in general anything that is not created as part of or in support of the design process for the ITEM described by the SAFETY CASE.

4.2.27 **ODD SUBSET**
a defined portion of an ODD

EXAMPLE: A particular all-weather ODD is broken up into subsets for fair weather, rain, snow, and ice.

NOTE: An ODD SUBSET might be defined to partition the operational space to ease design tasks, support phased deployment by adding additional subsets over time, or otherwise manage the complexity of a potentially large and varied ODD. The SAFETY CASE might ARGUE each ODD SUBSET independently for some aspects of the SAFETY CASE.

4.2.28 **OPERATIONAL DESIGN DOMAIN (ODD)**
set of environments and situations the ITEM is intended to operate within, including direct environmental conditions, geographic restrictions, and a characterization of the set of objects, events, and other conditions that will occur within that environment

NOTE: A system has a single ODD by definition. ASSESSMENT is made with regard to the entire ODD.

NOTE: This definition is compatible with, but expanded in scope compared to that given in J3016:2018.

See also: ODD SUBSET

4.2.29 **PLATOON**
collection of vehicles that travel together, actively coordinated in formation[a]

[a] Bergenhem, C., Shladover, S., Coelingh, E., Englund, C., and Tsugawa, S., "Overview of platooning systems," 2012

4.2.30 **PROMPT ELEMENT**
an individual prompt associated with a clause

NOTE: PROMPT ELEMENTS provide supporting detail for clauses in this standard.

4.2.31 **PROVENANCE**
traceable background of components, materials, supplies, software, and other aspects of the ITEM, especially as conferring distinction or quality

NOTE: In the context of UL 4600, this refers to establishing that COTS products and their components actually provide the required capabilities, and specifically excluding inferior, counterfeit, and other "unapproved" parts. This is different than variations in parts that provide ACCEPTABLE capabilities and other versioning activities.

4.2.32 **QUALITY FADE**
supply chain fault in which a component supplier degrades component quality over time via progressive use of substitute materials, substitute components, design changes, and/or elimination of protective components while apparently maintaining functionality and meeting tested parameter values

4.2.33 **RISK**
combination of the probability of occurrence of a LOSS event and the severity of that LOSS event

NOTE: In some circumstances RISK might be tied solely to severity, essentially assuming a probability of 1.

4.2.34 **ROBUST**
able to continue operation despite situations that are out of specification

EXAMPLES: A system that continues normal or degraded operation despite receiving out of specification inputs, encountering ODD violations, suffering component faults, and/or encountering data not represented in machine learning training sets could be considered to be ROBUST.

NOTE: ROBUSTNESS is often a matter of degree rather than an absolute property.

4.2.35 **SAFE**
having an ACCEPTABLE post-mitigation RISK at the ITEM level as defined by the SAFETY CASE

See Section 6.4.3.

4.2.36 **SAFETY ARGUMENT**
structured argument supporting safety CLAIMS

NOTE: An atomic argument is a single argument step in a SAFETY ARGUMENT, supporting a single conclusion directly from one or more premises. A compound argument comprises one or more atomic arguments that are combined, with the conclusion of some arguments serving as premises of others. An argument module comprises one or more compound argument on a single topic or related topics. Argument is used as a mass noun to refer to any collection of atomic argument, compound arguments, or argument modules, or the concept of an argument generally.

NOTE: A SAFETY ARGUMENT might be presented as a document or as electronic records accessed through a software tool. Arguments may reference other materials that are readily accessible to the ASSESSOR, e.g., plan documents, specifications, reports, or data in repositories.

4.2.37 **SAFETY CASE**
structured argument, supported by a body of evidence, that provides a compelling, comprehensible and valid case that a system is SAFE for a given application in a given environment

Source: Defence Standard 00-56 Issue 7 (Part 1): Safety Management Requirements for Defence Systems. UK Ministry of Defence. p. 26.

NOTE: In UL 4600, the use of the terms “argument” and “evidence” are generic, and do not preclude the use of SAFETY CASE notation or approaches that use a different strategy for SAFETY CASE construction [see 5.2.1.3.a].

NOTE: This term is broader in scope than, but compatible with, the use of “SAFETY CASE” in ISO 26262:2018.

4.2.38 **SAFETY CASE DEVIATION**
recorded exclusion of a PROMPT ELEMENT from consideration in accordance with the requirements of Section 4.1

4.2.39 **SAFETY PERFORMANCE INDICATOR (SPI)**
metric used to quantify safety performance

NOTE: This term is analogous to the term Key Performance Indicator (KPI), but is specific to SAFETY RELATED aspects of the ITEM.

4.2.40 **SAFETY RELATED**
directly or indirectly affecting safety

NOTE: This term is used in a way that is compatible with the ISO 26262:2018 definition of the term "safety-related function."

NOTE: This term is used in a substantially different way than MIL-STD-882E, which uses the term "safety-related function."

4.2.41 **STATUS TEST**
off-line test to detect latent faults

NOTE: A STATUS TEST is an extension of the concept of a proof test, which historically referred to mechanical testing that cannot be performed during normal system operation, such as ensuring pressure vessel integrity and the ability of emergency valves to actuate. STATUS TESTS can involve exercising failsafes, exercising sensors that are used to detect failures, executing an off-line Built-In Self-Test capability, inspecting equipment by a human maintenance technician, or otherwise performing activities to ensure ACCEPTABLE equipment status that require the system to be in a state of other-than-normal service. IEC 61508 requires proof tests, often used together with repair, to detect dangerous hidden failures and restore the system to the required MEL.

GENERAL TERMINOLOGY REFERENCES (INFORMATIVE):

• Avizienis, A., Laprie, J-C., Randell, B., Landwehr, C., "Basic Concepts and Taxonomy of Dependable and Secure Computing," IEEE Trans. Dependable and Secure Computing, 1(1), Jan.-Mar. 2004, pp. 1-23

• ISO 26262-1:2018

• ISO/IEC 15026:2011

• ISO 21448:2022

• MIL-STD-882E

• SAE J3016:2018

4.3 Abbreviations and Acronyms (Informative)

a) AD: Automated Driving

b) AI: Artificial Intelligence

c) ASIL: Automotive Safety Integrity Level

d) BIST: Built-In Self-Test

e) COTS: Commercial Off The Shelf

f) DAL: Design Assurance Level

g) DTC: Diagnostic Trouble Code

h) ECU: Electronic Control Unit

i) EOOC: ELEMENT OUT OF CONTEXT

j) FCR: FAULT CONTAINMENT REGION

k) FMVSS: Federal Motor Vehicle Safety Standards (USA)

l) GNSS: Global Navigation Satellite System

m) GPU: Graphics Processing Unit

n) GSN: Goal Structuring Notation

o) HARA: Hazard Analysis and RISK ASSESSMENT

p) ISO: International Standards Organization

q) MEL: MINIMUM EQUIPMENT LIST

r) NDI: NON-DEVELOPMENT ITEM

s) ODD: OPERATIONAL DESIGN DOMAIN

t) PSSA: Preliminary System Safety ASSESSMENT

u) SEBOK: System Engineering Body of Knowledge

v) SIL: Safety Integrity Level

w) SOUP: Software of Unknown PROVENANCE / Systems of Unknown PROVENANCE

x) SPI: SAFETY PERFORMANCE INDICATOR

y) SQA: Software Quality Assurance

z) SWEBOK: SoftWare Engineering Body of Knowledge

aa) V&V: Verification and Validation

bb) V2I: Vehicle-to-infrastructure

cc) V2V: Vehicle-to-vehicle

dd) V2X: Vehicle-to-other

5 Safety Case and Arguments

5.1 General

5.1.1 **The SAFETY CASE shall be a structured explanation in the form of CLAIMS, supported by argument and evidence, that justifies that the ITEM is acceptably SAFE for a defined OPERATIONAL DESIGN DOMAIN, and covers the ITEM's lifecycle.**

5.1.1.1 **MANDATORY:**

a) CONFORMANCE is demonstrated based upon a documented SAFETY CASE that:

1) Uses an ACCEPTABLE SAFETY CASE format (see Section 5.2)

2) Presents an acceptably complete argument arguing that the provided evidence supports the defined CLAIMS (see Section 5.3)

3) Presents ACCEPTABLE evidence (see Section 5.4)

4) Addresses accepted RISKS (see Section 5.5)

5) Addresses safety culture (see Section 5.6)

b) Configuration management of the SAFETY CASE (see Section 9.1.4)

c) Change management for the SAFETY CASE

5.1.1.2 **REQUIRED:**

a) Addresses ELEMENTS OUT OF CONTEXT (EOOC) (see Section 5.7.3)

b) *Deleted*

c) *Deleted*

d) Except to the extent permitted by Section 5.7.3.2.b.4., the SAFETY CASE encompasses all SAFETY RELATED aspects of the ITEM and its lifecycle including both bespoke and off-the-shelf components, software, and subsystems. This specifically includes, but is not limited to:

1) Sensors

2) Actuators

3) Computing components

EXAMPLES: Computing hardware, operating systems, libraries

4) Vehicle platforms used as a basis for adding “autonomy kits”

EXAMPLE: ASSESSMENT of an application of an autonomy kit to a vehicle platform includes the vehicle platform in its scope. One way to avoid gaps in coverage of vehicle properties is for the vehicle platform vendor to provide an EOOC interface for the vehicle platform.

5) On-line services

EXAMPLE: On-line map server provided from a cloud infrastructure to an operational vehicle on an as-needed basis

6) Logistical and maintenance support

7) Assumed infrastructure support

EXAMPLE: Road lane markings

NOTE: EOOC SAFETY CASES are permitted omissions if documented by the EOOC interface (see Section 5.7.3.2.b.4).

NOTE: For EOOC SAFETY CASES, the concept of an ITEM is interpreted to be the scope of the EOOC in light of scope limitations identified by the EOOC interface.

e) Any aspect of a purported SAFETY CASE that is not documented and/or written down, and is not provided to the ASSESSOR upon request, cannot be used in supporting a determination of CONFORMANCE.

NOTE: Verbal statements and other materials not part of the SAFETY CASE may be considered by the ASSESSOR to help with the ASSESSMENT process, but cannot be used as supporting argument and/or evidence in evaluation of the SAFETY CASE.

EXAMPLE: A developer verbally explains why a particular REQUIRED PROMPT ELEMENT is inapplicable, but that explanation is not in the actual SAFETY CASE, nor in any documentation referenced by the SAFETY CASE. The lack of documented explanation for SAFETY CASE DEVIATION of that REQUIRED PROMPT ELEMENT would cause a finding of non-CONFORMANCE even though the ASSESSOR might consider the verbal explanation reasonable. The non-CONFORMANCE finding could be remedied if the SAFETY CASE is updated, but it is not the ASSESSOR's responsibility to do so.

EXAMPLE: A tester verbally states that tests were performed on a particular configuration that is being deployed. However, documentation of the configuration that was actually tested has been lost and cannot be reconstructed with reasonable certainty. To the extent that documentation of the configuration tested is necessary for a well-formed SAFETY CASE, the ASSESSOR finds non-CONFORMANCE due to the missing documentation, and potentially the tests are re-run.

f) IDENTIFY any initial portion of lifecycle during which the SAFETY CASE is not intended to be valid

EXAMPLE: The SAFETY CASE does not apply until the time of the first public road testing within the lifecycle.

NOTE: This permits completing the SAFETY CASE in parallel with prototype development and potentially closed-course development testing. It is not intended to permit a CLAIM of CONFORMANCE for any vehicle operating on public roads at any time in its lifecycle.

g) *Deleted*

5.1.1.3 **HIGHLY RECOMMENDED – N/A**

5.1.1.4 **RECOMMENDED:**

a) IDENTIFY the audiences for the SAFETY CASE and the purposes for which it will be used.

EXAMPLE: ASSESSORS assess CONFORMANCE to UL 4600. Developers use the SAFETY CASE to record details of hazards, RISKS, mitigations, safety requirements, standards CONFORMANCE, and safety-relevant aspects of the ITEM, the process of developing it, and the chosen verification and validation activities. Current and future developers use the SAFETY CASE as an introduction and reference to these things, and to understand the potential safety implications revealed by INCIDENTS, LOSS events, discovered vulnerabilities, and analysis of collected data.

b) Where the SAFETY ARGUMENT is presented in distinct sections, IDENTIFY the role of each section of the SAFETY ARGUMENT.

EXAMPLE: The Top-Level section of the SAFETY ARGUMENT explains to a broad audience the overall safety concept, key hazards and mitigations, and key features of the engineering approach, referring readers to other SAFETY ARGUMENT sections and documents for details. The Hazards and Mitigations section serves as the primary documentation for mitigations, presenting each hazard and explaining its particular mitigations, identifying the safety requirements for those mitigations.

5.1.1.5 **CONFORMANCE:**

CONFORMANCE is checked via consideration of subsection CONFORMANCE checks, including traceable inclusion of each applicable PROMPT ELEMENT in the SAFETY CASE.

5.1.1.6.1 **NOTE:** Additional measures beyond CONFORMANCE to this standard are likely to be required to ensure operational safety. As an example, an end product standard might require CONFORMANCE not only to this standard, but to other standards involving topics such as electrical safety, fire safety, and passive occupant protection, among others.

5.1.1.6.2 **NOTE:** Access to the SAFETY CASE for ASSESSMENT purposes is discussed in Section 17 (see 17.1.1.1.f and 17.3.1.2.c.1).

5.1.1.6.3 **NOTE:** This clause is explicitly intended to require that the CONFORMANCE documentation package is in the form of a SAFETY CASE (i.e., everything other than administrative matters dealing with the ASSESSMENT itself is a part of the SAFETY CASE). As a practical matter there may be a variety of documents, repositories, and tools used to provide details such as evidence that make it impractical to have the entire SAFETY CASE in a single tool or uniformly formatted data set. However, ASSESSMENT requires ability to access any such material to perform an ASSESSMENT.

5.1.1.6.4 **NOTE:** In general, this section places requirements upon the structure and content of the SAFETY ARGUMENT. Other major sections generally place requirements upon determining the completeness of the SAFETY CASE.

REFERENCE: MISRA "Guidelines for automotive safety arguments," 2019.

5.2 Safety Case style and format

5.2.1 **The SAFETY CASE shall consistently use a limited number of defined formats.**

5.2.1.1 **MANDATORY:**

a) Definition of CLAIM and argument syntax, semantics, and any graphical elements used

NOTE: Notation might not have formally defined semantics. However, the best available information about the SAFETY CASE notation and approach is provided to facilitate interpretation that is as uniform as practical across the developer team and ASSESSORS.

b) Definition of evidence types, formats, data dictionaries, and schemas used (see 5.2.2)

c) *Deleted*

1) *Deleted*

5.2.1.2 **REQUIRED:**

a) Adherence to defined formats within SAFETY CASE

b) ASSESSOR access to any available browsing, searching, reporting, and analysis tools relevant to understanding the SAFETY CASE

NOTE: As a practical matter there are likely to be tools and other support used by the developers and self-ASSESSORS to make it easier to work with the SAFETY CASE. Those same tools and other support are made available upon request.

c) IDENTIFY reasoning approach regarding completeness of inductive elements of the SAFETY ARGUMENT

5.2.1.3 **HIGHLY RECOMMENDED:**

a) Use of an established SAFETY ARGUMENT notation, such as:

1) OMG Structured Assurance Case Metamodel (SACM)

2) Goal Structuring Notation (GSN)

3) Claims Argument Evidence (CAE)

4) Notations embodying established, relevant philosophical theories of argumentation

EXAMPLES: Those of Stephen Toulmin or Trudy Govier

b) Use of tool support to aid in SAFETY CASE comprehension and navigation

c) Use of the minimum feasible number of argument notations, such as:

1) Use of a single format within a portion of an argument (local homogeneity)

2) Use the same format for similar argument content when practical (global consistency of similar content)

NOTE: It can make sense to use multiple argument notations (within reason) when there are differing original sources of argument or when different argument notations are substantially better suited to addressing specific argument technical challenges.

d) Use of a limited number of different formats for SAFETY CASE material other than argument

e) Use of the same format for similar SAFETY CASE content

5.2.1.4 **RECOMMENDED:**

a) Use of a graphical interface for relevant parts of SAFETY CASE navigation where it increases navigability

b) Use of structured text-based notation rather than free-form text where it may facilitate the usage of appropriate tools

5.2.1.5 **CONFORMANCE:**

CONFORMANCE is checked by inspection of SAFETY ARGUMENT, evidence, and design records.

5.2.1.6.1 **NOTE:** Access to the SAFETY CASE for ASSESSMENT purposes is discussed in Section 17 (see 17.1.1.1.f and 17.3.1.2.c.1).

5.2.2 **The artifacts cited as evidence used shall conform to defined, auditable formats.**

5.2.2.1 **MANDATORY:**

a) A defined type for each set of evidence from a defined set of types used in the SAFETY CASE

EXAMPLES: Simulation output files, test plans, vehicle data logs

NOTE: “Type” is meant in a flexible, generic sense. However, each set of evidence is of a defined type (even if each set is a unique type different from all other sets of evidence). That type is then associated with metadata that permits interpreting the evidence, per the next PROMPT ELEMENT.

b) A defined format for each type of evidence, including at least:

1) Definition of syntax and semantics, data fields, metadata fields

NOTE: Semantics are defined to the degree practical given the type and nature of the evidence.

2) Specification of criteria for evidence consistency, correctness, and completeness suitable to make the evidence auditable

3) A defined means of validating any evidence that is not derived from ACCEPTABLE data

EXAMPLE: Subjective expert judgment might be validated by data collection over the ITEM lifecycle.

EXAMPLE: An expert opinion that lightning strikes are too rare to be relevant used as evidence for neglecting lightning strike RISK mitigation is backed up by argument that any lightning strikes that do occur to deployed vehicles will be recorded and that periodic analysis will be conducted to detect if lightning strike frequency in field data becomes too frequent to neglect. (It should be noted that it is well documented that lightning does in fact strike moving, occupied vehicles upon occasion, and a more suitable SAFETY ARGUMENT is likely to be that some form of RISK mitigation is in place to ensure that the post-mitigation RISK from lightning strikes is ACCEPTABLE.)

5.2.2.2 **REQUIRED – N/A**

5.2.2.3 **HIGHLY RECOMMENDED – N/A**

5.2.2.4 **RECOMMENDED:**

a) Descriptive or tutorial examples for interpreting each type of evidence

b) Avoidance of unconstrained free text as a data type for evidence

5.2.2.5 **CONFORMANCE:**

CONFORMANCE is checked by inspection of SAFETY ARGUMENT, evidence, and design records.

5.2.2.6.1 **NOTE:** The clause permits each piece of evidence to have a different, defined type. Using a smaller number of consistent types of evidence wherever practical can help streamline the SAFETY CASE for improved comprehensibility.

5.2.3 **The CLAIMS and argument in the SAFETY CASE shall be clear and consistent.**

5.2.3.1 **MANDATORY:**

a) Correct use of natural language

NOTE: Correct use of natural language requires, among other things, that a CLAIM be phrased as a proposition (i.e., a statement that is either true or false). Standards for argument notations and guidance for argument construction might offer more detailed advice. See, for example, Assurance Case Working Group, Goal Structuring Notation Community Standard, Version 2, Safety-Critical Systems Club. January, 2018.
http://scsc.uk/SCSC-141B

b) Use of a single natural language in the CLAIMS and argument

NOTE: The intent is that any natural language used in the entirety of the SAFETY CASE, except for evidence, is in a single natural language appropriate for use by the ITEM design team. Evidence can be in alternate natural languages. A reasonable workaround for legacy multi-lingual SAFETY CASES might be the use of EOOC-style component SAFETY CASE interfaces between different-language sections. This does not preclude the use of additional mathematical notation and formal languages accessible to speakers of the natural language selected.

NOTE: Component SAFETY CASES do not have to be in the same language as the ITEM level SAFETY CASE. However, a component safety interface must be included in the same language as the SAFETY CASE. For example, a Chinese language component SAFETY CASE can export an English language component safety interface that is used in an English language ITEM level SAFETY CASE (see Section 5.7.1).

c) Use of language reasonably understandable by a proficient speaker with general technical expertise in the ITEM domain

NOTE: This is intended to make the SAFETY CASE argument and safety CLAIMS accessible to an ASSESSOR who is not part of the design team. ASSESSORS are unlikely to be experts in the details of a particular ITEM, but can be assumed to have general understanding of the domain and relevant technologies.

d) Avoidance of substantive ambiguity

5.2.3.2 **REQUIRED:**

a) Explanation and consistent use of any defined notation

EXAMPLES: Formal specification language, mathematical notation

b) *Deleted*

5.2.3.3 **HIGHLY RECOMMENDED:**

a) IDENTIFICATION of, adoption of and CONFORMANCE to a technical writing style guide, including language use

b) IDENTIFICATION of, adoption of and CONFORMANCE to a visual design language for applicable aspects of SAFETY CASE related tool interfaces

c) Use of automated analysis tooling on the SAFETY CASE

1) Spell checker

2) Grammar checker

3) Detection and correction of excessively complex sentences

4) Technical writing specific text style checker

5) Traceability link checkers

d) Highlighting and careful use of qualifying statements and limiting phrases

EXAMPLES: Detect and exercise care with the phrases: “essentially all,” “should be,” “generally”

e) Highlighting and careful use of negations for clarity, especially multiple negations

EXAMPLES: “Not unlike,” “A SAFE outcome might NOT occur”

5.2.3.4 **RECOMMENDED**

a) Use of diagrams, illustrations, interactive drawings, and other graphical approaches when appropriate.

5.2.3.5 **CONFORMANCE:**

CONFORMANCE is checked by inspection of SAFETY ARGUMENT, evidence, and design records.

5.3 Claim and argument sufficiency

5.3.1 **The SAFETY CASE CLAIMS shall encompass all identified SAFETY RELATED hazards and RISKS.**

5.3.1.1 **MANDATORY:**

a) Definition of ITEM safety requirements

1) Safety requirements for intended functionality

2) Safety requirements for potentially unintended functionality

3) Safety requirements for unsafe behaviors and states that must be avoided

NOTE: For functional safety these are often identified via hazard analysis

4) Safety requirements for mitigating faults in the ITEM itself, including both design faults and operational faults

5) Safety requirements for mitigating exceptional and unspecified environmental conditions

6) Safety requirements for life cycle considerations, including updates, inspections, maintenance, and monitoring of changing operational environments

b) Satisfaction of defined ITEM safety requirements

c) Mapping of each identified hazard to a potential violation of at least one relevant safety requirement

NOTE: Clear mapping normally takes the form of arguing how hazards are addressed by mitigations, with the safety requirements corresponding to mitigations identified.

d) IDENTIFICATION of an ACCEPTABLE level of safety for each hazard identified

5.3.1.2 **REQUIRED – N/A**

5.3.1.3 **HIGHLY RECOMMENDED:**

a) Exclusion of CLAIMS unrelated to identified hazards and RISKS

NOTE: This is a backward traceability requirement to avoid CLAIMS that aren't actually relevant to ITEM safety.

NOTE: For EOOC SAFETY CASES it is ACCEPTABLE to trace SAFETY CASE fragments to the exported EOOC boundary interface under the presumption that at least some users of the EOOC will have a higher level ITEM SAFETY CASE that completes the tracing relationship to identified hazards and RISKS.

5.3.1.4 **RECOMMENDED – N/A**

5.3.1.5 **CONFORMANCE:**

CONFORMANCE is checked by inspection of SAFETY ARGUMENT, evidence, and design records.

5.3.1.6.1 **NOTE:** The term "CLAIMS" as used in this clause is a generic term, and includes sub-CLAIMS.

5.3.2 **The SAFETY ARGUMENT shall support all identified CLAIMS.**

5.3.2.1 **MANDATORY:**

a) Support SAFETY CASE CLAIMS by ACCEPTABLE argument

1) Argument explaining why readers should conclude that the evidence cited is an ACCEPTABLE basis for believing that the safety requirements are satisfied

NOTE: Where satisfaction of each of a set of similar requirements is shown by the same kinds of evidence, this argument need not be repeated for each individual requirement so long as it is clear which collective argument applies to each requirement.

NOTE: Kinds of evidence are discussed in 5.4.1.

2) Traceability from safety requirements to the artifacts used as evidence of their satisfaction

b) IDENTIFY criteria used to determine sufficiency of arguments

5.3.2.2 **REQUIRED:**

a) IDENTIFY strategy for addressing epistemic defeaters in arguments

NOTE: Many of the PROMPT ELEMENTS in other clauses are epistemic defeaters.

Additionally, reviews of the SAFETY ARGUMENT can include consideration as to whether any additional epistemic defeaters are applicable.

b) **Pitfall:** Taking credit for CONFORMANCE to a safety standard without specifically describing the limitations of the CONFORMANCE ASSESSMENT is prone to over-crediting safety attributes. See 5.3.2.6.1.

NOTE: This is in effect an argument gap that does not support the identified CLAIM(S). See also Section 5.7.

c) **Pitfall:** Taking credit for CONFORMANCE to a safety standard designed for human operated equipment is prone to missing fault management control obligations implicitly placed upon autonomy. See 5.3.2.6.1.

NOTE: This is a special, but important, case of the preceding Pitfall regarding limitations of CONFORMANCE ASSESSMENT to a safety standard.

EXAMPLE: Credit taken for ego vehicle controllability in assessing ISO 26262 CONFORMANCE places a corresponding controllability obligation upon autonomy functions to exercise that same level of control. The need to ARGUE a replacement for the human to provide the controllability assumed as part of ISO 26262 ASIL assignment might be missed if this Pitfall is not addressed.

5.3.2.3 **HIGHLY RECOMMENDED:**

a) Avoid including argument that is not relevant to the truth of any identified CLAIM

b) Avoid including evidence not in support of any CLAIM

c) Use of strategies to address epistemic defeaters including:

1) Argument reviewed to detect unidentified but applicable epistemic defeaters

2) IDENTIFICATION of substantive but unrebutted epistemic defeaters for each CLAIM

NOTE: Unrebutted epistemic defeaters can serve to inform a reviewer's perception of argument strength.

5.3.2.4 **RECOMMENDED – N/A**

5.3.2.5 **CONFORMANCE:**

CONFORMANCE is checked by inspection of SAFETY ARGUMENT, evidence, and design records.

5.3.2.6.1 **REFERENCE:** For discussion of the Pitfalls named in 5.3.2.2.b and 5.3.2.2.c, and Pitfalls generally, see: Koopman, P., Kane, A. & Black, J., "Credible Autonomy Safety Arguments," Safety-Critical Systems Symposium, Bristol UK, Feb. 2019.

5.3.3 **The SAFETY CASE shall avoid argument defects.**

EXAMPLE: William S. Greenwell, John C. Knight, C. Michael Holloway, and Jacob J. Pease, "A taxonomy of fallacies in system safety arguments," Proc. Int'l System Safety Conference (ISSC), Albuquerque, NM, 2006.
https://ntrs.nasa.gov/citations/20060027794

5.3.3.1 **MANDATORY:**

a) IDENTIFY logical fallacies to avoid

NOTE: This results in a checklist of logical fallacies to be avoided in the SAFETY CASE.

EXAMPLE: Checklist derived from: William S. Greenwell, John C. Knight, C. Michael Holloway, and Jacob J. Pease, "A taxonomy of fallacies in system safety arguments," Proc. Int'l System Safety Conference (ISSC), Albuquerque, NM, 2006.
https://ntrs.nasa.gov/citations/20060027794

b) IDENTIFY rhetorical devices to use and rhetorical devices to avoid

NOTE: This results in a checklist of rhetorical devices to be avoided in the SAFETY CASE.

5.3.3.2 **REQUIRED:**

a) Avoid identified logical fallacies

b) Avoid identified rhetorical devices

c) **Pitfall:** Taking credit for proven in use technology that is used in a different operational environment or for a different purpose is prone to over-crediting safety attributes. See 5.3.3.6.1.

1) This Pitfall specifically includes COTS, legacy, and EOOC components, including hardware and/or software

(**NOTE:** See Section 13.4)

2) This Pitfall specifically includes changes in ITEM operational parameters that might be relevant to the component

EXAMPLE: For an example of an accident where a change in the operational parameters for a flight guidance component led to the LOSS of a spacecraft, see Lions, J.L., Ariane 5 Flight 501 Failure, Report by the Inquiry Board, 1996

d) **Pitfall:** Discounting failures in FIELD ENGINEERING FEEDBACK because the failure cannot be reproduced, has not recurred, or has not led to an accident or other adverse outcome is prone to inductively discounting multiple failures that, if taken as a set, substantively demonstrate invalidity of a SAFETY CASE. See 5.3.3.6.1.

e) **Pitfall:** Arguing coverage of AUTONOMOUS failure analysis based on data from human-operated ITEM is prone to missing some types of failures, including: See 5.3.3.6.1.

1) Failures that are triggered by operational situations an AUTONOMOUS ITEM might enter that human operators typically avoid

2) Failures atypical of human mistakes that an AUTONOMOUS ITEM fault could trigger

f) **Pitfall:** Arguing RISK mitigation via analysis of operational data and/or test data based on arrival rate of INCIDENTS ("surprises" or other potential failures) is prone to: See 5.3.3.6.1.

1) Overlooking the additional compounding factor of the distribution of the means of different types of root causes

EXAMPLE: a heavy tail distribution of mean arrival rates of different types of triggering events for ITEM failures

2) Overlooking the potential effects of infrequent but inevitable common cause events

EXAMPLES: Leap seconds, GPS date rollover, daylight savings time changes, or other time keeping anomalies

g) **Pitfall:** Arguing test coverage based upon human-designed test planning is prone to overlooking edge cases that apply to AUTONOMOUS functions but would not generally be considered edge cases by a human operator. See 5.3.3.6.1.

h) **Pitfall:** Arguing ITEM correctness based upon use of formal methods is prone to overlooking any invalidities in underlying assumptions made by the proofs. See 5.3.3.6.1.

i) **Pitfall:** Arguing that RISK is low for a known hazard or variance from expected behavior based upon operational experience alone is prone to underestimating the possibility of catastrophic outcomes.

EXAMPLE: For an example of an accident preceded by deviations from expected behavior, see Rogers Commission Report; Report of Columbia Accident Investigation Board, 1986

j) **Pitfall:** Arguing low RISK based upon unvalidated simulation results alone is prone to missing RISKS due to simulation defects, modeling faults, and simplifications made in the abstraction process to create the simulation.

5.3.3.3 **HIGHLY RECOMMENDED – N/A**

5.3.3.4 **RECOMMENDED – N/A**

5.3.3.5 **CONFORMANCE:**

CONFORMANCE is checked by inspection of SAFETY ARGUMENT, evidence, and design records.

5.3.3.6.1 **REFERENCE:** For discussion of the Pitfalls named in 5.3.3.2.c, 5.3.3.2.d, 5.3.3.2.e, 5.3.3.2.f, 5.3.3.2.g, and 5.3.3.2.h, and Pitfalls generally, see: Koopman, P., Kane, A. & Black, J., "Credible Autonomy Safety Arguments," Safety-Critical Systems Symposium, Bristol UK, Feb. 2019.

5.3.4 **Defective construction patterns shall not be used in the development of the ITEM.**

5.3.4.1 **MANDATORY:**

a) IDENTIFY list of defective construction patterns of potential concern

NOTE: These include risky or otherwise unsuitable “patterns” in architecture, design, and implementation that are deemed unacceptable for the ITEM. (These are not patterns actually in use, but rather a catalog of patterns that are to be avoided.)

NOTE: A minimum list includes identifying patterns enumerated in Section 5.3.4.2. It is desirable that the list be further expanded based upon experience and literature research as determined by the design team.

5.3.4.2 **REQUIRED:**

a) Avoid identified defective construction patterns

b) **Pitfall:** Use of a “command override” pattern with a DOER/CHECKER architectural pattern is prone to failure to MITIGATE unsafe behaviors. See 5.3.4.6.1.

c) **Pitfall:** Use of a Checker that does not MITIGATE all hazards attributable to the associated Doer is prone to failure to MITIGATE unsafe behaviors. See 5.3.4.6.1.

5.3.4.3 **HIGHLY RECOMMENDED – N/A**

5.3.4.4 **RECOMMENDED – N/A**

5.3.4.5 **CONFORMANCE:**

CONFORMANCE is checked by inspection of SAFETY ARGUMENT, evidence, and design records.

5.3.4.6.1 **REFERENCE:** For discussion of the Pitfalls named in 5.3.4.2.b and 5.3.4.2.c, and Pitfalls generally, see: Koopman, P., Kane, A. & Black, J., "Credible Autonomy Safety Arguments," Safety-Critical Systems Symposium, Bristol UK, Feb. 2019.

5.4 Evidence sufficiency

5.4.1 **The SAFETY ARGUMENT shall be supported by evidence.**

5.4.1.1 **MANDATORY:**

a) Each CLAIM is traceable to supporting evidence.

5.4.1.2 **REQUIRED – N/A**

5.4.1.3 **HIGHLY RECOMMENDED:**

a) Use of the following categories of evidence:

1) Experimental data

2) Analytic data, reports, and justifications

3) Procedure definitions

4) Process compliance

5) Development and V&V process data

6) V&V data

EXAMPLES: Test plans, test results, experimental design methodology, formal verification

7) Qualitative analysis and subjective judgement

8) FIELD ENGINEERING FEEDBACK data

9) Placeholder for evidence that will be collected via FIELD ENGINEERING FEEDBACK

10) Accepted RISKS, including evidence that RISK is ACCEPTABLE

11) Assumptions for which no evidence is provided, including basis of support that the assumption is reasonable

EXAMPLE: An assumption that traffic signals never display signal indications that would cause traffic to flow on intersecting paths (save for permissive turns where permitted) might lack field data. However, the argument might IDENTIFY the authorities responsible for traffic signaling within the ODD and cite their commitment to ensuring that traffic signaling hardware is installed properly and complies with a standard such as National Electrical Manufacturers Association (NEMA) Standard 2 (2016), which requires hardware fail-safes to place all signal heads in a flashing red (major roadway) or yellow (minor roadway) mode if an unsafe condition is detected. The responsible authorities' commitment and the nature of the mitigations they require make it reasonable to assume that conflicting signal indications will very rarely happen in practice.

5.4.1.4 **RECOMMENDED – N/A**

5.4.1.5 **CONFORMANCE:**

CONFORMANCE is checked by inspection of SAFETY ARGUMENT, evidence, and design records.

5.4.1.6.1 **NOTE:** Traceability can be direct or can be indirect via sub-arguments and/or sub-CLAIMS

5.4.1.6.2 **NOTE:** Arguments that are not grounded either directly or indirectly in evidence are unacceptable. Assumptions can be considered evidence for this purpose per Section 5.4.1.3.a.11.

5.4.2 **The SAFETY ARGUMENT shall ARGUE the sufficiency of evidence.**

5.4.2.1 **MANDATORY:**

a) SAFETY CASE records the experimental design or other data collection strategy for evidence based on data collection

b) IDENTIFY criteria used to determine sufficiency of evidence

c) ARGUE evidence is sufficient to result in an ACCEPTABLE SAFETY CASE

1) Describe manner in which evidence is used to support or refute the validity of an argument and/or CLAIM.

2) Arguments that RISK of confirmation bias has been mitigated

5.4.2.2 **REQUIRED:**

a) Lifecycle monitoring that confirms CLAIMS that cannot adequately be supported by evidence at design time, such as:

1) Unsupported expert or subjective opinion

2) Existing practices that are not supported by data and are not supported by written public standards documents, public guidance documents, or similar cited sources

3) Assumptions

NOTE: Lifecycle monitoring is employed to monitor RISK to the degree that argument relies upon opinion, assumptions, or potentially weak evidence.

b) IDENTIFICATION of epistemic defeaters and accompanying defeasibility arguments, including at least:

1) Potentially confounding experimental variables

2) Potential biases in data

3) Potentially insufficient quantity of data samples

5.4.2.3 **HIGHLY RECOMMENDED:**

a) Inclusion of counter-evidence and accompanying arguments

b) Inclusion of process compliance arguments related to creation of evidence

c) Lifecycle monitoring of evidence is based on consensus-based public standard approaches

d) **Pitfall:** Data collected before argument has been created is prone to be inappropriately used as evidence for that argument

EXAMPLES: Through p-hacking, cherry-picking, and moving the goalposts

5.4.2.4 **RECOMMENDED:**

a) Use of adversarial sequential test design methodologies to maximize the chances of identifying failure modes and/or emergent behaviors within the ODD

5.4.2.5 **CONFORMANCE:**

CONFORMANCE is checked by inspection of SAFETY ARGUMENT, evidence, evidence gathering design, and design records.

5.4.2.6.1 **NOTE:** In practice, this clause may result in argument of the form: the argument is valid because (1) it is supported by evidence, and (2) the evidence itself is valid. The criteria defining what makes evidence valid depend on the kind of evidence, are often axiomatic within a discipline, and may change over time as the community of practice gains experience with that kind of evidence.

5.4.2.6.2 **NOTE:** The potential set of epistemic defeaters and potential types of counter-evidence that might be collected is effectively unbounded. Some defeaters and counter-evidence are more likely to be relevant in practice. Selection guidance is provided by PROMPT ELEMENTS in this standard and is additionally informed by developer experience.

5.4.2.6.3 **NOTE:** Unsupported expert opinion includes statements by domain experts that are not substantively supported by presented evidence. Opinions directly based upon scholarly papers, substantive data, and the like may be considered supported in this sense so long as that basis is stated as part of the evidence and the experimental context of the cited work is explicitly ARGUED to apply to its use in the SAFETY CASE.

5.4.3 **Evidence shall cover difficult-to-reproduce aspects of the ITEM.**

5.4.3.1 **MANDATORY:**

a) IDENTIFICATION of any nondeterministic and CHAOTIC aspects of the ITEM with regard to evidence (if none, so state)

5.4.3.2 **REQUIRED:**

a) Arguments and evidence to MITIGATE RISK of invalidity due to nondeterministic aspects of the ITEM and its operating environment (if none, so state)

EXAMPLE: Use of concurrency management mechanisms to MITIGATE timing-sensitive concurrent access faults, use of seeded pseudo-random number generators to reproduce intentionally nondeterministic behaviors for testing repeatability.

EXAMPLE: Fault injection used to fail elements which would otherwise fail infrequently during testing.

b) Arguments and evidence to MITIGATE RISK of invalidity due to CHAOTIC aspects of the ITEM and its operating environment (if none, so state)

EXAMPLE: Strategy for testing reproducibility when ITEM behavior changes dramatically based on small input differences. For some ITEMS a concrete example is whether the vehicle veers left or right when an obstacle appears exactly in front of it.

c) Any metrics used to conform to the characteristics relating to SPI Metrics (See Section 16).

5.4.3.3 **HIGHLY RECOMMENDED:**

a) Use of statistical significance approaches

b) Arguments and evidence mitigating the CHAOTIC aspects of the ITEM and its operating environment, if any, address:

1) Definition of control measures for variation

2) Testing of control measures

3) Justification for sufficiency of control measures and any other measures

5.4.3.4 **RECOMMENDED:**

a) Use of a digital twin or similar technique

5.4.3.5 **CONFORMANCE:**

CONFORMANCE is checked by inspection of SAFETY ARGUMENT, evidence, and design records.

5.4.3.6.1 **NOTE:** NONDETERMINISM means that an ITEM can behave in different ways for identical initial conditions (e.g., due to use of a pseudo-random algorithm). CHAOTIC means that the ITEM can behave in different ways due to perturbations in initial conditions that are smaller than the ability of validation methods to control. Both nondeterministic and CHAOTIC properties result in varying responses to testing, for example, but are caused by different sources. An ITEM can be both nondeterministic and CHAOTIC. It is not essential to differentiate between these causes if either is acknowledged to be present, but mitigation of both possibilities must be considered in establishing the validity of evidence. Non-deterministic and CHAOTIC behavior can undermine the validity and completeness of test results if tests pass by chance due to favorable system behavior in a particular test run.

See also Section 12.4.7 regarding fault injection.

5.5 Accepted risks

5.5.1 **Accepted RISKS shall be identified.**

5.5.1.1 **MANDATORY:**

a) IDENTIFY acceptance criteria for any RISK less than fully mitigated.

5.5.1.2 **REQUIRED:**

a) IDENTIFY any RISK that is less than fully mitigated to an ACCEPTABLE level as an "accepted RISK." These include but are not limited to:

1) Identified RISKS for which no mitigation measure has been taken

EXAMPLE: An ITEM level safety ASSESSMENT potential hazard that was determined to be extremely improbable or otherwise not something that could happen in the "real world" is an accepted RISK and is included in the SAFETY CASE as an unmitigated RISK.

2) Partially mitigated RISKS (i.e., RISKS not mitigated to a fully ACCEPTABLE level)

3) Unknown RISKS

EXAMPLES: "Known unknowns," "unknown unknowns"

EXAMPLE: CONFORMANCE with ISO 21448.

4) RISKS for which mitigation is based on unsupported expert opinion, assumptions, or other less-than-comprehensive evidence and also are not in CONFORMANCE with generally accepted industry practices such as standards documents

b) For less than fully mitigated RISKS (including accepted RISKS) characterize the level of mitigation and ARGUE that the level of mitigation, if any, is ACCEPTABLE in the context of the ITEM SAFETY CASE.

1) Include an evaluation of the expected outcomes of each such RISK across the ITEM lifecycle.

5.5.1.3 **HIGHLY RECOMMENDED:**

a) Characterization of post-mitigation level of RISKS that are deemed fully mitigated.

5.5.1.4 **RECOMMENDED – N/A**

5.5.1.5 **CONFORMANCE:**

CONFORMANCE is checked by inspection of SAFETY ARGUMENT, evidence, and design records.

5.5.1.6.1 **NOTE:** This includes acceptance of any remaining unknown RISKS, which should be broken down into categories if and as information exists to support doing so (e.g., "known unknowns"). It is understood that such RISKS might not be readily quantified, but yet they are being accepted when a decision to deploy is being made.

5.5.2 **Accepted RISKS shall be tracked through the ITEM lifecycle via FIELD ENGINEERING FEEDBACK**

5.5.2.1 **MANDATORY – N/A**

5.5.2.2 **REQUIRED:**

a) FIELD ENGINEERING FEEDBACK for accepted RISKS to ensure that the RISK in practice is less than or equal to the level of RISK stated as an expected outcome for ITEM operation

b) FIELD ENGINEERING FEEDBACK explicitly considers RISKS due to gaps in RISK analysis

5.5.2.3 **HIGHLY RECOMMENDED:**

a) Automated field data collection of INCIDENTS and LOSS events to determine if outcomes for accepted RISKS are within expectations on a per-RISK basis

5.5.2.4 **RECOMMENDED – N/A**

5.5.2.5 **CONFORMANCE:**

CONFORMANCE is checked by inspection of SAFETY ARGUMENT, evidence, and design records.

5.5.2.6.1 **NOTE:** An example approach is including "unknown RISKS" in the SAFETY CASE, noting them as "accepted RISKS," justifying why ACCEPTABLE RISK analysis has been performed to permit accepting the resultant RISKS, and arguing that the root cause analysis procedure explicitly includes identifying novel RISKS so as to add them to the SAFETY CASE in a timely fashion when they are encountered in the field.

5.6 Safety culture

5.6.1 **The role of safety culture of development, supply chain, maintenance and operations in RISK IDENTIFICATION and mitigation shall be identified.**

5.6.1.1 **MANDATORY:**

a) Definition of safety culture and role used as part of the RISK mitigation approach, addressing at least:

1) Role of safety staff in ensuring safety

2) Role of non-safety staff in ensuring safety

3) Role of management in ensuring safety

4) Role of safety management system in organization

5) Role of suppliers in ensuring safety

6) Role of lifecycle participants in ensuring safety

7) INDEPENDENCE of safety roles between engineering development stakeholders, deployment stakeholders, and business profitability stakeholders

8) Upper management visibility of and delegation of authority to safety roles

9) Role of transparency, reporting, and safety promotion in communicating safety culture

10) Role(s) of non-staff third parties

EXAMPLES: Emergency responders, INCIDENT investigators.

b) IDENTIFY process and activities that support the communication and tracking to resolution of potential SAFETY RELATED issues

c) IDENTIFY an ACCEPTABLE set of ongoing FIELD ENGINEERING FEEDBACK and continuous improvement activities related to safety culture

d) IDENTIFY an ACCEPTABLE set of ongoing activities to gather information on hazards and RISKS from publicly available sources

EXAMPLES: Monitoring recall notices from other developers, accident investigation reports, regulatory actions in the system domain plus related domains, published statistical analyses of hazards.

e) ARGUE that the execution of identified activities and other identified factors results in an ACCEPTABLE safety culture

f) IDENTIFY policies to prevent retribution for reporting a SAFETY RELATED issue

5.6.1.2 **REQUIRED:**

a) **Pitfall:** Organizational structures in which engineering, business, and/or operational management can exert control or pressure upon roles tasked with ensuring safety are prone to degraded safety outcomes.

EXAMPLE: Normalization of deviance

b) All activities that have access to SAFETY RELATED data identified as supporting the communication of potentially SAFETY RELATED issues

c) IDENTIFICATION of role in safety culture in ensuring that root cause analysis will be effective at identifying defects in SAFETY CASES and safety processes

5.6.1.3 **HIGHLY RECOMMENDED:**

a) IDENTIFICATION of metrics and feedback mechanisms used to evaluate and manage safety culture

b) IDENTIFICATION of roles and responsibilities accompanied by argument and evidence for suitable competency of staff

EXAMPLE: Evidence supporting an argument of Suitably Qualified and Experienced Personnel (SQEP)

5.6.1.4 **RECOMMENDED – N/A**

5.6.1.5 **CONFORMANCE:**

CONFORMANCE is checked by inspection of SAFETY ARGUMENT and evidence (e.g., design process documents, policy documents, training plans, and relevant records).

REFERENCE: See safety culture metrics for SPIS in Section 16.2.5

5.7 Item scope

5.7.1 **The SAFETY CASE shall IDENTIFY SAFETY RELATED aspects of the ITEM, including potential faults and failures, encompassing the ITEM lifecycle.**

5.7.1.1 **MANDATORY:**

a) IDENTIFY and describe SAFETY RELATED functionality and interfaces

b) IDENTIFY and describe SAFETY RELATED components

c) IDENTIFY and describe SAFETY RELATED properties

EXAMPLE: Real time deadline for responding to hazards when credit is taken for response time in RISK reduction argument

d) IDENTIFY and describe SAFETY RELATED aspects of non-operational lifecycle phases (See Section 14)

e) IDENTIFY and describe other aspects of the ITEM related to safety

EXAMPLE: Emergent properties such as total system weight if limited kinetic energy is part of the RISK mitigation strategy

f) Functional requirements for SAFETY RELATED functionality

g) ARGUE ACCEPTABLE coverage of identified fault models, including:

1) Detection, diagnosis, and management of runtime faults,

2) Verification and validation (V&V) of the absence of design faults (See Sections 6.2, 10.4, and 12.3 – 12.4)

h) Coverage analysis for the V&V of each identified element

NOTE: See subsections within 12.3 for more information

5.7.1.2 **REQUIRED:**

a) SAFETY RELATED exception handling capabilities

b) **Pitfall:** Components and functions that provide data to or otherwise affect the operation of SAFETY RELATED components and functions are themselves SAFETY RELATED, but are prone to being discounted as not SAFETY RELATED if traceability of data flows and other direct and indirect sources of interaction with SAFETY RELATED components is not performed rigorously.

5.7.1.3 **HIGHLY RECOMMENDED – N/A**

5.7.1.4 **RECOMMENDED – N/A**

5.7.1.5 **CONFORMANCE:**

CONFORMANCE is checked via consideration of subsection CONFORMANCE checks, including traceable inclusion of each applicable PROMPT ELEMENT in the SAFETY CASE.

5.7.1.6.1 **NOTE:** A primary goal of this clause is to ensure that aspects of the ITEM that can contribute to or MITIGATE RISK are identified to ensure inclusion in the SAFETY CASE. While the SAFETY ARGUMENT covers these matters, it might do so by reference to documentation such as specification and design documents. For example, argument explaining why readers should believe hazards are addressed or requirements are satisfied (see Sections 5.3.1 and 5.3.2) mentions interfaces and refers readers to primary documentation of those interfaces. Argument showing readers why fault model coverage is ACCEPTABLE refers to clear documentation of the fault models used.

5.7.2 **The SAFETY CASE shall describe the concept of operations for the ITEM.**

5.7.2.1 **MANDATORY:**

a) Overview of mission capabilities

b) Overview of safety objectives

c) Overview of RISK mitigation strategy

d) Overview of ITEM hardware, software, and functional architecture

e) Overview of SAFETY RELATED functionality

f) Overview of ITEM operational modes

g) Description of ODD (see Section 8)

5.7.2.2 **REQUIRED – N/A**

5.7.2.3 **HIGHLY RECOMMENDED:**

a) Overview of non-SAFETY RELATED functionality and overview of non-interference explanation with SAFETY RELATED functionality (See 10.3.3)

5.7.2.4 **RECOMMENDED – N/A**

5.7.2.5 **CONFORMANCE:**

CONFORMANCE is checked via inspection of the SAFETY CASE.

5.7.2.6.1 **NOTE:** This clause is intended to provide an overview of the ITEM and specifically to provide contextual information needed for understanding the SAFETY CASE.

5.7.3 **The boundary within the SAFETY CASE between any assessed ELEMENT OUT OF CONTEXT (EOOC) and the rest of the SAFETY CASE shall include a specified interface.**

5.7.3.1 **MANDATORY – N/A**

5.7.3.2 **REQUIRED:**

a) IDENTIFICATION of EOOC boundaries in the SAFETY CASE, if any

NOTE: Some NON-DEVELOPMENT ITEMS (NDIS) are not treated as EOOCS (See 13.4).

b) For each EOOC boundary:

1) List of EOOC properties that have been assessed

2) List of EOOC assumptions that are necessary conditions for assessed properties to hold true

3) EOOC fault model

4) IDENTIFY any UL 4600 clauses that were not considered in EOOC ASSESSMENT.

EXAMPLE: A purely hardware EOOC might consider software-related clauses out of scope for EOOC ASSESSMENT. This would put the full burden of CONFORMANCE with those clauses onto the SAFETY CASE incorporating the EOOC.

5) ASSESSMENT report for EOOC in same language as ITEM SAFETY CASE using the ASSESSMENT report

NOTE: This might require a translation of the EOOC safety report from the language used for the component SAFETY CASE to the language used in the ITEM SAFETY CASE

6) All other characteristics of the EOOC that are used as evidence by the SAFETY CASE for the ITEM that contains the EOOC.

7) When ITEMS incorporate assessed EOOCS, for each EOOC, IDENTIFY any unusual or possibly unexpected features of the ITEM or its environment and for each identified feature analyze the potential impact on the integrity and result of the safety analysis and SAFETY CASE.

5.7.3.3 **HIGHLY RECOMMENDED – N/A**

5.7.3.4 **RECOMMENDED – N/A**

5.7.3.5 **CONFORMANCE:**

CONFORMANCE is checked by inspection of SAFETY ARGUMENT, evidence, and design records.

5.7.3.6.1 **NOTE:** The concept of EOOC used here is generic and not limited to the functional safety aspects of the component as might be the case when the term Safety Element out of Context (SEooC) is used in a functional safety standard. ASSESSMENT of an EOOC is performed for the full scope of its use in the SAFETY ARGUMENT.

5.7.3.6.2 **NOTE:** Lists of EOOC assumptions and properties are documented to the degree they are known. As additional assumptions and properties are discovered over the component lifecycle, they are added. Addition of an additional assumption or property might invalidate higher level SAFETY CASES using the EOOC. **Reference:** Lions, J., Ariane 5 Flight 501 Failure Report by the Inquiry Board, 19 July 1996.

5.7.3.6.3 **NOTE:** Other approaches to modular SAFETY ARGUMENT, re-use of argument fragments, inter-domain argument reuse, argument patterns, and so on are not specifically prohibited. However, establishing ASSESSMENT criteria for alternate approaches is beyond the scope of this standard.

6 Risk Assessment

6.1 General

6.1.1 **The SAFETY CASE shall ARGUE RISKS are assessed and mitigated.**

6.1.1.1 **MANDATORY:**

a) IDENTIFICATION of fault models (See Section 6.2)

b) IDENTIFICATION of hazards (See Section 6.3)

c) RISK framework and evaluation (See Section 6.4)

d) RISK mitigation and evaluation of mitigation effectiveness (See Section 6.5)

6.1.1.2 **REQUIRED – N/A**

6.1.1.3 **HIGHLY RECOMMENDED:**

a) IDENTIFICATION and use of a total ITEM RISK summing approach

b) When using calculated probabilities, calculated probability of INCIDENT occurrence yields both probability and confidence for entire causal chain. Probability is considered unbounded unless coupled to confidence.

c) The validity of assumptions regarding accepted RISKS is monitored to ensure their validity holds true and does not turn out to be or become incorrect

6.1.1.4 **RECOMMENDED:**

a) Bayesian probability estimation for RISK approaches that require estimating probability of occurrence

6.1.1.5 **CONFORMANCE:**

CONFORMANCE is checked via consideration of subsection CONFORMANCE checks, including traceable inclusion of each applicable PROMPT ELEMENT in the SAFETY CASE.

6.1.1.6.1 **NOTE:** RISKS that are accepted are still monitored as a mitigation for a situation in which the acceptance was based on an assumption that is incorrect or becomes incorrect. Thus, all RISKS are said to have been “mitigated” in the final SAFETY CASE even if there is no explicit mitigation mechanism included in the ITEM design.

See also: Run-Time Monitoring, Section 12.5.

6.2 Fault model

6.2.1 **The SAFETY CASE shall IDENTIFY a fault model for SAFETY RELATED aspects of the ITEM.**

6.2.1.1 **MANDATORY:**

a) IDENTIFY a fault model for each SAFETY RELATED ITEM component, feature, or other aspect for which fault analysis is relevant

1) Systematic faults:

i) Design fault model

ii) Hardware fault model, including

iii) Software fault model

iv) Manufacturing fault model

v) Operational fault model

NOTE: Relates to systematic failures and intended as “using or performing an operation not suitable.” Random hardware faults are included separately below.

vi) Non-operational fault model

EXAMPLES: Age-related component degradation, degradation of ITEM due to lack of operation while in storage

vii) Maintenance fault model

viii) Procedural fault model

ix) ITEM operation fault model

x) Tool fault model

2) Random hardware faults

i) Permanent faults

ii) Transient faults

3) Fault multiplicity

i) Single fault

ii) Multiple faults due to a common cause

iii) Accumulation of multiple faults over the lifetime of the ITEM

iv) Sufficiently probable multi-point faults

NOTE: Use of the plausible dual-point faults and multiple-point faults approach defined in ISO 26262-5:2018 can be used to assess the probability of multi-point faults.

NOTE: Credit can be taken for diagnosis, recovery, degraded operational modes, and repair capabilities if supported by evidence

4) Mitigation

i) Detected

ii) Undetected (residual and/or latent, including masked) faults

b) Traceability from each fault model to fault mitigation for applicable components, functions, and other aspects of the ITEM.

6.2.1.2 **REQUIRED:**

a) Model of SAFETY RELATED expectations for correct or required operation for each SAFETY RELATED component or function.

NOTE: In some SAFETY CASES this might correspond to "safety requirements," but it is potentially broader if for example there are SAFETY RELATED non-functional aspects that need to be considered.

EXAMPLE: Abnormally excessive power consumption by a fail-SAFE device depletes a battery power supply, disabling a failsafe. In this case specified power consumption is a SAFETY RELATED expectation that might not normally be associated with a "safety requirement."

b) Fail arbitrary fault model at the component level for complex electronic components within an FCR.

EXAMPLES: Programmable hardware, components containing software

NOTE: An arbitrary failure mode at the component level is an arbitrary fault symptom at the next higher level of abstraction, and therefore is being referred to as an arbitrary fault in this context.

c) Inclusion of "other" fault modes for non-complex electronic component fault analysis

EXAMPLE: Beyond "short" and "open" to include resistive faults and capacitive faults for connectors and passive electronic components

d) **Pitfall:** Simplistic fault models are prone to being unacceptable for describing faults in computer-based ITEMS

EXAMPLES: Assumptions that all component failures result in clean fail-stop semantics, considering only input shorts to power/ground, considering only fail-stuck integrated circuit faults, and other fault models that are overly simplistic for some applications

6.2.1.3 **HIGHLY RECOMMENDED:**

a) Inclusion of electrical, mechanical, and other components that are relevant to ability of autonomy to operate safely

EXAMPLES: Sensor mounting structure, vehicle wheels

b) Encompassing detected but un-annunciated faults

c) Use of Byzantine component fault model (includes hazardously misleading information)

d) **Pitfall:** Fault models limited by conclusory and subjective statements that a typical type of fault is "unrealistic" or would "not happen in the real world" are prone to significantly understating faults that actually do happen in fielded ITEMS.

EXAMPLE: Exclusion of Byzantine faults in aerospace ITEM fault models (see Driscoll, K., "Real Item Failures," https://c3.nasa.gov/dashlink/resources/624/)

6.2.1.4 **RECOMMENDED – N/A**

6.2.1.5 **CONFORMANCE:**

CONFORMANCE is checked via consideration of subsection CONFORMANCE checks, including traceable inclusion of each applicable PROMPT ELEMENT in the SAFETY CASE.

See also: Fault Detection and Mitigation, Section 10.4

See also: Malicious Fault Model, Section 10.8.2

6.2.2 **The software fault model shall include an acceptably broad set of potential software faults and failures.**

6.2.2.1 **MANDATORY:**

a) Tool chain failure

EXAMPLE: Tool produces incorrect software image or configuration

b) Incorrect requirements and algorithms

c) Incorrectly built software image including nonvolatile data

EXAMPLE: Wrong version of library included

d) Incorrect installed software image, including nonvolatile data

EXAMPLE: Deployed software image differs from validated software image

e) Incorrect software image version information, including nonvolatile data

f) Corrupted software image, including nonvolatile data

g) Data reporting tool failure

EXAMPLE: INCIDENT data reporting tool reports incorrectly

h) Coding defects

i) Defects that corrupt data at run time

j) Defects that corrupt hardware runtime configuration

k) Timing faults

l) Race condition

m) Embedded software defects

6.2.2.2 **REQUIRED – N/A**

6.2.2.3 **HIGHLY RECOMMENDED:**

a) Other sources of systematic faults and errors

1) Faults resulting from use of hardware description languages (HDLs)

2) Third party component failure

EXAMPLE: RTOS defect, math library defect

3) Configuration reporting tool failure

4) Tool calibration and usage errors

EXAMPLES: Incorrect calibration variable definitions used with configuration data compilation tool, incorrect command line parameters used when running static analysis tool or compiler

5) Faults and failures associated with entry into and execution of supervisory sections of SAFETY RELATED software

EXAMPLES: Concurrency faults, timing faults, and data concurrency faults that might be caused by interrupt service routines and/or operating system scheduling; disabling interrupts in a way that disrupts system scheduling

6) Defects in software embedded in third party components

EXAMPLES: Sensor firmware defects, network adapter firmware defects, storage module firmware defects

7) Machine learning brittleness

8) Corrupted data

9) Boundary value faults

EXAMPLES: Off-by-one, incorrect handling of borderline cases, array index values at boundaries of array size

10) Exceptional value faults

EXAMPLES: Null pointers, floating point infinity, zero length character strings

11) Out of range value faults

EXAMPLES: Off-scale sensor values, buffer overflows, numeric overflow, floating point underflow

12) Software faults that corrupt memory

a) Data memory

i) Volatile memory

ii) Non-volatile memory

iii) Stack

iv) Heap

v) Statically allocated data

b) Program memory

13) Software faults that result in incorrect hardware performance

EXAMPLES: Software corruption of hardware configuration, including clock rate dividers, I/O pin configurations, and power management settings

14) Software defects that defeat integrity checks

EXAMPLE: Spurious watchdog timer kick

15) Software component crash

16) Software component hang

17) Software component misses real time deadline

18) Time keeping faults, including:

a) Time keeping overflow

EXAMPLE: Timer rollover

b) Time keeping roundoff

EXAMPLE: 32-bit floating point representation of time suffers roundoff error when incremented by tenths of a second

c) Incorrect handling of time keeping discontinuities

EXAMPLES: Leap second, leap year, “Y2K” bug, Unix time rollover in 2038, Daylight Savings Time, GPS week rollover

d) Incorrect handling of time zones

EXAMPLES: Crossing international date line, crossing time zone boundaries, changes to time zone boundaries and/or offsets

e) Incorrect calculation of local solar time and position

EXAMPLE: ODD precludes driving with the sun visible on the horizon due to potential camera issues, but ITEM’s calculation of apparent sun angle at the ITEM’s current position (latitude, longitude, altitude) and date is incorrect due to incorrect calculations and/or incorrect nautical almanac stored data values.

19) **Pitfall:** Use of a fail-crash software fault model is prone to overlooking fail active and other dangerous software failure behaviors.

b) A specified fault model for malicious data faults

EXAMPLE: An unbounded model for malicious data faults with the sole exception of adversary not knowing secret key values

6.2.2.4 **RECOMMENDED – N/A**

6.2.2.5 **CONFORMANCE:**

CONFORMANCE is checked via inspection of design and V&V evidence as well as DEMONSTRATION.

6.2.3 **The microelectronic and electronic hardware fault model shall cover an acceptably broad set of potential run-time as well as fabrication faults and failures.**

6.2.3.1 **MANDATORY:**

a) Power supply faults

b) Thermal faults

EXAMPLES: Clock throttling due to excessive temperature, early failure due to chronic operation at or above maximum rated temperature

c) IC fabrication faults

d) Embedded software defects

e) Single event effects

6.2.3.2 **REQUIRED:**

a) Clocking faults

b) Errata and design defects

c) Specific types of power supply faults:

1) Under-voltage (brownout)

EXAMPLES: Incorrectly configured brown-out protection; brown-out threshold voltage set for microcontroller requirements but is too low for associated non-volatile memory chip causing memory corruption when writing

2) Fast transient power spikes that do not activate brownout protection

3) Over-voltage

i) Due to regulator failure

ii) Due to external application of high voltage

EXAMPLE: Tow truck applies 24V jump voltage to 12V electrical system

iii) Due to power cabling faults

4) Power phasing and polarity faults

i) Reverse polarity battery installation

ii) Application of DC external voltage with reverse polarity

iii) Coupling with mismatch AC phase angle

iv) Coupling with incorrect AC 3-phase rotation direction

5) Loss of at least one supply voltage

6) Insufficient power supply capability

EXAMPLE: Weak programming of nonvolatile memory cell due to power supply voltage sag during programming due to high current demand

7) Voltage regulation failure

i) Off-chip regulator

ii) On-chip regulator

8) Back-feeding or parasitic power supplied via signal inputs

EXAMPLE: Processor fails to shut down or operates in an anomalous way due to backfeeding supply

d) Multiple adjacent single event upset faults for small geometry devices prone to such faults

e) Design changes

EXAMPLES: Mask change since most recent design validation, die shrink, temperature qualification range change, vendor changes for a specific component (e.g., produced with different mask or on different fab)

f) Excessive cycling of life-limited devices

EXAMPLE: Excessive EEPROM cycles that cause cell wearout

g) Memory degradation

EXAMPLES: Refresh fault, LOSS of stored charge over time for nonvolatile memory

6.2.3.3 **HIGHLY RECOMMENDED:**

a) Single event upset fault model includes:

1) Storage cell faults

NOTE: Small geometry cells can suffer upsets to multiple physically proximate bits from one single event upset

2) Random logic faults

3) Configuration register faults

4) Storage controller logic

NOTE: Upsets in controller logic can result in incorrect memory addressing and/or data handling, violating assumptions about single event upsets only affecting a small number of data bits.

b) IC damage

EXAMPLES: Damage due to voltage spikes, over-temperature operation

c) IC bias

EXAMPLE: MEMS inertial navigation sensor offset

6.2.3.4 **RECOMMENDED – N/A**

6.2.3.5 **CONFORMANCE:**

CONFORMANCE is checked via inspection of design and V&V evidence as well as DEMONSTRATION.

6.2.3.6.1 **NOTE:** This clause is intended primarily as a requirement for comprehensive electronic fault models. Consideration of multiple faults and sequences of faults is dealt with in other sections of this standard.

See also: Section 14.5 (e.g., counterfeit components), Section 14.4 (e.g., use of correct components in manufacturing)

6.2.4 **The sensor fault model shall include an acceptably broad set of potential run-time as well as fabrication faults and failures.**

6.2.4.1 **MANDATORY:**

a) Sensor failure

b) Faults, corruption, data LOSS, and integrity LOSS in sensor data

c) Physical sensor compromise (partial or total failure)

d) Sensor component degradation

EXAMPLES: Sensitivity, LOSS of calibration, violated temperature specification, abrasion, wear & tear

e) Adverse sensor environmental conditions in operation and in storage

EXAMPLES: Rain, water splash, mud, icing, dirt, low/high temperatures, low/high humidity

f) LOSS of sensor alignment or calibration

g) Transient sensor faults

h) Communication failure

i) Timing failure

EXAMPLES: Late or incorrectly time stamped data, excessive sensor reporting latency

j) Configuration fault

EXAMPLES: Configuration data, incompatible version

6.2.4.2 **REQUIRED:**

a) Malicious mechanical physical sensor compromise

EXAMPLES: Defacement, alignment compromise, gouged optics, blunt force impact

6.2.4.3 **HIGHLY RECOMMENDED:**

a) Malicious computer attack on externally accessible physical sensor compromise if within scope for the security plan

EXAMPLES: Via access to vehicle network; malicious sensor software update

6.2.4.4 **RECOMMENDED – N/A**

6.2.4.5 **CONFORMANCE:**

CONFORMANCE is checked via inspection of design and V&V evidence as well as DEMONSTRATION.

6.2.5 **The communication fault model shall include an acceptably broad set of potential run-time as well as fabrication faults and failures.**

6.2.5.1 **MANDATORY:**

a) Identified fault model for communications

6.2.5.2 **REQUIRED:**

a) Communication link LOSS

b) Communication packet errors

EXAMPLES: Random bit flips, burst errors, packet LOSS

c) Congestion

EXAMPLES: Excessive latency, repetition, insertion

d) Channel overload

EXAMPLE: Babbling idiot

e) Communication timing

EXAMPLES: High latency, early messages, large latency jitter

f) Faults, corruption, data loss, and integrity LOSS in data from external sources

EXAMPLES: Masquerade faults, message collisions, channel interference

g) Data integrity check failures

EXAMPLES: Error detection code has insufficient Hamming Distance for operational environment; error detection capability implemented incorrectly, providing reduced bit error detection ability

h) Time-based synchronization and time keeping failures

EXAMPLES: Inconsistent handling of time keeping discontinuities such as leap second, insufficiently ROBUST time synchronization in the presence of a faulty node, time keeping differences between ITEM and remote devices

i) Human communication malfunctions or failures, including at least:

1) Incorrect behavior of communication features

2) Obscuration of communication features

EXAMPLES: Due to ice, snow, lighting conditions, glare, mud, viewer use of polarized sunglasses, vehicle aspect, vehicle distance, vehicle speed

3) Failure of human to notice communication features

4) Incorrect interpretation by human of communication features

5) Errors of human omission

6) Errors of human commission

7) Human slip errors

NOTE: A “slip” is an incorrect action followed relatively quickly by human self-correction

8) Willful or defiant failure to comply with intent of communication features

6.2.5.3 **HIGHLY RECOMMENDED:**

a) Incompatible terminal in network

6.2.5.4 **RECOMMENDED:**

a) Malicious denial of service if within scope for the security plan

b) Malicious masquerade attacks if within scope for the security plan

c) Other malicious communication faults if within scope for the security plan

6.2.5.5 **CONFORMANCE:**

CONFORMANCE is checked via inspection of design and V&V evidence as well as DEMONSTRATION.

6.2.6 **The data fault model shall include an acceptably broad set of data-related faults and failures.**

6.2.6.1 **MANDATORY:**

a) Data storage faults

b) Data transmission faults

6.2.6.2 **REQUIRED:**

a) Data value faults

1) Detected data value corruption

2) Undetected data value corruption

3) Incorrect data value

4) Incorrect data format and/or units

5) Stale data value

EXAMPLE: Data value not updated despite incrementing time stamp

6) Malicious data value faults in accordance with security plan

7) Inadequate data value size

EXAMPLE: 64-bit floating point value converted to 16-bit integer value results in overflow

b) Metadata faults and related faults

1) Corrupted metadata

2) Incorrect metadata

3) Incorrect versioning and/or configuration information

4) Incorrect sender and/or receiver for message

5) Data routing faults

6) Invalid time stamp information

EXAMPLES: Inaccurate time stamp, time stamp rollover, time stamps go “backward” in time