AI-DAL: Towards Security Design Assurance for Artificial Intelligence Systems in Production

Susanna Cox

disesdi.susannacox@owasp.org

OWASP AI Exchange

08 October 2024

Introduction

Recent years have seen unprecedented development of machine learning (ML) and artificial intelligence (AI) applications. The explosive growth of AIML applications both in scale and scope has cemented AI technology as an integral feature in many aspects of society, including personal, social, and economic [1][2]. In parallel to this development, AI system-specific security threats have arisen [3][4][5][6][7]. These threats continue to evolve at a rapid pace, comparable to the overall rate of AI state-of-the-art (SOTA) advancement.

The wide and consequential application of AI systems, coupled with their risks of and consequences for failure, introduces the need for regulatory oversight [8][9][10][11]. However, the regulation of AI development controls, including metrics for sufficient application of mitigations where degrees or combinations of mitigations may exist, remains an open problem [12][13][14][15].

Background & Challenges

Regulating AI security controls is difficult for a number of reasons, including diversity of the field, its systems, and the explosive growth in industry; considerations for what is feasible & reasonable to require in an industry dominated by ad-hoc development practices; and the high-dimensional nature of the problem itself. Additional considerations include means of testing & verifying both efficacy and compliance, as well as how regulations can and should adapt to future SOTA developments. These factors contribute to a complexity which makes fairly, consistently and systematically determining when security mitigations are “good enough” — and doing so at regulatory scale — a highly technical and challenging policy problem.

Diversity of the field, its systems, and the explosive growth in industry. Due to the breadth of potential applications, as well as increases in resource allocation for AI development in industry, AI in production is an incredibly diverse field. This includes a wide variety of industry use cases, with differing failure modes & consequences. It also includes the diversity of systems which make up AI in production, each with its own ever-shifting attack surfaces and threat models [16][17]. As an example, regulating for AI supply chain risk alone is known to be a multifaceted and complex consideration [18]; factoring these threat models into the failure modes & effects analyses (FMEAs) [19] of up- and downstream interdependent system(s) creates a complexity that threatens to make regulation infeasible.

Considerations for what is possible & reasonable in a largely ad-hoc industry. Regulators must also take into account what is feasible and reasonable to require, particularly in an industry that is notoriously ad-hoc in its development methodologies [20][21][22]. Creating unnecessary burdens on practitioners may have negative consequences for the industry, hampering AI security innovation, and increasing incentives to game the system.

High-dimensional nature of the problem. Setting specificity of mitigation levels is difficult for the same reason that setting cohesive, applicable standards across all use cases is difficult, since in many cases requirements may become progressively more expensive or complex as security mitigations become more sophisticated, and because not all use cases may require such mitigations. The problem thus becomes standardization across intersecting

high-dimensional spaces: attacks, mitigations, and use cases. Collapsing these into a human-readable (and regulatory-friendly) system is imperative.

Testing & verification. Another problem is how to verify compliance, as well as the efficacy of mitigations and/or overall system robustness. Testing the security robustness of individual AI systems could quickly become resource-prohibitive at regulatory scale. Regulators need a standardized, universally applicable, rapidly reviewable, and legally actionable artifact set that is clearly defined and reasonable in its scope and application, such that it is feasible for developers to produce.

Regulatory brittleness. Because the AI security field is swiftly evolving and expanding, regulators face the challenge of developing a framework which is both comprehensive and specific enough to provide a measure of verifiable security assurance, while remaining flexible to respond to shifts in the global threat landscape and state-of-the-art [10]. If the regulations are not comprehensive or specific enough, they will be impossible to measure and enforce; if they are too specific, they will become obsolete before the metaphorical ink has dried. A means of understanding, systematizing, and adapting regulation within this constantly shifting space in a manner that is accessible to a wide variety of stakeholder audiences is thus required.

Safety-Critical Systems-of-Systems

The rapid growth and vast application potential of AI in industry & society brings with it the increasing possibility for the security failures of AI systems to cause harm to humans [23]. In this light, *safety*—a concept distinct from, but related to, *security*—should play an ever-increasing role in architectural considerations [24].

For software development practices where safety may be on the line, practitioners may look to the fields of Safety-Critical Systems (SCS), and safety-critical software engineering [25]. There is a large body of work, both theoretical and applied, in safety-critical software engineering in particular which may be of useful application to AI security regulation [26][27]. There is additionally work in the area of safety-critical artificial intelligence system (SCAIS) applications, including embedded & autonomous systems [28][29][30][31][32][33][34][35][36]; however, detailed discussion of SCAIS engineering principles is outside the scope of this paper.

While broadly requiring safety-critical software engineering standards in AI system development may seem like a panacea, such a solution may not be realistic given the current SOTA. It should be noted in this context that the problem of exactly how to conduct and document analysis of software systems remains an open research question, beyond the scope of this paper [37][38][39][40]. Additionally, implementing such a requirement without prescribing the production of structured artifacts for regulatory review would leave regulators with no metrics for compliance or enforcement mechanisms. Although there is a body of work on the integration of safety-critical engineering principles into modern software development practices, such as Agile methodologies [41][42][43], operationalizing such analyses and further adapting them to regulatory oversight is also beyond the scope of this paper. Finally, prescriptive, top-down requirements for security controls in the design phases may be antithetical to, and hinder progress of, the typically iterative development cycles of AIML systems in production.

A further consideration is the expense of safety-critical software creation. Development of safety-critical systems can be significantly more expensive than traditional software, with some estimates placing it as high as 20-30 times more costly [44]. Hastily introducing expensive requirements to the AI development process could create an undue burden on industry, and add incentives to game the system—producing the opposite of the desired regulatory effect and lessening overall industry security robustness.

Rather than reinventing approaches to software safety or its regulation, it may be useful to look to existing and heavily-regulated safety-critical software domains. One such domain is software engineering for aerospace, specifically commercial aviation.

Commercial aviation's safety record is staggering. According to the International Air Transport Association (IATA), 2023 saw 0.80 accidents per million flights operated. This represents a 61% ten-year decrease from 2014. According to the IATA Annual Safety Report "*...on average a person would have to travel by air every day for 103,239 years to experience a fatal accident,*" making air travel by far the safest mode of transit [45].

Airworthiness certification requirements, standardized operating procedures, and a voluntary culture of safety in industry are among factors contributing to commercial aviation's extraordinary record [46][47]. For aircraft software, the process(es) of airworthiness certification are of particular importance. Flight crews rely on software systems at virtually every level of aircraft operation. Even with the use of highly complex and interdependent mission-critical software, commercial aviation retains an astounding success rate for safe operations at scale.

Safety and security are not the same; they are conceptually related applications of *protection* which may occur at different phases of the mission lifecycle [48]. Security measures may create or contribute to mission safety [49]. While the global aviation industry is well known for its security operations, the focus of this paper is on non-security measures contributing to aviation's impressive safety record; specifically, the analysis of air missions as Systems of Systems (SoS), and the systematized qualification of component contribution to overall mission success [50][51][52]. In fact, SoS analysis remains critical to safety in many complex applications:

Safety uses systems theory and systems engineering to prevent foreseeable accidents and minimize the consequences of unforeseeable accidents...[It] is a planned, disciplined and systematic strategy for identifying, analyzing, evaluating, eliminating and controlling hazards throughout the system's life cycle in order to prevent or reduce the number of accidents [53].

Much like avionics and aircraft systems, artificial intelligence deployments in production, as well as the results of their interactions with users, subjects, and society, can and should be treated as systems-of-systems with complex and interacting failure modes and consequences.

Digitalization in Aerospace and the Rise of Software Airworthiness Certification

Early planes were controlled via systems of onboard cables, some of which might run the full length of the aircraft, adding significant weight. While providing near-direct connectivity between crew inputs and flight control surfaces, these systems required both additional maintenance and engineering considerations for their weight. With the advent of so-called “fly-by-wire" technologies, flight control inputs were mediated by digital controllers. At the same time, advancements in computing allowed for the increasing digitalization of aviation systems across the board. These fly-by-wire apparatuses, and various electronic aviation systems, or “avionics,” quickly became the design standard, reducing crew workloads, eliminating the need for heavy control cabling, and removing considerable weight requirements from aircraft design specifications [54]. The newest generations of fly-by-wire commercial jets have introduced further improvements in safety [55].

With digitalization in aviation, and the software that powered it, came the need for airworthiness certification of component systems [56][57][58]. Aircraft systems must operate deterministically and reliably, in real time, with minimal latency and quantifiable failure rates [53][57][59]. Regulatory validation of software requirements across manifold aviation components became a question analogous to the AI security controls issue in its complexity and importance.

Regulatory Assessment via Applied Design Assurance Levels

Airworthiness certification is handled primarily by regulatory agencies; e.g. in the EU, by the European Union Aviation Safety Agency (EASA), and by the Federal Aviation Administration (FAA) in the United States [60][61]. Additionally, the International Civil Aviation Organization (ICAO) plays an intergovernmental role in setting international civil aviation standards [62].

In 1992, a joint effort by the US-based RTCA, Inc. (previously known as the Radio Technical Commission for Aeronautics) and the European Organisation for Civil Aviation Equipment (EUROCAE) produced an international guideline set for the production of safety-critical software systems in aircraft, titled DO178-B [56]. This guideline was updated and replaced in 2012 by the DO178-C. For software airworthiness certification, both EASA and the FAA reference the DO178-C [63].

The DO178-B introduced the concept of software *Design Assurance Levels* (DALs), which are continued in the DO178-C [53][57][63][64]. There are five DALs, corresponding to five conceptual levels of impact to safety. These range from Level E, *No Effect*, in which component failure does not impact mission safety, to Level A*, Catastrophic*, in which component failure may result in death and/or loss of the aircraft (i.e. total mission failure).

Software components are evaluated more rigorously based on their contribution to overall mission safety/success. This contribution is considered across three primary vectors: *aircraft, crew,* and *passengers*. Systems or components are assigned an auditable DAL based on the worst-case potential safety impact they may be expected to have across each of these vectors. The five aerospace DALs are summarized in the table below.

DAL	Failure Condition	Resulting Aviation Condition
Level A	Catastrophic	Death, loss of aircraft.
Level B	Hazardous	Large negative impact on safety or performance. May reduce ability of crew to operate aircraft due to increased workload and/or physical stress. May cause serious or fatal passenger injuries.
Level C	Major	Failure significantly reduces safety margin OR significantly increases crew workload. May result in passenger discomfort or minor injuries.
Level D	Minor	Failure reduces safety margin or causes increase in crew workload. May result in passenger inconvenience.
Level E	No Effect	Failure causes no impact on safety, crew workload, or aircraft operation.

Table 1: Five Aerospace DALs from the DO178-C

Methods

AI for aerospace is a relatively new but growing field with a number of research avenues [33][51][52][65][66], including works acknowledging the difficulty of certifying AI systems under traditional airworthiness certification processes, and proposals for an AI-adapted certification process specific to SCAIS [67]. While analogies exist between mission-critical AIML systems and aerospace, AIML systems are not aircraft, nor do they (always) concern passenger transport. Adapting a design assurance tiering scheme to broad AI system application thus requires a process of structured abstraction and redefinition/reapplication. For reproducibility, that process is detailed here.

Abstracting Design Assurance Tiers to AIML System Application: Vectors

Design Assurance Levels take into account the effects of a software component or system's failure across three vectors: *aircraft, passengers,* and *crew*. Abstracting from each of these concepts a core representation of its relationship to safety within the context of airline transport yields three new conceptual fields, which may then be applied to create AIML system analogies.

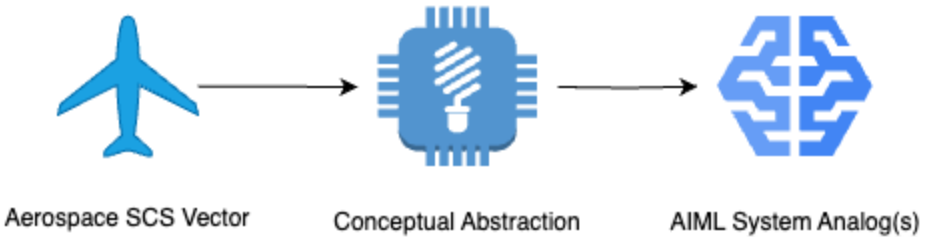

Figure 1: Design assurance tier abstraction process

Aircraft, when considered in the context of their role in safety, may be thought of as representing the overall mission, since loss of the aircraft will likely result in mission failure, injury or death. In an engineering context, aircraft also represent Systems of Systems (SoS), which have their own complexities and considerations, and which, for the purposes of this paper, may be considered analogous to the many components of a production AI system [68].

Application of the SoS abstraction to the context of production AIML systems yields an industry analog in *composite/constituent AIML systems*. Most production AI is composed of numerous subsystems, and AI systems themselves may be components. The analogous AIML system failure mode(s) for this vector include impacts to mission-critical technical and/or sociotechnical composite systems of which the AI system or subcomponent is a constituent, resulting in significant social, economic, or other harm. The focus of the analysis remains on system/mission failure, rather than individual model performance or loss due to security breach.

As an example of a production AI system whose performance & reliability impact larger sociotechnical systems, consider an AIML vendor providing biometric authentication services to banks. A failure or outage of such a service could have considerable negative effects on users' ability to perform banking tasks, and, depending on the size and nature of the vendor's user base, could potentially have measurable economic impact.

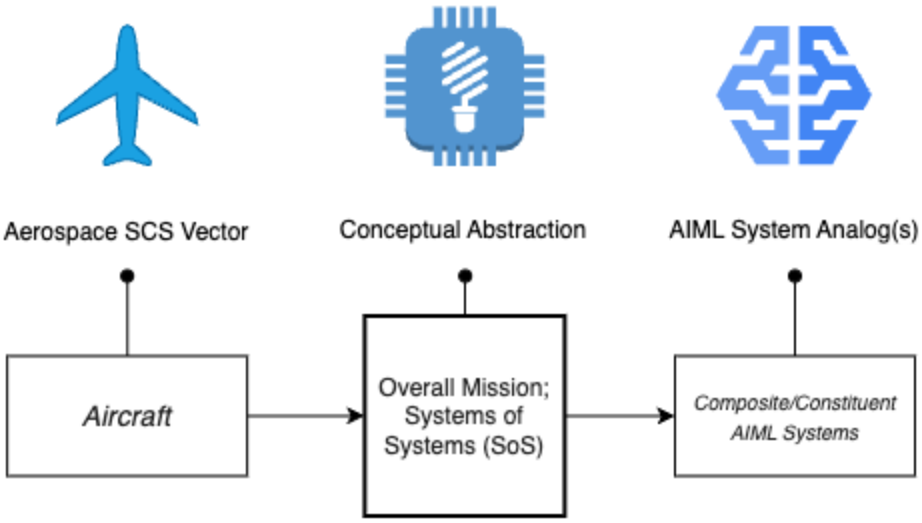

Figure 2: Mapping the contextual relationship between aircraft and system safety to constituent AIML systems

Aircraft crew/operators may be thought of as representing the system's operation and maintenance, and are the last line of responsibility for mission success. In mission-critical AIML contexts, the concept of operators refers to both human and organizational factors. In the context of production AIML this may be compared to both the human factors within a system, such as researchers, engineers, and data scientists, and non-human drivers contributing to mission success, such as organizational culture and standard operating procedures (SOPs).

At the highest level, aircraft crew represent the aircraft/system's capacity for continued operation. This representation is conceptually simplified here and applied to AIML systems broadly as the organization responsible for the system in production. Failure along this vector results in significant impacts to organizational operation, potentially resulting in economic loss, social harm, or other secondary damage.

Inclusion of this vector is designed in part to prevent circumstances in which organizations "too big to fail"--i.e. organizations whose size or footprint are so large that their potential economic distress might spill over beyond the organization itself--might otherwise be exempted from a more stringent AI-DAL due to their specific application, but whose failure might realistically cause problematic secondary effects.

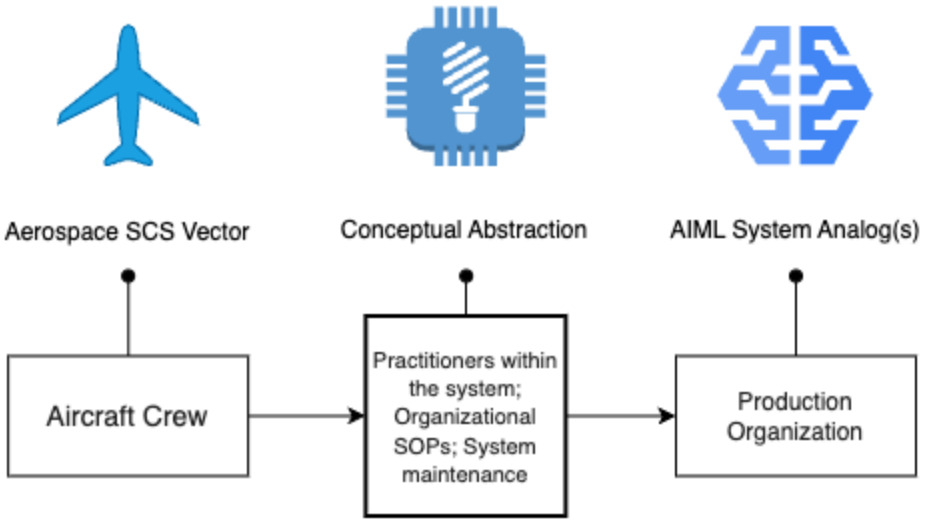

Figure 3: Crew as representative of system maintenance and the production organization

Within the airline transport analogy, the *passengers* vector represents, very simply, human and social factors affected by a system either through their interaction with said system, or via secondary effects. While airline transport considers mission success in the context of effects on passengers, crew, and aircraft which are being operated, the powerful and potentially far-reaching consequences of AIML applications require the additional consideration of *societal impacts.*

Due to the scale of AI systems' application, multiple vectors for human impacts exist. It should be noted here that humans interacting with an AIML system (outside the system's engineering and maintenance) may be users/consumers (as in the case of a commercial application), or they may be subjects, such as in the use of loan approval or criminal sentencing algorithms.

Humans may be affected by direct interaction with AI systems as users/consumers (*for whom* AI services are provided), as subjects (*to whom* AI services are applied), or as non-participants impacted by the unintended social, economic, or other consequences of these systems at scale. This vector is also defined so as to include societal harms which may arise outside of damage directly to composite sociotechnical systems.

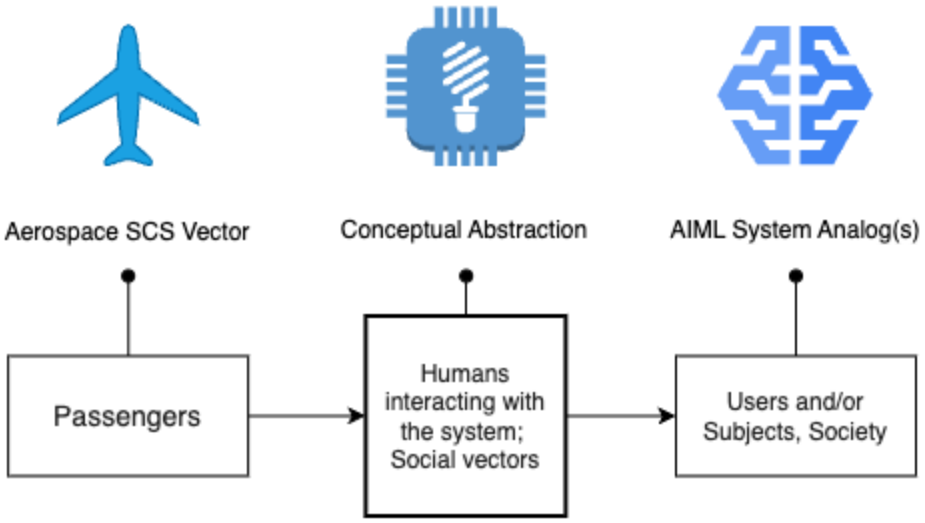

Figure 4: Mapping human factors, whether aircraft passengers or AIML system subject/users

Human Factors: Societal vs Personal Disruption, Users vs Subjects

The human factors, societal and personal, each have their own vectors of potential damage/disruption. Individuals may be subject to economic harm, including serious financial loss. They may suffer from mental or emotional harm, including serious emotional and/or social-emotional distress. Finally, in the most extreme of circumstances, they may be subject to physical harm, including injury or death.

Similarly, society-level risks include large-scale economic loss or destruction; mass emotional distress and/or community/relationship stress which may impact other societal systems; and civil unrest or disobedience, including the collapse of legal or financial structures. A realistic worst-case scenario analysis of failure modes & effects potentially places a number of currently widely-adopted AI applications in this risk category.

Abstracting & Re-Applying Conceptual Risk Analysis

In order to analogize aerospace design assurance tiers to AIML security requirements, a core representation of conceptual risk was abstracted from each DAL, and correlated to AIML analogs. Analog conceptual risk abstractions were then applied to analysis along their corresponding AIML vectors.

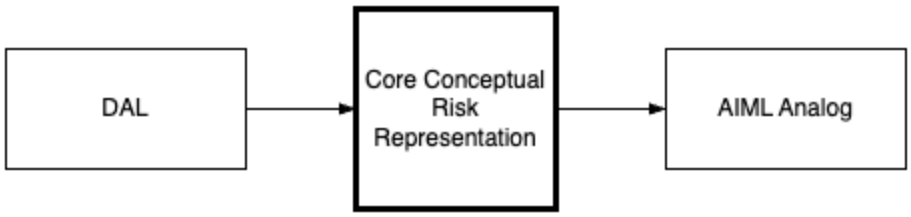

Figure 5: Core conceptual risk extraction and mapping

Design Assurance Level (DAL) E, *No Effect*, falls outside of regulatory scope. DAL D, *Minor*, refers to any failure which reduces safety margins, or causes increase in crew workload. Such failures may result in passenger inconvenience. This may be understood broadly as *safety reduction and inconvenience*.

Applying this conceptual risk abstraction along the AIML-analog vectors of *Constituent Systems, Production Organization* and *Human/Society* gives the AI Security Assurance Level (AI-DAL) D. The conditions resulting from failure of an AI system assigned to this tier may include inconvenience to users, and organizational costs incurred for incident response & recovery, model/data theft/loss, etc. Most organizations deploying AIML systems in production will at a minimum fall into this security assurance tier.

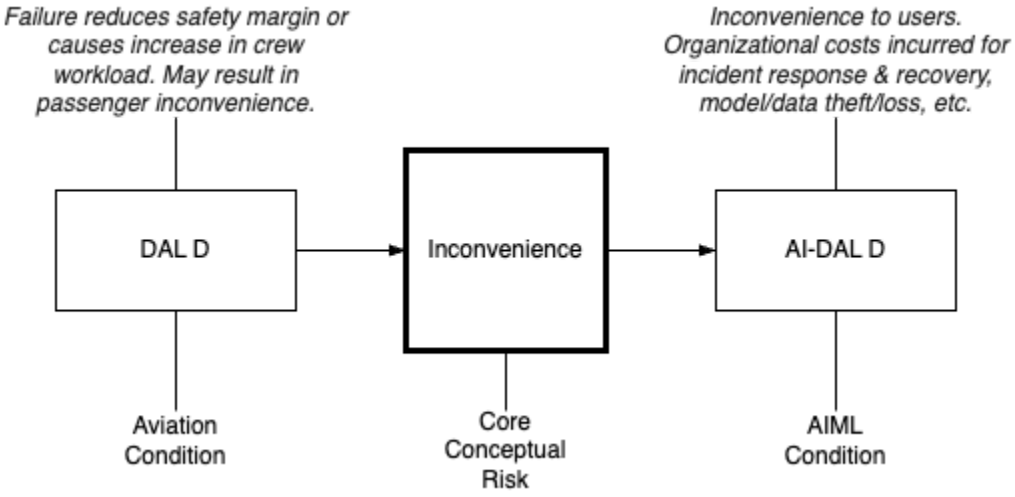

Figure 6: AI-DAL D conceptual risk mapping

DAL C, *Major*, refers to failures which significantly reduce the safety margin OR significantly increase crew workload, and/or may result in passenger discomfort or minor injuries. This may be understood as *danger, minor injury, discomfort, or generally contributing to an unsafe situation.*

Applying this concept to AIML analog vectors gives AI-DAL C. System failure impacts include economic, social, or emotional harm to humans (as users or subjects) at any scale; failure of composite non-critical systems; and/or organizational failure. Due to the increased potential for harm in the case of mission failure, organizations must bear a higher responsibility for security when their AI models are subcomponents of larger systems. Similarly, a greater degree of potential impact requires a greater degree of care when humans interacting with the system may include subjects as well as users. Finally, because total organizational failure often results in harms to a variety of stakeholders, systems with this potential scale of impact require additional security considerations.

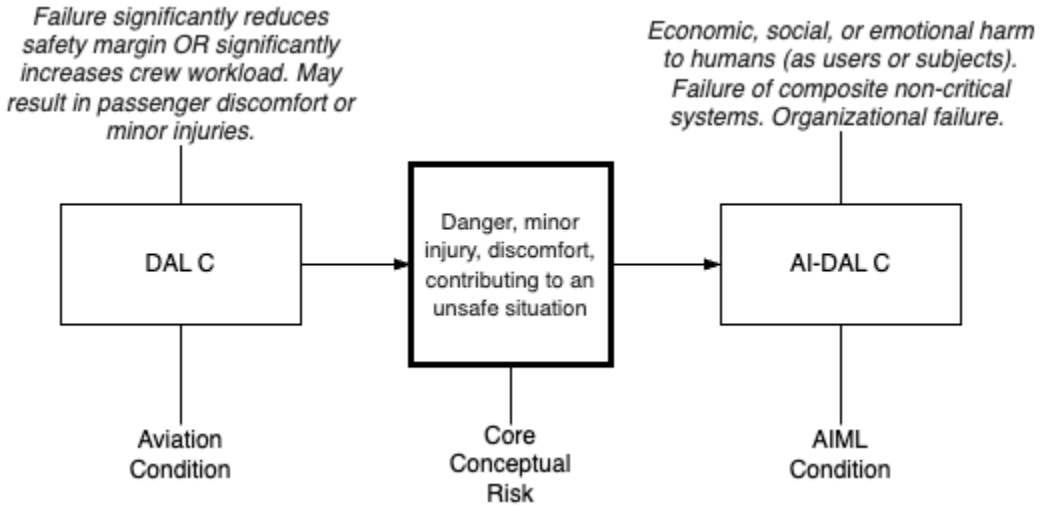

Figure 7: AI-DAL C conceptual risk mapping

DAL B, *Hazardous*, refers to failures which have a large negative impact on safety or performance, potentially reducing the ability of crew to operate due to increased workload and/or physical stress, and/or causing serious or fatal passenger injuries. This may be broadly understood as *significant danger, significant or fatal injury, and/or significant loss.* Analogous AIML failure conditions within the AI-DAL B tier include significant economic, social, or emotional harm to humans; organizational failure resulting in significant societal impacts; and failure of composite mission-critical systems. Because of the increased severity of their impacts, organizational failure with societal consequences, as well as dependencies in mission-critical composite systems, are treated as a distinct tier of security assurance requirements.

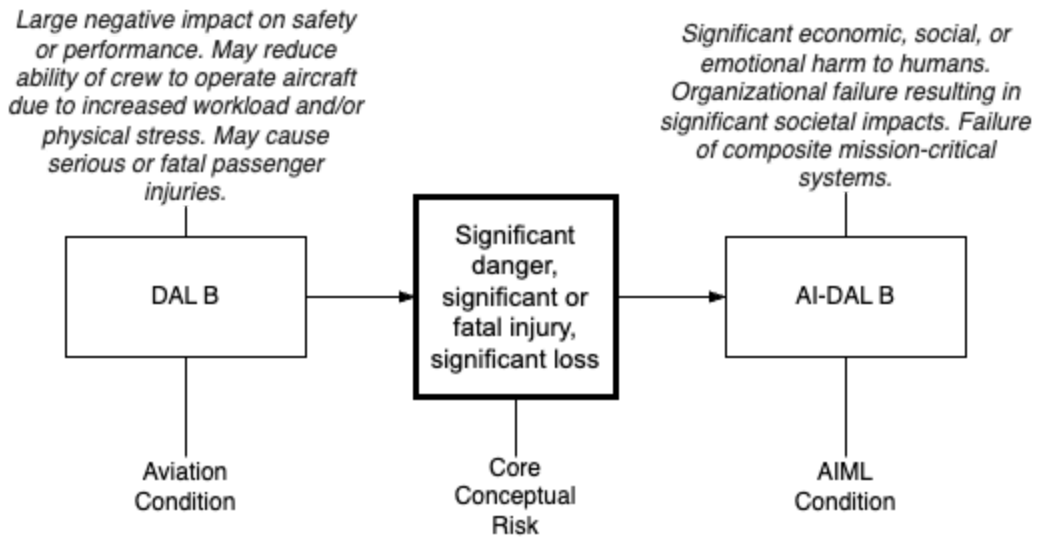

Figure 8: AI-DAL B conceptual risk mapping

Finally, DAL A, *Catastrophic*, refers to human death or loss of the aircraft. This may be conceptually represented as *total mission failure, catastrophic loss and/or death*. Applied as AI-DAL A, analogous AIML system failure impacts include physical harm to humans, up to death; and/or mass societal disruption including civil unrest, and legal or financial collapse. It should be noted that any degree of physical harm to humans or mass societal unrest caused by AI system failure should be considered unacceptable. A full mapping of aerospace risk profiles and tiers to AI-DALs is given in the table below.

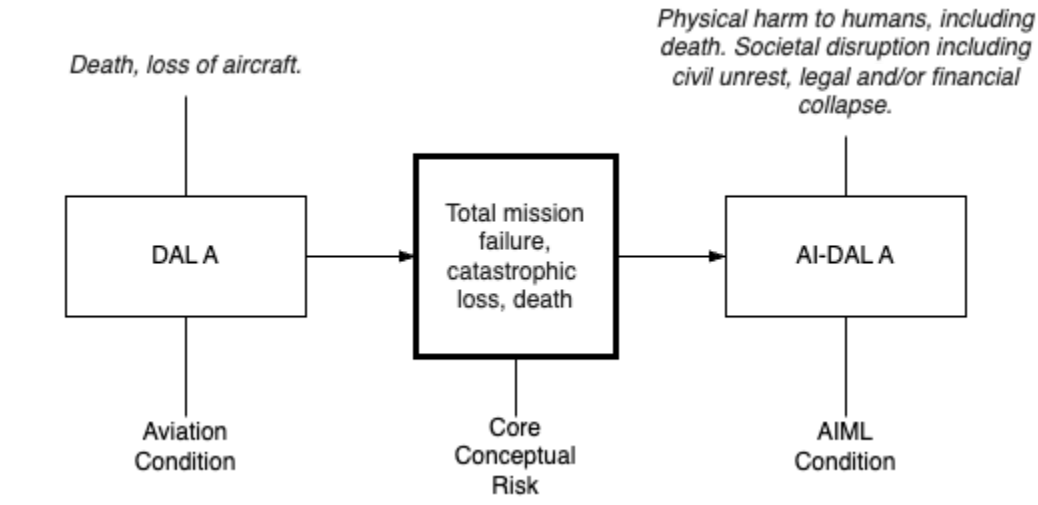

Figure 9: AI-DAL A conceptual risk mapping

DAL	Failure Condition	Resulting Aviation Condition	Conceptual risk	AI-DAL	Resulting AIML Condition
Level A	Catastrophic	Death, loss of aircraft.	Total mission failure, catastrophic loss, death	AI-DAL A	Physical harm to humans, including death. Societal disruption including civil unrest, legal and/or financial collapse.
Level B	Hazardous	Large negative impact on safety or performance. May reduce ability of crew to operate aircraft due to increased workload and/or physical stress. May cause serious or fatal passenger injuries.	Significant danger, significant or fatal injury, significant loss	AI-DAL B	Significant economic, social, or emotional harm to humans. Organizational failure resulting in significant societal impacts. Failure of composite mission-critical systems.
Level C	Major	Failure significantly reduces safety margin OR significantly increases crew workload. May result in passenger discomfort or minor injuries.	Danger, minor injury, discomfort, contributing to an unsafe situation	AI-DAL C	Economic, social, or emotional harm to humans (as users or subjects). Failure of composite non-critical systems. Organizational failure.
Level D	Minor	Failure reduces safety margin or causes increase in crew workload. May result in passenger inconvenience.	Inconvenience	AI-DAL D	Inconvenience to users. Organizational costs incurred for incident response & recovery, model/data theft/loss, etc.
Level E	No Effect	Failure causes no impact on safety, crew workload, or aircraft operation.	No Impact	AI-DAL E	No impact to organization or human/social vectors such as users, subjects, or society.

Table 2: AI-DAL/aerospace DAL tier mappings

Artifacts

This paper further proposes the production of two artifacts which are intended to streamline regulatory review, and serve as documentation of the personnel, processes, and procedures contributing to system development. First, organizations should produce a Plan for AI Software Aspects of Certification (PAISAC), analogous to the aerospace Plan for Software Aspects of Certification (PSAC), which gives information on development compliance efforts. Detailing the particular aspects of the PSAIC falls outside the scope of this paper.

Second, organizations should produce traceable documentation of AI-DAL tier-specific requirements compliance. This work leaves open both the methodologies for producing this

documentation, as well as the potential for future contribution of traceability mandates to regulation around AIML Bills of Materials (AIMLBOMs).

AI-DAL Benefits

Regulatory Flexibility. AI security is a rapidly developing field, with actors on all sides racing to secure—or attack—AI systems. A potential pitfall of attempting to specify prescriptive security remediations is an inability for regulators/regulations to adapt quickly enough. The application of design assurance levels to evaluate AI systems provides a framework which can be flexibly adjusted by threat and use case, as security threat landscapes evolve. Regulating by system use-case & accompanying impacts, and allowing for adjustment of required mitigations for each threat within the context of a system's AI-DAL, is intended to reduce the potential for future regulatory brittleness in the face of a rapidly-evolving technological landscape.

Security Focus. This analysis is not intended to address the ethical application of AI; only to provide regulatory ability to quantify requirements for security enforcement in mission-critical applications. We are not concerned with where the airplane is going when we analyze its constituent software components; similarly, the specific application of the AI system is of little concern to this analysis, except inasmuch as it pertains to the effects of mission failure on larger societal structures.

Simplicity. The goal of this analysis is to provide an accessible, fair, and socially beneficial means of applying AI security regulation, a particularly difficult challenge in a rapidly evolving and high-dimensional problem space. This is accomplished by compressing multiple dimensions into an analysis which focuses solely on the consequences for mission failure. We need not be concerned with risks arising from specific threat vectors, or their likelihood of occurrence, in order to complete this analysis. We need only to consider the likely effects of *total mission failure* of the AI system, quantifying the failure mode(s), and assigning an AI Security Assurance Level (AI-DAL) that is commensurate with the failure condition.

Conclusion

The rise of artificial intelligence applications in society, and their accompanying security concerns, has created a need for regulatory oversight that is auditable, actionable, and adaptable to a rapidly changing technological landscape. Methods from safety-critical software engineering, particularly aerospace, may be adapted to use in production AIML to aid both practitioners and regulators in establishing design thresholds for AIML system security. Assignment of AI Design Assurance Levels (AI-DAL) to projects/components, along with production of related compliance artifacts, is proposed as a means of consistently applying appropriate design requirements based on a system's potential adverse impact.

It is hoped that AI-DAL will serve as a basis upon which applied expertise can be used to iteratively determine appropriate requirements for security mitigations, in a manner which is fair, consistent, and flexible to accommodate future AIML innovations. As the SOTA advances, time is of the essence; the next great AI breakthrough, and the next AI security threat, lie just over the horizon.

References

1. Ashmore, Rob, Radu Calinescu and Colin Paterson. “Assuring the Machine Learning Lifecycle.” ACM Computing Surveys (CSUR) 54 (2019): 1 - 39.
2. Chivu, Stefania Ileana. “The Macroeconomic Impact of Artificial Intelligence.” International Journal of Sustainable Economies Management (2022): n. Pag.
3. Wang, Xianmin, Jing Li, Xiaohui Kuang, Yu-an Tan and Jin Li. “The security of machine learning in an adversarial setting: A survey.” J. Parallel Distributed Comput. 130 (2019): 12-23.
4. Chen, Huaming and M Ali Babar. “Security for Machine Learning-based Software Systems: a survey of threats, practices and challenges.” ArXiv abs/2201.04736 (2022): n. Pag.
5. Oseni, Ayodeji, Nour Moustafa, Helge Janicke, Peng Liu, Zahir Tari and Athanasios V. Vasilakos. “Security and Privacy for Artificial Intelligence: Opportunities and Challenges.” ArXiv abs/2102.04661 (2021): n. Pag.
6. Papernot, Nicolas, Patrick Mcdaniel, Somesh Jha, Matt Fredrikson, Z. Berkay Celik and Ananthram Swami. “The Limitations of Deep Learning in Adversarial Settings.” 2016 IEEE European Symposium on Security and Privacy (EuroS&P) (2015): 372-387.
7. Evtimov, I., Kevin Eykholt, Earlence Fernandes, Tadayoshi Kohno, Bo Li, Atul Prakash, Amir Rahmati and Dawn Xiaodong Song. “Robust Physical-World Attacks on Deep Learning Models.” arXiv: Cryptography and Security (2017): n. Pag.
8. Smuha, Nathalie A.. “From a ‘race to AI’ to a ‘race to AI regulation’: regulatory competition for artificial intelligence.” Law, Innovation and Technology 13 (2021): 57 - 84.
9. Hoffmann-Riem, Wolfgang. “Artificial Intelligence as a Challenge for Law and Regulation.” Regulating Artificial Intelligence (2019): n. Pag.
10. Bengio, Yoshua, Geoffrey Hinton, Andrew Yao, Dawn Song, Pieter Abbeel, Yuval Noah Harari, Ya-Qin Zhang, Lan Xue, Shai Shalev-Shwartz, Gillian K. Hadfield, Jeff Clune, Tegan Maharaj, Frank Hutter, Atilim Gunecs Baydin, Sheila A. McIlraith, Qiqi Gao, Ashwin Acharya, David Krueger, Anca Dragan, Philip Torr, Stuart Russell, Daniel Kahneman, Jan Markus Brauner and Sören Mindermann. “Managing AI Risks in an Era of Rapid Progress.” ArXiv abs/2310.17688 (2023): n. Pag.
11. Gil, Yolanda and Bart Selman. “A 20-Year Community Roadmap for Artificial Intelligence Research in the US.” ArXiv abs/1908.02624 (2019): n. Pag.
12. Mökander, Jakob, Marian Axente, Federico Casolari and L. Floridi. “Conformity Assessments and Post-market Monitoring: A Guide to the Role of Auditing in the Proposed European AI Regulation.” Minds and Machines 32 (2021): 241 - 268.
13. Jobin, Anna, Marcello Ienca and Effy Vayena. “Artificial Intelligence: the global landscape of ethics guidelines.” ArXiv abs/1906.11668 (2019): n. Pag.
14. Hagendorff, Thilo. “The Ethics of AI Ethics: An Evaluation of Guidelines.” Minds and Machines 30 (2019): 99 - 120.

15. Lucaj, Laura, Patrick van der Smagt and Djalel Benbouzid. “AI Regulation Is (not) All You Need.” Proceedings of the 2023 ACM Conference on Fairness, Accountability, and Transparency (2023): n. Pag.
16. Ruf, Philipp, Manav Madan, Christoph Reich and Djaffar Ould-Abdeslam. “Demystifying MLOps and Presenting a Recipe for the Selection of Open-Source Tools.” Applied Sciences (2021): n. Pag.
17. Sculley, D., Gary Holt, Daniel Golovin, Eugene Davydov, Todd Phillips, Dietmar Ebner, Vinay Chaudhary, Michael Young, Jean-François Crespo and Dan Dennison. “Hidden Technical Debt in Machine Learning Systems.” NIPS (2015).
18. Widder, David Gray and Dawn Nafus. “Dislocated accountabilities in the “AI supply chain”: Modularity and developers’ notions of responsibility.” Big Data & Society 10 (2022): n. Pag.
19. Schmittner, Christoph, Thomas Gruber, Peter P. Puschner and Erwin Schoitsch. “Security Application of Failure Mode and Effect Analysis (FMEA).” International Conference on Computer Safety, Reliability, and Security (2014).
20. Symeonidis, Georgios, Evangelos Nerantzis, Apostolos Kazakis and George A. Papakostas. “MLOps - Definitions, Tools and Challenges.” 2022 IEEE 12th Annual Computing and Communication Workshop and Conference (CCWC) (2022): 0453-0460.
21. Kreuzberger, Dominik, Niklas Kühl and Sebastian Hirschl. “Machine Learning Operations (MLOps): Overview, Definition, and Architecture.” IEEE Access 11 (2022): 31866-31879.
22. Mäkinen, Sasu, Henrik Skogström, Eero Laaksonen and Tommi Mikkonen. “Who Needs MLOps: What Data Scientists Seek to Accomplish and How Can MLOps Help?” 2021 IEEE/ACM 1st Workshop on AI Engineering - Software Engineering for AI (WAIN) (2021): 109-112.
23. Hendrycks, Dan, Mantas Mazeika and Thomas Woodside. “An Overview of Catastrophic AI Risks.” ArXiv abs/2306.12001 (2023): n. Pag.
24. Hendrycks, Dan, Nicholas Carlini, John Schulman and Jacob Steinhardt. “Unsolved Problems in ML Safety.” ArXiv abs/2109.13916 (2021): n. Pag.
25. Axelrod, C. Warren. “Applying lessons from safety-critical systems to security-critical software.” 2011 IEEE Long Island Systems, Applications and Technology Conference (2011): 1-6.
26. Ge, Xiaocheng, Richard F. Paige and John Mcdermid. “An Iterative Approach for Development of Safety-Critical Software and Safety Arguments.” 2010 Agile Conference (2010): 35-43.
27. Hatcliff, John, Alan Wassyng, Tim Kelly, Cyrille Comar and Paul L. Jones. “Certifiably safe software-dependent systems: challenges and directions.” Future of Software Engineering Proceedings (2014): n. pag.
28. Ozen, Elbruz and Alex Orailoglu. “Sanity-Check: Boosting the Reliability of Safety-Critical Deep Neural Network Applications.” 2019 IEEE 28th Asian Test Symposium (ATS) (2019): 7-75.
29. Ahmadilivani, Mohammad Hasan, Mahdi Taheri, Jaan Raik, Masoud Daneshtalab and Maksim Jenihhin. “A Systematic Literature Review on Hardware Reliability Assessment Methods for Deep Neural Networks.” ACM Computing Surveys 56 (2023): 1 - 39.

30. Bosio, Alberto, Paolo Bernardi, Annachiara Ruospo and Ernesto Sánchez. “A Reliability Analysis of a Deep Neural Network.” 2019 IEEE Latin American Test Symposium (LATS) (2019): 1-6.
31. Ruospo, Annachiara, Alberto Bosio, Alessandro Ianne and Ernesto Sánchez. “Evaluating Convolutional Neural Networks Reliability depending on their Data Representation.” 2020 23rd Euromicro Conference on Digital System Design (DSD) (2020): 672-679.
32. Xu, Dawen, Ziyang Zhu, Cheng Liu, Ying Wang, Shuang-xi Zhao, Lei Zhang, Huaguo Liang, Huawei Li and Kwang-Ting Cheng. “Reliability Evaluation and Analysis of FPGA-Based Neural Network Acceleration System.” IEEE Transactions on Very Large Scale Integration (VLSI) Systems 29 (2021): 472-484.
33. Athavale, Jyotika, Andrea Baldovin, Ralf Graefe, Michael Paulitsch and Rafael Rosales. “AI and Reliability Trends in Safety-Critical Autonomous Systems on Ground and Air.” 2020 50th Annual IEEE/IFIP International Conference on Dependable Systems and Networks Workshops (DSN-W) (2020): 74-77.
34. Cummings, Mary L.. “Rethinking the Maturity of Artificial Intelligence in Safety-Critical Settings.” AI Mag. 42 (2021): 6-15.
35. Cheng, Richard, Gábor Orosz, Richard M. Murray and Joel W. Burdick. “End-to-End Safe Reinforcement Learning through Barrier Functions for Safety-Critical Continuous Control Tasks.” AAAI Conference on Artificial Intelligence (2019).
36. Berkenkamp, Felix, Matteo Turchetta, Angela P. Schoellig and Andreas Krause. “Safe Model-based Reinforcement Learning with Stability Guarantees.” Neural Information Processing Systems (2017).
37. Vistbakka, Inna and Elena Troubitsyna. “Towards a Formal Approach to Analysing Security of Safety-Critical Systems.” 2018 14th European Dependable Computing Conference (EDCC) (2018): 182-189.
38. Quamara, Megha, Gabriel Pedroza and Brahim Hamid. “Multi-layered Model-based Design Approach towards System Safety and Security Co-engineering.” 2021 ACM/IEEE International Conference on Model Driven Engineering Languages and Systems Companion (MODELS-C) (2021): 274-283.
39. Miguel, Miguel A. de, Javier Fernández Briones, Juan Pedro Silva and Alejandro Alonso. “Integration of safety analysis in model-driven software development.” IET Softw. 2 (2008): 260-280.
40. Rouland, Quentin, Brahim Hamid and Jason Jaskolka. “Specification, detection, and treatment of STRIDE threats for software components: Modeling, formal methods, and tool support.” J. Syst. Archit. 117 (2021): 102073.
41. Gary, Kevin A., Andinet Enquobahrie, Luis Ibáñez, Patrick Cheng, Ziv Rafael Yaniv, Kevin Robert Cleary, Shylaja Kokoori, Benjamin Muffih and John Heidenreich. “Agile methods for open source safety‐critical software.” Software: Practice and Experience 41 (2011): n. pag.
42. Górski, Janusz and Katarzyna Lukasiewicz. “Assessment of Risks introduced to Safety Critical Software by Agile Practices - a Software Engineer's Perspective.” Comput. Sci. 13 (2012): 165-182.

43. Heeager, Lise Tordrup and Peter Axel Nielsen. “A conceptual model of agile software development in a safety-critical context: A systematic literature review.” Inf. Softw. Technol. 103 (2018): 22-39.
44. Zoughbi, Gregory, Lionel Claude Briand and Yvan Labiche. “Modeling safety and airworthiness (RTCA DO-178B) information: conceptual model and UML profile.” Software & Systems Modeling 10 (2011): 337-367.
45. “IATA Annual Safety Report Executive Summary and Safety Overview – 60th Edition.” 2023. IATA. https://www.iata.org/contentassets/a8e49941e8824a058fee3f5ae0c005d9/safety-report-executive-and-safety-overview-2023.pdf.
46. “FAA Aviation Safety Agency Priority Goal, Action Plan, FY 2023.” 2023. Performance.gov. https://assets.performance.gov/APG/files/2023/june/FY2023_June_DOT_Progress_Aviation_Safety.pdf.
47. Key, Kylie N., Peter T. Hu, Inchul Choi, and David J. Schroeder. 2023. “Safety Culture Assessment and Continuous Improvement in Aviation: A Literature Review.” Federal Aviation Administration.
48. Firesmith, Donald. “Engineering Safety and Security Related Requirements for Software Intensive Systems.” 29th International Conference on Software Engineering (ICSE '07 Companion) (2007): 169-169.
49. Steiner, Max and Peter Liggesmeyer. “Combination of Safety and Security Analysis - Finding Security Problems That Threaten The Safety of a System.” DECS@SAFECOMP (2013).
50. Vidot, Guillaume, Christophe Gabreau, Ileana Ober and Iulian Ober. “Certification of embedded systems based on Machine Learning: A survey.” ArXiv abs/2106.07221 (2021): n. Pag.
51. Henderson, Alex, Steven D. Harbour and Kelly Cohen. “Toward Airworthiness Certification for Artificial Intelligence (AI) in Aerospace Systems.” 2022 IEEE/AIAA 41st Digital Avionics Systems Conference (DASC) (2022): 1-10.
52. Raz, Ali K., Erik P. Blasch, Cesare Guariniello and Zohaib T. Mian. “An Overview of Systems Engineering Challenges for Designing AI-Enabled Aerospace Systems.” AIAA Scitech 2021 Forum (2020): n. Pag.
53. Baron, Claude and Vincent Louis. “Towards a continuous certification of safety-critical avionics software.” Comput. Ind. 125 (2021): 103382.
54. NASA. 2011. “Fly-by-Wire Systems Enable Safer, More Efficient Flight.” NASA Spinoff. https://spinoff.nasa.gov/Spinoff2011/t_5.html.
55. “Keeping Safety First: A Statistical Analysis of Commercial Aviation Accidents.” 2023. Airbus. https://www.airbus.com/en/newsroom/stories/2023-03-keeping-safety-first-a-statistical-analysis-of-commercial-aviation.
56. Lougee, Hoyt. “SOFTWARE CONSIDERATIONS IN AIRBORNE SYSTEMS AND EQUIPMENT CERTIFICATION.” (2001).
57. Kornecki, Andrew J. and Janusz Zalewski. “Software Certification for Safety-Critical Systems: A Status Report.” (2008).

58. Kornecki, Andrew J. and Janusz Zalewski. “Certification of software for real-time safety-critical systems: state of the art.” Innovations in Systems and Software Engineering 5 (2009): 149-161.
59. Alves-Foss, Jim, Paul W. Oman, Carol A. Taylor and Scott Harrison. “The MILS architecture for high-assurance embedded systems.” Int. J. Embed. Syst. 2 (2006): 239-247.
60. European Union. n.d. “European Union Aviation Safety Agency (EASA).” European Union. Accessed August 19, 2024. https://european-union.europa.eu/institutions-law-budget/institutions-and-bodies/search-all-eu-institutions-and-bodies/european-union-aviation-safety-agency-easa_en.
61. Federal Aviation Administration. n.d. “A Brief History of the FAA.” Federal Aviation Administration. Accessed August 19, 2024. https://www.faa.gov/about/history/brief_history.
62. The International Civil Aviation Organization. n.d. “About ICAO.” ICAO. Accessed August 19, 2024. https://www.icao.int/about-icao/Pages/default.aspx.
63. Youn, Wonsang, Seung Bum Hong, Kyung-Ryoon Oh and Oh-Sung Ahn. “Software certification of safety-critical avionic systems: DO-178C and its impacts.” IEEE Aerospace and Electronic Systems Magazine 30 (2015): 4-13.
64. Hilderman, Vance and Len Buckwalter. “Avionics Certification: A Complete Guide to DO-178 (Software), DO-254 (Hardware).” (2007).
65. Sutthithatip, Sujitra, Suresh Perinpanayagam, Sohaib Aslam and Andrew J. Wileman. “Explainable AI in Aerospace for Enhanced System Performance.” 2021 IEEE/AIAA 40th Digital Avionics Systems Conference (DASC) (2021): 1-7.
66. Chai, Runqi, Antonios Tsourdos, Al Savvaris, Senchun Chai, Yuanqing Xia and C. L. Philip Chen. “Review of advanced guidance and control algorithms for space/aerospace vehicles.” Progress in Aerospace Sciences 122 (2021): 100696.
67. Gariel, Maxime, Brian Shimanuki, Robert Eugene Johnston Timpe and E. Wilson. “Framework for Certification of AI-Based Systems.” ArXiv abs/2302.11049 (2023): n. Pag.
68. Nielsen, Claus Ballegaard, Peter Gorm Larsen, John S. Fitzgerald, Jim Woodcock and Jan Peleska. “Systems of Systems Engineering.” ACM Computing Surveys (CSUR) 48 (2015): 1 - 41.