FINAL DRAFT

PUBLICLY AVAILABLE SPECIFICATION

# ISO/DPAS 8926

ISO/TC **22**/SC **32**

Secretariat: **JISC**

Voting begins on:
**2023-09-05**

Voting terminates on:
**2023-10-31**

# Road vehicles – Functional safety - Use of pre-existing software architectural elements

RECIPIENTS OF THIS DRAFT ARE INVITED TO SUBMIT, WITH THEIR COMMENTS, NOTIFICATION OF ANY RELEVANT PATENT RIGHTS OF WHICH THEY ARE AWARE AND TO PROVIDE SUPPORTING DOCUMENTATION.

IN ADDITION TO THEIR EVALUATION AS BEING ACCEPTABLE FOR INDUSTRIAL, TECHNOLOGICAL, COMMERCIAL AND USER PURPOSES, DRAFT INTERNATIONAL STANDARDS MAY ON OCCASION HAVE TO BE CONSIDERED IN THE LIGHT OF THEIR POTENTIAL TO BECOME STANDARDS TO WHICH REFERENCE MAY BE MADE IN NATIONAL REGULATIONS.

Reference number
ISO/DPAS 8926:2023(E)

ISO copyright office
CP 401 • Ch. de Blandonnet 8
CH-1214 Vernier, Geneva
Phone: +41 22 749 01 11
Email: copyright@iso.org
Website: www.iso.org

Published in Switzerland

# Contents

# Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular, the different approval criteria needed for the different types of ISO document should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives).

ISO draws attention to the possibility that the implementation of this document may involve the use of (a) patent(s). ISO takes no position concerning the evidence, validity or applicability of any claimed patent rights in respect thereof. As of the date of publication of this document, ISO had not received notice of (a) patent(s) which may be required to implement this document. However, implementers are cautioned that this may not represent the latest information, which may be obtained from the patent database available at www.iso.org/patents. ISO shall not be held responsible for identifying any or all such patent rights.

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation of the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT), see www.iso.org/iso/foreword.html.

This document was prepared by Technical Committee ISO/TC 22 , *Road vehicles*, Subcommittee SC 32, *Electrical and electronic components and general system aspects*

Any feedback or questions on this document should be directed to the user’s national standards body. A complete listing of these bodies can be found at www.iso.org/members.html.

# Introduction

This document addresses the use of pre-existing software architectural elements not originally developed in accordance with the ISO 26262:2018 series in the context of functional safety project conformant to the ISO 26262:2018 series. It describes criteria for a pre-existing software architectural element for its integration to achieve functional safety. The evidence supporting confidence is kept up to date as part of the safety case and is subject to confirmation measures.

Establishing confidence in a pre-existing software architectural element enables its use in safety-related embedded software developed in accordance with the ISO 26262:2018 series when:

- it meets the needs of a target software architectural design because it provides required safety-related functionalities and properties (including safety mechanisms); and
- it meets the needs of a target software architectural design because of its static and dynamic design, its interfaces and its resources used.

**PUBLICLY AVAILABLE SPECIFICATION** **ISO/DPAS 8926:2023(E)**

# Road vehicles – Functional safety - Use of pre-existing software architectural elements

## 1 Scope

This document describes a framework for functional safety to enable the use of pre-existing software architectural elements not originally developed in accordance with the ISO 26262:2018 series, but intended to be integrated into safety-related embedded software conformant with the ISO 26262:2018 series by:

- determining relevant criteria when using the pre-existing software architectural element as a safety-related element of safety-related embedded software;
- determining relevant criteria inherent to the pre-existing software architectural element, e.g. needs for external safety mechanisms to detect and control failures caused by the pre-existing software architectural element;
- supporting the provision of suitable evidence and arguments for use of the pre-existing software architectural element that can include applicable procedures, techniques and safety measures;
- supporting the fulfilment of software safety requirements when using the pre-existing software architectural element as a safety-related element of safety-related embedded software; and
- supporting the integration of the pre-existing software architectural element as a safety-related element of safety-related embedded software.

## 2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO 26262-1:2018, *Road vehicles — Functional safety — Part 1: Vocabulary*

ISO 26262-2:2018, *Road vehicles — Functional safety — Part 2: Management of functional safety*

ISO 26262-6:2018, *Road vehicles — Functional safety — Part 6: Product development at the software level*

ISO 26262-8:2018, *Road vehicles — Functional safety — Part 8: Supporting processes*

ISO 26262-9:2018, *Road vehicles — Functional safety — Part 9: Automotive safety integrity level (ASIL)-oriented and safety-oriented analyses*

## 3 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO 26262-1 and the following apply.

ISO and IEC maintain terminology databases for use in standardization at the following addresses:

- ISO Online browsing platform: available at https://www.iso.org/obp
- IEC Electropedia: available at https://www.electropedia.org/

**3.1**
**complexity**
degree to which a software or a software architectural element has a design, implementation and/or functionalities that are difficult to understand and verify

[SOURCE: ISO/IEC/IEEE 24765:2017, 3.694.3, modified — The phrase "system or component" was replaced by "software or software architectural element" and "and/or functionalities" was added.]

**3.2**
**complexity measure**
variable to which a value is assigned as a result of measurement concerning *complexity* (3.1)

[SOURCE: ISO/IEC 25000:2014, 4.18, modified — The original term was "measure", the phrase "concerning complexity" has been added and the Note 1 to entry has been deleted.]

**3.3**
**pre-existing software architectural element**
**PSAE**
already available commercial off-the-shelf or custom software element not specifically built-to-order, and not developed to conform with ISO 26262:2018 series

**3.4**
**provenance**
information regarding the origins, custody and ownership of a software and its associated data

[SOURCE: Reference [6], modified — The phrase "item or collection" has been replaced by "software and its associated data".]

**3.5**
**target software architectural design**
software architectural design, developed in accordance with ISO 26262:2018 series, into which the *pre-existing software architectural element (PSAE)* (3.3) is intended to be integrated

# 4 Use of pre-existing software architectural elements into safety-related embedded software conformant with the ISO 26262 series

## 4.1 Objectives

This clause applies to PSAE (3.3) with the following objectives:

a) to provide evidence that functional safety is achieved for the target software architectural design (3.5) after integration of the PSAE (3.3);

b) to provide evidence that the PSAE (3.3), once integrated, fulfils the requirements allocated to the PSAE (3.3) in accordance with the target software architectural design (3.5);

c) to manage PSAE (3.3) failure modes relevant to the integration of the PSAE (3.3) in the target software architectural design;

d) to identify and apply appropriate safety measures required to support the achievement of functional safety when using the PSAE (3.3); and

e) to identify foreseeable limitations and to confirm known limitations when using the PSAE (3.3).

## 4.2 General

A PSAE (3.3) is safety-related if it is a safety element in the target software architectural design (3.5), i.e. if software safety requirements derived from the technical safety requirements are allocated to it or if errors of its software functions and/or properties could lead to a violation of the safety requirements.

EXAMPLE 1 An operating system (OS) that is used to host safety-related software applications can have safety-related properties for the correct execution with partitioning to achieve freedom from interference and an OS strategy for fault handling.

EXAMPLE 2 A safety-related device driver can include hardware diagnostics, a client software interface that enables freedom from interference and a strategy for fault handling.

An examination of PSAE (3.3) used in the target software architectural design (3.5) is performed to assess the functional safety implications, including:

- the functionalities and properties of the PSAE (3.3) including identifying those mechanisms that comply with the allocated safety requirements;
- the implementation and interfaces of the PSAE (3.3) that comply with the static and dynamic design aspects of the target software architectural design (3.5);
- that the target environment has sufficient hardware and software resources to meet the software safety requirements of the target software architectural design (3.5);
- that unused functionalities and properties of the PSAE (3.3) do not interfere with the achievement of functional safety or can be excluded from integration (e.g. selected configuration settings during build-process); and
- that unintended behaviours are absent or the risk introduced is sufficiently low.

Annex A provides examples of PSAE (3.3) including the implications of its use on functional safety.

A classification is used to determine whether software qualification is applicable (in accordance with ISO 26262-8:2018, Clause 12) or whether specific safety activities are to be tailored (in accordance with ISO 26262-2:2018, 6.4.5.1 and 6.4.5.2) and planned (in accordance with ISO 26262-2:2018, 6.4.6.7).

NOTE 1 Subclauses 4.4.4 and 4.4.5 describe requirements to tailor the specific safety activities in accordance with ISO 26262-2:2018, 6.4.5.1 and 6.4.5.2 and planned in accordance with ISO 26262-2:2018, 6.4.6.7.

NOTE 2 The confirmation measures defined in ISO 26262-2:2018, 6.4.9 can apply to prevent any anomalies resulting from an inappropriate classification of PSAE (3.3) or impact analysis.

For this purpose, the classification (see 4.4.2) is used to justify the tailoring of specific safety activities to mitigate the risk of integrating the PSAE (3.3) in the target software architectural design (3.5).

The classification is based on criteria that considers:

- the possibility that the uncertainty related to the PSAE (3.3) development may increase the likelihood of systematic faults; and
- the possibility that the complexity (3.1) of the PSAE (3.3) may make finding systematic faults more difficult.

To address the complexity (3.1) of PSAE (3.3), a set of complexity measures (3.2) is selected. Then, the complexity (3.1) of PSAE (3.3) is justified with the reasoning documented as part of the impact analysis report.

NOTE 3 Complexity (3.1) can depend on the use case and the reasoning can vary as well. In some cases, numerical methods, such as cyclomatic complexity or number of lines, can be used while in some other cases qualitative methods can be used to evaluate complexity.

NOTE 4 Criteria for the use of these complexity measures (3.2) can be established to determine whether the activities in ISO 26262-8:2018, Clause 12 provide a suitable risk reduction and improve the detection of systematic faults in the PSAE (3.3).

NOTE 5 Criteria for the use of these complexity measures (3.2) can be established to define the organizational (or project) upper bound for the application of this document's additional safety activities, where the application of the PSAE (3.3) becomes excessively unmanageable and thus not recommended.

Figure 1 illustrates the role of the classification and the dependencies with the target software architectural design (3.5).

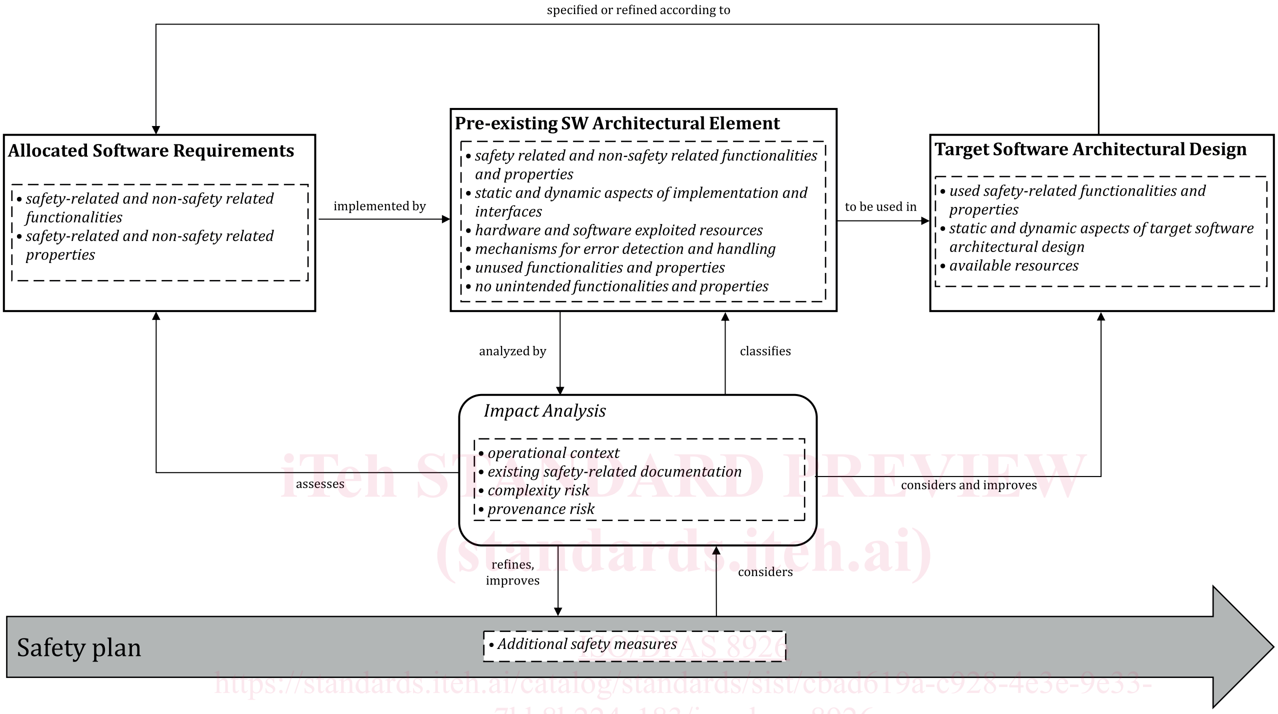
specified or refined according to
Allocated Software Requirements
• safety-related and non-safety related functionalities
• safety-related and non-safety related properties
implemented by
Pre-existing SW Architectural Element
• safety related and non-safety related functionalities and properties
• static and dynamic aspects of implementation and interfaces
• hardware and software exploited resources
• mechanisms for error detection and handling
• unused functionalities and properties
• no unintended functionalities and properties
to be used in
Target Software Architectural Design
• used safety-related functionalities and properties
• static and dynamic aspects of target software architectural design
• available resources
analyzed by
classifies
Impact Analysis
• operational context
• existing safety-related documentation
• complexity risk
• provenance risk
assesses
considers and improves
refines, improves
considers
Safety plan
• Additional safety measures

**Figure 1 — Overview of impact analysis extended by classification**

## 4.3 Input to this clause

### 4.3.1 Prerequisites

The following information shall be available:

- software safety requirements specification for the target software architectural design (3.5) in accordance with ISO 26262-6:2018, 6.5.1;
- safety analysis report for the target software architectural design (3.5) in accordance with ISO 26262-6:2018, 7.5.2;
- documentation of the software development environment related to the target software architectural design (3.5) in accordance with ISO 26262-6:2018, 5.5.1; and
- organization-specific rules and processes for functional safety in accordance with ISO 26262-2:2018, 5.5.1.

### 4.3.2 Further supporting information

The following information can be considered:

- technical safety requirements specification in accordance with ISO 26262-4:2018, 6.5.1;
- technical safety concept in accordance with ISO 26262-4:2018, 6.5.2;
- system architectural design specification in accordance with ISO 26262-4:2018, 6.5.3;
- hardware-software interface (HSI) specification in accordance with ISO 26262-4:2018, 6.5.4;
- rules and processes applied for the design, the implementation and the verification of the PSAE (3.3) (from an external source);
- design specification of the PSAE (3.3) (from an external source);
- specification of non-safety-related functions and properties of the PSAE (3.3) (from an external source);
- PSAE (3.3) implementation (from an external source);
- previous verification report of the PSAE (3.3) (from an external source);
- requirements of the PSAE (3.3) (from an external source);
- specification of functionalities and properties of the PSAE (3.3) (from an external source); and
- other existing information useful for conducting an impact analysis (from an external source).

## 4.4 Requirements and recommendations

### 4.4.1 General

All claims made for the PSAE (3.3) shall be against the specific version and configuration, if applicable, of the PSAE (3.3) that is proposed for integration of the PSAE (3.3) into the target software architectural design (3.5).

### 4.4.2 Classification of a PSAE

**4.4.2.1** A PSAE (3.3) shall be classified based on provenance (3.4) and complexity (3.1) of the PSAE (3.3).

**4.4.2.2** A set of complexity measures (3.2) shall be established to determine the likelihood of PSAE (3.3) systematic faults occurring, as indicated by ISO 26262-6:2018, 7.4.3, including the rationale for its appropriateness.

NOTE Annex B provides examples of complexity measures (3.2).

**4.4.2.3** The values of the complexity measures (3.2) shall be determined for the PSAE (3.3).

**4.4.2.4** The existing supporting information of the PSAE (3.3), if available together with its source and its object code, shall be analysed and evaluated to determine:

a) the provenance-related uncertainty that the PSAE (3.3) can contain systematic faults impacting the target software architectural design (3.5). This is expressed as follows:

   - P1 shall be selected when there is evidence that the software development process for the PSAE (3.3) is based on an appropriate national or international standard (e.g. ISO/IEC/IEEE 12207) or a different functional safety standard (e.g. IEC 61508, RTCA DO-178C[7]);