



IEEE Standard for Functional Safety Data Format for Interoperability within the Dependability Lifecycle

IEEE Computer Society

Developed by the
Functional Safety Standards Committee

IEEE Std 2851™-2023

IEEE Standard for Functional Safety Data Format for Interoperability within the Dependability Lifecycle

Developed by the

Functional Safety Standards Committee
of the
IEEE Computer Society

Approved 21 September 2023

IEEE SA Standards Board

Abstract: A dependability lifecycle of products with focus on interoperable activities related to functional safety and its interactions with reliability, security, operational safety and time determinism are defined in this standard. The standard also describes methods, description languages, data models, and database schema that have been identified as necessary or critical, to enable the exchange/interoperability of data across all steps of the lifecycle encompassing activities executed at intellectual property (IP), system-on-chip (SoC), system and item levels, in a way that allows integration in different application domains such as automotive, industrial, medical and avionics safety critical systems.

Keywords: dependability, functional safety, IEEE 2851™, interoperability

The Institute of Electrical and Electronics Engineers, Inc.
3 Park Avenue, New York, NY 10016-5997, USA

Copyright © 2023 by The Institute of Electrical and Electronics Engineers, Inc.
All rights reserved. Published 15 December 2023. Printed in the United States of America.

IEEE is a registered trademark in the U.S. Patent & Trademark Office, owned by The Institute of Electrical and Electronics Engineers, Incorporated.

PDF: ISBN 979-8-8557-0295-8 STD26602
Print: ISBN 979-8-8557-0296-5 STDPD26602

IEEE prohibits discrimination, harassment, and bullying.

For more information, visit <https://www.ieee.org/about/corporate/governance/p9-26.html>.

No part of this publication may be reproduced in any form, in an electronic retrieval system or otherwise, without the prior written permission of the publisher.

Important Notices and Disclaimers Concerning IEEE Standards Documents

IEEE Standards documents are made available for use subject to important notices and legal disclaimers. These notices and disclaimers, or a reference to this page (<https://standards.ieee.org/ipr/disclaimers.html>), appear in all standards and may be found under the heading “Important Notices and Disclaimers Concerning IEEE Standards Documents.”

Notice and Disclaimer of Liability Concerning the Use of IEEE Standards Documents

IEEE Standards documents are developed within IEEE Societies and subcommittees of IEEE Standards Association (IEEE SA) Board of Governors. IEEE develops its standards through an accredited consensus development process, which brings together volunteers representing varied viewpoints and interests to achieve the final product. IEEE Standards are documents developed by volunteers with scientific, academic, and industry-based expertise in technical working groups. Volunteers are not necessarily members of IEEE or IEEE SA and participate without compensation from IEEE. While IEEE administers the process and establishes rules to promote fairness in the consensus development process, IEEE does not independently evaluate, test, or verify the accuracy of any of the information or the soundness of any judgments contained in its standards.

IEEE makes no warranties or representations concerning its standards, and expressly disclaims all warranties, express or implied, concerning this standard, including but not limited to the warranties of merchantability, fitness for a particular purpose and non-infringement. In addition, IEEE does not warrant or represent that the use of the material contained in its standards is free from patent infringement. IEEE standards documents are supplied “AS IS” and “WITH ALL FAULTS.”

Use of an IEEE standard is wholly voluntary. The existence of an IEEE Standard does not imply that there are no other ways to produce, test, measure, purchase, market, or provide other goods and services related to the scope of the IEEE standard. Furthermore, the viewpoint expressed at the time a standard is approved and issued is subject to change brought about through developments in the state of the art and comments received from users of the standard.

In publishing and making its standards available, IEEE is not suggesting or rendering professional or other services for, or on behalf of, any person or entity, nor is IEEE undertaking to perform any duty owed by any other person or entity to another. Any person utilizing any IEEE Standards document, should rely upon his or her own independent judgment in the exercise of reasonable care in any given circumstances or, as appropriate, seek the advice of a competent professional in determining the appropriateness of a given IEEE standard.

IN NO EVENT SHALL IEEE BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO: THE NEED TO PROCURE SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE PUBLICATION, USE OF, OR RELIANCE UPON ANY STANDARD, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE AND REGARDLESS OF WHETHER SUCH DAMAGE WAS FORESEEABLE.

Translations

The IEEE consensus development process involves the review of documents in English only. In the event that an IEEE standard is translated, only the English version published by IEEE is the approved IEEE standard.

Official statements

A statement, written or oral, that is not processed in accordance with the IEEE SA Standards Board Operations Manual shall not be considered or inferred to be the official position of IEEE or any of its committees and shall not be considered to be, nor be relied upon as, a formal position of IEEE. At lectures, symposia, seminars, or educational courses, an individual presenting information on IEEE standards shall make it clear that the presenter's views should be considered the personal views of that individual rather than the formal position of IEEE, IEEE SA, the Standards Committee, or the Working Group. Statements made by volunteers may not represent the formal position of their employer(s) or affiliation(s).

Comments on standards

Comments for revision of IEEE Standards documents are welcome from any interested party, regardless of membership affiliation with IEEE or IEEE SA. However, **IEEE does not provide interpretations, consulting information, or advice pertaining to IEEE Standards documents.**

Suggestions for changes in documents should be in the form of a proposed change of text, together with appropriate supporting comments. Since IEEE standards represent a consensus of concerned interests, it is important that any responses to comments and questions also receive the concurrence of a balance of interests. For this reason, IEEE and the members of its Societies and subcommittees of the IEEE SA Board of Governors are not able to provide an instant response to comments, or questions except in those cases where the matter has previously been addressed. For the same reason, IEEE does not respond to interpretation requests. Any person who would like to participate in evaluating comments or in revisions to an IEEE standard is welcome to join the relevant IEEE working group. You can indicate interest in a working group using the Interests tab in the Manage Profile & Interests area of the [IEEE SA myProject system](#).¹ An IEEE Account is needed to access the application.

Comments on standards should be submitted using the [Contact Us](#) form.²

Laws and regulations

Users of IEEE Standards documents should consult all applicable laws and regulations. Compliance with the provisions of any IEEE Standards document does not constitute compliance to any applicable regulatory requirements. Implementers of the standard are responsible for observing or referring to the applicable regulatory requirements. IEEE does not, by the publication of its standards, intend to urge action that is not in compliance with applicable laws, and these documents may not be construed as doing so.

¹ Available at: <https://development.standards.ieee.org/myproject-web/public/view.html#landing>.

² Available at: <https://standards.ieee.org/content/ieee-standards/en/about/contact/index.html>.

Data privacy

Users of IEEE Standards documents should evaluate the standards for considerations of data privacy and data ownership in the context of assessing and using the standards in compliance with applicable laws and regulations.

Copyrights

IEEE draft and approved standards are copyrighted by IEEE under US and international copyright laws. They are made available by IEEE and are adopted for a wide variety of both public and private uses. These include both uses, by reference, in laws and regulations, and use in private self-regulation, standardization, and the promotion of engineering practices and methods. By making these documents available for use and adoption by public authorities and private users, neither IEEE nor its licensors waive any rights in copyright to the documents.

Photocopies

Subject to payment of the appropriate licensing fees, IEEE will grant users a limited, non-exclusive license to photocopy portions of any individual standard for company or organizational internal use or individual, non-commercial use only. To arrange for payment of licensing fees, please contact Copyright Clearance Center, Customer Service, 222 Rosewood Drive, Danvers, MA 01923 USA; +1 978 750 8400; <https://www.copyright.com/>. Permission to photocopy portions of any individual standard for educational classroom use can also be obtained through the Copyright Clearance Center.

Updating of IEEE Standards documents

Users of IEEE Standards documents should be aware that these documents may be superseded at any time by the issuance of new editions or may be amended from time to time through the issuance of amendments, corrigenda, or errata. An official IEEE document at any point in time consists of the current edition of the document together with any amendments, corrigenda, or errata then in effect.

Every IEEE standard is subjected to review at least every 10 years. When a document is more than 10 years old and has not undergone a revision process, it is reasonable to conclude that its contents, although still of some value, do not wholly reflect the present state of the art. Users are cautioned to check to determine that they have the latest edition of any IEEE standard.

In order to determine whether a given document is the current edition and whether it has been amended through the issuance of amendments, corrigenda, or errata, visit [IEEE Xplore](#) or [contact IEEE](#).³ For more information about the IEEE SA or IEEE's standards development process, visit the IEEE SA Website.

Errata

Errata, if any, for all IEEE standards can be accessed on the [IEEE SA Website](#).⁴ Search for standard number and year of approval to access the web page of the published standard. Errata links are located under the

³ Available at: <https://ieeexplore.ieee.org/browse/standards/collection/ieee>.

⁴ Available at: <https://standards.ieee.org/standard/index.html>.

Additional Resources Details section. Errata are also available in [IEEE Xplore](#). Users are encouraged to periodically check for errata.

Patents

IEEE Standards are developed in compliance with the [IEEE SA Patent Policy](#).⁵

Attention is called to the possibility that implementation of this standard may require use of subject matter covered by patent rights. By publication of this standard, no position is taken by the IEEE with respect to the existence or validity of any patent rights in connection therewith. If a patent holder or patent applicant has filed a statement of assurance via an Accepted Letter of Assurance, then the statement is listed on the IEEE SA Website at <https://standards.ieee.org/about/sasb/patcom/patents.html>. Letters of Assurance may indicate whether the Submitter is willing or unwilling to grant licenses under patent rights without compensation or under reasonable rates, with reasonable terms and conditions that are demonstrably free of any unfair discrimination to applicants desiring to obtain such licenses.

Essential Patent Claims may exist for which a Letter of Assurance has not been received. The IEEE is not responsible for identifying Essential Patent Claims for which a license may be required, for conducting inquiries into the legal validity or scope of Patents Claims or determining whether any licensing terms or conditions provided in connection with submission of a Letter of Assurance, if any, or in any licensing agreements are reasonable or non-discriminatory. Users of this standard are expressly advised that determination of the validity of any patent rights, and the risk of infringement of such rights, is entirely their own responsibility. Further information may be obtained from the IEEE Standards Association.

IMPORTANT NOTICE

IEEE Standards do not guarantee or ensure safety, security, health, or environmental protection, or ensure against interference with or from other devices or networks. IEEE Standards development activities consider research and information presented to the standards development group in developing any safety recommendations. Other information about safety practices, changes in technology or technology implementation, or impact by peripheral systems also may be pertinent to safety considerations during implementation of the standard. Implementers and users of IEEE Standards documents are responsible for determining and complying with all appropriate safety, security, environmental, health, and interference protection practices and all applicable laws and regulations.

⁵ Available at: <https://standards.ieee.org/about/sasb/patcom/materials.html>.

Participants

At the time this standard was completed, the P2851 Working Group had the following membership:

Jyotika Athavale, Chair
Nir Maor, Vice Chair
Wei-Ren Chen, Secretary

<i>Organization Represented</i>	<i>Name of Representative</i>
Accellera	Darren Galpin
Advanced Micro Devices (AMD)	Bala Chavali
ARM	Luca Di Mauro
Arteris Inc.	Stefano Lorenzini
Cadence Design Systems, Inc.	Francesco Lertora
Ford.....	Kyle Post
Hitachi	Hiroaki Itsuji
Horizon AI	Ziyu Meng
Infineon.....	Ekaterina Vlasova
Intel.....	Giuseppe Capodanno
IrocTech	Thierry Bonnoit
Micron	Marco Montagnana
NVIDIA	Hui Jin
NXP	Franck Galtie
ON Semiconductor	Yves Renard
Qualcomm Technologies, Inc.....	Nir Maor
SGS TUV	Wei-Ren Chen
Siemens-EDA	Jake Wiltgen
ST Microelectronics.....	Om Ranjan
Synopsys, Inc.....	Jyotika Athavale
Texas Instruments, Inc.	Prasanth Viswanathan Pillai

The following members of the entity Standards Association balloting group voted on this standard. Balloters may have voted for approval, disapproval, or abstention.

Advanced Micro Devices (AMD)	Cadence Design Systems, Inc. CRRZ Zhuzhou Institute Co. Ltd.	Intel
ARM	IBM	NVIDIA
Arteris Inc.	Infineon Technologies	ON Semiconductor
Beijing Horizon Robotics Technology Research and Development Co., Ltd.	Institute of Biomedical Engineering	Qualcomm Technologies, Inc
		Siemens Corporation
		Synopsys, Inc.
		Texas Instruments, Inc.

When the IEEE SA Standards Board approved this standard on 21 September 2023, it had the following membership:

David J. Law, *Chair*
Ted Burse, *Vice Chair*
Gary Hoffman, *Past Chair*
Konstantinos Karachalios, *Secretary*

Sara R. Biyabani
Doug Edwards
Ramy Ahmed Fathy
Guido R. Hiertz
Yousef Kimiagar
Joseph L. Koepfinger*
Thomas Koshy
John D. Kulick

Joseph S. Levy
Howard Li
Johnny Daozhuang Lin
Gui Lin
Xiaohui Liu
Kevin W. Lu
Daleep C. Mohla
Andrew Myles

Paul Nikolich
Annette D. Reilly
Robby Robson
Lei Wang
F. Keith Waters
Karl Weber
Philip B. Winston
Don Wright

*Member Emeritus

Introduction

This introduction is not part of IEEE Std 2851-2023, IEEE Standard for Functional Safety Data Format for Interoperability within the Dependability Lifecycle.

Dependability is a fundamental characteristic of intelligent machines. In the context of this document, dependability is intended as the property of the machine to perform its mission reliably, safely, securely, and in an available, maintainable, serviceable, and time-deterministic manner. Dependability is achieved by performing specific activities during the whole lifecycle of the machine, in particular, analysis and verification activities. Standards such as ISO 26262, IEC 61508, and many others define the complete set of safety-related activities that need to be performed, requiring companies at various levels of their respective supply chains to tailor the safety-related analysis and verification activities that apply to them and deliver results to other levels of the chain for which other requirements are applicable. In addition, these standards also propose various details of development lifecycles, which vary in scope, format, terminology, and consistency of language. This lack of uniformity already presents industry challenges to developments that need to address multiple standards. The scope of IEEE Std 2851 spans industry domains; therefore, it is essential to define the lifecycle in a manner that provides a common language and understanding to enable translation with lifecycles defined by other standards.

Electronic design automation (EDA) vendors have already started to provide tools to automate those activities. However, currently, there are no common methodologies, languages, or formats to provide those results. Furthermore, today there is not a cohesive view of all the activities that are executed, often in a disjointed way. As a result of this gap, companies are struggling with many different types of methodologies and description languages and are investing valuable time and effort to reconsolidate, compare, integrate, and combine the data. For this reason, the safety critical community is urgently asking for a solution to accelerate the safety engineering process while reducing risks and costs.

In that context, the IEEE P2851 standardization initiative was started to define and deliver a comprehensive view of all those activities. One objective was to define a format for exchange/interoperability for safety-related analysis and verification activities in the context of dependability and facilitate companies in delivering results in a consistent way. Additionally, the work products of this standardization initiative will hopefully enable interoperability between tools.

The interoperable activities to be performed are identified by this standard within a dependability lifecycle to address the dependability of intelligent machines during their development and usage in the field, with safety as the primary scope, while considering other dependability attributes that could impact safety. The definition of needs is included for each activity, in terms of methodologies, description languages, data models, and databases that have been identified as necessary or critical to perform those activities.

IEEE Std 2851-2023 has been inspired by two standards related to lifecycles: ISO/IEC/IEEE 15288 [B12]⁶ and ISO/IEC/IEEE 24748 [B13]. ISO/IEC/IEEE 15288 defines a process as a set of interrelated activities that transform inputs into outputs. A process is described in terms of five characteristics—title, purpose, outcome, activities, and tasks necessary to execute the transformation. This level of description reduces the potential for misinterpretation and improves the ability to translate with other standards. ISO/IEC/IEEE 15288 [B12] goes on to define a complete set of processes necessary to enable the development of a product through conception to retirement. These processes into the following four categories:

- *Agreement processes*: Deals with supply and acquisition between organizations and within organizations.
- *Technical processes*: Deals with technical actions that transform the product from one form to another form.

⁶ The numbers in brackets correspond to those of the bibliography in Annex D.

- *Technical management processes*: Deals with managing resources and assets to enable the execution of the technical processes in a manner that meets what has been agreed in the agreement processes.
- *Organizational project enabling processes*: Deals with establishing the environment in which development projects are conducted.

Figure 1 summarizes these processes. The intent is not to say that this is the only set of processes by which a product may be developed. Indeed, ISO/IEC/IEEE 15288 [B12] allows for tailoring of processes and acknowledges that an organization may identify additional useful processes.

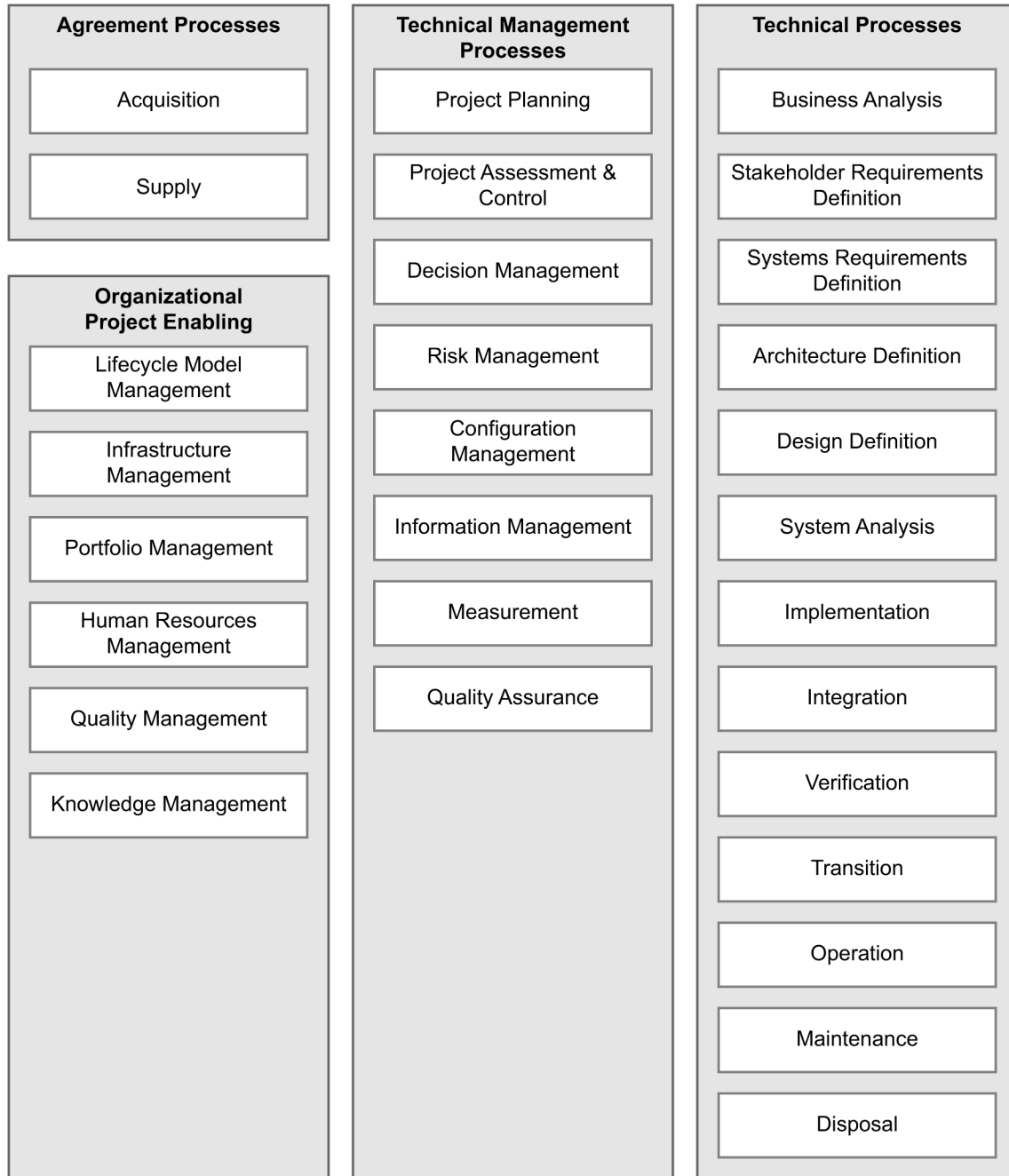


Figure 1—ISO/IEC/IEEE 15288 system lifecycle processes

ISO/IEC/IEEE 24748 [B13] provides guidelines on how ISO/IEC/IEEE 15288 [B12] processes may be arranged to enable product development. A lifecycle model is expressed in terms of the specific processes employed and the relationships and interdependencies between them. It defines a stage, to represent a major lifecycle period which relates to the state of the products description or realization. Entry and exit criteria may be defined for each stage, enabling support for strong project control and monitoring.

Both ISO/IEC/IEEE 15288 [B12] and ISO/IEC/IEEE 24748 [B13] highlight the importance of the iterative and recursive nature of processes in product development.

Recursive application of process, i.e., the same process applied at various levels of granularity, is expected, as requirements at one level create requirements on lower levels. Figure 2 provides an automotive centric example of this recursive nature, taking an automated driving system as an example. The automated driving subsystem may place requirements on the localization system. The localization system in turn comprises of various subsystems including a global navigation satellite system (GNSS) and an inertial measurement unit (IMU), each of which inherit requirements. The IMU in turn is comprised of multiple integrated circuit and software components, all of which inherit their own requirements.

NOTE—To describe in a consistent way the various levels of granularity, this document adopts the taxonomy of development defined in ISO 26262-1, i.e., item, system, component, hardware (HW) part, and software (SW) unit. In some descriptions, this document also makes use of IEC 61508 taxonomy. Moreover, this document also uses terminology such as SoC, FPGA (that are equivalent to HW component level of ISO 26262 taxonomy), IP (that is equivalent to HW part level of ISO 26262 taxonomy) and firmware (FW) (that is equivalent to SW component level of ISO 26262 taxonomy). A “product” refers to any of those levels.

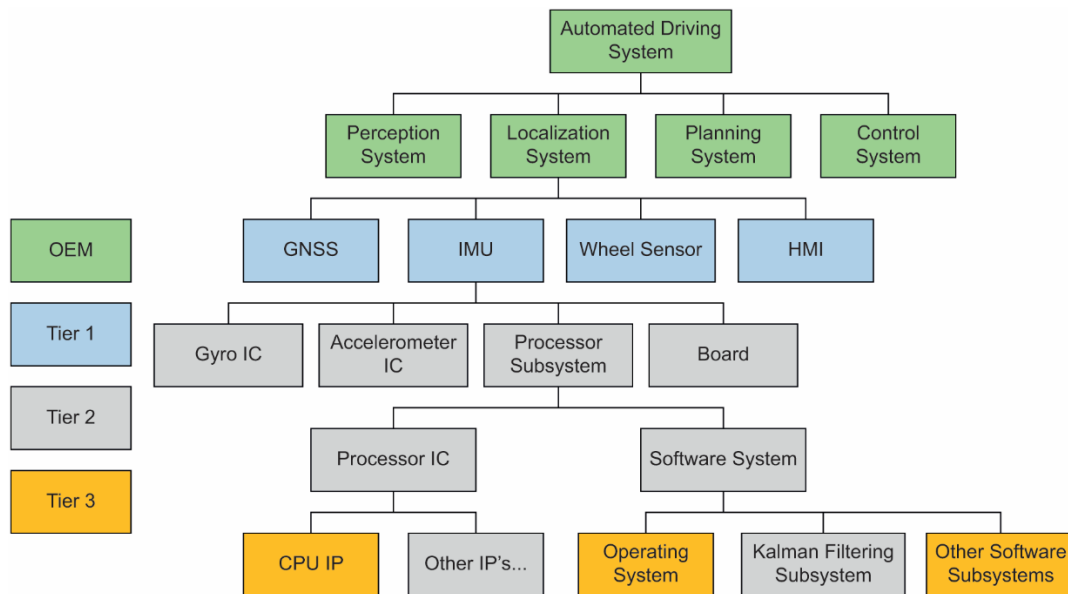


Figure 2—Example of recursive application of processes

It can be recognized that each of these developments may go through its own development lifecycle and interface at the appropriate points with the levels of granularity above and below.

Iterative application of process, i.e., a process repeated on the same level of product granularity, is not only appropriate but also expected to account for evolving knowledge and trade-offs. Figure 3 presents a simplified view of the iterative relationship between the requirements, architecture, and design processes. The intent is to highlight that certain requirements cannot be derived until some portion of the architecture or design evolves. This iterative nature applies to all processes, not only the requirements process.

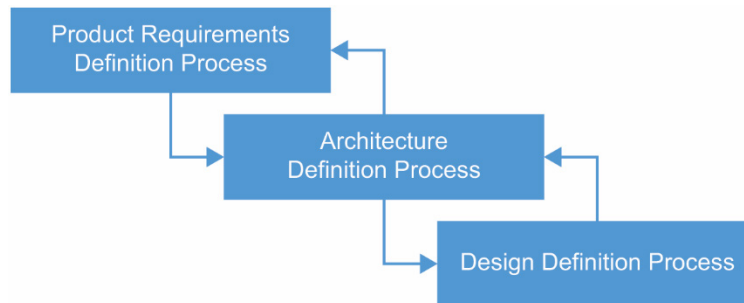


Figure 3—Iterative relationship between requirements, architecture, and design processes

ISO/IEC/IEEE 24748 further introduces the notion of a development strategy, which prescribes certain processes and activities that may be performed sequentially, repeated, and/or combined. The lifecycle model and processes are mapped to the desired development strategy. Examples of development strategies include waterfall, V-model, and agile developments.

The IEEE 2851 landscape finds it useful to introduce the notion of a product dependability lifecycle (PDLC).

This standard provides structures and directions to allow a seamless exchange of information and interoperability between activities at the same or different level of abstraction.

The IEEE 2851 standardization initiative was launched with the goal of defining a dependability lifecycle as well as methods and formats for exchange/interoperability of data across all steps of the lifecycle in a consistent way. Additionally, the work products of this standardization initiative can be expected to help enable interoperability between tools. The goal of this work was to provide structures and directions to allow a seamless exchange of information and interoperability between activities at the same or different level of abstraction, as well as activities across lifecycles related to different dependability attributes across application domains such as automotive, industrial, medical, and avionics safety critical systems.

The purpose of IEEE Std 2851 is to provide a framework and infrastructure for execution and transfer of information between the lifecycle activities identified by existing standards such as ISO 26262 and IEC 61508.

To achieve that goal, this standard has the following objectives:

- OBJ1: Define a theoretical representation of a generic dependability lifecycle and supporting infrastructure required for implementing the dependability lifecycle

NOTE—The supporting infrastructure is the dependability lifecycle framework and the activities within each phase.

- OBJ2: Identify description languages, methodologies, and databases that facilitate the exchange of information in between the lifecycle activities identified by existing standards such as ISO 26262 and IEC 61508:
 - 1) OBJ2.1: Between tiers working on same lifecycle activities of the same standard (e.g., HSI in ISO 26262).
 - 2) OBJ2.2: Between tiers working on different lifecycle activities (or different levels of abstraction) of the same standard.

- 3) OBJ2.3: Between tiers working on same lifecycle activities of different standards (e.g., safety analysis in ISO 26262 vs. IEC 61508).
- 4) OBJ2.4: Between tiers working on similar activities but for different dependability properties (e.g., TARA vs. HARA).

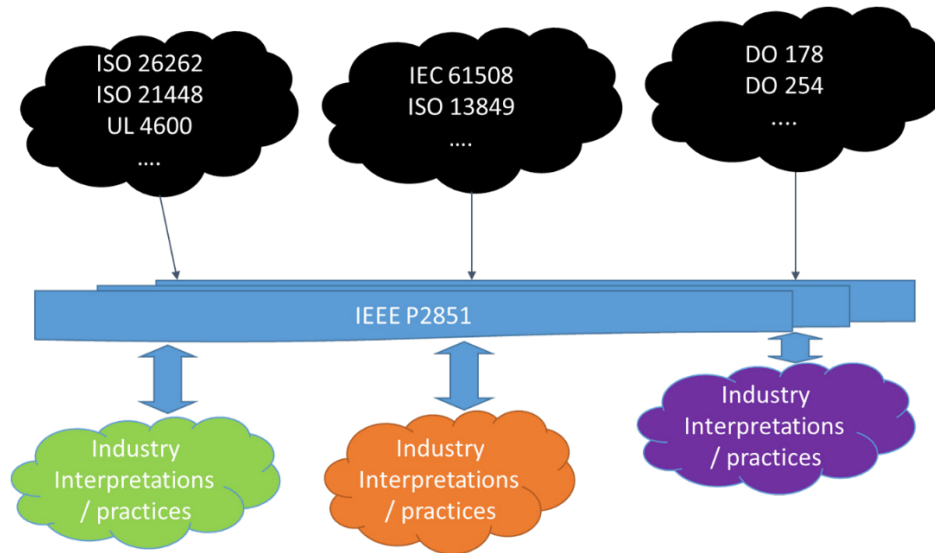


Figure 4—Purpose of IEEE Std 2851

The clauses in this standard represent the various phases of the dependability lifecycle during product development as follows:

- a) Dependability management
- b) Product definition
- c) Generic requirements elicitation process of dependability lifecycle
- d) Generic design of dependable architecture
- e) SW, HW co-design and the interface to non-E/E technology
- f) Implementation aspects
- g) Verification, integration, and analysis
- h) Dependability validation
- i) Post-release activities
- j) Dependability evaluation

Each phase lists specific requirements related to methodology (ME), description language (DL), and databases (DBs) for ensuring interoperability across levels, application domains and dependability attributes, thereby providing an overall framework of interoperable activities within the dependability lifecycle. The dependability attributes in scope of each requirement are listed within each clause. Each ME/DL/DB classified requirement is tagged with a unique identifier linking it to the detailed descriptions in C.3.

Informative descriptions for each ME/DL/DB requirement are found in C.3 in the following format:

- Description
- Problem statement
- Proposal
- Limitations
- Levels applicable
- References

A “priority,” according to the following scale, is also assigned to each requirement:

0 = HIGH: Important to implement this requirement as essential to help ensure interoperability across levels, dependability attributes and applications domains.

1 = MEDIUM-HIGH: Significant gaps that may be addressed for interoperability across levels and dependability attributes and can be relevant across use condition domains.

2 = MEDIUM: Implementing these requirements may be beneficial, but not critical to help ensure interoperability.

3 = LOW: There may be value in pursuing this requirement, however it is not considered a significant gap.

The following dependability attributes are considered in scope: reliability, availability, maintainability, real time, SOTIF, functional safety, cybersecurity.

NOTE—Quality is a pre-requisite for any dependability attribute, including the establishment of a quality management system based on standards such as IATF 16949, ISO 9001 or equivalent.

Figure 5 presents a simplified view of the need to help ensure requirements consistency between the dependability attributes within the iteration of an activity. The iteration could require jumping back to any higher level of the flow as needed. The staggering of functional safety (FuSa), SOTIF, cybersecurity, reliability, availability, maintainability, and real time requirements in this example is to convey that first, the function may be understood to an adequate level, which then allows performance to be revealed, which in turn allows the safety requirements to be revealed via the appropriate analysis (FMEDA, FTA, etc.). However, this intent is not to suggest that this is the only possible flow, rather it is an example.

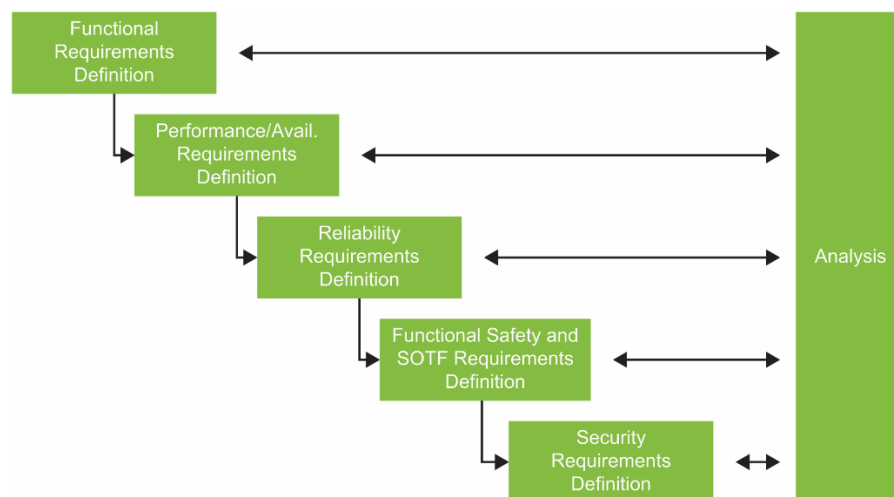


Figure 5—Need to find an appropriate trade-off between requirements

Annex A discusses an alignment between functional safety and security requirements and a possible flow is proposed.

Annex B discusses the suppliers, inputs, process, outputs, customers (SIPOC) analysis that needs to be planned and performed with respect to the involved dependability attributes (e.g., safety and security) to make the results of the analysis comparable.

Annex C contains informative content related to each dependability clause and corresponding requirements within the main body of the standard.

The subclause on generic E/E system description in C.2 formally defines the notion of the E/E system and the subclause on generic dependability life cycle definition in C.3 describes the dependability lifecycle introduced by this standard. This allows for the consideration of implications on functional safety from interaction with other dependability attributes and vice versa with a final goal of overall dependability.

Contents

1. Overview	20
1.1 Scope	20
1.2 Purpose	20
1.3 Word usage	20
2. Normative references	21
3. Definitions, acronyms, and abbreviations	21
3.1 Definitions	21
3.2 Acronyms and abbreviations	23
4. Dependability management	25
4.1 Objectives	25
4.2 Requirements and recommendations	25
5. Product definition	30
5.1 Objectives	30
5.2 General	31
5.3 Inputs	31
5.4 Requirements and recommendations	31
6. Generic requirements elicitation process of dependability lifecycle	35
6.1 Objectives	35
6.2 General	35
6.3 Inputs	35
6.4 Requirements and recommendations	35
7. Generic design of dependable architecture	41
7.1 Objectives	41
7.2 General	41
7.3 Inputs	42
7.4 Requirements and recommendations	42
8. Software, hardware co-design and the interface to non-electrical/electronic system (E/E) technology ...	54
8.1 Objectives	54
8.2 General	54
8.3 Inputs	55
8.4 Requirements and recommendations	55
9. Implementation aspects	56
9.1 Objectives	56
9.2 General	56
9.3 Inputs	57
9.4 Requirements and recommendations	57
10. Verification, integration, and analysis	59
10.1 Objectives	59
10.2 General	59
10.3 Inputs	59
10.4 Requirements and recommendations	60

11. Dependability validation of the system integrated into the overall product with regard to the feature behavior in the operational domain	74
11.1 Objectives	74
11.2 General	74
11.3 Inputs	74
11.4 Requirements and recommendations	75
12. Post-release activities	80
12.1 Objectives	80
12.2 General	80
12.3 Inputs	80
12.4 Requirements and recommendations	81
13. Dependability evaluation	82
13.1 Objectives	82
13.2 General	82
13.3 Requirements and recommendations	83
Annex A (informative) Proposed safety security alignment flow	86
A.1 Overview	86
A.2 Alignment flow at item, system and IP/SoC levels.....	86
A.3 Safety security alignment matrix	88
Annex B (informative) SIPOC analysis-based needs traceability	91
B.1 Objectives	91
B.2 General.....	91
B.3 Inputs	92
B.4 Recommendations.....	92
B.5 Work products.....	92
Annex C (informative) Dependability clauses and requirements	93
C.1 Introduction.....	93
C.2 Generic E/E system description	93
C.3 Generic dependability lifecycle definition	96
C.4 Dependability management.....	106
C.5 Product definition.....	111
C.6 Generic requirements elicitation process of dependability lifecycle.....	117
C.7 Generic design of dependable architecture	129
C.8 Software and hardware co-design and the interface to non-E/E technology.....	145
C.9 Implementation aspects.....	146
C.10 Verification, integration, and analysis	147
C.11 Dependability validation of the system integrated into the overall product with regard to the feature behavior in OD	167
C.12 Post-release activities.....	169
C.13 Dependability evaluation	171
Annex D (informative) Bibliography	180

List of Figures

Figure 1—ISO/IEC/IEEE 15288 system lifecycle processes	10
Figure 2—Example of recursive application of processes	11
Figure 3—Iterative relationship between requirements, architecture, and design processes.....	12
Figure 4—Purpose of IEEE Std 2851	13
Figure 5—Need to find an appropriate trade-off between requirements	14
Figure A.1—Safety-security alignment flow at all design levels	87
Figure B.1—Suppliers, inputs, process, outputs, customers (SIPOC) diagram.....	91
Figure C.1—E/E system functional block diagram.....	95
Figure C.2—Sense—Plan and control—Act architecture	98
Figure C.3—Automotive dependability lifecycle.....	100
Figure C.4—Design cone and the test cone forming the dual-cone-V-model (DCVM)	100
Figure C.5—Activities flow of intersection pilot	119
Figure C.6—Requirements elicitation process	120
Figure C.7—Depiction of object element.....	131
Figure C.8—Depiction of directed relation.....	132
Figure C.9—Depiction of action and activity.....	132
Figure C.10—Depiction of a sequence.....	133
Figure C.11—Two-layer V diagram	149

List of Tables

Table A.1—Safety-security alignment matrix example	88
Table A.2—Example of safety/security impacts and potential conflicts	89
Table C.1—ME and DL for dependability management	107
Table C.2—DB for dependability management	109
Table C.3—ME, DL, and DB for user story, user feature, and addressable market	113
Table C.4—ME, DL, and DB for OD, system feature behavior, system/item definition	116
Table C.5—ME and DL for requirement elicitation	123
Table C.6—DB for requirement elicitation	128
Table C.7—ME for dependable architecture	133
Table C.8—DL for dependable architecture	137
Table C.9—DB for dependable architecture	141
Table C.10—DL for SW, HW co-design	145
Table C.11—Methodology for implementation aspects	146
Table C.12—DL for implementation aspects	147
Table C.13—ME and DL for verification, integration, and analysis	153
Table C.14—DB for verification, integration, and analysis	164
Table C.15—ME for dependability validation	167
Table C.16—DL for dependability validation	168
Table C.17—ME for post-release activities	169
Table C.18—ME for dependability evaluation	176
Table C.19—DL for dependability evaluation	177

IEEE Standard for Functional Safety Data Format for Interoperability within the Dependability Lifecycle

1. Overview

1.1 Scope

This standard defines a dependability lifecycle of products with focus on interoperable activities related to functional safety and its interactions with reliability, security, operational safety, and time-determinism. The standard also describes methods, description languages, data models, and databases that have been identified as necessary or critical, to enable the exchange/interoperability of data across all steps of the lifecycle encompassing activities executed at intellectual property (IP), system-on-chip (SoC), system, and item levels in a technology independent way across application domains such as automotive, industrial, medical, and avionics safety critical systems.

1.2 Purpose

The IEEE 2851 standardization initiative was started to define a dependability lifecycle as well as methods and formats for exchange/interoperability of data across all steps of the lifecycle in a consistent way. Additionally, the work products of this standardization initiative will help enable interoperability between tools. The goal of this work is to provide structures and directions to allow a seamless exchange of information and interoperability between activities at the same or different level of abstraction, as well as activities across lifecycles related to different dependability attributes across application domains such as automotive, industrial, medical, and avionics safety critical systems.

1.3 Word usage

The word *shall* indicates mandatory requirements strictly to be followed in order to conform to the standard and from which no deviation is permitted (*shall equals is required to*).^{7, 8}

⁷ The use of the word *must* be deprecated and cannot be used when stating mandatory requirements; *must* is used only to describe unavoidable situations.

⁸ The use of *will* is deprecated and cannot be used when stating mandatory requirements; *will* is only used in statements of fact.

The word *should* indicates that among several possibilities one is recommended as particularly suitable, without mentioning or excluding others; or that a certain course of action is preferred but not necessarily required (*should* equals *is recommended that*).

The word *may* is used to indicate a course of action permissible within the limits of the standard (*may* equals *is permitted to*).

The word *can* is used for statements of possibility and capability, whether material, physical, or causal (*can* equals *is able to*).

2. Normative references

The following referenced documents are indispensable for the application of this document (i.e., they must be understood and used, so each referenced document is cited in text and its relationship to this document is explained). For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments or corrigenda) applies.

IEC 61508:2010, Functional Safety of Electrical/Electronic/Programmable Electronic Safety-Related Systems.⁹

ISO 26262:2018, Road Vehicles—Functional Safety.¹⁰

3. Definitions, acronyms, and abbreviations

3.1 Definitions

For the purposes of this document, the following terms and definitions apply. The *IEEE Standards Dictionary Online* should be consulted for terms not defined in this clause.¹¹

activity: Sequence of engineering actions performed in a phase or sub-phase of the product dependability lifecycle.

availability: The probability that a product will operate acceptably at a given point in time when used under specified conditions. *See also:* **dependability attributes**.

cybersecurity (CS): The avoidance of hazards due to deliberate attacks. *See also:* **dependability attributes**.

database (DB): A collection or listing of parameters, characteristics, properties and/or data sets.

data model: Abstract model that organizes the description, semantics, and consistency constraints of data resulting from an activity.

⁹ IEC publications are available from the International Electrotechnical Commission (<http://www.iec.ch>) and the American National Standards Institute (<http://www.ansi.org/>).

¹⁰ ISO publications are available from the International Organization for Standardization (<http://www.iso.org/>) and the American National Standards Institute (<http://www.ansi.org/>).

¹¹ *IEEE Standards Dictionary Online* is available at: <http://dictionary.ieee.org>. An IEEE account is required for access to the dictionary, and one can be created at no charge on the dictionary sign-in page.

dependability: Capability of an electrical or electronic system and its function to show one or more of the following characteristics (attributes): reliability, availability, maintainability, real time, SOTIF, functional safety, cybersecurity.

NOTE 1—Dependability may be tailored to the needs of the product.

NOTE 2—The needs presented in this standard address interdependence of functional safety and with other attributes of dependability, i.e., it does not cover interdependences between attributes like reliability and security.

dependability attributes: Characteristics of a dependable system that include reliability, availability, maintainability, real time, SOTIF, functional safety, and cybersecurity.

NOTE—The characteristics include reliability, availability, maintainability, real time, SOTIF, functional safety, cybersecurity.¹²

dependability audit: Examination of the project specific process tailoring of the dependability lifecycle and the conformance of the products to the tailored process and project objectives.

dependability confirmation review: Confirmation whether the work products are developed according to the engineering standards of safety, while considering the implications of other dependability attributes.

dependability manager: The owner of the dependability case who manages and provides oversight to the development of the work products.

NOTE—The dependability manager is responsible for the setup of the **dependability plan**.

dependability plan (DP): A plan to manage and guide the execution of functional safety activities while considering the impact of the dependability attributes.

description language (DL): Language or document template used to describe the framework, syntax, and semantics to fully express and relate the content.

discipline/engineering discipline: Engineering skillset and technical responsibility for one or more dependability attributes to help ensure the product function and performance as well as the product dependability attributes.

domain: A specific industry that uses or develops dependable products.

domain agnostic: A technology or approach that is not specific to any domain or industry.

functional safety (FuSa): The avoidance of hazards due to malfunctions. *See also:* **dependability attributes**.

interoperability and exchangeability: The capability to easily share technical information between domains and participants in the supply chain.

maintainability (M): The probability that a failed product will be restored to a specified condition within a specified period. *See also:* **dependability attributes**.

methodology (ME): Best known methods, guidelines, and principles that describe the detailed steps needed to implement an activity.

¹² Notes in text, tables, and figures are given for information only and do not contain requirements needed to implement the standard.

operation conditions: The set of external factors like temperature, voltage and vibration that affect the different dependability attributes.

operational domain (OD): The functions needed to operate dependably. Operational domain consists of the following aspects:

- Operator
- Scenario
- System
- Environment
- Infrastructure

operating modes: Conditions of functional state that arise from the use and application of product.

operating situations: All relevant conditions and options that determine an appropriate behavior of the system under consideration at a particular point in time.

phase: Dedicated stage in the product dependability lifecycle.

process: Set of interrelated or interacting activities that transform inputs into outputs.

product: A product can be an item, system, sub-system, HW, or SW component, HW part, or SW unit.

product dependability lifecycle: The entirety of phases covering activities for ensuring dependability of the product function.

real time (RT): Ensuring deterministic performance, i.e., ensuring that timing can be guaranteed within a certain margin of error. *See also:* **dependability attributes**.

reliability (R): The probability that a product, system, or service will operate in a defined environment without failure. *See also:* **dependability attributes**.

safety of the intended functionality (SOTIF): The nominal performance necessary to avoid hazards. *See also:* **dependability attributes**.

stage: A period during the lifecycle of a product that relates to the state of its description or realization.

user story: A documentation of needs, expectations, and desired outcomes that serves as an input for safety teams designing and implementing features in an overall product.

3.2 Acronyms and abbreviations

A	availability
AoU	assumption of use
ABS	antilock braking system
ADAS	advanced driver assistance system
AI	artificial intelligence
APQP	advanced product quality planning
ASIL	automotive safety integrity level
ASR	anti-slip regulation
BFR	base failure rate
BKM	best known method

CAD	computer aided design
CAE	computer aided engineering
CMA	common mode analysis
COTS	commercial off-the-shelf
CS	cybersecurity
DB	database
DC	diagnostic coverage
DFA	dependent failure analysis
DFI	dependent failure initiator
DFMEA	design FMEA
DFT	discrete Fourier transform
DIA	development interface agreement
DL	description language
DNN	deep neural network
DP	dependability plan
E/E	electrical/electronic system
EDA	electronic design automation
FFI	freedom from interference
FIA	functional interface analysis
FIT	failure in time
FMEA	failure modes and effects analysis
FMEDA	failure modes effects and diagnostics analysis
FSC	functional safety concept
FW	firmware
FuSa	functional safety
HARA	hazard analysis and risk assessment
HW	hardware
IP	intellectual property
ISP	image signal processor
LiDAR	light detection and ranging
M	maintainability
ME	method, methodology
NoC	network-on-chip
OD	operational domain
PCB	printed circuit board
PDL	product dependability lifecycle
QM	quality management
R	reliability
RAS	reliability-availability-and-serviceability
RT	real time
SAD	software architectural design
SCDL	safety concept description language
SEE	single event effects
SEU	single event upset
SIL	safety integrity level
SIPOC	suppliers inputs process outputs customers
SoC	system-on chip
SOTIF	safety of the intended functionality
SW	software
SWaP-C	size, weight, power, and cost
TARA	threat analysis and risk assessment
TCA	triggering condition analysis
TCL	tool confidence level
TSC	technical security concept
UAM	urban air mobility
UAV	unmanned aerial vehicle

VEIA	verification, integration, and analysis
WLAN	wireless local area network
WCET	worst case execution time

4. Dependability management

4.1 Objectives

The following objectives shall be achieved:

- a) Implement rules and processes to help achieve functional safety goals while considering the impact of each product dependability attributes on functional safety.
- b) Institute and maintain processes to help ensure an adequate resolution of identified conflicts (e.g., a cybersecurity requirement impacting the functional safety requirement).
- c) Plan and execute activities at various phases of the dependability lifecycle.
- d) Plan for confirmation activities, e.g., audits and assessments.
- e) Institute and maintain adequate rules and processes to achieve goals for each product dependability attribute.

4.2 Requirements and recommendations

4.2.1 Dependability management process

The organization involved in the execution of the dependability lifecycle shall implement a dependability management process.

NOTE—The dependability management process could be derived from the functional safety management processes described in ISO 26262-2 or IEC 61508-1, enlarging the scope of respective requirements to cover all dependability attributes, including functional safety.

4.2.2 Product impact analysis—ME, DB

Dependability attributes covered are as follows:

- Functional safety
- Safety of the intended functionality (SOTIF)
- Cybersecurity
- Reliability
- Availability
- Maintainability
- Real time

4.2.2.1 Product impact analysis—ME

A methodology shall be defined for an impact analysis at the product level to determine whether the product is a new development, a modification of an existing product or an existing product with a modified operational domain (OD).

NOTE—An informative reference guidance for a methodology [DPMG-ME-1] for evaluating the implications of the modifications with reference to various dependent attributes associated with product development and identifying and describing the activities to be performed based on the impact of the modifications is included in Table C.1.

4.2.2.2 Product impact analysis—DB

In the case of a modification of a product or its OD, a description of the modifications applied to the element shall be documented.

NOTE—An informative description of all modifications applied to the element, including modifications to the design, implementation, and environment database [DPMG-DB-1] is included in Table C.2.

4.2.3 Development interface agreement (DIA)—DL

Dependability attributes covered are as follows:

- Functional safety
- SOTIF
- Cybersecurity
- Reliability
- Availability
- Maintainability
- Real time

A description language shall be defined that documents interactions and dependencies between customers and suppliers, that defines the allocation of responsibilities of customer and supplier and identifies work products to be exchanged between customer and supplier.

NOTE—An informative description of a description language [DPMG-DL-1] for objectively capturing the activities and expected dates for those activities to be performed by the vendor and integrator, such that the required activities for ensuring the product dependability are performed, is included in Table C.1.

4.2.4 Software tools—ME, DB

Dependability attributes covered are as follows:

- Functional safety
- SOTIF
- Cybersecurity
- Reliability
- Availability

- Maintainability
- Real time

4.2.4.1 Software tools—DB

Software tools used for product development, including the tools used for ensuring the various dependability attributes, should be documented.

NOTE—A proposal for a database (DB) containing an informative description of software tools used for product development, including the tools used for ensuring the various dependability attributes [DPMG-DB-3] is included in Table C.2.

4.2.4.2 Software tools—ME

A methodology shall be defined for qualification of the tools used for product development to improve dependability of the software tool output.

NOTE—An informative reference guidance for a methodology [DPMG-ME-3] for qualification of the tools used for product development to improve dependability of the software tool output, is described in Table C.1.

4.2.5 Monitoring of parameters for prognostics—DB

Dependability attributes covered are as follows:

- Functional safety
- SOTIF
- Reliability
- Availability
- Maintainability

Reliability parameters that need to be monitored in the field for prognostics shall be documented.

NOTE—A proposal for an informative database of parameters for prognostics [DPMG-DB-4] is included in Table C.9.

4.2.6 Tailoring of dependability activities—ME, DL, DB

Dependability attributes covered are as follows:

- Functional safety
- SOTIF
- Cybersecurity
- Reliability
- Availability
- Maintainability
- Real time

4.2.6.1 Tailoring of dependability activities—ME

A methodology shall be defined for project specific tailoring of dependability activities.

NOTE—An informative reference guidance for a methodology [DPMG-ME-4] for project specific tailoring of dependability activities is described in Table C.1.

4.2.6.2 Tailoring of dependability activities—DL

A description language shall be defined for project specific tailoring of dependability activities.

NOTE—An informative summary of a description language [DPMG-DL-2] for project specific tailoring of dependability activities is included in Table C.1.

4.2.6.3 Tailoring of dependability activities—DB

The objective criteria that was used for tailoring of dependability activities shall be documented.

NOTE—An informative description for a database [DPMG-DB-6] containing the criteria used for tailoring of dependability activities data is included in Table C.2.

4.2.7 Dependability case—DB

Dependability attributes covered are as follows:

- Functional safety
- SOTIF
- Cybersecurity
- Reliability
- Availability
- Maintainability
- Real time

Work products developed in accordance with the dependability plan (DP) that provide the argument for the achievement of dependability shall be documented.

NOTE—An informative description for a database [DPMG-DB-7] containing work products developed in accordance with the DP that provide the argument for the achievement of dependability is included in Table C.2.

4.2.8 Confirmation reviews—ME

Dependability attributes covered are as follows:

- Functional safety
- SOTIF
- Cybersecurity
- Reliability

- Availability
- Maintainability
- Real time

A methodology shall be developed for reviewing work products and other evidence to be used as confirmation that dependability was achieved.

NOTE—Table C.1 includes an informative reference guidance for a methodology [DPMG-ME-5] for performing confirmation reviews that provides an independent judgement of whether the corresponding objectives of IEEE Std 2851 are achieved.

4.2.9 Dependability audit—ME

Dependability attributes covered are as follows:

- Functional safety
- SOTIF
- Cybersecurity
- Reliability
- Availability
- Maintainability
- Real time

A methodology shall be defined to perform a dependability audit.

NOTE—An informative reference guidance for a methodology [DPMG-ME-6] to perform a dependability audit to judge whether the process related objectives of IEEE Std 2851 are achieved is included in Table C.1.

4.2.10 Dependability assessment—ME

Dependability attributes covered are as follows:

- Functional safety
- SOTIF
- Cybersecurity
- Reliability
- Availability
- Maintainability
- Real time

A methodology shall be defined to judge the achieved dependability considering the impact of other dependability attributes on functional safety.

NOTE—An informative reference guidance for a methodology [DPMG-ME-7] to judge the achieved dependability considering the impact of other dependability attributes on functional safety is included in Table C.1.

4.2.11 Release for production—DB

Dependability attributes covered are as follows:

- Functional safety
- SOTIF
- Cybersecurity
- Reliability
- Availability
- Maintainability
- Real time

The dependability case, confirmation measure reports, and dependability assessment report shall be available prior to the release for production.

NOTE—An informative description containing the dependability case in accordance with 4.2.7, confirmation measure reports and dependability assessment report, if applicable, in accordance with database [DPMG-DB-8] is included in Table C.2.

5. Product definition

5.1 Objectives

The objective of this clause is to enable proper capture of product definition across all domains while considering the various aspects of dependability attributes.

The product definition addresses all aspects of the product relevant for guaranteeing dependable functional behavior of the product operational domain including humans, standards, laws, regulations, stakeholders, use/user story, and features.

The product definition consists of defining main functions and constraint functions by looking at the product as a black box. The product definition is not intended to discuss the implementation or the “how-to” descriptions.

The main function is to define the services that the product is providing, in other words, the need for the product creation.

Product definitions need to define specifically which level of the hierarchy the product is targeting.

The constraint function defines the limits and conditions in which the product can be subjected to operate. For example, it addresses regulations, standardization, energy, environment, and many other factors that can be defined in a mission profile (e.g., temperature, mechanical stress, EMI).

Certain constraint function database aspects need to be considered and defined. To make sure that the SOTIF is covered, a database of use-cases can be considered.

5.2 General

A few ways in which the previous functions can be represented are as follows:

- Functional analysis using a list of features (product seen as black box)
- Statistics
- Databases of misuse scenarios and customer use cases
- Operation domain
- Size, weight, power, and cost (SWaP-C) optimization

5.3 Inputs

Primary inputs required for product definition are market needs, innovations, and user concept.

5.4 Requirements and recommendations

5.4.1 User story, user feature, and addressable market

The definitions are as follows:

- a) *User story*: Description of feature (system capability) written from the point of view of a user or customer.
- b) *User feature*: Description of a feature (system capability) written from the point of view of the product.
- c) *Addressable market*: Reference to the target market that defines boundaries for user features and environment conditions.

NOTE—User story, user feature(s), and addressable market(s) can be specific to a particular product or common between products and include specific scenarios and behaviors.

5.4.2 System/life profile—DL, DB

Dependability attributes covered are as follows:

- Functional safety
- SOTIF
- Reliability
- Availability
- Real time
- Maintainability

5.4.2.1 System/life profile—DL

System or life profile shall list the operating conditions and measurable stress seen by the product during its lifetime.

NOTE—An informative summary of a proposal for a description language [PRDF-DL-1] needed for listing the product operating conditions, lifetime, measurable stress seen by the product during its lifetime is described in Table C.3.

5.4.2.2 System/life profile—DB

The thermal, electrical, operating life, operating conditions, and chemical profiles for a safety analysis should be documented.

NOTE—An informative description of a proposal for a database containing the thermal, electrical, lifetime, operating conditions, and chemical profiles for a safety analysis [PRDF-DB-1] is described in Table C.3.

5.4.3 Hardware/software/system triggering conditions—DB

Dependability attributes covered are as follows:

- Functional safety
- SOTIF

A database of HW/SW/system triggering conditions that could lead to hazardous behavior for a safety system shall be documented.

NOTE—An informative proposal for a database containing HW/SW/system triggering conditions for a safety system [PRDF-DB-4] is described in Table C.3.

5.4.4 Operating situations and operating modes—DL, ME

Dependability attributes covered are as follows:

- Functional safety
- SOTIF
- Cybersecurity
- Reliability
- Availability
- Maintainability
- Real time

5.4.4.1 Operating situations and operating modes—DL

A description language shall be defined to describe operating situations and operating modes.

NOTE—An informative summary of a proposal for a description language [PRDF-DL-2] needed for operating situations and operating modes is described in Table C.3.

5.4.4.2 Operating situations and operating modes—ME

A methodology shall be defined for operating situations and operating modes.

NOTE—Informative reference guidance for a proposed methodology [PRDF-ME-2] to cover operating situations and operating modes is described in Table C.3.

5.4.5 Item interdependency—DL

Dependability attributes covered are as follows:

- Functional safety
- SOTIF
- Cybersecurity
- Reliability
- Availability
- Real time

A description language shall be defined to describe the functional description, interdependencies of functions, and external mechanisms of the item.

NOTE—An informative summary of a description language proposal [PRDF-DL-3] needed to describe the functional description, interdependencies of functions, and external mechanisms of the item is described in Table C.3. The intent is to provide a description language or guidelines to describe item information.

5.4.6 External measures for safety and security—DL, DB

Dependability attributes covered are as follows:

- Functional safety
- SOTIF
- Cybersecurity
- Availability

NOTE—External measures are located outside of the system or at a higher level than the current level of the safety analysis.

5.4.6.1 External measures for safety and security—DL

A description language shall be defined to describe representative external measures across systems.

NOTE—An informative summary of a proposed description language [PRDF-DL-5] to describe common external measures for a safety and security analysis is described in Table C.3.

5.4.6.2 External measures for safety and security—DB

Commonly used external measures across safety critical systems shall be documented.

NOTE—An informative description of common external measures for a safety analysis database proposal [PRDF-DB-8] is described in Table C.4.

5.4.7 System level considerations for enabling deterministic performance—DB

Dependability attributes covered are as follows:

- Functional safety
- SOTIF
- Reliability
- Real time

Performance profile shall list the operating conditions that are seen by the product during its lifetime.

NOTE—An informative description containing thermal, electrical, operating life, operating conditions, and chemical profiles for a safety analysis proposed database [PRDF-DB-6] is described in Table C.3.

5.4.8 Operating situations, operating modes, performance metrics, and dynamic conditions—DB

Dependability attributes covered are as follows:

- Functional safety
- SOTIF
- Cybersecurity
- Reliability
- Availability
- Maintainability
- Real time

A database shall be created containing the operating situations, operating modes, performance metrics, and dynamic conditions consistent with product definition.

NOTE—An informative description to cover operating situations, operating modes, performance metrics, and dynamic conditions [PRDF-DB-7] is included in Table C.4.

5.4.9 Environment use—DB

Dependability attributes covered are as follows:

- Functional safety
- Reliability
- Availability
- Maintainability

A database shall be created containing the environmental use details of the product.

NOTE—An informative description of environmental use details such as location of the product, temperature range, weather, and other relevant information database proposal [PRDF-DB-9] is described in Table C.4.

6. Generic requirements elicitation process of dependability lifecycle

6.1 Objectives

The objective of this clause is to enable a proper requirements definition process for a product in a manner that considers the different aspects of dependability and helps ensure the resulting requirements are balanced and consistent with each other.

6.2 General

Requirement specification is the best way of exchanging information on the product envisaged between the parties involved in a product development.

For every functional and technical requirement, at least one dependability requirement can exist that states whether the functional and technical requirement may be implemented in a dependable way.

6.3 Inputs

Inputs for a product within the supply chain and a standalone product are as follows:

- a) For a product within the supply chain:
 - Set of dependable requirements from engineering level above
 - Dependable architecture from engineering level above
- b) For a standalone product:
 - Product definition: use cases, use features
 - Dependability analyses [e.g., Safety—hazard analysis and risk assessment (HARA), Security—threat analysis and risk assessment (TARA)] (refer to Clause 10 for details)

6.4 Requirements and recommendations

6.4.1 Safety requirements—ME, DL

Dependability attributes covered are as follows:

- Functional safety
- SOTIF

6.4.1.1 Safety requirements—ME

A methodology shall be defined for good requirement practice.

NOTE—An informative reference guidance for a proposed methodology [REQE-ME-1] for good requirement definition practice, in alignment with ISO 26262 and other relevant standards related to the use of natural, semi-formal and formal languages is described in Table C.5.

6.4.1.2 Safety requirements—DL

A description language shall be defined to describe specification insufficiencies as well as performance limitations related to SOTIF.

NOTE—An informative summary of a description language [REQE-DL-1] is defined to allow for the description of specification insufficiencies as well as performance limitations related to SOTIF in described in Table C.5.

6.4.2 Concept—DL

Dependability attributes covered are as follows:

- Functional safety
- SOTIF

A description language shall be defined for describing the safety element being developed including environmental description, assumptions of use, high-level functional description, list of potential sources of hazard, and effects description. The list of applicable statutory or regulatory requirements should also be included.

NOTE—An informative summary of a description language proposal [REQE-DL-2] for describing the safety element being developed including environmental description, assumptions of use, high-level functional description, list of potential sources of hazard, and effects description (including interaction with other elements) is described in Table C.5.

6.4.3 Risk level/ automotive safety integrity level (ASIL) alignment matrix—ME

Dependability attributes covered are as follows:

- Functional safety
- SOTIF
- Cybersecurity

A methodology shall be defined for the creation of an alignment matrix between functional safety and cybersecurity.

NOTE—An informative description of a proposed methodology [REQE-ME-2] for the creation of an alignment matrix between functional safety and cybersecurity to help ensure that safety considerations are given to the security implementation is described in Table C.5.

6.4.4 Safety goal—DL

Dependability attributes covered are as follows:

- Functional safety
- SOTIF

A description language shall be defined for describing safety goals in a manner common to functional safety and SOTIF.

NOTE 1—It may be possible to derive a high-level common safety goal, with sub-goals incorporating functional safety specific information, such as failure rate.

NOTE 2—An informative summary of a description language proposal [REQE-DL-3] for describing safety goals in a manner common to functional safety and SOTIF is described in Table C.5.

6.4.5 Hazards and risks—ME, DB

Dependability attributes covered are as follows:

- Functional safety
- SOTIF
- Cybersecurity
- Reliability
- Availability

6.4.5.1 Hazards and risks—ME

A methodology shall be defined to identify and classify hazards and risks, across all operational domains, operational situations, and operational conditions, for consideration in the safety assessment and formulation of safety goals.

NOTE—An informative description of a methodology proposal [REQE-ME-3] to identify and classify the hazardous events and formulate safety goals is included in Table C.13, considering ISO 26262 and SAE J2980 [B21] for ASIL hazard classification, as well as other relevant methods including the application of STPA (systems theoretic process analysis) to help ensure no duplication.

6.4.5.2 Hazards and risks—DB

All hazards and risks, across all operations domains, operational situations, and operational conditions, which were considered in the safety assessment shall be documented.

NOTE—A proposed database of all hazards and risks that can be considered in the safety assessment, across domains [REQE-DB-1] is described in Table C.14.

6.4.6 Impact assessment vs. requirements in case of reuse of element(s)—ME, DL

Dependability attributes covered are as follows:

- Functional safety

6.4.6.1 Impact assessment vs. requirements in case of reuse of element(s)—ME

A methodology shall be defined for describing the required activities of derivative elements.

NOTE—An informative description of a proposed methodology [REQE-ME-4] to keep track of the reuse activity and provide example use cases is described in Table C.7.

6.4.6.2 Impact assessment vs. requirements in case of reuse of element(s)—DL

A description language shall be defined for the activities tailoring due to reuse.

NOTE—An informative summary of a proposed description language [REQE-DL-4] so that the activities tailoring due to reuse can be standardized, focusing on the changes needed for an already compliant element, is described in Table C.8.

6.4.7 Threat and risk—DL

Dependability attributes covered are as follows:

- Functional safety
- Cybersecurity

A description language shall be defined for describing cybersecurity threats and risks that impact functional safety.

NOTE—An informative summary of a proposed description language [REQE-DL-5] for describing cybersecurity threats and risks that impact safety is described in Table C.5.

6.4.8 Platform tuning for safety critical real time applications—ME

Dependability attributes covered are as follows:

- Functional safety
- Real time

A methodology shall be defined to support the approaches and mitigations to help ensure that safety critical applications are able to meet worst case execution time (WCET) targets.

NOTE—An informative description of a proposed methodology [REQE-ME-5] to support the approaches and mitigations to help ensure that safety critical applications can meet WCET targets is described in Table C.5.

6.4.9 Real time considerations for safety analysis—ME

Dependability attributes covered are as follows:

- Functional safety
- Real time

A methodology shall be defined to provide guidance for coherent, synergetic management of the interactions between real time and functional safety.

NOTE—An informative description of a proposed methodology [REQE-ME-6] to provide guidance for coherent, synergetic management of the interactions between real time and safety is described in Table C.5.

6.4.10 Modeling of the radiation working environment—ME

Dependability attributes covered are as follows:

- Functional safety
- Reliability

A methodology shall be defined to describe the radiation working environment for soft error rate modeling.

NOTE—An informative description of a proposed methodology [REQE-ME-7] for the radiation working environment (in terms of energetic particles) for soft error rate modeling is described in Table C.5.

6.4.11 HW and SW requirements traceability—DL

Dependability attributes covered are as follows:

- Functional safety
- SOTIF
- Cybersecurity
- Reliability

A description language shall be defined to enable requirements traceability harmoniously and synergistically in each design stage.

NOTE—An informative summary of a proposed description language [REQE-DL-6] to enable requirements traceability harmoniously and synergistically, and to be able to address them with a coherent methodology in each layer or design stage is described in Table C.5.

6.4.12 Ensuring deterministic performance for safety applications—ME

Dependability attributes covered are as follows:

- Functional safety
- Real time
- SOTIF
- Reliability

A methodology shall be defined for enabling a system-level implementation to help ensure deterministic performance of critical safety applications.

NOTE—An informative description of a proposed methodology [REQE-ME-9] to express requirements in a standard format and to produce, use, and present data and metrics, helping enable a system-level implementation to ensure a deterministic performance of critical safety applications, is described in Table C.5.

6.4.13 Common mode failures—DB

Dependability attributes covered are as follows:

- Functional safety
- Reliability

Schemes, techniques, and features to help mitigate the effects of common faults, errors, and failures, including the capabilities, advantages or disadvantages of each scheme shall be documented.

NOTE—An informative description of a proposal for schemes, techniques, and features to help mitigate the effects of common faults, errors, and failures database [REQE-DB-2] is described in Table C.6.

6.4.14 Radiation testing requirements for safety analysis—DB

Dependability attributes covered are as follows:

- Functional safety
- Reliability

Radiation testing requirements for safety analysis shall be documented.

NOTE—An informative description of a proposal for a database of links, citations or applicable documents, standards, or best known methods (BKMs) for radiation testing requirements for safety analysis [REQE-DB-3] is described in Table C.6.

6.4.15 Parameters for real time and safety metrics—DB

Dependability attributes covered are as follows:

- Functional safety
- Real time

All parameters that are needed to reduce jitter/interference to help ensure freedom from interference (FFI), and be compliant with WCET targets shall be documented.

NOTE—An informative description of a proposal for a database containing parameters needed to reduce contention, help ensure FFI, and be compliant with WCET targets in a mixed criticality system [REQE-DB-4] is described in Table C.6.

6.4.16 Contention/Shared resources of functional safety real time intersections—DB

Dependability attributes covered are as follows:

- Functional safety
- Real time

Sources of contention/shared resources that need to be considered for functional safety and real time intersections shall be documented.

NOTE—An informative description of contention/shared resources database [REQE-DB-5] that need to be considered for mitigations to help ensure safety metrics and real time compliance is achieved, is described in Table C.6.

6.4.17 System level considerations for enabling deterministic performance—DB

Dependability attributes covered are as follows:

- Functional safety
- Real time
- SOTIF
- Reliability

Refer to 5.4.7.

7. Generic design of dependable architecture

7.1 Objectives

The objectives of this clause are as follows:

- a) To provide technical architectures, activity diagrams, and sequence diagrams of the product, while considering all dependability attributes.
- b) To provide evidence that the specific concepts coming from all dependability attributes can function as specified and do not interfere with each other and lead to new and not considered risks.
- c) To provide evidence that the functional architectures and the technical architectures are appropriate for achieving the dependability attributes for the system integrated into the overall product.

7.2 General

Modeling starts with setting up a use or user story, which explains what the product is needed for in the context of its application. Use or user stories are normally composed in natural language and are descriptive in an epic style.

As use or user stories try to cover the full range of application, they are typically very extensive. It is recommended to break them down into use cases, which—as a whole set—allows the composition of any story episode as a repetitive concatenation of use cases.

This standard recommends that the architecture description comprises a static view and a dynamic view. The standard recommends the use of functional block diagrams, activity diagrams, and sequence diagrams for the architectural description.

7.3 Inputs

7.3.1 Prerequisites

Prerequisites are as follows:

- User stories and use cases as already composed.
- Product requirements that need to be achieved for functional safety in addition to requirements on other dependability attributes in order to achieve an arbitrated and balanced dependably product.

7.4 Requirements and recommendations

7.4.1 Failure mode—DL

Dependability attributes covered are as follows:

- Functional safety
- Reliability

A description language for failure modes shall be provided that contains the most relevant characteristics associated with the failure mode.

NOTE 1—The main goal is to define a set of critical characteristics that represent the failure mode and allow taking the failure mode into account in the next higher-level safety analysis is described in Table C.7.

NOTE 2—An informative summary of a proposed description language [ARCD-DL-1] to enable requirements traceability harmoniously and synergistically and to be able to address them with a coherent methodology in each layer or design stage is included in Table C.8.

7.4.2 Functional safety HW design—DL

Dependability attributes covered are as follows:

- Functional safety

A description language of HW design shall be defined to standardize the description of the design with all the critical parameters and characteristics related to functional safety management. This should include characteristics such as the expected type of monitoring, expected safe state, expected fault handling time interval, and safety indicators.

NOTE—An informative summary of a proposed description language [ARCD-DL-2] to enable requirements traceability harmoniously and synergistically and to standardize the description of the design with all the critical parameters and characteristics related to functional safety management is included in Table C.8.

7.4.3 Cybersecurity software testing—ME

Dependability attributes covered are as follows:

- Functional safety
- Cybersecurity

A methodology shall be defined for cybersecurity SW testing as part of the functional safety SW test flow to assess if a threat has an impact on safety.

NOTE—An informative summary of a proposed methodology [ARCD-ME-1] for consideration in the functional safety SW test flow to assess if the threat has impact on safety is included in Table C.7.

7.4.4 Software architectural design (SAD)—ME, DL

Dependability attributes covered are as follows:

- Functional safety
- SOTIF
- Cybersecurity

The software architectural design (SAD) represents the architecture of the software, its elements, and their interactions. The SAD also allocates the input requirements to the different elements composing the software.

7.4.4.1 Software architectural design (SAD)—ME

A methodology shall be defined for the SW architectural design, with considerations for functional safety, SOTIF, and cybersecurity.

NOTE—An informative summary of a proposed methodology [ARCD-ME-2] is included in Table C.7 to address synergies and trade-offs to be considered for SW architectural design.

7.4.4.2 Software architectural design (SAD)—DL

A description language shall be defined for the SW architectural design, with considerations for functional safety, SOTIF, and cybersecurity.

NOTE—An informative summary of a proposed description language [ARCD-DL-3] for the SW architectural design is included in Table C.8, including functional safety, cybersecurity, and SOTIF considerations.

7.4.5 Artificial intelligence (AI) software configuration and calibration—ME, DL

Dependability attributes covered are as follows:

- Functional safety
- SOTIF

NOTE 1—AI-based elements need to be trained with massive data to achieve their intended functionality such as audio recognition, object detection, and object classification.

NOTE 2—Training of AI elements requires complex systems for data management.

7.4.5.1 Artificial intelligence software configuration and calibration—ME

A methodology shall be defined to capture requirements and considerations necessary to build a sufficient training data set. AI training should be considered as a special type of software configuration and calibration.

NOTE—An informative summary of a proposed methodology [ARCD-ME-3] to include requirements and considerations for how to build a sufficient training data set is described in Table C.7.

7.4.5.2 Artificial intelligence software configuration and calibration—DL

Dependability attributes covered are as follows:

- Functional safety
- SOTIF

To address challenges associated with proper implementation of data harvest, management, and data quality control, there shall be a description language for configuration and calibration data.

NOTE—An informative summary of a proposed description language [ARCD-DL-5] needed for configuration and calibration data is described in Table C.7.

7.4.6 Safety mechanism and safety envelope—DL

Dependability attributes covered are as follows:

- Functional safety
- SOTIF
- Cybersecurity
- Real time
- Reliability

A description language shall be defined for describing safety mechanism characteristics.

NOTE—An informative summary of a proposed description language [ARCD-DL-4] is described in Table C.8. Safety mechanisms benefit from being described in a formal way with various characteristics: detection time, reaction time, the way an issue is flagged, the entered safe state, diagnostic coverage, and whether it can be used to detect SPF or LF.

7.4.7 Automotive safety integrity level (ASIL) decomposition—ME

Dependability attributes covered are as follows:

- Functional safety

A methodology shall be defined for ASIL decomposition considering minimum requirements for the decomposed elements to comply with dependability.

NOTE—An informative summary of a proposed methodology [ARCD-ME-5] for ASIL decomposition considering minimum requirements for the decomposed elements to comply with dependability attributes is described in Table C.7.

7.4.8 System architectural design—DL

Dependability attributes covered are as follows:

- Functional safety
- SOTIF
- Cybersecurity
- Reliability
- Availability
- Maintainability
- Real time

A description language shall be defined for system architectural design to be applicable across levels.

NOTE 1—Description of system architecture includes system composition and system behavior.

NOTE 2—An informative summary of a proposed description language [ARCD-DL-6] for system architectural design, which can be used on multiple levels, is described in Table C.8.

7.4.9 SW design/algorithm impact analysis—ME

Dependability attributes covered are as follows:

- Functional safety

A methodology shall be defined for gap analysis automation of SW design/algorithm impact analysis.

NOTE 1—Impact analysis allows tailoring of the engineering lifecycle based on the idea of reuse of the work products that retain their actuality.

NOTE 2—An informative summary of a proposed methodology [ARCD-ME-6] for impact analysis to allow automation of the gap analysis is described in Table C.7.

7.4.10 Confidence in use of software tools evaluation—ME, DL

Dependability attributes covered are as follows:

- Functional safety

7.4.10.1 Confidence in use of software tools evaluation—ME

A methodology shall be defined for expressing confidence in the use of SW tools used in the development of a system or its software or hardware elements.

NOTE—An informative summary of a proposed methodology [ARCD-ME-7] for evaluating confidence in the use of SW tools from electronic design automation (EDA) vendors or internal tools and testing is described in Table C.7.

7.4.10.2 Confidence in use of software tools evaluation—DL

A description language shall be defined for expressing confidence in the use of SW tools used in development of a system or its software or hardware elements.

NOTE—An informative summary of a proposed description language [ARCD-DL-7] for parameters in a tool confidence level (TCL) description data element for each SW tool in use and the parameters for describing the TCL of a tool chain is described in Table C.7.

7.4.11 Criteria for coexistence of elements—ME

Dependability attributes covered are as follows:

- Functional safety

A methodology shall be defined for the coexistence criteria of safety-related sub-elements with non-safety related sub-elements within the same element, or safety-related sub-elements with different ASILs assigned within the same element.

NOTE 1—Definition of a methodology is needed to show that an element implementing requirements with lower ASIL cannot impact an element with requirements with a higher ASIL.

NOTE 2—It is also needed to show that an element implementing non-safety-related requirements cannot impact an element with safety-related requirements.

NOTE 3—The methodology could define rules for coexistence of sub-elements of a different safety integrity level in one element by presenting the mitigation for each potential effect, starting from dependent failure analysis (DFA).

NOTE 4—An informative summary of a proposed methodology [ARCD-ME-8] for coexistence criteria of safety-related sub-elements with non-safety related sub-elements or sub-elements with different ASILs assigned within the same element is described in Table C.7.

7.4.12 Impact assessment vs. requirements in case of reuse of element(s)—ME, DL

Dependability attributes covered are as follows:

- Functional safety

Refer to 6.4.6.

7.4.13 Threat and risk—DL

Dependability attributes covered are as follows:

- Functional safety
- Cybersecurity

Refer to 6.4.7.

7.4.14 Software development environment—DL

Dependability attributes covered are as follows:

- Functional safety

A description language shall be defined for the SW development environment, including development model, languages, and tools. The DL should describe all the parts of SW development environment.

NOTE—An informative summary of a proposed description language [ARCD-DL-9] for SW development environment, including development model, languages, and tools is described in Table C.8.

7.4.15 Model interoperability—ME, DL

Dependability attributes covered are as follows:

- Functional safety
- SOTIF
- Cybersecurity
- Availability

NOTE—Models are used to describe and verify a product's functionalities at a high abstraction level.

7.4.15.1 Model interoperability—ME

There should be a methodology to write models to improve interoperability and data exchange between tools.

NOTE—An informative summary of a proposed methodology [ARCD-ME-9] for the writing of models to improve interoperability and data exchange between tools is described in Table C.7.

7.4.15.2 Model interoperability—DL

There should be a description language for describing models to improve the interoperability and data exchange between tools.

NOTE—An informative summary of a proposed description language [ARCD-DL-10] for describing models to improve interoperability and data exchange between tools is described in Table C.8.

7.4.16 Platform tuning for safety critical real time applications—ME

Dependability attributes covered are as follows:

- Functional safety
- Real time

Refer to 6.4.8.

7.4.17 Real time considerations for safety analysis—ME

Dependability attributes covered are as follows:

- Functional safety
- Real time

Refer to 6.4.9.

7.4.18 Modeling of the radiation working environment—ME

Dependability attributes covered are as follows:

- Functional safety
- Reliability

Refer to 6.4.10.

7.4.19 Ensuring deterministic performance for safety applications—ME

Dependability attributes covered are as follows:

- Functional safety
- Real time
- SOTIF
- Reliability

Refer to 6.4.12.

7.4.20 System level considerations for deterministic performance—ME

Dependability attributes covered are as follows:

- Functional safety
- SOTIF
- Real time
- Reliability

There should be a coherent system-level design methodology to address deterministic performance requiring the capability to express requirements in a standard format and to produce, use, and present data and metrics related to these requirements.

NOTE—Critical safety applications require a system-level implementation to help ensure deterministic performance (CAST-32A). An informative summary of a proposed methodology [ARCD-ME-10] to address deterministic performance requiring the capability to express requirements in a standard format and to produce, use, and present data and metrics related to these requirements is described in Table C.7.

7.4.21 Module design, integration, and testing report—DL

Dependability attributes covered are as follows:

- Functional safety
- SOTIF
- Reliability

There should be a description language and guidance to improve the design, integration, and testing report.

NOTE 1—The design process is highly hierarchical and builds upon smaller modules to build a larger system. In the classical design flow, functional, performance, and cost metrics are clearly expressed and made available to the user of the module, for efficient re-use and integration. Safety and reliability information and metrics need a similar treatment to improve the design, integration and testing activities, and reporting or documentation.

NOTE 2—An informative summary of a proposed description language [ARCD-DL-11] to improve the design, integration, and the testing report is described in Table C.8.

7.4.22 Dependent failure initiators (DFIs)—DB

Dependability attributes covered are as follows:

- Functional safety
- Real time
- Reliability

Description of events that may serve as initiators for dependent failures or common cause/common mode failures shall be documented.

NOTE—The completeness of the DFA depends on the completeness of the list of initiators. An informative description of events that may serve as initiators for dependent failures or common cause/common mode failures proposed database [ARCD-DB-1] is included in Table C.9.

7.4.23 Interface behavior parameters—DB

Dependability attributes covered are as follows:

- Functional safety
- Real time

Parameters influencing the safety of interface behaviors, including malfunctions that may result from faulty interface behavior, as well as their potential consequences and hazards, shall be documented.