Proc. of the European Safety and Reliability Conference (ESREL 2026)
Edited by José C. Matos, Paulo B. Lourenço, Michael Beer, Daniel Oliveira,
Edoardo Patelli, Hélder S. Sousa, Rui A. Silva and Monica Santamaria
 Published by Research Publishing, Singapore.
doi: 10.3850/ESREL2026061419_esrel26-p26092-cd

From fragments of information to evidence. Strengthening collision investigation for software defined vehicles

Thor Myklebust

SESS, SINTEF Digital, Norway. E-mail: thor.myklebust@sintef.no

Modern passenger vehicles are increasingly software-defined, with safety-critical behaviour implemented through networks of electronic control units, sensors, and complex software. When severe incidents occur, such as sudden unintended acceleration (SUA) or unexpected braking, investigators often face fragmentary event logs, incomplete capture of sensor and actuator signals, and limited transparency into manufacturers' safety analyses.
We examine these gaps and present an investigation-oriented framework, including a failure-mode taxonomy, to transform "data fragments" into defensible safety evidence.
Drawing on real-world cases in which recorded vehicle data indicated high accelerator-pedal (throttle) demand while other evidence pointed to deceleration or limited driver intent, our analysis identifies several plausible electronic and software-related failure modes. These include transient power or sampling anomalies, signal-processing misinterpretation, and unintended system activation. It also shows why short pre-crash windows and sparse sampling hinder causal determination without supporting diagnostics and engineering evidence.
Our workflow
• couples engineering reconstruction with Safety Cases and common safety analyses
• motivates an EDR+ oriented set of essential evidence needs for next-generation incident logging and investigation, including pre- and post-event recording duration, sampling rates, and priority signals spanning driver inputs, supervisory functions, power and voltage health, and automated interventions.
Our recommendations align with BSI PAS 1882:2021, Data collection and management for automated vehicle trials for the purpose of incident investigation, as well as with the Data Safety Guidance (2025). We recommend that regulators and investigators extend and standardise incident data capture and require traceable safety evidence, including safety cases and supporting analyses, for critical subsystems, while taking account of human–automation interaction in investigations and approvals. Taken together, these measures will reduce uncertainty, improve causal fairness, and accelerate safety gains across the vehicle life cycle.

Keywords: Collision investigation, Software defined vehicles, Advanced Driver Assistance Systems, Sudden Unintended Acceleration, Event Data Recorder (EDR) and Human–machine interface (HMI).

1. Introduction

Modern road vehicles are best understood as complex software-defined systems rather than traditional mechanical vehicles. Safety-critical behaviour is now realised through tightly integrated electronics, advanced semiconductor devices, and software executing across distributed platforms, including data-driven and AI based functions for perception, monitoring, and decision support. From a functional safety perspective, AI functionality is typically integrated within the software domain and executed on dedicated hardware such as systems-on-chip and AI accelerators. An AI accelerator is hardware designed to run AI workloads efficiently, whose behavior depends on the interaction between the chip, software, and the AI model. Functional safety therefore depends on correct operation of both hardware and software, individually and in combination across the vehicle life cycle, with failures in either domain directly affecting post-incident diagnosability and causal determination. When severe incidents occur, resulting from e.g. sudden unintended acceleration or unexpected braking, this architecture creates significant challenges for collision investigation. Investigators frequently face low-rate and fragmentary event data, incomplete capture of safety-critical sensor and actuator signals, and limited transparency

into manufacturers' safety analyses. Similar observable vehicle behaviour may arise from different electronic, software, or system-level mechanisms, complicating causal determination and undermining learning and fairness.

In safety-critical engineering, such challenges are traditionally addressed through a safety case, understood as a structured, evidence-based argument that a system is acceptably safe for its intended use under defined assumptions (Myklebust et al., 2025). In the automotive domain, systematic safety arguments and supporting evidence have been required since the introduction of the ISO 26262 functional safety standard series in 2010.

Safety cases and related AI assurance standards establish confidence by documenting hazards, assumptions, risk controls, and verification activities. However, this paper examines their limitations in post-incident contexts and proposes a structured investigation framework for software defined vehicles (SDV). Drawing on anonymized real-world cases, it shows how sparse data and system integration effects can complicate analysis and hinder causal determination. While safety cases support approval and certification, they do not by themselves ensure post-incident diagnosability. Effective investigation therefore requires pairing safety claims with explicit evidence-readiness provisions. The proposed failure-mode taxonomy provides a structured basis for identifying which categories of failure must be discriminated after an incident and what runtime evidence must be preserved, enabling more transparent and defensible conclusions.

The core contribution of this paper is an investigation-oriented framework that transforms fragmented post-incident vehicle data into defensible safety evidence by addressing non-unique causation, linking taxonomy-based hypotheses to evidentiary needs, and making uncertainty explicit. Although safety case concepts are well established, investigator-accessible safety cases are generally unavailable for operational road vehicles and are therefore used here as an analytical reference rather than an assumed artefact.

The proposed framework consists of three interacting components. First, the failure-mode taxonomy introduced in Chapter 3 structures the space of plausible causes in software-defined vehicles. Second, the analysis-first investigation approach described in Chapter 5 provides the methodological basis for evaluating these hypotheses using engineering analysis and targeted testing. Third, the investigation workflow presented in Chapter 6 integrates evidence from multiple sources and supports traceable reasoning from observations to conclusions. Together, these components support transparent and defensible post-incident analysis of software-defined vehicles.

Cybersecurity, adversarial threats, and intentional misuse are outside the scope of this paper.

2. Background

Post-incident investigation of SDV operates within a landscape of safety standards and regulatory requirements. Functional safety and AI standards shape safety analyses during development, while EDR regulations determine what evidence is available after incidents. This chapter provides the background for the proposed framework: Section 2.1 summarises relevant safety and AI standards, and Section 2.2 reviews EDR regulations.

2.1 *AI and safety standards*

IEC 61508:2010 is the generic functional safety standard, introducing a lifecycle-based approach and strong documentation requirements to support hazard analysis, safety requirements, design, verification and validation (V&V), operation, modification, and decommissioning. Documentation is treated as essential evidence that functional safety has been achieved and maintained, with Annexes providing example documentation structures across system and software lifecycles. The standard series ISO 26262 "*Automotive*" builds on principles for the automotive domain, defining numerous mandatory work products (a.k.a. documents, logs etc.) across its parts and emphasizing "shippable documents" (Myklebust et al 2022, 2026) such as hazard analyses, safety manuals, tool qualification reports, and lifecycle compliance reports. These documents enable regulatory review, integration by OEMs

(original equipment manufacturers, e.g., Volvo), and certification readiness, including adaptations for machine learning components.

New AI-focused standards are: The generic ISO/IEC TR 5469:2024 "*AI and functional safety*" and ISO/PAS 8800:2024 "*AI and functional safety for automotive*" extend functional safety concepts to AI systems, stressing that safety arguments rely heavily on documented evidence from data management, training, testing, validation, and operational monitoring. ISO 21448:2022 "*Safety of the intended functionality (SOTIF)"* further emphasizes documentation of assumptions, operational design domains, scenarios, and residual risk justification where systems function as intended but remain imperfect.

2.2. *EDR regulations and guidelines*

BSI PAS 1882:2021 (BSI 2021) specifies systematic data collection and management for incident investigation during automated vehicle trials. It includes requirements for pre- and post-event data windows, software and hardware version traceability, and the establishment of information management plans to support post-incident analysis. UNECE, EU, NHTSA (UNECE 2021, EU 2019 and NHTSA 2024) mandates standardized, crash-focused data elements as part of vehicle type approval, ensuring comparability, survivability, and retrievability of core vehicle dynamics and driver input data across manufacturers. The 2024 update to the NHTSA EDR regulation extends pre-crash recording to 20 seconds at higher sampling rates, substantially improving temporal resolution of driver actions and vehicle responses prior to collisions. Compared with earlier EDR requirements, this represents a significant improvement in capturing rapid input changes and short-duration events immediately preceding an incident.

3. Taxonomy in Software Defined Vehicles

This chapter introduces a failure mode taxonomy tailored to SDV, with the objective of explaining why identical observed vehicle behaviour can arise from multiple, non-unique causal mechanisms. The taxonomy is intended to support post-incident technical investigation, rather than design-time optimisation or certification, by structuring plausible failure modes across the electronic, software, and system-integration domains that characterise modern vehicle architectures. The taxonomy builds on established failure-mode approaches but extends them explicitly for post-incident investigation by treating data quality, algorithmic behaviour, and end-to-end integration effects as primary causal categories. Prior research (Yildirim et al 2024, Thieme et al 2020 and Stanton 2020) has shown that hardware- and software-centric taxonomies alone are insufficient to capture the complexity introduced by data-driven perception, algorithmic decision-making, and tightly coupled cyber–physical integration. These limitations motivate explicit consideration of data quality, algorithmic behaviour, and end-to-end effects when interpreting post-incident evidence.

Camera-based vision systems play a central role in perception and driver monitoring in SDV. For post-incident investigation, vision-related evidence, such as camera availability, data quality, preprocessing, and perception outputs, is critical for assessing system behaviour (Ceccarelli et al 2023) and human–automation interaction, but must be interpreted carefully due to sensitivity to lighting, occlusion, calibration, and algorithmic confidence.

Our taxonomy comprises eight categories: (1) hardware faults, (2) software faults, (3) data quality issues, (4) algorithmic limitations, (5) end-to-end integration effects, (6) SOTIF limitations, (7) system-level architectural faults, and (8) human–machine interface and interaction issues. Other contributing domains, such as Operational Design Domain limitations, purely mechanical failures, and human errors, are acknowledged but treated as out of scope except where they interact with software defined functionality or human–machine interfaces. The taxonomy is used to structure the interpretation of observed vehicle behaviour across multiple plausible causal pathways, thereby highlighting why causal determination based on isolated signals or fragmentary logs is inherently unreliable in SDV and providing a transparent basis for defensible post-incident analysis.

Table 3.1 shows that identical vehicle behaviour can arise from multiple failure categories, highlighting non-unique causal determination in SDV.

Taxonomy category	Typical manifestation in incidents	Investigation relevance
Hardware faults	Sensor dropouts, actuator anomalies	May leave transient or no physical trace
Software faults	Logic errors, race conditions, unexpected state	Often indistinguishable from hardware or data faults
Data quality issues	Corrupted, biased sensor data	Can silently drive unsafe decisions
Algorithmic limitations	Unsafe edge case behaviour	Not detectable as faults
End-to-end integration effects	Timing or latency conflicts	Components appear correct alone
SOTIF limitations	Unsafe behaviour without malfunction	Often misclassified as driver error
System-level architectural faults	Redundancy deficiency, fault containment	Root cause may lie above component level
HMI interaction issues	Misleading alerts or unclear system state	Affects causal determination and liability

4. Weaknesses in current EDR regulations and guidelines

While current EDR regulations (UNECE, EU, and NHTSA) and guidelines such as BSI PAS 1882 represent important progress in improving the availability and comparability of vehicle data after serious incidents, they were largely conceived for mechanically dominated vehicles and do not fully reflect the complexity of software-defined vehicle architectures. As a result, significant limitations remain when these regimes are applied to post-incident investigation.

Traditional EDRs primarily capture observable vehicle responses, such as speed and driver inputs, which were often sufficient to explain crashes in earlier vehicle generations. In software-defined vehicles, however, behaviour increasingly emerges from interactions between software, algorithms, sensor data, system integration, and human supervision. Consequently, EDR data often show what the vehicle did but not why, leaving multiple plausible causal explanations, including driver error, software faults, sensor degradation, or algorithmic limitations. This ambiguity complicates accident reconstruction, causal determination, liability assessment, and safety learning.

Despite recent improvements, existing regimes remain largely outcome-oriented, providing limited insight into internal system states, decisions, and failure mechanisms. Hardware faults are typically only inferable indirectly, with limited visibility into power anomalies, ECU resets, or transient failures. Software faults, configuration errors, data quality degradation, algorithmic limitations, and SOTIF-related conditions are weakly represented or not observable, while end-to-end integration effects and system-level architectural faults remain difficult to reconstruct without synchronized multi-ECU evidence. Human–machine interaction is similarly reduced to basic driver inputs, without recording alerts, handovers, or automation state.

To address these limitations, we propose an EDR+ concept that extends conventional EDR by capturing a structured set of additional system-level information, including automation state, timing relationships, software and configuration identity, and indicators of system and data health. EDR+ captures a targeted set of contextual information that should be sufficient to distinguish between competing causal hypotheses in modern software-defined vehicles.

5. Analysis, testing, and compliance-driven verification

Analysis and testing are complementary but distinct activities in post-incident investigation of SDV. Analysis is used to develop and evaluate causal hypotheses based on available evidence, system architecture, and engineering knowledge. Testing is subsequently applied to probe specific hypotheses, boundary conditions, or inconsistencies identified during analysis. Testing alone is rarely sufficient to explain unexpected behaviour, particularly where failure mechanisms are transient, cross-layer, configuration-dependent, or non-deterministic. Consequently, analysis plays a central role in interpreting fragmentary data, identifying inconsistencies, and forming defensible causal hypotheses. In functional safety practice, V&V activities performed during development

establish the expected behaviour of the system and its safety mechanisms. In the present framework these verification results serve as reference evidence against which observed behaviour is interpreted. However, as highlighted in recent work on explainable testing and validation of intelligent automotive systems (Eris et al 2025), compliance-driven V&V must often be complemented by additional analysis and exploratory testing to diagnose unexpected behaviour, integration effects, and residual risk that are not fully captured by requirements alone.

In this work, analysis is performed before, during, and after testing, following an analysis-first mindset (Myklebust et al 2019). Engineering analyses such as hazard analysis, functional failure analysis, change-impact analysis, and system-level reasoning are used to identify plausible failure mechanisms across the taxonomy categories and to determine what evidence should be available if a given hypothesis is correct. This includes evaluating both observed signals and the absence of expected diagnostics, fault flags, or state transitions against design intent, defined safety mechanisms, and assumed diagnostic coverage, as documented in safety analysis such as FMEA (Failure Mode and Effect Analysis) in accordance with ISO 26262. Although complex software and AI-based functions are often described as "black boxes," or opaque, post-incident investigation rarely starts from a position of complete opacity. In practice, partial knowledge of system architecture, inputs, outputs, configuration, timing behaviour, and safety assumptions is usually available. Such systems are therefore more appropriately treated as grey boxes, enabling analysis that combines observable behaviour with known design constraints, process-level failure modes, and documented mitigations, including those identified through ML (Machine Learning) FMEA approaches (Schmitt et al 2025) applied to the machine-learning development pipeline.

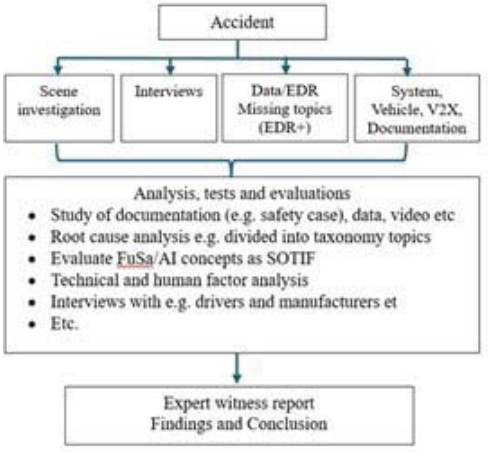
From a functional safety perspective, the analysis explicitly considers whether detected behaviour is consistent with assumed fault detection mechanisms, fault reaction times, and safe-state transitions, and whether the observed data supports or contradicts the claimed diagnostic coverage. Missing or ambiguous diagnostic indications are therefore treated as evidence to be analysed, not as proof of fault absence. Testing is applied selectively to probe specific hypotheses, boundary conditions, or sensitivities identified during analysis, rather than to replace the verification of requirements performed during development. Typical testing activities include cross-signal consistency checks, plausibility assessments against physical constraints, timing and load sensitivity evaluations, and correlation of independent data sources. Where direct reproduction of the event is not possible, testing is used to bound behaviour and assess plausibility rather than to force a binary fault/no-fault conclusion.

6. Evidence-based collision investigation workflow

This chapter presents a structured, investigation-oriented workflow for post-incident analysis of SDV. The workflow supports transparent, traceable, and defensible findings, consistent with principles from forensic collision investigation and independent safety investigations, as reflected in ENFSI (The European Network of Forensic Science Institutes) best practice (ENFSI, 2015) and BSI PAS 1882 guidance. Following an accident or serious incident, information is typically collected from multiple parallel sources, including scene investigation, interviews with involved parties and witnesses, recorded vehicle data (e.g. EDR and related logs), and examination of the vehicle, its systems, and associated documentation. Each source provides partial evidence and may include uncertainty, limited resolution, or gaps in coverage; taken in isolation, such sources rarely support reliable causal determination. Figure 6.1 illustrates how the failure-mode taxonomy, analysis-first investigation approach, and investigation workflow interact to integrate multiple information streams through a dedicated phase of analysis, testing, and evaluation. In this workflow, analysis precedes testing and is used to formulate hypotheses that guide targeted testing activities. This phase includes systematic review of available documentation (such as safety cases, hazard analyses, and design assumptions where accessible), correlation of recorded data and video material, structured root-cause analysis, and integrated technical and human-factor evaluation. Where appropriate, supplementary interviews with drivers, manufacturers, operators, or other stakeholders are conducted to clarify operational context and system assumptions. The workflow follows an analysis-first approach, in which

plausible hypotheses are developed from available evidence before targeted tests or evaluations are undertaken. Testing is applied selectively to assess specific hypotheses rather than as a generic verification activity. All intermediate assessments and results are treated as evidentiary material and must be traceable, reproducible, and suitable for scrutiny. The outcome is a set of findings and conclusions explicitly linked to the underlying evidence, assumptions, and limitations, supporting expert witness reporting and investigation conclusions that are proportionate to the available evidence and reducing the risk of unsupported causal conclusions.

Figure 6.1 Integrated post-incident investigation workflow for SDV.



In practical application, the investigation workflow shown in Figure 6.1 is intentionally evidence-driven rather than strictly linear, allowing repeated synthesis of evidence, refinement of competing hypotheses, and reassessment of uncertainty as new information becomes available during the investigation. Initial evidence collection from the scene, interviews, and recorded vehicle data typically provides only a partial and sometimes contradictory picture of the incident. The workflow therefore emphasises early synthesis of these inputs to formulate multiple plausible hypotheses, explicitly informed by the failure-mode taxonomy introduced in Chapter 3.

For each hypothesis, the workflow guides investigators to identify what additional evidence would be required to confirm or refute that explanation. For example, hypotheses related to hardware faults or power instability imply the need for voltage, reset, or diagnostic indicators, while hypotheses involving software faults, algorithmic limitations, or end-to-end integration effects require time-aligned internal states, control prioritization outcomes, and configuration information. Human–machine interaction hypotheses similarly motivate closer examination of alerts, control transitions, and driver context.

Analysis precedes targeted testing or evaluation. Rather than attempting to reproduce the event indiscriminately, tests are selected to probe specific assumptions, boundary conditions, or inconsistencies revealed during analysis. This may include signal consistency checks across independent channels, plausibility assessments against physical constraints, or controlled evaluation of system responses under representative conditions. Where testing cannot reproduce the event, the workflow still supports defensible conclusions by documenting why certain hypotheses remain plausible or cannot be excluded. A key feature of the workflow is explicit documentation of uncertainty and evidentiary gaps. Instead of forcing a single root cause, findings are expressed in terms of supported, weakened, or unresolved hypotheses, each linked to the available evidence and its limitations. This approach aligns with forensic best practice and supports transparent expert reporting, fair causal determination, and meaningful safety learning even when definitive causation cannot be established.

7. Anonymized case illustrations from Post-Incident Investigations

This chapter presents two anonymized case illustrations drawn from real post-incident investigations of modern passenger vehicles. The cases are deliberately de-identified and abstracted to prevent any linkage to specific vehicles, manufacturers, jurisdictions, or court proceedings. They are included solely to illustrate recurring investigation challenges and to demonstrate the applicability of the proposed framework.

Case A concerns a severe low-speed collision in an urban environment where recorded vehicle data indicated sustained accelerator demand immediately prior to impact. Scene evidence, witness statements, and physical damage patterns, however, were inconsistent with continuous intentional acceleration by the driver. The available event data were limited to a short pre-impact window with coarse temporal resolution, and no

diagnostic or system-level logs were accessible. Analysis therefore had to consider multiple plausible hypotheses, including transient electronic effects, signal interpretation anomalies, and integration effects between driver inputs and control systems. The case illustrates how reliance on isolated EDR parameters can lead to early causal conclusions when supporting evidence is absent.

Case B involves a roadway departure following passage over an uneven surface, where the driver reported unexpected vehicle behaviour. Recorded data showed conflicting indications of braking, acceleration, and vehicle deceleration. No definitive mechanical faults were identified in post-incident inspections. The investigation was constrained by the absence of manufacturer safety analyses (e.g. FMEA) and by limited insight into system assumptions and failure handling. As in Case A, several competing causal explanations remained technically plausible.

Taken together, these cases demonstrate that post-incident investigation of SDV must accommodate non-unique causation, integrate heterogeneous evidence sources, and explicitly document uncertainty. They motivate the need for evidence-ready systems, extended incident data capture, and traceable safety documentation to support defensible conclusions.

8. Discussion. Implications, Limitations, and Open Challenges

The framework presented in this paper has several implications for post-incident investigation of SDV. First, it demonstrates that collision analysis can no longer rely on isolated signals or single-cause explanations. The combination of distributed electronics, complex software, data-driven algorithms, and human–automation interaction makes non-unique causation a normal condition rather than an exception. The proposed taxonomy and investigation workflow provide a structured means to manage this complexity by preserving multiple plausible hypotheses and linking them explicitly to evidentiary requirements. The need to explicitly manage uncertainty and avoid premature causal closure is reinforced by recent maturity models for accident investigation, which show that breakdowns in shared understanding can remain invisible even in technically and procedurally sound analyses (Steen et al, 2025).

Second, the framework highlights a shift from outcome-based reconstruction toward evidence-based reasoning. By integrating engineering reconstruction with safety cases, classical safety analyses, and explicit evidence-readiness considerations, investigators can better explain why causal determination remains uncertain in some cases, rather than defaulting to simplified explanations such as driver error. This has implications not only for technical investigations, but also for fairness, transparency, and learning in regulatory and judicial contexts.

The work also has limitations. The framework depends on the availability and accessibility of vehicle-internal data and supporting documentation. In many current cases, investigators lack access to extended logs, diagnostic data, or manufacturer safety artefacts such as FMEA, which constrains the achievable level of certainty. Legacy EDR implementations, with short pre-event windows and low sampling rates, further limit reconstruction capability. In addition, the framework does not address cybersecurity threats or intentional misuse, which are explicitly out of scope but increasingly relevant for future vehicles.

Several open challenges remain. Standardisation of incident data capture beyond current EDR requirements, practical mechanisms for sharing safety-relevant documentation while protecting intellectual property, and training of investigators in software and system-oriented analysis are all necessary to realise the full potential of the approach. Future work should also examine how the framework can be operationalised across jurisdictions and extended to higher levels of vehicle automation.

9. Conclusion

This paper has examined the challenges of post-incident investigation in SDV, where safety-critical behaviour increasingly emerges from interactions between electronics, software, data, algorithms, and human–automation interfaces. By focusing on transforming data fragments into defensible evidence, the proposed framework supports management of non-unique causation, links observed behaviour to safety-relevant evidence and enables transparent conclusions that acknowledge residual uncertainty. Analysis of anonymised real-world cases shows how fragmentary event data and limited access to

manufacturer safety analyses can lead to ambiguous or early causal conclusions in incidents such as sudden unintended acceleration or unexpected braking.

To address these challenges, the paper introduces a structured investigation framework centred on a failure-mode taxonomy tailored to SDV. The taxonomy explains why similar observed behaviour may arise from multiple plausible causes and supports disciplined reasoning by preserving competing hypotheses and linking them to evidentiary requirements. The associated workflow integrates engineering reconstruction with safety cases, classical safety analyses such as FMEA, and analysis-first testing, consistent with forensic investigation practice and guidance such as BSI PAS 1882.

The paper also identifies limitations in current EDR regulations. Despite recent improvements, they remain largely outcome-oriented and insufficient for discriminating between complex electronic, software, and system-integration mechanisms. This motivates the need for extended incident data capture, explicit evidence-readiness provisions, and traceable safety documentation.

Overall, the proposed framework supports fairer causal determination, clearer communication of uncertainty, and improved learning from incidents. For practitioners, the framework provides a practical way to reason about incidents when definitive causation cannot be established, without defaulting to oversimplified explanations. For regulators and investigators, priorities include standardising incident data capture, improving access to safety-relevant documentation, and strengthening software and system-oriented investigation competence to enhance safety across the vehicle life cycle.

Acknowledgement

This work has been supported by Expert witness projects and the SINTEF project *HEART 2026*.

References

Yildirim, U., Campean, F., & Younus, H. (2024). Robustness of autonomous driving systems: Extending the function failure modes taxonomy. Procedia CIRP, 128, 722–727.

T. Myklebust, T. Stålhane and D. Vatn (2025, January). Which chapters and topics should be included in a safety case? Paper presented at 71st Annual Reliability and Maintainability Symposium (RAMS 2025), Miramar Beach, Florida.

Thieme, C. A., Mosleh, A., Utne, I. B., & Hegde, J. (2020). Incorporating software failure in risk analysis—Part 1: Software functional failure mode classification. Reliability Engineering & System Safety, 197, Article 106803.

Stanton, N. A. (2022). Road Collision Investigation Project: Development of taxonomies and meta-analysis. RAC Foundation

Schmitt, P., Seifert, H. B., Bijelic, M., Pennar, K., Lopez, J., & Heide, F. (2025). Introducing the ML FMEA: A safe machine learning approach (SAE Technical Paper No. 2025-01-8078). SAE International.

Data safety guidance. Version 3.7: 2025. SCSC Publication No. SCSC 127J).

T. Myklebust, T. Stålhane and G. K. Hanssen. Analysis first development for agile development of safety critical software (AFD). ESREL 2019 Germany

BSI PAS 1882:2021 Data for Automated Vehicle Trials Incident Investigation

UNECE UN Regulation No. 160: Uniform provisions concerning the approval of motor vehicles with regard to the event data recorder (EDR) (E/ECE/TRANS/505/Rev.3/Add.159).

Regulation EU 2019/2144 on type-approval requirements for motor vehicles and their trailers, and systems, components and separate technical units intended for such vehicles. Official Journal of the European Union, L 325, 1–40.

National Highway Traffic Safety Administration (NHTSA). (2024, December 18). Event Data Recorders (Final Rule No. NHTSA–2024–0084, RIN 2127–AM12, 49 CFR Part 563). Doc identification: NHTSA-2024-0084.

ENFSI 2015. Best practice manual for road accident reconstruction. ENFSI-BPM-RAA-01. V01. November 2015.

T. Myklebust and P. Okoh. Functional safety. Sprints and Kanban approach in practice. ISSC 40 International Conference on System Safety. Cincinnati 2022

Eris, H., & Wagner, S. (2025). Revolutionizing validation and verification: Explainable testing methodologies for intelligent automotive decision-making systems.

T. Myklebust. Shippable document and market access for high-risk AI systems. ESREL 2026 To be published.

Steen, R., & Johnsen, S. O. (2025). A maturity model for accident investigation: Beyond technical and functional analysis. Safety Science, To be published, 107108.

Ceccarelli, A., & Secci, F. (2023). RGB cameras failures and their effects in autonomous driving applications. IEEE Transactions on Dependable and Secure Computing, 20(4), 2731–2745.