drones

MDPI

Article

Regulatory Frameworks and Development Standards for Civilian Unmanned Aircraft Systems: From Regulatory Safety Intent to Development Lifecycles

Adina Aniculaesei [1,2]

[1] Institute for Software and Systems Engineering, TU Clausthal, 38678 Clausthal-Zellerfeld, Germany; adinaan@chalmers.se
[2] Department of Computer Science and Engineering, University of Gothenburg and Chalmers University of Technology, 41258 Gothenburg, Sweden

Highlights

What are the main findings?

- European regulations for unmanned aircraft systems (UAS) express safety intent primarily through operational approval artifacts (e.g., STS, PDRA, SORA). Existing international UAS standards address specific aspects of operation and classification but do not provide an integrated system and software development lifecycle comparable to those established in adjacent safety-critical domains such as automotive or avionics.
- Regulatory compliance for UAS operations does not translate directly into development and assurance goals for software-intensive UAS, leaving a gap between operational approval frameworks and systematic safety engineering practices at the system and software level.

What are the implications of the main findings?

- A structured engineering method can be used to systematically extract explicit system-level and software-level safety requirements from regulatory artifacts under stated operational assumptions.
- A software-centered, risk-proportionate development lifecycle can align regulatory safety intent with established assurance principles from avionics and automotive domains, enabling the identification of transferable practices and necessary adaptations for UAS.

Abstract

The rapid growth of civilian unmanned aircraft systems (UAS) for various applications, such as logistics, inspection and surveillance has enabled increasingly complex UAS operations in shared airspace and in close proximity to third parties. European regulations for civilian UAS provide a comprehensive framework for operational approval, based on operational rules, risk-based approval processes, and airspace management concepts. While regulatory frameworks and current international standards provide detailed guidance for operational authorization for UAS, they do not prescribe how UAS should be developed and verified at a system and software level to support safety assurance in a structured and traceable manner. This paper addresses this gap by proposing a method for extracting system-level and software-level safety requirements from regulatory artifacts. The method interprets regulatory safety intent–expressed through operational constraints, mitigation measures, and robustness expectations–and translates it into development-relevant safety requirements under explicit operational assumptions. Building on these requirements,

Check for updates

Academic Editor: Kimon P. Valavanis

Received: 13 January 2026
Revised: 4 April 2026
Accepted: 6 April 2026
Published: 9 April 2026

Copyright: © 2026 by the author. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

the paper introduces a software-centered system lifecycle for UAS development. The proposed lifecycle integrates regulatory safety intent, risk-proportionate assurance, and staged verification. Finally, through a cross-domain analysis, the paper positions the proposed approach relative to established practices from the automotive and the avionics domains, aiming to identify transferable and necessary adaptations for the development of unmanned aircraft systems.

Keywords: civilian unmanned aircraft systems; regulations and standards; European UAS regulations; UAS standards; safety requirements; system development process; system lifecycle

1. Introduction

Advances in autonomous navigation and control, sensing, and communication have led to the rapid growth of civilian unmanned aircraft systems (UAS) for applications such as logistics, inspection, and surveillance. These advances have also enabled increasingly complex UAS operations in shared airspace and in close proximity to third parties. In response, regulatory authorities—particularly within the European Union—have established an extensive regulatory framework intended to ensure that UAS operations can be conducted safely alongside manned aviation. This framework is primarily oriented toward the safe integration of drones into civilian airspace through operational rules, risk-based approval processes, and airspace management concepts.

A number of international standards contribute to this landscape by addressing specific aspects of UAS operation and classification. For example, ISO 21384-3 [1] specifies operational procedures for UAS, while ISO 21895 [2] defines categories and classes of civilian UAS based on physical and operational characteristics. Complementary work, such as ISO/TR 23629-1 [3], surveys concepts for UAS traffic management (UTM) and reflects the infrastructural perspective required for large-scale integration of drones into civilian airspace.

While these standards play an essential role in structuring safe operations and regulatory approval for UAS, they do not provide an integrated system or software development lifecycle comparable to those established in adjacent safety-critical domains, such as avionics or automotive engineering. In particular, they do not define development assurance levels, software lifecycle rigor, or traceability mechanisms linking operational safety objectives to system and software artifacts. An early attempt to articulate operational and functional safety objectives for UAS is found in RTCA DO-344 [4], which synthesizes safety objectives and requirements for UAS. Nevertheless, this document remains interpretive in nature and does not constitute a normative development standard.

In parallel to regulatory developments, research in safety-critical domains has examined certification standards and safety assurance practices for software-intensive systems [5,6]. In aviation and automotive engineering, development standards such as DO-178C [7] and ISO 26262 [8] define rigorous lifecycle processes, assurance levels, and traceability mechanisms linking safety objectives to system and software artifacts. Several studies have examined how such standards structure development assurance across safety-critical domains and how assurance levels influence system and software development activities [9,10]. More recent work has explored the applicability of these assurance approaches to unmanned aerial vehicles (UAVs) and compared safety assurance frameworks across industrial domains [11,12]. However, these studies primarily focus on certification methodologies or cross-domain comparisons and do not address how regulatory safety

intent embedded in UAS operational approval frameworks can be systematically translated into development-level system and software requirements and integrated within an appropriate development lifecycle.

Research Questions. Regulatory compliance for UAS operations does not directly translate into development-time assurance of software-intensive UAS, leaving a gap between operational approval frameworks and systematic software engineering and safety assurance practices. Regulatory artifacts—such as operational rules, risk assessments, and approval procedures—express safety expectations for UAS operations but do not define how these expectations should be transformed into concrete system- and software-level safety requirements.

In this paper, we refer to *regulatory safety intent* as the set of safety objectives, operational constraints, and risk mitigation expectations embedded in regulatory frameworks and operational approval procedures. These elements describe the level of safety that UAS operations must achieve but typically do not specify how such objectives should be realized at the level of system and software development.

Addressing this gap requires both a structured method for deriving development-level safety requirements from regulatory artifacts and a development lifecycle capable of integrating such requirements within established safety assurance practices.

Thus, the research questions of this paper are:

- **RQ-1** How can regulatory safety intent embedded in operational approval artifacts be systematically translated into system- and software-level safety requirements for UAS?
- **RQ-2** How can the resulting safety requirements be integrated into a development lifecycle aligned with established safety assurance practices?

Contributions. This paper addresses the identified gap and the corresponding research questions through the following contributions:

- It proposes a structured method for systematically translating regulatory safety intent—expressed in operational approval artifacts such as STS, PDRA, and SORA—into explicit system-level and software-level safety requirements under well-defined operational assumptions, explicitly bridging operational approval frameworks and system/software engineering practices.
- It introduces a software-centered system lifecycle that integrates these derived safety requirements into a structured development and assurance process, enabling traceability from regulatory constraints to system design, implementation, and verification activities.
- It provides a cross-domain analytical perspective that relates UAS regulatory constructs (e.g., SAILs and OSOs) to established safety assurance principles from avionics and automotive domains, identifying both transferable assurance concepts and necessary domain-specific adaptations for UAS without asserting direct equivalence.

Paper Structure. This paper is organized as follows: Section 2 reviews related work, and Section 3 outlines the methodology used to analyze the regulatory and standardization landscape for civilian transportation drones. Section 4 identifies the regulatory safety intent underlying European and selected national UAS frameworks, while Section 5 reviews safety assurance practices from avionics and automotive domains to establish relevant assurance principles. Section 6 addresses the research questions by deriving UAS safety requirements from regulatory artifacts and proposing a software-centered system lifecycle for their integration. Finally, Section 7 concludes the paper and outlines directions for future work.

2. Related Work

This section reviews existing research relevant to the relationship between regulatory frameworks for UAS operations and safety assurance practices in software-intensive sys-

tems. The discussion is structured along three complementary perspectives: (i) regulatory frameworks and operational approval processes for UAS (Section 2.1), (ii) certification and safety engineering practices in safety-critical domains (Section 2.2), and (iii) cross-domain analyses of safety assurance approaches (Section 2.3). Together, these perspectives provide the context for understanding how regulatory safety intent can be related to development-time safety assurance.

2.1. Regulations for Unmanned Aircraft Systems

Recent research has examined several aspects of the regulatory landscape governing the deployment of unmanned aircraft systems (UAS). Alamouri et al. [13] provide an overview of the evolving European regulatory framework for UAS operations and analyze how recent regulatory changes influence the deployment and economic potential of drone technologies across different civilian sectors, including scientific and commercial applications. Complementing this regulatory perspective, Du et al. [14] review existing risk assessment methods—including safety management processes, causal models, collision risk models, and ground risk models—and discuss the challenges of assessing and mitigating operational risks in civilian UAS operations. Other studies have explored technological approaches for supporting regulatory compliance. For example, Fakhraian et al. [15] investigate the potential use of digital twin technologies to assist certification and compliance processes under European UAS regulatory frameworks.

Several works have also focused specifically on the operational authorization mechanisms used in European drone regulation. Habibi et al. [16] analyze the requirements of the Specific Operations Risk Assessment (SORA) framework used by the European Union Aviation Safety Agency (EASA) to authorize higher-risk UAV operations and highlight the complexity of the associated approval procedures. To support drone operators in navigating this process, the authors propose workflow improvements and outline an initial concept for automating parts of the authorization procedure. Similarly, Capitán et al. [17] apply the SORA methodology to evaluate operational risks in a drone-based media production scenario and identify limitations of the framework for multi-UAS operations. Practical tools have also been proposed to support this authorization process; for instance, Schnüriger et al. [18] present a web-based tool designed to guide operators through the SORA approval procedure and automate parts of the required documentation. Finally, Nawaz [19] examines regulatory challenges related to human oversight in European drone law, identifying conceptual tensions arising from increasingly autonomous drone operations and questioning the adequacy of existing regulatory concepts such as the role of the remote pilot.

While these studies provide important insights into regulatory frameworks, operational risk assessment methods, and governance challenges for UAS operations, they primarily focus on operational approval and risk evaluation processes. They do not address how safety expectations embedded in these frameworks can be systematically translated into development-level system and software requirements or integrated into structured safety assurance processes.

2.2. Certification and Safety Engineering

In the aviation domain, software assurance standards such as DO-178C play a central role in ensuring the safety of airborne software systems. Youn et al. [5] provide a practitioner-oriented overview of the DO-178C standard and its supplementary documents, describing the key changes introduced compared to the earlier DO-178B standard and discussing their practical implications for software development and certification processes in avionics systems.

Building on these certification frameworks, several studies have examined verification approaches for demonstrating compliance with safety-critical software standards. Moy et al. [6] analyze the role of formal methods in the DO-178C certification context and explain how the DO-333 supplement allows formal verification techniques to replace certain testing activities. Drawing on industrial experience from Airbus and Dassault Aviation, the authors show that formal methods can support requirements analysis and software verification while remaining compatible with certification processes for safety-critical avionics software.

More recent work has explored how such assurance techniques can be applied to unmanned aerial vehicles. Zrelli et al. [11] propose an integrated methodology combining formal methods with automated verification tools to support the development of DO-178C-compliant UAV software. Their approach demonstrates how formal specification and automated analysis can improve traceability, verification efficiency, and the generation of certification evidence across the software lifecycle of safety-critical UAV systems.

Complementing research on certification and software assurance, other studies have investigated safety assessment methods for UAV operations. Allouch et al. [20] propose a safety assessment methodology for Internet-of-Drones environments that combines qualitative hazard analysis based on established safety standards with quantitative probabilistic modelling using Bayesian networks. Their approach integrates hazard identification, risk assessment, and mitigation analysis to evaluate safety risks in UAV missions.

Further research has examined specific safety mechanisms intended to support certification and regulatory approval of UAV operations. Guerin et al. [21] investigate safety requirements for emergency landing capabilities in urban UAV operations and analyze the limitations of the SORA framework when applied to such scenarios. The authors propose a landing zone selection module supported by runtime monitoring to demonstrate how safety mechanisms could facilitate the certification of higher-risk urban drone operations.

Finally, research has explored decision-support mechanisms for managing UAV operations at scale. Alharbi et al. [22] propose an explainable artificial intelligence framework for demand–capacity management in UAV traffic management systems operating in low-altitude urban airspace. Their approach integrates machine-learning models with data analytics to support trajectory allocation, flight planning, congestion prediction, and airspace capacity estimation while accounting for operational uncertainties such as weather conditions, dynamic obstacles, and emergency scenarios. To address the transparency challenges associated with AI-based decision support, the authors combine black-box and white-box models and introduce explainability metrics to evaluate the interpretability of the resulting traffic management recommendations.

These studies demonstrate how certification standards and safety engineering techniques provide rigorous mechanisms for ensuring the correctness and safety of software-intensive systems, including emerging UAV applications. However, they primarily focus on development-time assurance processes and verification techniques, assuming that safety requirements are already defined within the framework of a given standard. They do not address how safety expectations originating from regulatory and operational approval frameworks can be systematically translated into a development-level system and software requirements, nor how such requirements can be integrated within a unified assurance lifecycle for UAS.

2.3. Cross-Domain Safety Assurance

Recent research has also examined safety assurance practices across different industrial domains in order to identify common principles and potential opportunities for cross-domain reuse of certification approaches. Hawkins et al. [23] examine different approaches

to software certification in safety-critical systems by comparing prescriptive, process-based certification standards with argument-based assurance approaches relying on explicit safety cases. Through a case study involving aircraft braking software, the authors show how assurance arguments can complement prescriptive certification processes by providing a structured explanation of how development evidence supports safety claims.

This discussion highlights broader questions regarding how safety assurance requirements are structured across industrial domains, which has motivated comparative studies of safety assurance practices and development assurance levels in different safety-critical sectors. Machrouh et al. [9] investigate how Development Assurance Levels (DAL) and Safety Integrity Levels (SIL) influence system development activities across multiple safety-critical domains, including aviation, automotive, space, nuclear, railway, and industrial automation. By comparing standards such as ARP4754, ARP4761, ISO 26262, ECSS, and IEC 61508, the authors identify both common principles and domain-specific differences in system-level safety assurance practices.

Complementing this system-level perspective, Ledinot et al. [10] present a cross-domain comparison of software development assurance standards used in the same safety-critical industries. Their study analyzes how development assurance levels—such as DAL, SIL, and ASIL—affect the rigor of software lifecycle activities, including development processes, verification techniques, and supporting evidence required by each standard. Their study highlights both common principles and significant differences in how software safety assurance is achieved across domains and shows that direct equivalence between assurance levels is difficult to establish due to differences in regulatory frameworks, system contexts, and safety objectives.

Building on these comparative analyses, Zeller et al. [24] propose a cross-domain safety assurance process for safety-critical embedded software systems that can be applied across different industrial domains and development methodologies. By emphasizing the reuse of safety analysis techniques, tools, and assurance artifacts, their approach aims to reduce certification effort and support safety assessment in heterogeneous systems and software product-line environments.

Complementary research has examined the broader landscape of safety standards in order to identify opportunities for harmonization. Baufreton et al. [12] compare safety standards across several industrial domains, including aviation, nuclear, railway, automotive, and industrial automation, and analyze similarities and differences in their certification regimes and safety assurance approaches. Their study identifies common principles in safety engineering practices—such as the deterministic demonstration that software behavior satisfies system safety objectives—while also highlighting domain-specific characteristics that limit harmonization. In particular, the authors note significant differences in regulatory regimes and certification schemes across industries, with domains such as aviation and nuclear relying on strict regulatory certification processes, whereas automotive and industrial automation follow different assurance and compliance models.

While these studies provide important insights into regulatory frameworks, operational risk assessment methods, and governance challenges for UAS operations, they remain largely focused on operational approval processes and risk evaluation at the mission level. As a result, the translation of regulatory safety intent into development-level system and software requirements remains insufficiently addressed. In particular, there is a lack of systematic approaches that connect regulatory constructs—such as SORA elements, operational limitations, and safety objectives—to structured safety assurance processes across the system lifecycle. This disconnect limits the ability to integrate regulatory expectations into engineering practices for software-intensive UAS and motivates the need for a development-oriented interpretation of regulatory frameworks.

Taken together, these studies address regulatory frameworks for UAS operations, technical approaches for demonstrating compliance with safety standards, and cross-domain analyses of safety assurance practices. However, the existing work typically treats these aspects in isolation: regulatory studies focus on operational approval and risk assessment, certification-oriented research addresses development-time assurance processes, and cross-domain analyses examine similarities between standards without linking them to regulatory contexts.

As a result, there is limited work that systematically connects regulatory safety intent embedded in UAS operational frameworks with the development of system- and software-level safety requirements and their integration into structured assurance lifecycles.

This paper addresses this gap by providing a structured analytical method for examining official regulations and development standards governing civilian transport drone systems and explicitly linking regulatory safety intent to development-level safety requirements and lifecycle integration. In doing so, it establishes a systematic bridge between operational approval frameworks and software/system engineering practices for UAS. Unlike prior work that focuses either on regulatory analysis or on domain-specific certification practices, this paper explicitly connects regulatory safety intent to development-level safety requirements and integrates this connection within a structured lifecycle perspective.

3. Analytical Approach

This section describes the analytical approach used to examine the regulatory and standardization landscape for civilian transport drones. The approach, depicted in Figure 1, combines structured collection of relevant regulatory and standardization documents with a consistent classification framework and a set of analytical steps.

The objective is to provide a transparent and reproducible basis for analyzing how regulatory frameworks and safety standards define safety expectations for UAS operations and how these can be related to system and software development concerns. The selection of sources focuses on authoritative regulatory frameworks and widely adopted standards that define or influence safety expectations for civilian UAS operations and their development.

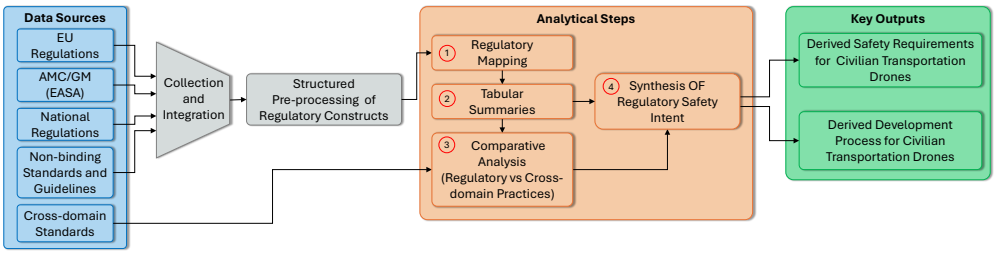


Figure 1. Methodological framework for analyzing the regulatory and standardization landscape of civilian transportation drones.

3.1. Data Sources

The analysis draws on a set of regulatory and standardization documents spanning multiple categories. At the core are binding regulations at the level of the European Union, in particular (EU) 2019/945 and (EU) 2019/947 and their amendments, which define the legal framework for UAS operations. These are complemented by acceptable means of compliance and guidance material (AMC/GM) published by European Union Aviation Safety Agency (EASA), which provide further implementation guidance.

To capture national perspectives, selected national regulations are included, such as Germany's Aviation Act (*Ger.:* Luftverkehrsgesetz, abbrv. LuftVG) and Air Traffic Act (*Ger.:* Luftverkehrs-Ordnung, abbrv. LuftVO). In addition, non-binding sources are consid-

ered, including industry guidelines (e.g., BDLI), ICAO-aligned publications (e.g., Skybrary), and relevant technical standards issued by ISO and RTCA.

Finally, cross-domain standards from adjacent safety-critical domains are incorporated, including avionics standards such as DO-178C, DO-278A, DO-331, DO-332, DO-333, and DO-337, as well as automotive standards such as ISO 26262 and ISO 21448. These standards provide established models for development assurance and serve as a basis for cross-domain comparison.

The selection of these sources is guided by their relevance to operational safety approval and development assurance for civilian UAS. Priority is given to binding regulatory frameworks, official guidance documents, and widely recognized industry standards. Cross-domain standards are selected based on their established role in safety-critical system development and their relevance for comparison with UAS-specific regulatory practices.

3.2. Classification Framework

To enable systematic comparison across these heterogeneous sources, all documents are organized within a structured classification framework. As illustrated in Figure 2, each document is annotated with a consistent set of metadata fields, including author or issuing organization, publication year, title, regulatory domain and jurisdiction, and document type (e.g., regulation, AMC/GM, guideline, or standard).

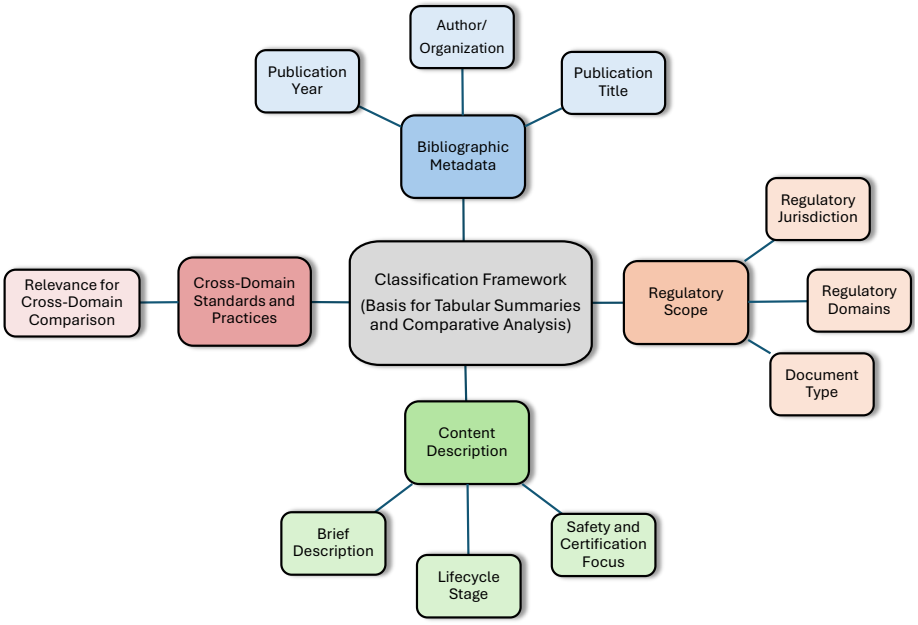


Figure 2. Taxonomy for the systematic classification of regulation and standards for unmanned aerial vehicles.

Additional descriptors capture the document's content and focus, including the lifecycle stages addressed, the emphasis on safety or certification, and notes on cross-domain relevance. This structured classification supports consistent treatment of diverse materials and enables systematic comparison across regulatory and standardization sources. It further provides a basis for identifying safety-relevant concepts that can be related to system and software development concerns (RQ-1) and to structured assurance practices (RQ-2).

For transparency, the detailed list of classification categories is provided in Table A1 in Appendix A.

3.3. Analytical Steps

The analysis proceeds in several steps, combining descriptive mapping with comparative and interpretive analysis.

First, the European regulatory framework is systematically mapped to capture its structure and key elements. This includes distinguishing between operational rules and product requirements and linking AMC/GM documents to their corresponding regulations.

Second, tabular summaries are constructed to consolidate and make explicit key regulatory constructs defined in the EU framework. These include drone classes, operational categories, standard scenarios (STS), pre-defined risk assessments (PDRAs), and the Specific Operations Risk Assessment (SORA) methodology with its core elements such as Ground Risk Class (GRC), Air Risk Class (ARC), Specific Assurance and Integrity Level (SAIL), and Operational Safety Objectives (OSOs). These summaries provide a structured representation of regulatory concepts and relationships, enabling consistent comparison and subsequent interpretation. These tabular summaries are distinct from the classification framework used to organize source materials; instead, they represent the regulatory constructs themselves in a structured and comparable form.

The results of the first two steps are presented in Section 4, where descriptive mapping and structured consolidation are combined to provide a coherent representation of the regulatory framework.

Third, a comparative analysis is conducted in Section 5 to examine similarities and differences between established safety standards in avionics and automotive domains. Building on these results, Section 6.3 provides a structured interpretation of UAS regulatory frameworks, relating safety objectives, risk classifications, and assurance mechanisms across domains.

Finally, based on this analysis, a system- and software-oriented development process is derived in Section 6.2. This derivation is based on interpreting regulatory requirements in terms of the safety objectives and constraints they express and relating these to established lifecycle and assurance principles from cross-domain standards. In this way, regulatory safety intent is systematically connected to development-level requirements and structured assurance processes.

The approach is not intended as an exhaustive survey, but as a structured and reproducible analysis of representative and authoritative sources that capture the core regulatory and safety assurance principles relevant to civilian transportation drones. By combining explicit source selection, consistent classification, and stepwise analysis, the methodology ensures transparency in interpretation and provides a reproducible basis for the derivation of safety requirements and development processes presented in this paper.

4. Regulatory Framework for Unmanned Aircraft Systems at National and EU Level

This section presents the regulatory framework that forms the basis for identifying regulatory safety intent in civilian UAS operations. It focuses on the EU legal framework, associated EASA guidance material, and selected national provisions that define the operational constraints, approval pathways, and risk assessment mechanisms relevant to civilian transport drones. These elements provide the regulatory input for the subsequent derivation of development-level safety requirements. The section first outlines the evolution of the regulatory framework (Section 4.1), then introduces UAS classes and operational categories (Section 4.2), and finally reviews the approval pathways in the Specific category (Section 4.3), with particular emphasis on the SORA methodology (Section 4.4). Section 4.5 consolidates the key findings and discusses their implications, bridging the regulatory analysis with development-level safety assurance considerations.

4.1. Evolution of the Regulatory Frameworks for Unmanned Aircraft Systems

The regulation of unmanned aircraft in the European Union evolved from the broader development of European aviation law. Regulation (EC) No. 1592/2002 [25] established the European Aviation Safety Agency (EASA), followed by Regulation (EC) No. 216/2008 [26] and, most importantly for UAS, Regulation (EU) 2018/1139 (the "Basic Regulation") [27], which explicitly brought unmanned aircraft within the scope of EU aviation safety rules. The detailed drone-specific framework was introduced in 2019 through two complementary acts: Commission Delegated Regulation (EU) 2019/945 [28], which defines product requirements, and Commission Implementing Regulation (EU) 2019/947 [29], which establishes operational rules. Together, these regulations form the basis of the harmonised EU framework for UAS. Since then, the framework has been refined through amendments, including (EU) 2020/639 [30] on standard scenarios and (EU) 2020/1058 [31] introducing drone classes C5 and C6, as well as further updates in 2021 [32], 2022 [33], and 2024 [34,35], together with successive EASA AMC/GM documents [36–39]. Figure 3 summarises this regulatory evolution.

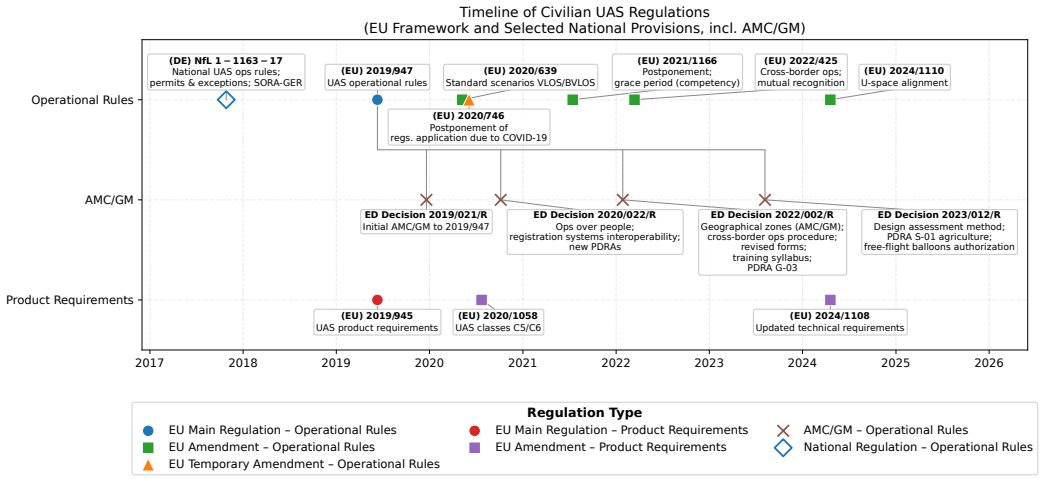


Figure 3. Evolution of national and EU regulatory framework for unmanned aircraft systems from 2017 to the present.

During the transition to this harmonised framework, national regulations continued to play an important role. In Germany, UAS were explicitly incorporated into aviation law in 2012 through the Fourteenth Act Amending the Aviation Act [40], which classified UAS as aircraft under § 1 LuftVG. Today, § 1 LuftVG [41] lists UAS and model aircraft as aircraft types, while §§ 21a and 21b LuftVO [42,43] define operational restrictions and approval requirements. In particular, § 21b(3) LuftVO introduced an "experimental clause" that enabled exemptions for UAS operations and provided the legal basis for so-called *Reallabore* (real-world laboratories). With the entry into force of Regulation (EU) 2018/1139 and subsequently Regulation (EU) 2019/947, however, responsibility for UAS regulation largely shifted to the EU level, and these national provisions were progressively superseded by the harmonised operational categories of the EU framework.

4.2. Classification of Unmanned Aircraft Systems and Their Operational Categories

The European UAS regulatory framework is structured along two complementary dimensions: product requirements and operational rules. Product requirements, defined in (EU) 2019/945 [28], specify the technical characteristics that drones must meet, while operational rules, defined in (EU) 2019/947 [29], determine the conditions under which they may be flown.

On the product side, drones are classified into classes C0–C6 based on parameters such as mass, performance, and built-in safety-related features. These classes establish manda-

tory technical requirements and serve as the reference for determining the permissible operational context. Table 1 summarizes the main characteristics of these classes.

Table 1. Drone Classes (cf. (EU) 2019/945 [28] and (EU) 2020/1058 [31]).

Drone Class ID	C0	C1	C2	C3	C4	C5	C6
Maximum Take-off Mass	250 g	900 g	4 kg	25 kg	25 kg	25 kg	25 kg
Maximum Speed	19 m/s	19 m/s	low-speed mode $\leq$ 3 m/s [b]	–	–	–	50 m/s [e]
Maximum Flight Height	120 m [a]	120 m [a]	120 m [a]	120 m [a]	–	–	–
Adjustable Height	–	yes	yes	yes	–	–	–
Altimeter	–	yes	yes	yes	–	yes	yes
Remote Identification	–	yes	yes	yes	no [c]	yes [d]	yes [d]
Geo-awareness	–	yes	yes	yes	no [c]	yes [d]	yes [d]

Notes. [a] operational altitude limit specified in (EU) 2019/947; not a class requirement. [b] C2 low-speed mode. [c] C4 no remote ID/geo-awareness. [d] C5/C6 (STS–01/02) include remote ID, geo-awareness, flight-termination. C5/C6 (STS–01/02) include remote ID, geo-awareness, flight-termination. [e] C6 max speed $\leq$ 50 m/s.

The classification evolved incrementally, with classes C0–C4 introduced in 2019 and classes C5 and C6 added in 2020. Across these classes, requirements scale with increasing operational capability and risk, including constraints on mass and performance as well as the introduction of features such as altitude limitation, remote identification, and geo-awareness.

Complementing the product classification, the regulatory framework distinguishes three operational categories—*Open*, *Specific*, and *Certified*—which reflect increasing levels of risk and regulatory oversight. The relationship between product classes and operational categories ensures that drones are operated within environments consistent with their capabilities (Figure 4).

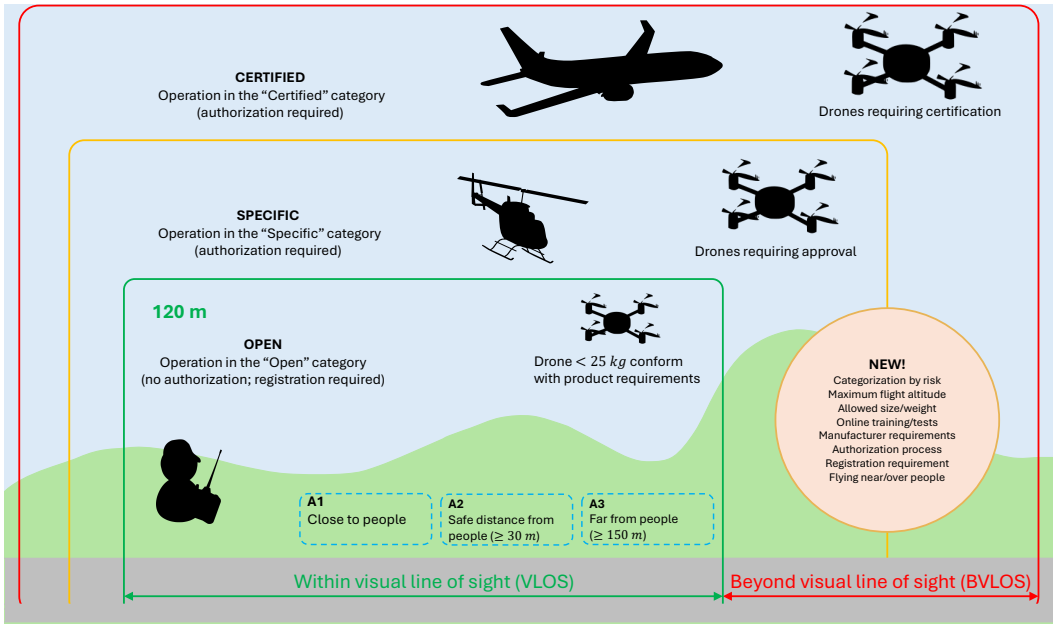


Figure 4. Overview of the three operational categories for civilian drones in the EU: Open (subcategories A1–A3), Specific, and Certified (adapted from (EU) 2019/947 [29], Articles 4–6 (Operational Categories)).

The *Open* category covers low-risk operations that do not require prior authorization and are subject to predefined operational limitations, including visual line-of-sight (VLOS) operation and restrictions on proximity to people. It is subdivided into A1–A3, which differ

primarily in their allowed proximity to uninvolved persons and corresponding technical and training requirements.

The *Specific* category applies to operations that exceed the limitations of the Open category, including beyond visual line-of-sight (BVLOS) operations or flights over populated areas. In this case, operators must obtain authorization based on one of three pathways: standard scenarios (STS), predefined risk assessments (PDRA), or a dedicated Specific Operations Risk Assessment (SORA).

The *Certified* category covers high-risk operations requiring a level of safety assurance comparable to manned aviation, including the transport of people or dangerous goods. These operations require certification of both the system and the operator, and remote pilots may need licensing comparable to manned aviation standards. This category therefore represents the highest level of regulatory oversight in the EU framework.

ISO 21895 [2] provides a complementary terminology for categorizing UAS operations, but the operational categories and class-based structure defined in the EU regulatory framework remain the primary reference for authorization.

These classifications and operational categories define how regulatory safety intent is expressed in terms of operational constraints and risk-based approval mechanisms, but they do not directly specify corresponding system- or software-level requirements.

4.3. Approval Processes for the Operation of Unmanned Aircraft Systems

Operations that cannot be conducted within the Open category fall into the Specific category, where regulatory approval is required. The EU framework provides three approval pathways with increasing flexibility: standard scenarios (STS), predefined risk assessments (PDRA), and, where neither applies, a dedicated Specific Operations Risk Assessment (SORA).

Standard Scenarios (STSs) define predefined types of operations for which a complete set of risk mitigations has been established by EASA. In such cases, operators may submit a declaration of compliance rather than applying for a full authorization. This approach reduces administrative effort by standardizing both the operational context and the associated safety measures. Currently defined STSs cover typical VLOS and BVLOS operations under constrained conditions (Table 2).

Table 2. Standard Test Scenarios (cf. (EU) 2020/639 [30], Annex 1).

STS ID	Edition/Date	UAS Characteristics	BVLOS/VLOS	Overflown Area	Max Range	Max Height	Airspace
STS–01	June 2020	C5 marking; MCD $\leq$ 3 m; MTOM $\leq$ 25 kg [a]	VLOS	Controlled; may be in populated area	VLOS	120 m	Controlled/ uncontrolled; low encounter risk [c]
STS–02	June 2020	C6 marking; MCD $\leq$ 3 m; MTOM $\leq$ 25 kg [a]	BVLOS	Controlled; entirely in sparsely populated area	2 km with AO; 1 km if no AO [b]	120 m	Controlled/ uncontrolled; low encounter risk [c]

Notes. [a] C5/C6 requirements introduced by (EU) 2020/1058. [b] "AO" = Airspace Observer, required for BVLOS under STS-02. [c] "Low risk of encounter with manned aircraft" per AMC/GM to (EU) 2019/947.

Predefined Risk Assessments (PDRAs) extend this concept to a broader set of recurring operations. They provide reusable risk assessments derived from the SORA methodology, specifying acceptable conditions and mitigations. Compared to STSs, PDRAs offer greater flexibility while still reducing the need for a fully operation-specific risk assessment. Both scenario-derived ("S-type") and generic ("G-type") PDRAs are defined for typical VLOS and BVLOS operations (Table 3).

If neither STSs nor PDRAs are applicable, operators must perform a Specific Operations Risk Assessment (SORA) to demonstrate that the operation can be conducted safely. In this case, safety justification is based on an explicit assessment of operational risks and the identification of appropriate mitigation measures.

Table 3. Pre-Defined Risk Assessments (cf. ED Decision 2020/022/R [37] and ED Decision 2023/012/R [39].

PDRA ID	Edition/Date	UAS Characteristics	BVLOS/VLOS	Overflown Area	Max Range	Max Height	Airspace	AMC ID
PDRA–S01 [a]	Amend. 3/October 2023	MCD $\leq$ 3 m; MTOM $\leq$ 25 kg	VLOS	Controlled; may be in populated area (agricultural ops included)	VLOS	120 m	Controlled/uncontrolled; low encounter risk	AMC 4
PDRA–S02	1.0/July 2020	MCD $\leq$ 3 m; MTOM $\leq$ 25 kg	BVLOS	Controlled; entirely in sparsely populated area	2 km with AO; 1 km if no AO [c]	120 m	Controlled/uncontrolled; low encounter risk	AMC 5
PDRA–G01 [b]	Rev. October 2023	MCD $\leq$ 3 m; KE $\leq$ 34 kJ	BVLOS	Sparsely populated area	up to 1 km if no AO [c]	150 m (operational volume)	Uncontrolled; low encounter risk	AMC 2
PDRA–G02 [b]	Rev. October 2023	MCD $\leq$ 3 m; KE $\leq$ 34 kJ	BVLOS	Sparsely populated area	N/A	As established for reserved airspace	As reserved for the operation	AMC 3
PDRA–G03 [b]	Rev. October 2023	MCD $\leq$ 3 m; KE $\leq$ 34 kJ	BVLOS	Sparsely populated area	up to 1 km, with mitigations	120–150 m (per reserved volume)	Uncontrolled; low encounter risk	AMC 6

Notes. [a] Originally in ED Decision 2020/022/R; amended by ED Decision 2023/012/R—S01 scope expanded to agricultural operations; containment aligned with AMC 1 SORA. [b] ED Decision 2023/012/R—Characterization/conditions updated for G01/G02/G03 to align with AMC 1 SORA (containment, consistency tweaks). [c] "AO" = Airspace Observer.

Together, these approval pathways define a structured spectrum ranging from fully predefined operations to fully risk-based justification. They express regulatory safety intent in terms of acceptable operational scenarios, required mitigations, and risk assessment procedures, but do not directly prescribe how these safety expectations should be translated into system- and software-level requirements.

4.4. Specific Operations Risk Assessment

The Specific Operations Risk Assessment (SORA), developed by the Joint Authorities for Rulemaking on Unmanned Systems (JARUS), is the central risk-based approval mechanism for operations in the Specific category when neither a standard scenario nor a predefined risk assessment applies. It provides a structured method for evaluating operational risks and determining the mitigations and safety objectives required for authorization. In the context of this paper, SORA is particularly important because it expresses regulatory safety intent in a structured form, through explicit operational assumptions, risk classes, assurance levels, and safety objectives. An overview of the SORA process flow is shown in Figure 5.

Concept of Operations. The process begins with defining the Concept of Operations (ConOps), which describes the planned UAS operation in sufficient technical, operational, and procedural detail to allow assessment of the associated risks. It also specifies the interaction with the air navigation service provider (ANSP) and is typically refined iteratively as additional mitigations or restrictions are identified during the SORA process. The ConOps aligns with the broader notion of operational specification in UAS operational guidance [1] and anticipates integration with UTM concepts [3].

Determination of Intrinsic Ground Risk Class. The intrinsic ground risk represents the likelihood of a person being struck by a UAS in the event of a loss of control. It is determined based on the maximum characteristic dimension of the aircraft and the intended operational scenario. This baseline ground risk class (GRC) is established using the criteria in Table 4. The GRC can then be adjusted through sequential mitigations (M1–M3), with correction factors applied according to the level of robustness, as shown in Table 5. A positive correction increases the risk level, while a negative correction decreases it. All mitigation measures must be applied in sequence, and competent authorities may prescribe additional measures

and associated correction factors. The final GRC is obtained by adding all correction factors to the intrinsic GRC. If the resulting value exceeds seven, the operation falls outside the scope of the Specific category and is therefore not supported under the SORA process.

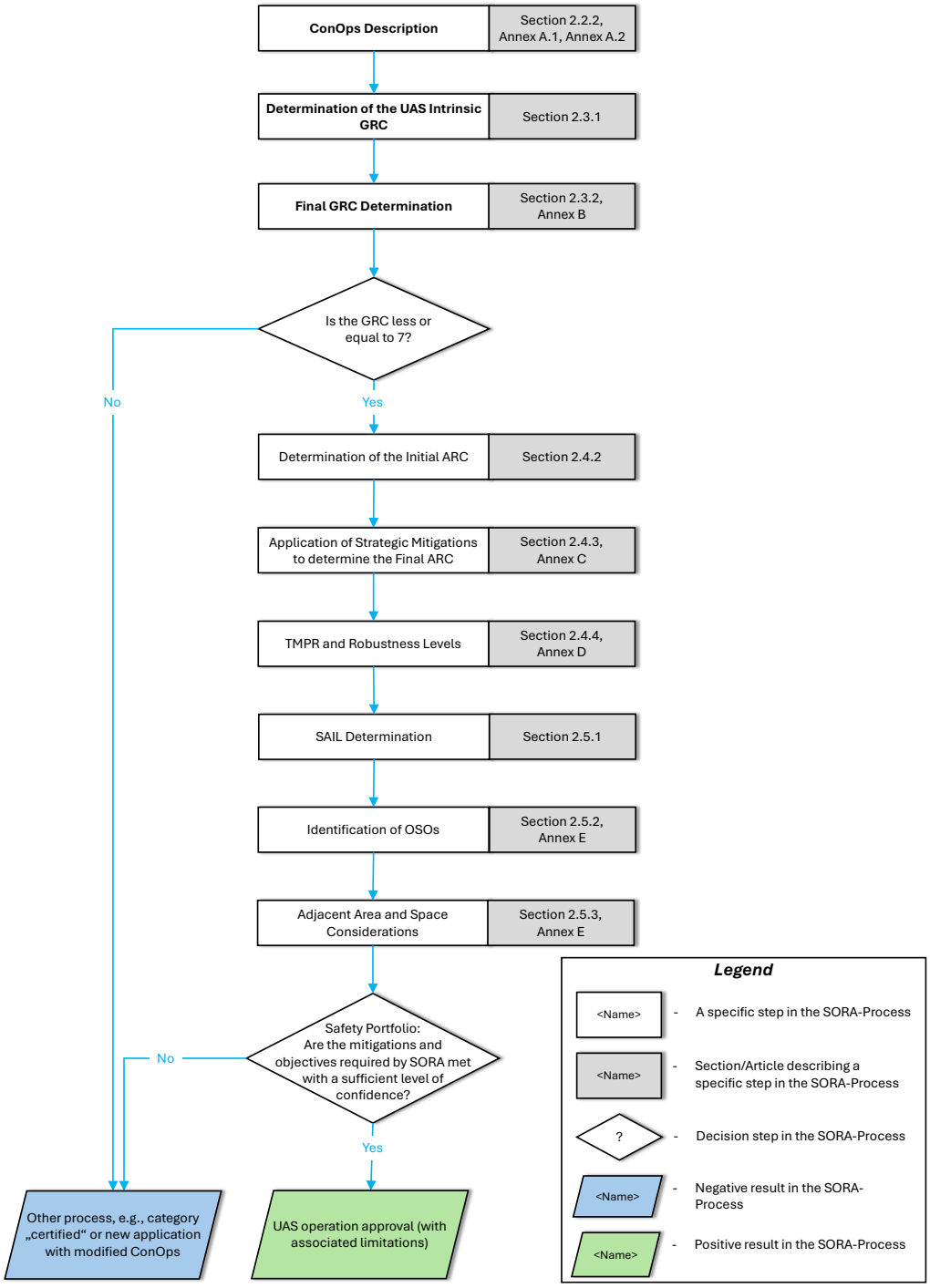


Figure 5. Overview of the SORA process flow (adapted from JARUS SORA v2.5, Annex F [44]).

Determination of Initial Air Risk Class. Based on the operational airspace defined in the ConOps, SORA assesses the intrinsic risk of mid-air collisions and determines the initial Air Risk Class (ARC). The ARC is a qualitative classification of the expected encounter rate with manned aircraft and is grouped into four levels, ARC-a to ARC-d, representing increasing collision risk. Strategic mitigations may be applied to reduce the initial ARC, while any residual risk must be addressed through tactical mitigations such as visual separation or detect-and-avoid (DAA) systems. Figure 6 summarizes the decision logic for

determining the ARC. As part of the SORA process, operators are expected to coordinate with the competent airspace authority and obtain the necessary approvals. In some cases, the authority may provide a static or dynamic air collision risk map, in which case the process shown in Figure 6 is no longer required.

Table 4. Determination of the Intrinsic Ground Risk Class (cf. JARUS SORA v2.5, Annex F [44], Table F-1).

Operational Scenario	MCD < 1 m KE < 700 J	1 m ≤ MCD < 3 m KE < 34 kJ	3 m ≤ MCD < 8 m KE < 1084 kJ	MCD ≥ 8 m KE ≥ 1084 kJ
VLOS/BVLOS over a controlled ground area [a]	1	2	3	4
VLOS over a sparsely populated area [c]	2	3	4	5
BVLOS over a sparsely populated area [c]	3	4	5	6
VLOS over a populated area [c]	4	5	6	7
BVLOS over a populated area [c]	5	6	7	8
VLOS over an assembly of people	7	8 [b]	9 [b]	10 [b]
BVLOS over an assembly of people	8	9 [b]	10 [b]	10 [b]

Notes. [a] Controlled ground area: operator ensures no uninvolved persons in the area of operation. [b] Although Intrinsic GRC values are provided for VLOS and BVLOS operations over assemblies of people, these scenarios generally exceed the risk envelope of the "specific" category under (EU) 2019/947 and are expected to fall into the "certified" category. The values are retained in the table for completeness of the SORA v2.5 methodology but should not be interpreted as permitting such operations in the "specific" category. [c] Population density distinctions (sparsely vs. populated area) are qualitative in SORA v2.5 and require national guidance for precise thresholds.

Table 5. Mitigations for the Final Determination of the Intrinsic Ground Risk Class (cf. JARUS SORA v2.5, Annex F [44], Table F-2).

Mitigation Sequence	Mitigations for Ground Risk	Robustness [d]		
		Low/None	Medium	High
1	M1—Strategic mitigations for ground risk [a]	0: None; −1: Low	−2	−4
2	M2—Effects of ground impact are reduced [b]	0	−1	−2
3	M3—An emergency response plan (ERP) is in place, the UAS operator is validated and effective [c]	+1	0	−1

Notes. [a] Examples: flight geography, flight path management, avoidance of overflight of people. [b] Examples: design features reducing impact energy (parachutes, frangible structures, energy-absorbing materials). [c] ERP effectiveness: e.g., coordination with local emergency services, communication protocols, drills. [d] Values are cumulative adjustments applied to the intrinsic GRC to yield the final GRC.

Strategic and Tactical Mitigations. If the initial ARC is too high, strategic mitigations—such as airspace segregation or operational limitations—may be applied to reduce it. Residual collision risk is then addressed by tactical mitigations, including VLOS-based "see and avoid" or DAA systems. For BVLOS operations, these mitigations must satisfy the applicable Tactical Mitigation Performance Requirements (TMPRs), whose required robustness depends on the residual ARC (Table 6).

Determination of Specific Assurance and Integrity Level. The combination of the final GRC and residual ARC yields the Specific Assurance and Integrity Level (SAIL), ranging from I (lowest) to VI (highest). The SAIL is not a quantitative probability but rather a qualitative measure of the confidence required that UAS operation will remain under control. Table 7 shows how SAIL is determined from the final GRC and residual ARC.

Identification of Operational Safety Objectives. The final step in the SORA process is to derive Operational Safety Objectives (OSOs) from the resulting SAIL and to assign the corresponding robustness levels. These OSOs define the defense mechanisms expected for a given operation and represent a structured set of safety expectations derived from the

assessed operational risks. Table 8 presents a representative subset of OSOs illustrating how robustness requirements increase with SAIL; the complete set is provided in Table A2 in Appendix B. Competent authorities issuing the operating permit may require additional OSOs or impose higher robustness levels for given SAILs, depending on the specifics of the operation and the assessed risk.

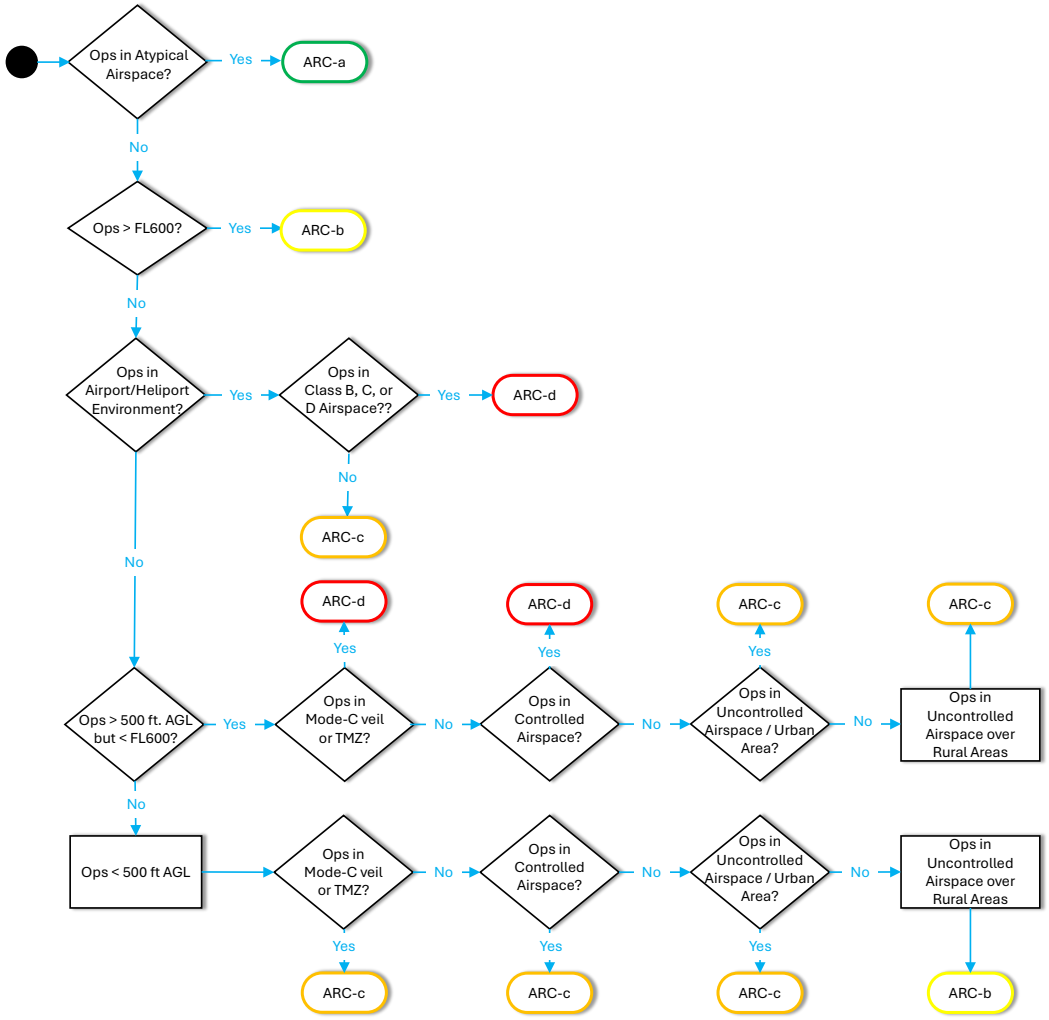


Figure 6. Decision logic for determining the Air Risk Class (cf. JARUS SORA v2.5, Annex F [44]). *Notes.* (1) *Atypical airspace*: segregated/restricted state areas (e.g., danger/military ranges). (2) FL600 = flight level 600 (∼60,000 ft); above this, UAS ops are isolated from conventional traffic. (3) *Airport/heliport environment*: aerodrome-defined controlled surfaces. (4) Mode C veil/TMZ: state-defined transponder-mandatory zones; dimensions vary nationally. (5) Urban vs. rural: per national population-density guidance and encounter-risk assumptions.

Table 6. Assignment of TMPR Level of Robustness (cf. JARUS SORA v2.5, Annex F [44], Table F-3).

Residual ARC	TMPRs	TMPR Level of Robustness [a]
ARC–d	High	High
ARC–c	Medium	Medium
ARC–b	Low	Low
ARC–a	No requirement [b]	No requirement

Notes. [a] TMPR robustness level indicates the minimum assurance required for detect-and-avoid (DAA) or strategic mitigations to lower residual air risk. [b] For ARC–a, no TMPR is required as operations are assumed to be segregated from other air traffic, e.g., atypical or very high-level airspace.

Table 7. SAIL determination (cf. JARUS SORA v2.5, Annex F [44], Table F-4).

Final GRC	Residual ARC			
	a	**b**	**c**	**d**
≤2	I [a]	II [a]	IV [a]	VI [a]
3	II [a]	II [a]	IV [a]	VI [a]
4	III [a]	III [a]	IV [a]	VI [a]
5	IV [a]	IV [a]	IV [a]	VI [a]
6	V [a]	V [a]	V [a]	VI [a]
7	VI [a]	VI [a]	VI [a]	VI [a]
>7 [b]	Category C operation			

Notes. [a] The resulting *Specific Assurance and Integrity Level (SAIL)* determines both the number and the robustness level of the Operational Safety Objectives (OSOs) to be applied. [b] For operations with final GRC > 7, the risk is considered outside the "specific" category and falls into the "certified" category under (EU) 2019/947.

Table 8. Representative Set of Operational Safety Objectives (OSOs) Illustrating Robustness Scaling with Specific Assurance and Integrity Level (SAIL) (cf. JARUS SORA v2.5, Annex E [45]).

OSO No.	OSO Description	SAIL					
		I	**II**	**III**	**IV**	**V**	**VI**
Technical issue with the UAS							
OSO#01	Ensure the UAS operator is competent and/or proven	O	L	M	H	H	H
...	...	...	...	...	...	...	...
OSO#04	UAS developed to authority-recognized design standards	O	O	L	L	M	H
OSO#05	UAS is designed considering system safety and reliability	O	O	L	M	H	H
...	...	...	...	...	...	...	...
OSO#08	Operational procedures are defined, validated and adhered to	L	M	H	H	H	H
...	...	...	...	...	...	...	...
OSO#10	Safe recovery from a technical issue	L	L	M	M	H	H
Deterioration of external systems supporting UAS operations							
OSO#11	Procedures in-place to handle deterioration of external systems supporting UAS operations	L	M	H	H	H	H
...	...	...	...	...	...	...	...
Human error							
...	...	...	...	...	...	...	...
OSO#18	Automatic protection of the flight envelope from human error	O	O	L	M	H	H
...	...	...	...	...	...	...	...
Adverse operating conditions							
...	...	...	...	...	...	...	...
OSO#23	Environmental conditions for safe ops are defined, measurable and adhered to	L	L	M	M	H	H
OSO#24	UAS designed/qualified for adverse environmental conditions	O	O	M	H	H	H

Notes. **O** = Optional objective, **L** = Low robustness, **M** = Medium robustness, **H** = High robustness.

Adjacent Area and Airspace Considerations. Operators must also ensure containment of the operation within its intended operational volume and prevent infringement of adjacent ground areas or airspace. Enhanced containment requirements apply for higher-risk operations.

Safety Portfolio. Taken together, the outputs of the SORA process—ConOps assumptions, ground and air risk classifications, mitigation measures, SAIL, and OSOs—form a safety portfolio that supports the authorization decision. For the purposes of this paper, these elements are especially relevant because they make regulatory safety intent explicit in operational and risk-based terms. At the same time, they remain formulated primarily for authorization and compliance, rather than for development. In other words, SORA defines what level of safety must be demonstrated for a given operation, but not how these expectations should be translated into system requirements, software requirements, lifecycle activities, or assurance artifacts. This makes SORA a key regulatory input to the derivation step developed later in this paper.

4.5. Synthesis and Implications

Taken together, the EU regulatory framework for UAS operations establishes a comprehensive, risk-based structure spanning the open, specific, and certified categories. Each category introduces increasing levels of operational complexity and assurance, ranging from prescriptive operational constraints in the open category, to structured risk assessment and safety objectives in the specific category through SORA, and ultimately to certification-oriented assurance processes in the certified category.

Across all three categories, safety is consistently expressed in terms of operational constraints, risk classes, mitigation measures, and high-level safety objectives. However, these elements are primarily formulated to support authorization and regulatory compliance, rather than to guide the systematic development of UAS software and systems. In particular, while mechanisms such as SORA provide a structured representation of operational risk and required safety objectives, they do not define how these should be translated into concrete system requirements, software design decisions, or lifecycle assurance activities.

This observation contrasts with existing research discussed in Section 2. Prior studies on UAS regulation predominantly focus on regulatory frameworks, operational approval processes, and risk assessment methodologies [13,14,16], while work in safety-critical domains such as aviation and automotive concentrates on development-time assurance processes assuming pre-defined safety requirements [5,6,11]. Cross-domain analyses further identify common principles across standards but do not establish a systematic link between regulatory constructs and development-level artifacts [9,10,12].

In contrast to these strands of work, this paper explicitly bridges these perspectives. It provides a structured method for translating regulatory safety intent—expressed through operational categories, SORA elements, and safety objectives—into development-level system and software requirements. Rather than treating regulatory analysis, certification practices, and cross-domain comparisons in isolation, the approach integrates them into a unified, development-oriented interpretation of UAS safety assurance.

This gap between regulatory safety intent and development-level realization motivates the need to examine complementary standards and practices from related domains. In the following section, we therefore analyze established development and certification standards from automotive and aviation, in order to understand how high-level safety objectives can be systematically refined into implementable and verifiable system requirements.

5. Cross-Domain Comparison of Safety Standards for Avionics and Automotive Systems

This section supports the third step of the analytical approach by examining established safety standards from the avionics (Section 5.1) and automotive domains (Section 5.2) and identifying their underlying assurance concepts, lifecycle structures, and verification principles. Section 5.3 then compares these cross-domain standards with the regulatory framework for unmanned aircraft systems.

5.1. Overview of Development Standards for Commercial and General Avionics Systems

The aviation domain has relied for more than three decades on the DO-178 family of standards as the principal framework governing software development for airborne systems and equipment. The most recent edition, DO-178C, published by RTCA in 2011 (and harmonized with EUROCAE ED-12C), refines earlier versions by providing clearer guidance on requirements traceability, tool qualification, and the integration of model-based and formal development methods. Figure 7 offers a visual intuition of the software lifecycle processes defined in DO-178C [7] and their relationship to the system and hardware life-cycle processes specified in ARP4754A [46] and ARP4761A [47].

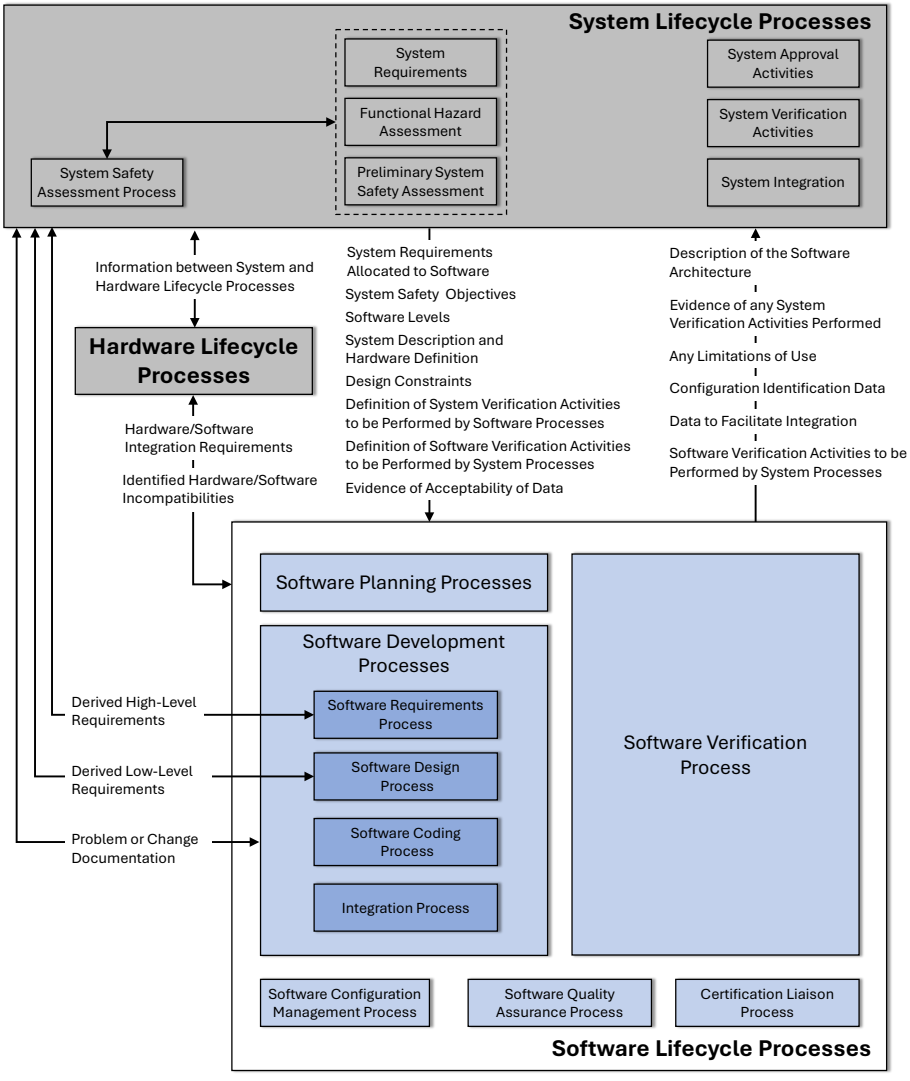


Figure 7. Software Lifecycle Process in DO-178C and its Relation to System and Hardware Lifecycle Processes (adapted from [48]).

At the system level, ARP4754A [46] defines the processes for system requirements, system architecture, system integration, and system verification. Complementing these processes, ARP4761A describes the safety-assessment activities—functional hazard assessment (FHA), preliminary system safety assessment (PSSA), and system safety assessment (SSA)—that identify system-level hazards, allocate safety objectives, and determine assurance levels for the different system components and subsystems. These activities produce the system requirements, safety objectives, and design constraints that form essential inputs to the software life-cycle processes described in DO-178C.

DO-178C structures the software life cycle into three main process groups: planning, development, and integral processes. The planning process defines how development and verification are to be conducted and produces the core planning artifacts, including development, verification, configuration management, and quality assurance plans.

The development process encompasses software requirements, design, implementation, and integration. System-level requirements are refined into software requirements allocated to components and subsystems, while issues identified during software development may generate derived requirements that must be reviewed at system level. Throughout these activities, safety and security objectives must be addressed explicitly and traceability between requirements, design, implementation, and verification must be maintained.

The integral processes—software verification, configuration management, quality assurance, and certification liaison—operate across the lifecycle to ensure correctness, control, and confidence in both the process and its outputs. Verification provides evidence that each development artifact satisfies its objectives through reviews, analyses, and testing, while configuration management, quality assurance, and certification liaison ensure change control, process compliance, and communication with certification authorities.

The standard introduces the concept of design assurance levels (DALs) A to E, which reflect the contribution of a software component to potential failure conditions identified through the system safety assessment process. Each failure condition corresponds to a software level, determining the rigor of verification and independence required. The failure conditions and their corresponding DAL are detailed in Table 9.

Table 9. Failure Conditions and corresponding Design Assurance Levels (DAL) for Avionics Systems.

Failure Condition	Description	DAL
Catastrophic	Failure conditions that could lead to multiple fatalities, usually with the loss of the aircraft	A
Hazardous	Conditions that severely reduce the crew's ability to cope with adverse situations, leading to a major reduction in safety margins or serious injuries	B
Major	Conditions that significantly increase crew workload or reduce functional capability, possibly leading to discomfort or minor injuries	C
Minor	Conditions with limited impact on safety or workload; the crew can readily manage them	D
No safety effect	Faults with no impact on operational safety or crew workload	E

Design assurance levels are derived through the system safety assessment. In this process, software malfunctions are analyzed from the perspective how these malfunctions may contribute to system-level hazards. When a software component can contribute to multiple failure conditions, the most severe applicable DAL is selected.

DO-178C also encourages architectural practices that support safe operation, such as software monitoring. Monitoring functions are intended to detect anomalies in safety-critical behavior and mitigate their effects. For example, a monitor supervising a flight control function must be developed to the same or higher assurance level as the monitored function and must demonstrate sufficient fault coverage, including independence from common-cause failures affecting both monitor and controlled function.

The described lifecycle structure and assurance mechanisms provide a reference model for how safety objectives are translated into development and verification processes in a highly regulated domain. These characteristics are used in the subsequent comparison to identify transferable principles for UAS development.

5.2. Overview of Development Standards for Automotive Systems

ISO 26262, first published in 2011 and revised in 2018, is an adaptation of IEC 61508 to the automotive domain and addresses the specific requirements of electrical and electronic (E/E) systems in road vehicles [8]. The standard defines a comprehensive automotive safety lifecycle and provides detailed guidance for the development of safety-related systems at system, hardware, and software levels.

Complementing ISO 26262, the ISO 21448 standard extends the development process to address safety of the intended functionality (SOTIF), focusing on hazardous behavior that may arise in the absence of system faults, for example due to performance limitations or insufficient specification of the operational environment [49].

ISO 26262 distinguishes development at the system, hardware, and software levels. At a system level, development begins with the concept phase, in which the item definition is established, including system boundaries, interfaces, assumptions about interactions with other systems, and relevant operational and environmental constraints [50]. A hazard analysis and risk assessment (HARA) is then performed to identify hazardous events resulting from malfunctioning behavior and to assess their associated risks. The outputs of this phase are the functional concept and the functional safety concept, from which technical safety requirements are derived.

These requirements guide the subsequent system design. The logical architecture defines system functions, interfaces, and communication structures, while the technical architecture allocates technical safety requirements to hardware and software components and specifies the relevant hardware–software interfaces [51]. Once system-level design is completed, development proceeds in parallel at the hardware and software levels.

Figure 8 illustrates the software development process defined in ISO 26262 and its relationship to system-level development activities.

At the software level, development begins with planning activities that coordinate software development and verification with the system- and hardware-level processes. Software safety requirements are derived from the technical safety requirements and from the logical and technical system architecture, taking into account the constraints imposed by the hardware platform [52].

The software architectural design captures both static and dynamic aspects of the software, including component decomposition, interfaces, control and data flow, concurrency, and timing behavior. Software safety requirements are allocated to software components, each of which is developed in accordance with the highest ASIL associated with any allocated requirement. Safety analyses such as FMEA or FTA are applied to identify safety-relevant components, software-level hazards, and corresponding safety mechanisms [52].

Implementation and verification proceed through software unit design, coding or model realization, unit testing, and incremental integration. Verification activities include

reviews, static analysis, requirements-based testing, interface testing, and fault-injection or robustness testing, supported where appropriate by model-, software-, processor-, or hardware-in-the-loop environments [52,53]. The final software-level verification demonstrates compliance with the software safety requirements in the target environment.

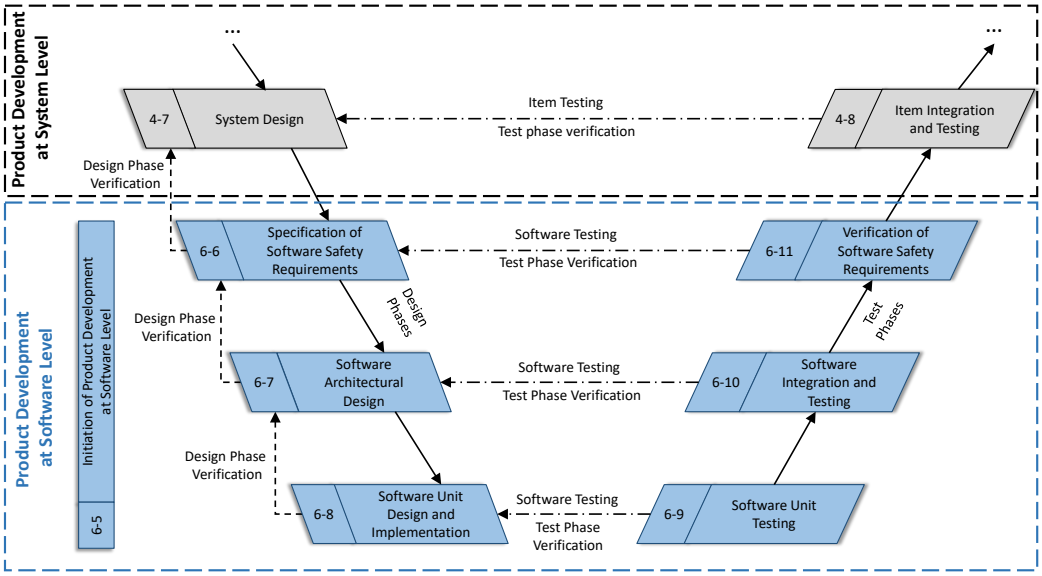


Figure 8. Software Development Process in ISO 26262, its Relation to the System-level Activities and their corresponding Verification Phases [52].

The hazard analysis and risk assessment (HARA) identifies hazardous events resulting from malfunctioning behavior and evaluates them based on severity, exposure, and controllability [50]. Hazard identification considers both correct use and reasonably foreseeable misuse of the system and may be supported by systematic techniques such as FMEA, FTA, or HAZOP.

Based on the combination of severity (S), exposure (E), and controllability (C), each hazardous event is assigned an Automotive Safety Integrity Level (ASIL), ranging from A to D, with D representing the highest safety requirements, as shown in Table 10. The assigned ASIL determines the rigor of the corresponding development and verification activities. In addition, the class QM (quality management) denotes situations not subject to functional safety requirements beyond standard quality practices.

Table 10. Automotive Safety Integrity Levels according to ISO 26262.

ASIL	Description	Relative Rigor
A	Lowest safety integrity level	Lowest safety-related rigor
B	Moderate safety integrity level	Increased safety measures
C	High safety integrity level	Strong safety measures
D	Highest safety integrity level	Most stringent safety measures

The automotive safety lifecycle complements the avionics perspective by incorporating risk-based reasoning and context-dependent assurance. Together, these characteristics provide a basis for cross-domain comparison with UAS regulatory constructs.

5.3. Comparative Analysis of Automotive and Avionics Standards

Figure 9 juxtaposes the software development processes defined in DO-178C and ISO 26262 and provides the visual basis for the cross-domain comparison. For the automotive domain, the figure follows the V-model explicitly adopted in ISO 26262. For the

avionics domain, DO-178C does not prescribe an equivalent V-model; instead, the depiction follows the interpretive synthesis proposed in [54], which reorganizes its planning, development, and verification activities into a form suitable for comparison.

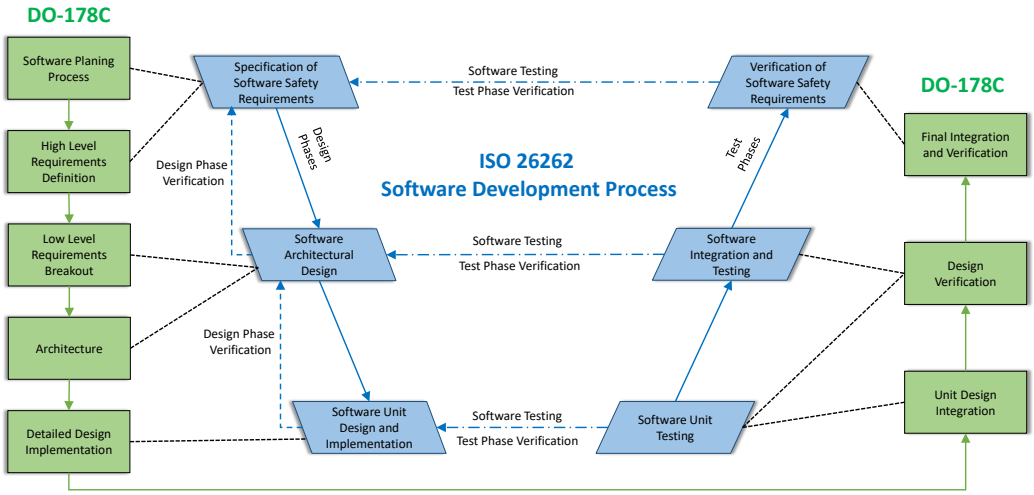


Figure 9. Comparison between the Software Development Processes defined in DO-178C and ISO 26262.

Despite their differences, both standards are requirements-driven and organize software development along a structured lifecycle in which verification activities are associated with each development stage to ensure traceability and completeness. This shared foundation provides the starting point for the comparison below.

A fundamental difference between the two standards lies in their *safety assurance concepts*. In the avionics domain, DO-178C derives assurance requirements from predefined failure conditions identified through system safety assessment. Software is assigned a DAL based on the severity of the consequences to which its anomalous behavior may contribute, rather than on the probability of occurrence during operation. This reflects a safety philosophy centered on preventing catastrophic outcomes through strict design and verification rigor.

In contrast, ISO 26262 adopts a hazard- and risk-driven assurance concept. Hazardous situations are identified through HARA and evaluated in terms of severity, exposure, and controllability. As a result, software components and systems are assigned an ASIL based on a contextualized assessment of risk rather than solely on failure severity.

These differing assurance concepts are also reflected in how *assurance targets are allocated*. In the avionics domain, DALs are assigned to software functions and components according to the most severe failure condition to which they may contribute. Once assigned, the DAL governs the rigor of development and verification throughout the lifecycle, supporting a conservative strategy driven by worst-case consequences.

By contrast, ISO 26262 assigns ASILs to hazardous events identified during HARA and propagates them to technical safety requirements and to the hardware and software components involved in mitigating those hazards. This event-centric allocation supports mixed-ASIL designs and distributes assurance according to risk contribution rather than solely according to failure severity.

These differences in DAL and ASIL allocation are rooted in the assumptions each domain makes about its operational environment and the role of human operators.

Avionics and automotive standards are based on markedly different assumptions about operational environment and human involvement. Avionics systems are developed for operation within a highly regulated and structured airspace, supported by certified in-

frastructure, standardized procedures, and trained operators. Within this context, DO-178C relies primarily on design-time assurance and does not depend heavily on real-time human intervention or environmental mitigation for high-severity failure conditions.

By contrast, ISO 26262 and ISO 21448 are tailored to road traffic environments characterized by openness, stronger variability, and a high degree of uncertainty. Automotive systems operate in close proximity to other vehicles and vulnerable road users under dynamic conditions that cannot be fully controlled or standardized. Accordingly, these standards explicitly incorporate assumptions about driver intervention, exposure, and controllability. This leads to a safety argument that distributes assurance according to contextualized risk rather than worst-case consequence alone.

These differences are also reflected in how verification is structured, how traceability is enforced, and how certification evidence is interpreted. In both DO-178C and ISO 26262, verification activities are tightly coupled to development stages and serves as a primary source of certification evidence. However, its role in the overall assurance argument differs.

In the avionics domain, DO-178C defines prescriptive objectives for each DAL, and verification demonstrates direct compliance with those objectives. Requirements-based certification and strict bidirectional traceability between system requirements, software requirements, design artifacts, source code, and verification results form the backbone of the certification process. Certification authorities assess whether the prescribed objectives have been satisfied and whether the resulting evidence is sufficient for airworthiness approval.

In the automotive domain, ISO 26262 likewise requires comprehensive verification and traceability across system, hardware, and software levels. Verification evidence is reviewed within a broader safety argument demonstrating that hazardous events have been mitigated to an acceptable level. Thus, while both standards rely on systematic verification and rigorous traceability, avionics emphasizes objective-driven compliance, whereas automotive places greater weight on safety-argument-based justification of acceptable residual risk.

These cross-domain differences highlight alternative ways of structuring safety assurance, ranging from consequence-driven to risk-based approaches and from prescriptive compliance to argument-based justification. These insights provide the basis for interpreting the safety intent of UAS regulations and for deriving corresponding safety requirements and development processes in the next section.

6. Results

This section presents the results of the analytical approach: (i) a method for deriving system- and software-level safety requirements for UAS from regulatory artifacts (Section 6.1), (ii) a software-centered system lifecycle aligned with these requirements (Section 6.2), and (iii) an interpretation of UAS assurance constructs in relation to established avionics and automotive standards (Section 6.3).

6.1. Deriving Safety Requirements from Regulations for Unmanned Aircraft Systems

Safety requirements for UAS are not provided as explicit development-level specifications but must be derived from a combination of regulatory artifacts, including STS, PDRA, and SORA, as well as supporting guidance such as AMC/GM. These artifacts encode safety intent in terms of operational constraints, mitigation measures, and risk-reduction objectives, rather than as directly implementable system or software requirements.

The analysis shows that regulatory safety intent for UAS is consistently expressed through three complementary elements: (i) operational constraints defining admissible conditions of use, (ii) risk classifications and associated safety objectives capturing acceptable levels of safety, and (iii) mitigation measures and robustness expectations specifying how risks are to be controlled. These elements are distributed across regulatory instruments

but form a coherent structure that can be systematically transformed into development-level artifacts.

Consequently, a structured interpretation process is required to translate regulatory safety intent into system and software requirements. This process synthesizes hazard-related information, operational constraints, and mitigation expectations into explicit safety objectives and requirements.

Supporting standards such as ISO 21384-3 [1] and ISO 21895 [2] provide operational and classification context, while documents such as DO-344 contribute early formulations of UAS safety objectives. Although these standards do not define a development lifecycle, they inform the interpretation of regulatory safety intent and the constraints under which system and software requirements must be defined.

Figure 10 illustrates the resulting method for extracting safety requirements from regulatory artifacts. Although STS, PDRA, and SORA are depicted as symmetric inputs, the analysis reveals distinct conceptual roles: SORA drives hazard identification and safety objective derivation, whereas STS and PDRA constrain the admissible solution space through predefined operational patterns and compliance conditions.

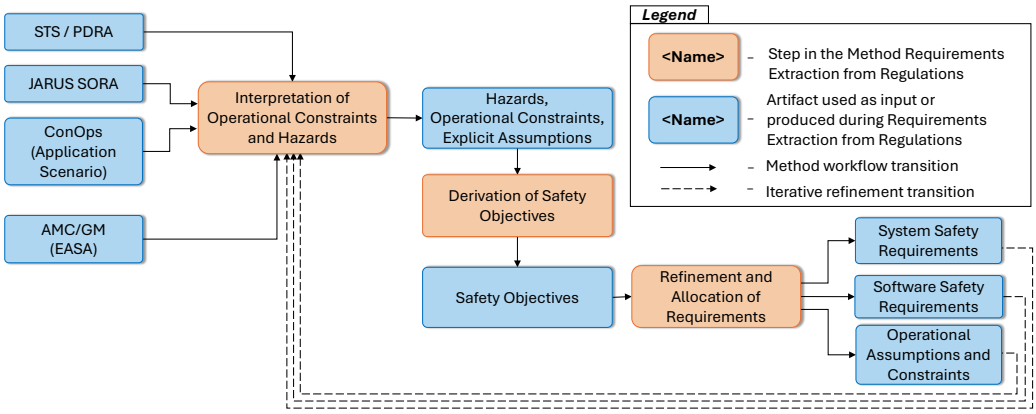


Figure 10. Engineering Method for Extraction of Safety Requirements from UAS Regulatory Artifacts.

The derived method proceeds in three steps. First, operational constraints, hazardous events, and mitigation measures are extracted from regulatory artifacts together with their underlying assumptions, establishing the operational context and risk structure of the intended UAS operation.

Second, these elements are abstracted into technology-independent safety objectives expressing the conditions required to achieve acceptable operational safety. These objectives capture the regulatory safety intent underlying STS, PDRA, and SORA outcomes.

Third, safety objectives are refined into explicit system-level and software-level safety requirements, including allocation to system components, identification of monitoring and mitigation mechanisms, and formalization of operational assumptions that must hold during execution.

The refinement process is iterative and revisited when inconsistencies arise, such as infeasible requirements under the assumed operational envelope, conflicting mitigation measures, or verification assumptions not supported by regulatory constraints. This ensures consistency between regulatory intent, system design, and verification feasibility.

Throughout the process, traceability to originating regulatory sources is maintained to support verification and regulatory justification.

The resulting artifacts—explicit operational assumptions, system-level safety requirements, and software safety requirements—constitute the primary output of the analysis and provide the foundation for the lifecycle defined in the following section.

Illustrative Example: From SORA to Software Requirements

To illustrate the application of the proposed method, consider a UAS operation in the Specific category involving beyond visual line-of-sight (BVLOS) inspection of infrastructure in a semi-urban environment.

Based on SORA, a key hazard is loss of containment leading to ground impact. The corresponding operational safety objective requires maintaining an acceptable level of containment reliability under defined environmental conditions.

Applying the proposed method, we follow the steps:

1. **Extraction of regulatory elements:**
   - Hazard: uncontrolled descent or fly-away
   - Operational constraint: operation within defined geographic boundaries
   - Mitigation: geofencing, fail-safe landing mechanisms
2. **Derivation of safety objectives:**
   - Ensure containment under nominal and degraded conditions
   - Detect and mitigate navigation failures within bounded time
3. **Translation into system/software requirements:**
   - The system shall implement geofencing with enforcement latency < X ms
   - The system shall detect GNSS degradation and trigger fail-safe behavior
   - The flight control software shall ensure controlled descent upon loss of positioning

This example illustrates how regulatory safety intent expressed in SORA can be systematically transformed into implementable system and software requirements.

6.2. Defining a Software-Centered System Lifecycle for Unmanned Aerial Vehicles

This section presents a software-centered system lifecycle for UAS systems derived from the safety requirements and operational assumptions identified in Section 6.1. The lifecycle integrates operational specification, hazard and risk assessment, and architectural allocation, while emphasizing software development and verification as primary means of implementing safety mechanisms.

The analysis indicates that a suitable UAS lifecycle must integrate early operational specification, risk-based hazard analysis, explicit allocation of mitigation responsibilities, staged verification, and structured evidence packaging to support regulatory approval under risk-based frameworks.

The lifecycle does not replace existing regulatory frameworks nor does it define a new certification standard. Instead, it structures development and assurance activities in a way that is consistent with regulatory expectations derived from STS, PDRA, and SORA, and enables systematic traceability from regulatory safety intent to system and software artifacts. Inspired by avionics and automotive practices, the lifecycle adopts a risk-proportionate assurance philosophy grounded in UAS-specific regulatory constructs.

6.2.1. Design Goals

Given the fragmented and risk-driven nature of the regulatory landscape, the lifecycle must be scalable across operational contexts rather than imposing a uniform level of rigor.

It must support both process- and product-oriented evidence, enabling traceability from regulatory sources to requirements and verification results.

Operational constraints from STS, PDRA, and SORA must be integrated early, as they directly shape requirements, architecture, and verification.

Finally, the lifecycle must accommodate mixed criticality and increasing autonomy, allowing differentiated assurance levels within a single system.

6.2.2. Risk-Proportionate Assurance

A central result of the analysis is that development and verification rigor in UAS systems must be scaled in proportion to operational risk. This principle is reflected in regulatory constructs such as SORA, where SAIL and associated OSOs define graduated assurance expectations.

The lifecycle leverages these constructs to modulate requirements specification, verification depth, and evidence generation based on operational risk factors, including SAIL, operational category, exposure, and reliance on autonomy. In this way, it aligns with established safety-critical domains while remaining grounded in UAS-specific regulatory mechanisms.

6.2.3. Phases of the Development Lifecycle

The proposed lifecycle defines a structured development and assurance process for UAS. Throughout its phases, regulatory safety intent is operationalized and progressively transformed into implementable and verifiable system artifacts.

Planning and Assurance Definition. This is the first phase in the UAS lifecycle, which establishes the organizational and technical framework for development. This phase defines development and verification plans, configuration management and quality assurance arrangements, and an initial outline of the safety and compliance argument. Its purpose is to ensure that subsequent activities are conducted in a controlled and traceable manner, with roles, responsibilities, and evidence expectations aligned with the intended operational category and risk profile of the UAS system.

ConOps and ODD Specification. In this phase, the intended operation of the UAS system is specified through a concept of operations (ConOps) and an ODD-style operational description. This includes the operational environment, mission profiles, interaction with other airspace users, environmental conditions, and external constraints derived from regulatory artifacts. These specifications make explicit the operational assumptions under which safety arguments are constructed and provide a foundation for hazard identification and architectural decisions.

Hazard Analysis and Risk Assessment. Based on the operational specification, hazard analysis and risk assessment activities are performed in alignment with SORA or PDRA principles. This phase identifies hazardous events associated with UAS operation, evaluates ground and air risk, and derives safety objectives and required mitigation measures. The outcome is a structured understanding of operational risk and regulatory safety intent, which drives both architectural choices and the rigor applied in subsequent lifecycle phases.

System Architecture and Allocation. System-level architecture is developed to enable the implementation of the identified safety objectives and mitigation strategies. This includes defining functional decomposition, allocating responsibilities to hardware, software, human operators, and external services, and identifying safety mechanisms such as monitoring, fallback, or containment strategies. Architectural allocation ensures that safety requirements are systematically assigned and that interactions between components are explicitly managed under the assumed operational constraints.

Software Lifecycle: Requirements, Architecture, and Implementation. The software lifecycle phase refines allocated system requirements into software safety and functional requirements, which are then realized through software architectural design and implementation. Software architecture captures both static and dynamic aspects of software components, including interfaces, data flows, control logic, and timing behavior. Implementation follows established coding and modeling practices appropriate to the required assurance rigor, with particular attention to safety-critical and autonomy-related components.

Stage-based Verification. Verification activities are conducted incrementally and are tightly coupled to each development stage. Requirements, architectural models, source

code, and integrated software are verified using a combination of reviews, analyses, testing, and simulation, with the depth and formality of verification scaled according to operational risk. This staged verification approach supports early detection of inconsistencies and provides structured evidence that safety requirements are correctly implemented.

Operational Validation and Evidence Packaging. The final phase focuses on validating system behavior in representative operational contexts and consolidating evidence for regulatory or third-party assessment. This includes demonstrating that the UAS system satisfies its safety objectives under the defined operational assumptions and that mitigation measures perform as intended. Verification results, assumptions, and traceability information are packaged into a coherent body of evidence that supports approval, deployment, and continued operation within the approved operational envelope.

6.2.4. Consideration of UAS Operational Infrastructure Context

While the proposed lifecycle focuses on the UAS and its associated software, real-world operations are inherently embedded within a broader infrastructure ecosystem, particularly in emerging Urban Air Mobility (UAM) scenarios.

Elements such as vertiports, vertistops, and low-altitude airspace corridors introduce additional operational constraints and safety considerations [55], including precision landing requirements, communication protocols, traffic density management, and geofencing constraints.

These infrastructure-level characteristics act as boundary conditions for the derivation of system and software requirements. Within the proposed method, they can be incorporated as part of the regulatory and operational inputs from which safety objectives are derived.

Consequently, the approach remains applicable in infrastructure-rich contexts, provided that such constraints are explicitly captured and reflected in the requirements extraction process.

6.3. Cross-Domain Interpretation of Assurance Levels and Safety Objectives

The results presented above are informed by established safety engineering practices from avionics and automotive domains while remaining grounded in UAS-specific regulatory constructs. In particular, they reflect principles of development assurance, staged verification, and risk-proportionate rigor without introducing new integrity levels or claiming equivalence with prescriptive standards.

In avionics and automotive engineering, DALs and ASILs determine development rigor based on failure severity. UAS regulations do not define equivalent development-centric levels, but SORA introduces assurance expectations through SAILs and OSOs at the operational level.

The analysis shows that SAILs and OSOs can be interpreted as indicators of required confidence in safe operation under defined assumptions. They modulate mitigation robustness, system analysis depth, and evidence requirements, playing a role analogous—though not equivalent—to assurance levels in other domains.

The requirements extraction method translates these constructs into system- and software-level requirements, enabling their realization through established development practices. For example, OSOs related to reliability and containment translate into architectural and monitoring requirements that drive software design and verification activities, with rigor scaled according to SAIL.

Similarly, the lifecycle reflects cross-domain principles while adapting them to a regulatory environment centered on operational approval. It combines system-level grounding

and staged verification from avionics with context-driven hazard analysis from automotive, aligning both with UAS regulatory structures.

These results demonstrate that UAS regulations, while operational in nature, implicitly define a structured safety assurance logic. By making this logic explicit and relating it to established engineering practices, the proposed approach provides a systematic basis for developing and assuring software-intensive UAS.

7. Conclusions and Future Work

This paper addressed a fundamental challenge in the development of civilian UAS: the absence of an integrated system and software development lifecycle that aligns regulatory approval frameworks with established safety assurance practices. While European UAS regulations provide a comprehensive basis for operational authorization through risk-based approval mechanisms, they do not prescribe how software-intensive UAS should be developed and verified in a systematic and traceable manner to support these approvals.

The analysis reveals that UAS regulatory frameworks implicitly encode a structured safety assurance logic—through operational constraints, risk classifications, and mitigation requirements—but leave its translation into development-level processes largely unspecified. Making this implicit structure explicit is essential for enabling systematic development, verification, and regulatory justification of UAS systems.

With respect to **RQ-1**, the paper demonstrates how regulatory safety intent embedded in operational approval artifacts can be systematically translated into system- and software-level safety requirements. This is achieved through an engineering method that extracts operational constraints, hazard information, and mitigation expectations from STS, PDRA, and SORA, and refines them into explicit safety objectives and development-level requirements under well-defined operational assumptions.

With respect to **RQ-2**, the paper shows how the resulting safety requirements can be integrated into a development lifecycle aligned with established safety assurance practices. The derived software-centered lifecycle combines operational specification, risk-based hazard analysis, architectural allocation, staged verification, and evidence packaging, while remaining consistent with existing regulatory frameworks and supporting risk-proportionate assurance.

In addition, the paper provides a cross-domain interpretive perspective that relates UAS assurance constructs such as OSOs and SAILs to development assurance principles from avionics and automotive engineering. Rather than establishing direct equivalence, this perspective clarifies how regulatory approval constructs can inform development rigor and evidence expectations.

Taken together, these results demonstrate that regulatory approval mechanisms for UAS can be systematically connected to development and assurance practices. Rather than treating regulation and development as separate concerns, the paper establishes a structured pathway from regulatory safety intent to implementable and verifiable system and software artifacts.

The scope of this study is subject to several limitations. The analysis focuses primarily on European regulations and associated guidance material and does not address variations across other regulatory frameworks, such as those in the United States. Furthermore, the proposed lifecycle is conceptual and does not constitute a certification standard or claim regulatory endorsement. Finally, while the paper draws on established cross-domain practices, it does not provide quantitative mappings or formal equivalence between UAS approval constructs and avionics or automotive safety integrity levels.

Several directions for future work are possible following this study. A first direction concerns empirical validation: applying the proposed requirements extraction method

and lifecycle to concrete UAS development projects. In particular, in the Specific and Certified operational categories, it would be important to gather insights into the practical applicability and scalability of the proposed method for requirements extraction and UAS lifecycle. A second direction involves tool support, especially for managing traceability between regulatory sources, derived safety requirements, lifecycle artifacts, and verification evidence. Automating parts of this traceability could significantly reduce the effort required to build coherent assurance arguments.

In addition, emerging AI-based techniques, including large language models (LLMs), offer potential support for scaling parts of the proposed method, such as parsing regulatory documents, identifying safety-relevant constraints, and supporting traceability between regulatory artifacts and derived requirements [56]. This is particularly relevant in infrastructure-rich UAM contexts, where regulatory, operational, and spatial constraints interact. However, given the safety-critical nature of UAS operations, such techniques would require careful validation and human oversight. Their integration therefore represents a promising direction for future work rather than a replacement for the structured, expert-driven approach presented in this paper.

Further work could also investigate how the proposed lifecycle adapts to increasing levels of autonomy, including learning-enabled components, and how it interfaces with emerging standards for UAS traffic management and cooperative airspace operations.

Funding: This research was partially supported by the Mobil-e-Hub project "Mobil-e-Hub: drohnenbasierte Lieferlogistik–Teilvorhaben: Laufzeitabsicherung mit Dependability Cages und Energiemanagement in einem E-Mobility Logistics System" ("Mobil-e-Hub: Drone-Based Delivery Logistics–Sub-Project: Runtime Assurance with Dependability Cages and Energy Management in an E-Mobility Logistics System"), funded under grant number 01ME19007G by the German Federal Ministry for Economic Affairs and Climate Action (BMWK), January 2020–June 2023, carried out at Institute for Software and Systems Engineering, TU Clausthal. The author acknowledges the support by Open Access Publishing Fund of Clausthal University of Technology for the open access publication of this work.

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author.

Conflicts of Interest: The author declares no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

Abbreviation	Meaning
Core UAS/Regulatory Concepts	
UAS	Unmanned Aircraft System
UAV	Unmanned Aerial Vehicle
EU	European Union
EASA	European Union Aviation Safety Agency
JARUS	Joint Authorities for Rulemaking on Unmanned Systems
Operational Categories and Approval Instruments	
STS	Standard Scenario
PDRA	Pre-Defined Risk Assessment
SORA	Specific Operations Risk Assessment
AMC	Acceptable Means of Compliance
GM	Guidance Material
Risk and Assurance Concepts	
GRC	Ground Risk Class

ARC	Air Risk Class
SAIL	Specific Assurance and Integrity Level
OSO	Operational Safety Objective
TMPR	Tactical Mitigation Performance Requirement
Operational Concepts	
ConOps	Concept of Operations
VLOS	Visual Line of Sight
BVLOS	Beyond Visual Line of Sight
EVLOS	Extended Visual Line of Sight
AO	Airspace Observer
DAA	Detect and Avoid
UTM	Unmanned Aircraft System Traffic Management
Aircraft and Technical Characteristics	
MTOM	Maximum Take-Off Mass
MCD	Maximum Characteristic Dimension
KE	Kinetic Energy
C2/C3 link	Command and Control Link
Standards and Guidance Bodies	
ISO	International Organization for Standardization
RTCA	Radio Technical Commission for Aeronautics
Cross-domain Assurance Concepts	
DAL	Design Assurance Level (avionics)
ASIL	Automotive Safety Integrity Level (automotive)

Appendix A. Classification Categories Used in the Analysis

Table A1 lists the classification categories and admissible values used to organize and analyze the regulatory and standardization sources considered in this paper.

Table A1. Classification categories and admissible values used in the analysis.

Field	Categories
Regulatory Domains	Airworthiness and Certification; Operations and Flight Rules; Airspace Management and UTM (U-Space); Safety Management and Risk Assessment; Environmental and Noise Protection; Security and Data Protection; Training and Licensing; Infrastructure and Support Systems; Cross-domain Harmonization
Regulatory Jurisdiction	National (Country Name); European Economic Area; International
Document Type	Regulation; Acceptable Means of Compliance and Guidance Material (AMC/GM); Non-binding Guideline; Standard
Lifecycle Stage	Concept and Requirements; Design and Development; Verification and Validation; Operations; Decommissioning
Safety/Certification Focus	Safety; Certification; Safety and Certification; None/Not explicit
Cross-Domain Relevance	Not applicable; Informative only; Comparable concepts; Transferable assurance principles
Brief Description	Free-text summary of the document scope and purpose

Table A1. *Cont.*

Field	Categories
Author/Organization	Issuing regulatory authority, standards body, industry association, or author(s)
Publication Year	Year of publication, adoption, or revision
Publication Title	Official title of the regulation, standard, guideline, or other source

Appendix B. Complete Set of Operational Safety Objectives

Table A2 reproduces the complete set of Operational Safety Objectives (OSOs) and associated robustness levels as defined in JARUS SORA v2.5, Annex E [45]. It is provided here for completeness and reference, complementing the representative subset of OSOs discussed in the main body of the paper in Table 8 in Section 4.4 in order to illustrate how robustness expectations scale with the SAIL. The full table serves as a normative background for the regulatory safety intent underlying SORA-based approval. It is not intended to be interpreted as a prescriptive development standard.

Table A2. Recommended Operational Safety Objectives (OSOs) and Robustness Levels by Specific Assurance and Integrity Level (SAIL) (cf. JARUS SORA v2.5, Annex E [45]).

OSO No.	OSO Description	SAIL					
		I	**II**	**III**	**IV**	**V**	**VI**
Technical issue with the UAS							
OSO#01	Ensure the UAS operator is competent and/or proven	O	L	M	H	H	H
OSO#02	UAS manufactured by competent and/or proven entity	O	O	L	M	H	H
OSO#03	UAS maintained by competent and/or proven entity	L	L	M	M	H	H
OSO#04	UAS developed to authority-recognized design standards	O	O	L	L	M	H
OSO#05	UAS is designed considering system safety and reliability	O	O	L	M	H	H
OSO#06	C3 link performance is appropriate for the operation	O	L	L	M	H	H
OSO#07	Inspection of the UAS to ensure consistency with the ConOps	L	L	M	M	H	H
OSO#08	Operational procedures are defined, validated and adhered to	L	M	H	H	H	H
OSO#09	Remote crew trained and current and able to control the abnormal situation	L	L	M	M	H	H
OSO#10	Safe recovery from a technical issue	L	L	M	M	H	H
Deterioration of external systems supporting UAS operations							
OSO#11	Procedures in-place to handle deterioration of external systems supporting UAS operations	L	M	H	H	H	H
OSO#12	UAS designed to manage deterioration of external systems supporting UAS operations	L	L	M	M	H	H
OSO#13	External services supporting UAS operations are adequate for the operation	L	L	M	H	H	H

Table A2. *Cont.*

OSO No.	OSO Description	SAIL					
		I	II	III	IV	V	VI
Human error							
OSO#14	Operational procedures are defined, validated and adhered to	L	M	H	H	H	H
OSO#15	Remote crew trained/current and able to control abnormal situation	L	L	M	M	H	H
OSO#16	Multi-crew coordination	L	L	M	M	H	H
OSO#17	Remote crew is fit to operate	L	L	M	M	H	H
OSO#18	Automatic protection of the flight envelope from human error	O	O	L	M	H	H
OSO#19	Safe recovery from human error	O	O	L	M	M	H
OSO#20	Human factors evaluation and HMI appropriateness	O	L	L	M	M	H
Adverse operating conditions							
OSO#21	Operational procedures for adverse conditions	L	M	H	H	H	H
OSO#22	Remote crew trained to identify/avoid critical environmental conditions	L	L	M	M	M	H
OSO#23	Environmental conditions for safe ops are defined, measurable and adhered to	L	L	M	M	H	H
OSO#24	UAS designed/qualified for adverse environmental conditions	O	O	M	H	H	H

Notes. **O** = Optional objective, **L** = Low robustness, **M** = Medium robustness, **H** = High robustness.

References

1. *ISO 21384-3:2023*; Unmanned Aircraft Systems—Part 3: Operational Procedures. International Organization for Standardization: Geneva, Switzerland, 2023.
2. *ISO 21895:2020*; Categorization and Classification of Civil Unmanned Aircraft Systems. International Organization for Standardization: Geneva, Switzerland, 2020.
3. *ISO/TR 23629-1:2020*; Unmanned Aircraft Systems Traffic Management (UTM)—Part 1: Survey of UTM Concepts. International Organization for Standardization: Geneva, Switzerland, 2020.
4. *RTCA-DO-344:2013*; Operational and Functional Requirements and Safety Objectives for Unmanned Aircraft Systems Standards. RTCA: Washington, DC, USA, 2013.
5. Youn, W.K.; Hong, S.B.; Oh, K.R.; Ahn, O.S. Software Certification of Safety-Critical Avionic Systems: DO-178C and Its Impacts. *IEEE Aerosp. Electron. Syst. Mag.* **2015**, *30*, 4–13. [CrossRef]
6. Moy, Y.; Ledinot, E.; Delseny, H.; Wiels, V.; Monate, B. Testing or Formal Verification: DO-178C Alternatives and Industrial Experience. *IEEE Softw.* **2013**, *30*, 50–57. [CrossRef]
7. *RTCA-DO-178C:2011*; DO-178C: Software Considerations in Airborne Systems and Equipment Certification. RTCA: Washington, DC, USA, 2011.
8. *ISO 26262-1:2018*; Road Vehicles—Functional Safety. Part 1: Vocabulary. International Organization for Standardization: Geneva, Switzerland, 2018.
9. Machrouh, J.; Blanquart, J.P.; Baufreton, P.; Boulanger, J.L.; Delseny, H.; Gassino, J.; Ladier, G.; Ledinot, E.; Leeman, M.; Astruc, J.M. Cross-Domain Comparison of System Assurance. In Proceedings of the 6th European Congress on Embedded Real Time Software and Systems (ERTS 2012), Toulouse, France, 29 January–1 February 2012.
10. Ledinot, E.; Astruc, J.M.; Blanquart, J.P.; Baufreton, P.; Delseny, H.; Gassino, J.; Ladier, G.; Leeman, M.; Machrouh, J.; Quéré, P.; et al. A Cross-Domain Comparison of Software Development Assurance Standards. In Proceedings of the International Conference on Embedded Real Time Software and Systems (ERTS 2012), Toulouse, France, 29 January–1 February 2012.
11. Zrelli, R.; Misson, H.A.; Kamkuimo, S.; Ben Attia, M.; Shabah, A.; de Magalhaes, F.G.; Nicolescu, G. Integrating Formal Methods and Automated Tools for DO-178C Compliance in UAV Software. *Inf. Softw. Technol.* **2026**, *194*, 108068. [CrossRef]

12. Baufreton, P.; Derrien, J.C.; Ricque, B.; Blanquart, J.P.; Boulanger, J.L.; Delseny, H.; Gassino, J.; Ladier, G.; Ledinot, E.; Leeman, M.; et al. Multi-Domain Comparison of Safety Standards. *Rev. Electr. Electron.* **2011**, *2*, 13–25.
13. Alamouri, A.; Lampert, A.; Gerke, M. An Exploratory Investigation of UAS Regulations in Europe and the Impact on Effective Use and Economic Potential. *Drones* **2021**, *5*, 63. [CrossRef]
14. Du, S.; Zhong, G.; Wang, F.; Pang, B.; Zhang, H.; Jiao, Q. Safety Risk Modelling and Assessment of Civil Unmanned Aircraft System Operations: A Comprehensive Review. *Drones* **2024**, *8*, 354. [CrossRef]
15. Fakhraian, E.; Semanjski, I.; Semanjski, S.; Aghezzaf, E.H. Towards Safe and Efficient Unmanned Aircraft System Operations: Literature Review of Digital Twins' Applications and European Union Regulatory Compliance. *Drones* **2023**, *7*, 478. [CrossRef]
16. Habibi, H.; Rao, D.M.K.K.V.; Sanchez-Lopez, J.L.; Voos, H. On SORA for High-Risk UAV Operations under New EU Regulations: Perspectives for Automated Approach. In Proceedings of the International Conference on Unmanned Aircraft Systems (ICUAS), Warsaw, Poland, 6–9 June 2023; pp. 213–220. [CrossRef]
17. Capitán, C.; Capitán, J.; Castaño, A.R.; Ollero, A. Risk Assessment Based on SORA Methodology for a UAS Media Production Application. In Proceedings of the International Conference on Unmanned Aircraft Systems (ICUAS), Atlanta, GA, USA, 11–14 June 2019; pp. 451–459. [CrossRef]
18. Schnüriger, P.; Schreiber, J.; Widmer, K.; Lenhart, P.M. SORA Tool—A Specific Operation Risk Assessment Tool for Civilian Drone Operations. *Drone Syst. Appl.* **2025**, *13*, 1–11. [CrossRef]
19. Nawaz, S.A. Towards Regulating Human Oversight: Challenges for EU Drone Law. *Inf. Commun. Technol. Law* **2025**, 1–17. [CrossRef]
20. Allouch, A.; Koubaa, A.; Khalgui, M.; Abbes, T. Qualitative and Quantitative Risk Analysis and Safety Assessment of Unmanned Aerial Vehicle Missions Over the Internet. *IEEE Access* **2019**, *7*, 53392–53410. [CrossRef]
21. Guerin, J.; Delmas, K.; Guiochet, J. Certifying Emergency Landing for Safe Urban UAV. In Proceedings of the 51st Annual IEEE/IFIP International Conference on Dependable Systems and Networks Workshops (DSN-W), Taipei, Taiwan, 21–24 June 2021; pp. 55–62. [CrossRef]
22. Alharbi, A.; Petrunin, I.; Panagiotakopoulos, D. Assuring Safe and Efficient Operation of UAV Using Explainable Machine Learning. *Drones* **2023**, *7*, 327. [CrossRef]
23. Hawkins, R.; Habli, I.; Kelly, T.; McDermid, J. Assurance Cases and Prescriptive Software Safety Certification: A Comparative Study. *Saf. Sci.* **2013**, *59*, 55–71. [CrossRef]
24. Zeller, M.; Höfig, K.; Rothfelder, M. Towards a Cross-Domain Software Safety Assurance Process for Embedded Systems. In *Proceedings of the Computer Safety, Reliability, and Security (SAFECOMP)*; Springer: Berlin/Heidelberg, Germany, 2014; pp. 396–400.
25. European Union. Regulation (EC) No 1592/2002 of the European Parliament and of the Council of 15 July 2002 on common rules in the field of civil aviation and establishing a European Aviation Safety Agency (Text with EEA relevance). *Off. J.* **2002**, *240*, 1–21.
26. European Union. Regulation (EC) No 216/2008 of the European Parliament and of the Council of 20 February 2008 on common rules in the field of civil aviation and establishing a European Aviation Safety Agency, and repealing Council Directive 91/670/EEC, Regulation (EC) No 1592/2002 and Directive 2004/36/EC (Text with EEA relevance). *Off. J.* **2008**, *79*, 1–49.
27. European Union. Regulation (EU) 2018/1139 of the European Parliament and of the Council of 4 July 2018 on common rules in the field of civil aviation and establishing a European Union Aviation Safety Agency, and amending Regulations (EC) No 2111/2005, (EC) No 1008/2008, (EU) No 996/2010, (EU) No 376/2014 and Directives 2014/30/EU and 2014/53/EU of the European Parliament and of the Council, and repealing Regulations (EC) No 552/2004 and (EC) No 216/2008 of the European Parliament and of the Council and Council Regulation (EEC) No 3922/91 (Text with EEA relevance). *Off. J.* **2018**, *212*, 1–122.
28. European Union. Commission Delegated Regulation (EU) 2019/945 of 12 March 2019 on unmanned aircraft systems and on third-country operators of unmanned aircraft systems. *Off. J.* **2019**, *152*, 1–40.
29. European Union. Commission Implementing Regulation (EU) 2019/947 of 24 May 2019 on the rules and procedures for the operation of unmanned aircraft (Text with EEA relevance). *Off. J.* **2019**, *152*, 45–71.
30. European Union. Commission Implementing Regulation (EU) 2020/639 of 12 May 2020 amending Implementing Regulation (EU) 2019/947 as regards standard scenarios for operations executed in or beyond the visual line of sight (Text with EEA relevance). *Off. J.* **2020**, *150*, 1–31.
31. European Union. Commission Delegated Regulation (EU) 2020/1058 of 27 April 2020 amending Delegated Regulation (EU) 2019/945 as regards the introduction of two new unmanned aircraft systems classes. *Off. J.* **2020**, *232*, 1–27.
32. European Union. Commission Implementing Regulation (EU) 2021/1166 of 15 July 2021 amending Implementing Regulation (EU) 2019/947 as regards postponing the date of application for standard scenarios for operations executed in or beyond the visual line of sight (Text with EEA relevance). *Off. J.* **2021**, *253*, 49–50.
33. European Union. Commission Implementing Regulation (EU) 2022/425 of 14 March 2022 amending Implementing Regulation (EU) 2019/947 as regards postponing the transition dates for the use of certain unmanned aircraft systems in the 'open' category and the date of application for standard scenarios for operations executed in or beyond the visual line of sight (Text with EEA relevance). *Off. J.* **2022**, *87*, 20–21.

34. European Union. Commission Implementing Regulation (EU) 2024/1110 of 10 April 2024 amending Regulation (EU) No 748/2012 as regards the initial airworthiness of unmanned aircraft systems subject to certification and Implementing Regulation (EU) 2019/947 as regards the rules and procedures for the operation of unmanned aircraft. *Off. J.* **2024**. Available online: http://data.europa.eu/eli/reg_impl/2024/1110/oj (accessed on 5 April 2026).
35. European Union. Commission Delegated Regulation (EU) 2024/1108 of 13 March 2024 amending Regulation (EU) No 748/2012 as regards the initial airworthiness of unmanned aircraft systems subject to certification and Delegated Regulation (EU) 2019/945 as regards unmanned aircraft systems and third-country operators of unmanned aircraft systems. *Off. J.* **2024**. Available online: http://data.europa.eu/eli/reg_del/2024/1108/oj (accessed on 5 April 2026).
36. European Union Aviation Safety Agency. *ED Decision 2019/021/R Issuing Acceptable Means of Compliance and Guidance Material to Commission Implementing Regulation (EU) No 2019/947 'Rules and Procedures for the Operation of Unmanned Aircraft'*; Official Publication of the European Union Aviation Safety Agency: Cologne, Germany, 2019.
37. European Union Aviation Safety Agency. *ED Decision 2020/022/R Issuing Regular Update of the Acceptable Means of Compliance and Guidance Material to Regulation (EU) 2019/947*; Official Publication of the European Union Aviation Safety Agency: Cologne, Germany, 2020.
38. European Union Aviation Safety Agency. *ED Decision 2022/002/R Issuing Regular Update of the Acceptable Means of Compliance and Guidance Material to Regulation (EU) 2019/947—Issue 1, Amendment 2*; Official Publication of the European Union Aviation Safety Agency: Cologne, Germany, 2022.
39. European Union Aviation Safety Agency. *ED Decision 2023/012/R Issuing Regular Update of the Acceptable Means of Compliance and Guidance Material to Regulations (EU) 2019/945 and 2019/947 (Drones in the 'Open' and 'Specific' Category)*; Official Publication of the European Union Aviation Safety Agency: Cologne, Germany, 2023.
40. Bundesministerium der Justiz und für Verbraucherschutz. Vierzehntes Gesetz zur Änderung des Luftverkehrsgesetzes—Hinzufügung von „Unbemannten Luftfahrtsystemen" als Luftfahrzeuge (§ 1 Abs. 1 Nr. 2 LuftVG). BGBl. I 2012, S. 606; New Sentence Added to § 1 Abs. 1 LuftVG. 2012. Available online: https://www.buzer.de/gesetz/10159/index.htm (accessed on 15 October 2025).
41. Bundesministerium der Justiz und für Verbraucherschutz. § 1 Luftverkehrsgesetz (LuftVG)—Luftfahrzeuge (Inkl. Flugmodelle und Unbemannte Luftfahrtsysteme). BGBl. I 1959, S. 733; Last Amended by Art. 4 G v. 31.1.2019 I 54. 2019. Available online: https://www.gesetze-im-internet.de/luftvg/__1.html (accessed on 20 October 2025).
42. Bundesministerium der Justiz und für Verbraucherschutz. § 21a Luftverkehrs-Ordnung (LuftVO)—Erlaubnisbedürftiger Betrieb von Unbemannten Luftfahrtsystemen und Flugmodellen. Inserted by Verordnung v. 30.3.2017 I 626, in Force Since 7.4.2017. 2017. Available online: https://www.gesetze-im-internet.de/luftvo_2015/__21a.html (accessed on 30 October 2025).
43. Bundesministerium der Justiz und für Verbraucherschutz. § 21b Luftverkehrs-Ordnung (LuftVO)—Betriebsverbote für Unbemannte Luftfahrtsysteme und Flugmodelle. Inserted by Verordnung v. 30.3.2017 I 626, in Force Since 7.4.2017. 2017. Available online: https://www.gesetze-im-internet.de/luftvo_2015/__21b.html (accessed on 5 November 2025).
44. JARUS Working Group. JARUS SORA Annex F: Specific Assurance and Integrity Level (SAIL) Requirements. 2017. Available online: https://jarus-rpas.org/publications/ (accessed on 5 April 2026).
45. JARUS Working Group. JARUS SORA Annex E: Tactical Mitigation Performance Requirements (TMPR). 2017. Available online: https://jarus-rpas.org/publications/ (accessed on 5 April 2026).
46. SAE International. *ARP4754A: Guidelines for Development of Civil Aircraft and Systems*; EUROCAE ED-79A; SAE International: Warrendale, PA, USA, 2010.
47. SAE International. *ARP4761A: Guidelines and Methods for Conducting the Safety Assessment Process on Civil Aircraft*; EUROCAE ED-135A; SAE International: Warrendale, PA, USA, 2023.
48. Parasoft Corporation. What Is DO-178C? Software Considerations in Airborne Systems and Equipment Certification. 2023. Available online: https://www.parasoft.com/learning-center/do-178c/what-is/ (accessed on 10 December 2025).
49. *ISO 21448:2022*; Road Vehicles—Safety of the Intended Functionality (SOTIF). International Organization for Standardization: Geneva, Switzerland, 2022.
50. *ISO 26262-3:2018*; Road Vehicles—Functional Safety. Part 3: Concept Phase. International Organization for Standardization: Geneva, Switzerland, 2018.
51. *ISO 26262-4:2018*; Road Vehicles—Functional Safety. Part 4: Product Development at the System Level. International Organization for Standardization: Geneva, Switzerland, 2018.
52. *ISO 26262-6:2018*; Road Vehicles—Functional Safety. Part 6: Product Development at the Software Level. International Organization for Standardization: Geneva, Switzerland, 2018.
53. Kapinski, J.; Deshmukh, J.V.; Jin, X.; Ito, H.; Butts, K. Simulation-Based Approaches for Verification of Embedded Control Systems: An Overview of Traditional and Advanced Modeling, Testing, and Verification Techniques. *IEEE Control Syst.* **2016**, *36*, 45–64. [CrossRef]
54. Crots, K.; Skentzos, P.; Bartz, D. A Comparative Analysis of Aviation and Ground Vehicle Software Development Standards. In *Proceedings of the 2014 Ground Vehicle Systems Engineering and Technology Symposium*; National Defense Industrial Association Michigan Chapter: Novi, MI, USA, 2014. [CrossRef]

55. Chen, Z.; Shum, H.Y.; Cao, X.; Hansen, M. Engineering and technology for low-altitude economy infrastructure. *Front. Inf. Technol. Electron. Eng.* **2025**, *26*, 2393–2396. [CrossRef]
56. Weigang, L.; Maia, J.A.; Stenzel, E.; Siefert, L.R. Eixão-UAM: LLM-assisted iterative design of a low-altitude urban air mobility corridor in Brasilia. *Front. Inf. Technol. Electron. Eng.* **2025**, *26*, 2421–2439. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.