

Automotive SPICE[®]

Process Reference Model

Process Assessment Model

Version 3.991

Title: Automotive SPICE Process Assessment / Reference Model
Author(s): VDA QMC Working Group 13
Version: 3.991
Date: 2023-06-06
Status: Draft (Do not use for assessments)
Confidentiality:
Revision ID:

DISCLAIMER:

This published version is a draft version of the upcoming Automotive SPICE Process Assessment / Reference Model in its version 4.0.

It is intended to provide a preview of the new model and shall not be used for assessments.

Copyright notice

This document is a revision of the Automotive SPICE process assessment model and process reference model 3.1, which has been developed by the Working Group 13 of the Quality Management Center (QMC) in the German Association of the Automotive Industry.

This document reproduces relevant material from:

- **ISO/IEC 33020:2015**
Information technology -- Process assessment -- Process measurement framework for assessment of process capability

ISO/IEC 33020:2015 provides the following copyright release statement:

'Users of this International Standard may reproduce subclauses 5.2, 5.3, 5.4 and 5.6 as part of any process assessment model or maturity model so that it can be used for its intended purpose.'

- **ISO/IEC 15504-5:2006**
Information Technology – Process assessment – Part 5: An exemplar Process Assessment Model

ISO/IEC 15504-5:2006 provides the following copyright release statement:

'Users of this part of ISO/IEC 15504 may freely reproduce the detailed descriptions contained in the exemplar assessment model as part of any tool or other material to support the performance of process assessments, so that it can be used for its intended purpose.'

Relevant material from one of the mentioned standards is incorporated under the copyright release notice.

Acknowledgement

The VDA, the VDA QMC and the Project Group 13 explicitly acknowledge the high quality work carried out by the members of the intacs™ working groups. We would like to thank all involved people, who have contributed to the development and publication of Automotive SPICE®.

Derivative works

You may not alter, transform, or build upon this work without the prior consent of the VDA Quality Management Center. Such consent may be given provided ISO copyright is not infringed.

The detailed descriptions contained in this document may be incorporated as part of any tool or other material to support the performance of process assessments, so that this process assessment model can be used for its intended purpose, provided that any such material is not offered for sale.

All distribution of derivative works shall be made at no cost to the recipient.

Document distribution

The Automotive SPICE® process assessment model may only be obtained by download from the www.vda-qmc.de web site. It is not permitted for the recipient to further distribute the document.

Change requests

Any problems or change requests should be reported through the defined mechanism at the www.vda-qmc.de web site.

Trademark notice

Automotive SPICE® is a registered trademark of the Verband der Automobilindustrie e.V. (VDA)
 For further information about Automotive SPICE® visit www.vda-qmc.de.

Document history

Version	Date	By	Notes
2.0	2005-05-04	AutoSIG / SUG	DRAFT RELEASE, pending final editorial review
2.1	2005-06-24	AutoSIG / SUG	Editorial review comments implemented Updated to reflect changes in FDIS 15504-5
2.2	2005-08-21	AutoSIG / SUG	Final checks implemented: FORMAL RELEASE
2.3	2007-05-05	AutoSIG / SUG	Revision following CCB: FORMAL RELEASE
2.4	2008-08-01	AutoSIG / SUG	Revision following CCB: FORMAL RELEASE
2.5	2010-05-10	AutoSIG / SUG	Revision following CCB: FORMAL RELEASE
3.0	2015-07-16	VDA QMC WG13	Changes: See release notes
3.1	2017-11-01	VDA QMC WG13	Changes: See www.automotivespice.com
3.991	2023-06-06	VDA QMC WG13	First draft release

Release notes

The release notes will be updated with the final version of Automotive SPICE 4.0.

Table of contents

Copyright notice	2
Acknowledgement.....	2
Derivative works.....	2
Document distribution	2
Change requests.....	2
Trademark notice	3
Document history	3
Release notes.....	3
Table of contents	4
List of Figures	6
List of Tables	6
1. Introduction.....	7
1.1. Scope.....	7
1.2. Terminology.....	7
1.3. Abbreviations.....	10
2. Statement of compliance.....	11
3. Process capability determination.....	12
3.1. Process reference model.....	13
3.1.1. Primary life cycle processes category	13
3.1.2. Supporting life cycle processes category	15
3.1.3. Organizational life cycle processes category	15
3.2. Measurement framework.....	17
3.2.1. Process capability levels and process attributes	17
3.2.2. Process attribute rating	18
3.2.3. Rating and aggregation method.....	20
3.2.4. Process capability level model	22
3.3. Process assessment model.....	23
3.3.1. Assessment indicators	23
3.3.2. Understanding information items and work products.....	24
3.3.3. Understanding the level of abstraction of a PAM.....	26
3.3.4. Why a PRM and PAM are not a lifecycle model	27
4. Process reference model and performance indicators (Level 1).....	28
4.1. Acquisition process group (ACQ).....	29
4.1.1. ACQ.4 Supplier Monitoring	29
4.2. Supply process group (SPL).....	31
4.2.1. SPL.2 Product Release.....	31
4.3. System engineering process group (SYS).....	33
4.3.1. SYS.1 Requirements Elicitation	33
4.3.2. SYS.2 System Requirements Analysis.....	35
4.3.3. SYS.3 System Architectural Design	37
4.3.4. SYS.4 System Integration and Integration Verification	39
4.3.5. SYS.5 System Verification	42

4.4.	Software engineering process group (SWE)	44
4.4.1.	SWE.1 Software Requirements Analysis	44
4.4.2.	SWE.2 Software Architectural Design	46
4.4.3.	SWE.3 Software Detailed Design and Unit Construction	48
4.4.4.	SWE.4 Software Unit Verification	50
4.4.5.	SWE.5 Software Component Verification and Integration Verification	51
4.4.6.	SWE.6 Software Verification	55
4.5.	Validation process group (VAL)	57
4.5.1.	VAL.1 Validation	57
4.6.	Machine Learning Engineering process group (MLE)	60
4.6.1.	MLE.1 Machine Learning Requirements Analysis	60
4.6.2.	MLE.2 Machine Learning Architecture	62
4.6.3.	MLE.3 Machine Learning Training	64
4.6.4.	MLE.4 Machine Learning Model Testing	66
4.7.	Hardware Engineering process group (HWE)	69
4.7.1.	HWE.1 Hardware Requirements Analysis	69
4.7.2.	HWE.2 Hardware Design	72
4.7.3.	HWE.3 Verification against Hardware Design	75
4.7.4.	HWE.4 Verification against Hardware Requirements	77
4.8.	Supporting process group (SUP)	79
4.8.1.	SUP.1 Quality Assurance	79
4.8.2.	SUP.8 Configuration Management	81
4.8.3.	SUP.9 Problem Resolution Management	84
4.8.4.	SUP.10 Change Request Management	86
4.8.5.	SUP.11 Machine Learning Data Management	87
4.9.	Management process group (MAN)	90
4.9.1.	MAN.3 Project Management	90
4.9.2.	MAN.5 Risk Management	93
4.9.3.	MAN.6 Measurement	95
4.10.	Process improvement process group (PIM)	96
4.10.1.	PIM.3 Process Improvement	96
4.11.	Reuse process group (REU)	99
4.11.1.	REU.2 Management of Products for Reuse	99
5.	Process capability levels and process attributes	101
5.1.	Process capability level 0: Incomplete process	101
5.2.	Process capability Level 1: Performed process	102
5.2.1.	PA 1.1 Process performance process attribute	102
5.3.	Process capability Level 2: Managed process	102
5.3.1.	PA 2.1 Process performance management process attribute	103
5.3.2.	PA 2.2 Work product management process attribute	106
5.4.	Process capability Level 3: Established process	107
5.4.1.	PA 3.1 Process definition process attribute	108
5.4.2.	PA 3.2 Process deployment process attribute	110
5.5.	Process capability Level 4: Predictable process	112
5.5.1.	PA 4.1 Quantitative analysis process attribute	112

5.5.2.	PA 4.2 Quantitative control process attribute	115
5.6.	Process capability Level 5: Innovating process.....	117
5.6.1.	PA 5.1 Process innovation process attribute	117
5.6.2.	PA 5.2 Process innovation implementation process attribute	119
Annex A	Conformity statements	121
Annex A.1	Introduction	121
Annex A.2	Conformance to the requirements for process reference models	121
Annex A.3	Conformance to the requirements for process assessment models	122
Annex A.4	Conformance to the requirements for measurement frameworks	124
Annex B	Information item characteristics	125
Annex C	Key concepts and guidance	165
Annex D	Reference standards	166

List of Figures

Figure 1	— Process assessment model relationship	12
Figure 2	— Automotive SPICE process reference model - Overview	13
Figure 3	— Relationship between assessment indicators and process capability.....	24
Figure 4	— Possible levels of abstraction for the term "process"	26
Figure 5	— Performing a process assessment for determining process capability	27

List of Tables

Table 1	— Terminology	10
Table 2	— Abbreviation List	11
Table 3	— Primary life cycle processes – ACQ process group.....	14
Table 4	— Primary life cycle processes – SPL process group.....	14
Table 5	— Primary life cycle processes – SYS process group	14
Table 6	— Primary life cycle processes – VAL process group.....	14
Table 7	— Primary life cycle processes – SWE process group	14
Table 8	— Primary life cycle processes – MLE process group	15
Table 9	— Primary life cycle processes – HWE process group	15
Table 10	— Supporting life cycle processes - SUP process group	15
Table 11	— Organizational life cycle processes - MAN process group.....	15
Table 12	— Organizational life cycle processes - PIM process group	16
Table 13	— Organizational life cycle processes - REU process group	16
Table 14	— Process capability levels	17
Table 15	— Process attributes	18
Table 16	— Rating scale	19
Table 17	— Rating scale percentage values	19
Table 18	— Refinement of rating scale	19
Table 19	— Refined rating scale percentage values	19
Table 20	— Process capability level model	22
Table 21	— Template for the process description	28
Table 22	— Template for the process description	101
Table B.1	— Structure of IIC tables.....	125
Table B.2	— Information item characteristics	164

1. Introduction

1.1. Scope

Process assessment is a disciplined evaluation of an organizational unit's processes against a process assessment model.

The Automotive SPICE process assessment model (PAM) is intended for use when performing conformant assessments of the process capability on the development of embedded automotive systems. It was developed in accordance with the requirements of ISO/IEC 33004:2015.

Automotive SPICE has its own process reference model (PRM), which was developed based on the Automotive SPICE process reference model 4.5. It was further developed and tailored considering the specific needs of the automotive industry. If processes beyond the scope of Automotive SPICE are needed, appropriate processes from other process reference models such as ISO/IEC 12207:2017 or ISO/IEC/IEEE 15288:2015 may be added based on the business needs of the organization.

The PRM is incorporated in this document and is used in conjunction with the Automotive SPICE process assessment model when performing an assessment.

This Automotive SPICE process assessment model contains a set of indicators to be considered when interpreting the intent of the Automotive SPICE process reference model. These indicators may also be used when implementing a process improvement program.

1.2. Terminology

Automotive SPICE follows the following precedence for use of terminology:

- ISO/IEC 33001 for assessment related terminology
- ISO/IEC/IEEE 24765, ISO/SAE 21434 and ISO/IEC/IEEE 29119 terminology (as contained in Annex C)
- Terms introduced by Automotive SPICE (as contained in Annex C)
- PMBOK® Guide – Fourth Edition
- PAS 1883:2020

Term	Origin	Description
Acceptance testing	ISO/IEC/IEEE 24765	Formal testing conducted to enable a user, customer, or authorized entity to determine whether to accept a system or component.
Activity	Automotive SPICE	Execution of a task by a stakeholder or an involved party.
Application parameter	Automotive SPICE V3.1	An application parameter is a parameter containing data applied to the system or software functions, behavior or properties. The notion of application parameter is expressed in two ways: firstly, the logical specification (including name, description, unit, value domain or threshold values or characteristic curves, respectively), and, secondly, the actual quantitative data value it receives by means of data application.

Approval	Automotive SPICE V4.0	Written statement that a deliverable is capable for its intended use.
Baseline	ISO/IEC/IEEE 24765	A specification or product that has been formally reviewed and agreed upon, that thereafter serves as the basis for further development, and can be changed only through formal change control procedures.
Black-box testing	Automotive SPICE V4.0	Method of testing where tests are developed without knowledge of the internal structure and mechanisms of the tested object-under-test.
Coding	ISO/IEC/IEEE 24765	The transforming of logic and data from design specifications (design descriptions) into programming language.
Deliverable	PMBOK® Guide – Fourth Edition	Any unique and verifiable product, result, or capability to perform a service that must be produced to complete a process, phase, or project. Often used more narrowly in reference to an external deliverable, which is a deliverable that is subject to approval by the project sponsor or customer.
Error	ISO/IEC/IEEE 24765	The difference between a computed, observed, or measured value or condition and the true, specified, or theoretically correct value or condition.
Fault	ISO/IEC/IEEE 24765	A manifestation of an error in software.
Functional requirement	ISO/IEC/IEEE 24765	A statement that identifies what a product or process must accomplish to produce required behavior and/or results.
Functional specification	ISO/IEC/IEEE 24765	A document that specifies the functions that a system or component must perform. Often part of a requirements specification.
Functional testing	ISO/IEC/IEEE 24765	Testing conducted to evaluate the compliance of a system or component with specified functional requirements.
Hardware	ISO/IEC/IEEE 24765	Physical equipment used to process, store, or transmit computer programs or data.
Hyperparameter	Automotive SPICE V4.0	In machine learning, a hyperparameter is a parameter whose value is used to control the training of the ML model. Its value must be set between training iterations. Examples: learning rate, loss function, model depth, regularization constants.
Integration	Automotive SPICE V3.1	A process of combining items to larger items up to an overall system.
Integration testing	Automotive SPICE V3.1	Testing in which items (software items, hardware items, or system items) are combined and tested to evaluate the interaction among them.
Information need		The need for characterizing process or product related effectiveness and efficiency.
Machine Learning (ML)	Automotive SPICE V4.0	In Automotive SPICE, Machine Learning (ML) describes the ability of software to learn from specific training data and to apply this knowledge to other similar tasks
Measure	Automotive SPICE V4.0	An activity to achieve a certain intent
Measurement	Oxford Dictionary	“The activity to find the size, quantity or degree of something”

Metric	Automotive SPICE V4.0	A quantitative or qualitative measurable indicator that matches information needs.
ODD	PAS 1883:2020	Operational Design Domain (ODD) is operating conditions under which a given overall system or feature thereof is specifically designed to function. This includes, but is not limited to, environmental, geographical, and time-of-day restrictions, and/or the requisite presence or absence of certain traffic or roadway characteristics.
Project	ISO/IEC/IEEE 24765	Endeavor with defined start and finish dates undertaken to create a product or service in accordance with specified resources and requirements
Quality assurance	ISO/IEC/IEEE 24765	A planned and systematic pattern of all actions necessary to provide adequate confidence that an item or product conforms to established technical requirements.
Release		A physical product compilation to be transferred to the customer
Regression verification	Automotive SPICE V4.0	Selective re-verification of elements to verify that modifications have not caused unintended effects
Risk	ISO/IEC/IEEE 24765	The combination of the probability of occurrence and the consequences of a given future undesirable event
Software component	Automotive SPICE V4.0	As a result of the decomposition of the architecture at software level, the software is decomposed across appropriate hierarchical levels of components down to the lowest-level software components in a conceptual model; or a representation of a software component under verification such as source code, an object file, a library, an executable, or executable model as input to verification
Software Unit	Automotive SPICE V4.0	Can be a representation of a software element, which is decided not to be further subdivided and that is a part of a software component at the lowest level, in a conceptual model or a representation of a software unit under verification such as source code, an object file, a library, an executable, or an executable model as input to verification
Software element	Automotive SPICE V4.0	Refers to a →Software Component or a →Software Unit
Stakeholder requirements	Automotive SPICE V4.0	Any type of requirement for the stakeholders in the given context, e.g. customer requirement, supplier internal requirements (product-specific, platform etc.), legal requirements, regulatory requirements, statutory requirements, industry sector requirements, international standards, codes of practice etc. ...
System	Automotive SPICE V4.0	A collection of interacting system elements organized to accomplish a specific functionality or set of functionalities within a specific environment.

System Element	Automotive SPICE V4.0	<p>System elements can be logical and structural objects at the architectural and design level. Such elements can be further decomposed into more fine-grained sub-elements of the architecture or design across appropriate hierarchical levels;</p> <p>or physical representations of these objects, or a combination, e.g. peripherals, sensors, actuators, mechanical parts, software executables.</p> <p>Note: An aspect of semiconductor development (systems-on-chip or microcontrollers) are systems in that they comprise a mechanical housing, EE hardware, and firmware) pre-silicon verification is performed. This can be done in the context of SYS.3 and, also, SYS.4 and SYS.5, respectively (e.g. by back-to-back tests).</p>
Unit test	Automotive SPICE V4.0	Testing of individual software units
Task		A definition, but not the execution, of a coherent and set of atomic actions.
Validation measure	Automotive SPICE V4.0	Validation measure is an umbrella term for e.g. operational use case testing under real-life conditions, highly accelerated life testing (HALT), simulations under real-life conditions, end user trials, panel or blind tests, or expert panels
Verification	Automotive SPICE V4.0	Verification is confirmation, through the provision of objective evidence, that an element fulfils the specified requirements
Verification measure	Automotive SPICE V4.0	Verification measure is an umbrella term for test cases, measurements, calculations, simulations, reviews, and analyses. In particular domains particular verification measures may not be applicable, e.g. software units generally cannot be verified by means of calculations or analyses.
White-box testing	Automotive SPICE V4.0	Verification method where verification measures are developed based on the knowledge of the element's internal structure and mechanisms

Table 1 — Terminology

1.3. Abbreviations

BP	Base Practice
CAN	Controller Area Network
CASE	Computer-Aided Software Engineering,
CCB	Change Control Board
CPU	Central Processing Unit
DFA	Dependent Failure Analysis
ECU	Electronic Control Unit
EEPROM	Electrically Erasable Programmable Read Only Memory

EOL	End-of-Line
FMEA	Failure Mode and Effect Analysis
FTA	Fault Tree Analysis
GP	Generic Practice
GR	Generic Resource
IEC	International Electrotechnical Commission
IEEE	Institute of Electrical and Electronics Engineers
I/O	Input / Output
ISO	International Organization for Standardization
LIN	Local Interconnect Network
MISRA	Motor Industry Software Reliability Association
MOST	Media Oriented Systems Transport
PA	Process Attribute
PAM	Process Assessment Model
PRM	Process Reference Model
PWM	Pulse Width Modulation
RAM	Random Access Memory
ROM	Read Only Memory
SPICE	Systems Process Improvement and Capability dEtermination
SUG	Spice User Group
USB	Universal Serial Bus
WP	Work Product
WPC	Work Product Characteristic

Table 2 — Abbreviation List

2. Statement of compliance

The Automotive SPICE process reference model and process assessment model are conformant with the ISO/IEC 33004 and can be used as the basis for conducting an assessment of process capability.

An ISO/IEC 33003 compliant Measurement Framework is defined in section 5.

A statement of compliance of the process assessment model and process reference model with the requirements of ISO/IEC 33004 is provided in Annex A.

A statement of compliance of the measurement framework with the requirements of ISO/IEC 33003 is provided in Annex A.

3. Process capability determination

The concept of process capability determination by using a process assessment model is based on a two-dimensional framework. The first dimension is provided by processes defined in a process reference model (process dimension). The second dimension consists of capability levels that are further subdivided into process attributes (capability dimension). The process attributes provide the measurable characteristics of process capability.

The process assessment model selects processes from a process reference model and supplements with indicators. These indicators support the collection of objective evidence which enable an assessor to assign ratings for processes according to the capability dimension.

The relationship is shown in Figure 1.

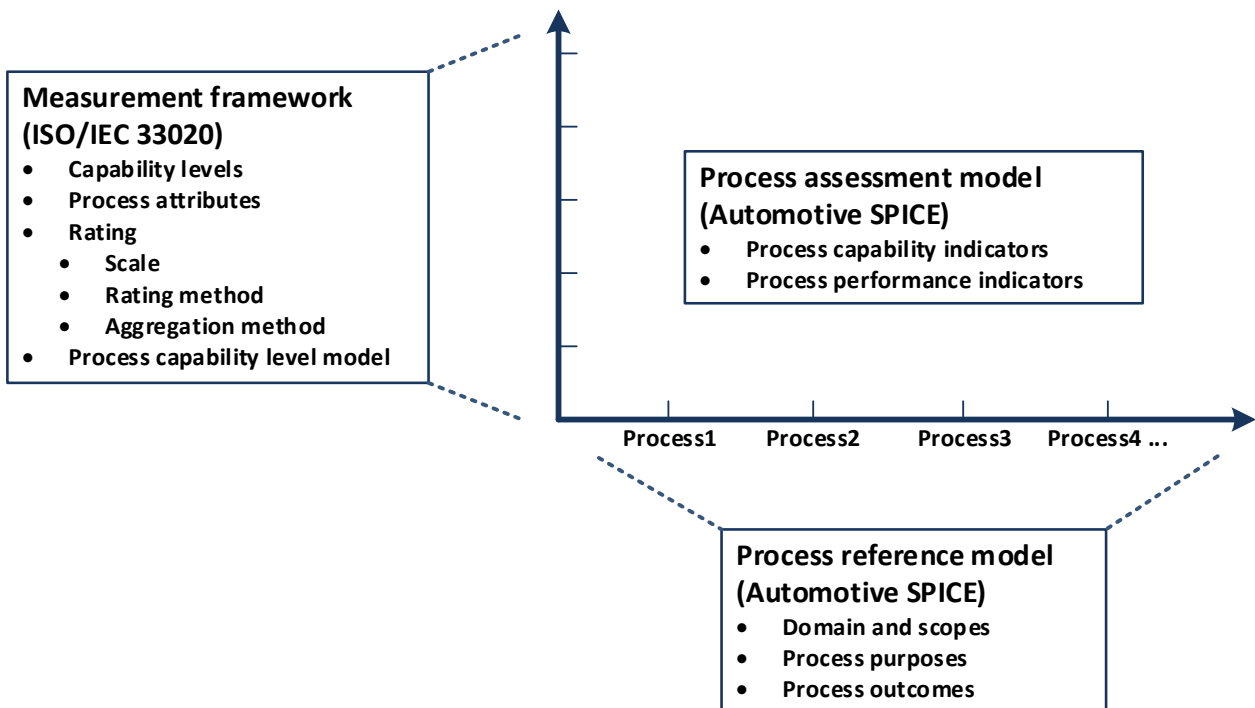


Figure 1 — Process assessment model relationship

3.1. Process reference model

Processes are collected into process groups according to the domain of activities they address.

These process groups are organized into 3 process categories: Primary life cycle processes, Organizational life cycle processes and Supporting life cycle processes.

For each process a purpose statement is formulated that contains the unique functional objectives of the process when performed in a particular environment. For each purpose statement a list of specific outcomes is associated, as a list of expected positive results of the process performance.

For the process dimension, the Automotive SPICE process reference model provides the set of processes shown in Figure 2.

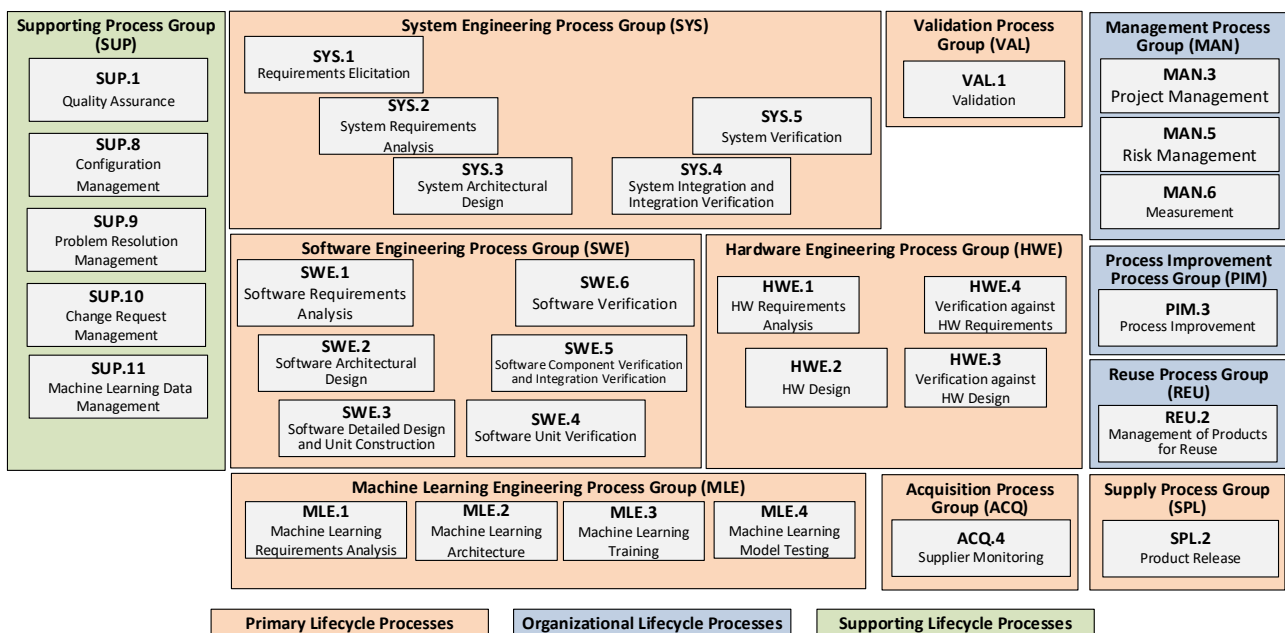


Figure 2 — Automotive SPICE process reference model - Overview

3.1.1. Primary life cycle processes category

The primary life cycle processes category consists of processes that may apply for an acquirer of products from a supplier or may apply for product development when responding to stakeholder needs and delivering products including the engineering processes needed for specification, design, implementation, integration and verification.

The primary life cycle processes category consists of the following groups:

- the Acquisition process group;
- the Supply process group;
- the System engineering process group;
- the Validation process group;
- the Software engineering process group;
- the Machine learning engineering process group and
- the Hardware engineering process group.

The Acquisition process group (ACQ) consists of one process that is performed by the customer, or by the supplier when acting as a customer for its own suppliers, in order to acquire a product and/or service.

ACQ.4 Supplier Monitoring

Table 3 — Primary life cycle processes – ACQ process group

The Supply process group (SPL) consists of one process performed by the supplier in order to supply a product and/or a service.

SPL.2 Product Release

Table 4 — Primary life cycle processes – SPL process group

The System Engineering process group (SYS) consists of processes addressing the elicitation and management of customer and internal requirements, the definition of the system architecture and the integration and verification on the system level.

- SYS.1** Requirements Elicitation
- SYS.2** System Requirements Analysis
- SYS.3** System Architectural Design
- SYS.4** System Integration and Integration Verification
- SYS.5** System Verification

Table 5 — Primary life cycle processes – SYS process group

The Validation process group (VAL) consists of one process that is performed to provide evidence that the product to be delivered satisfies the expectations for its intended use.

VAL.1 Validation

Table 6 — Primary life cycle processes – VAL process group

The Software Engineering process group (SWE) consists of processes addressing the management of software requirements derived from the system requirements, the development of the corresponding software architecture and design as well as the implementation, integration and verification of the software.

- SWE.1** Software Requirements Analysis
- SWE.2** Software Architectural Design
- SWE.3** Software Detailed Design and Unit Construction
- SWE.4** Software Unit Verification
- SWE.5** Software Component Verification and Integration Verification
- SWE.6** Software Verification

Table 7 — Primary life cycle processes – SWE process group

The Machine Learning Engineering process group (MLE) consists of processes addressing the management of ML requirements derived from the software requirements, the development of the corresponding ML architecture, the training of ML model, and testing of ML model against ML requirements.

- MLE.1** Machine Learning Requirements Analysis
- MLE.2** Machine Learning Architecture

- MLE.3** Machine Learning Training
- MLE.4** Machine Learning Model Testing

Table 8 — Primary life cycle processes – MLE process group

The Hardware Engineering process group (HWE) consists of processes addressing the management of hardware requirements derived from the system requirements, the development of the corresponding hardware architecture and design as well as the verification of the hardware.

- HWE.1** Hardware Requirements Analysis
- HWE.2** Hardware Design
- HWE.3** Verification against Hardware Design
- HWE.4** Verification against Hardware Requirements

Table 9 — Primary life cycle processes – HWE process group

3.1.2. Supporting life cycle processes category

The supporting life cycle processes category consists of processes that may be employed by any of the other processes at various points in the life cycle.

- SUP.1** Quality Assurance
- SUP.8** Configuration Management
- SUP.9** Problem Resolution Management
- SUP.10** Change Request Management
- SUP.11** Machine Learning Data Management

Table 10 — Supporting life cycle processes - SUP process group

3.1.3. Organizational life cycle processes category

The organizational life cycle processes category consists of processes that develop process, product, and resource assets which, when used by projects in the organization, may help the organization achieve its business goals.

The organizational life cycle processes category consists of the following groups:

- the Management process group;
- the Process Improvement process group;
- the Reuse process group.

The Management process group (MAN) consists of processes that may be used by anyone who manages any type of project or process within the life cycle.

- MAN.3** Project Management
- MAN.5** Risk Management
- MAN.6** Measurement

Table 11 — Organizational life cycle processes - MAN process group

The Process Improvement process group (PIM) covers one process that contains practices to improve the processes performed in the organizational unit.

PIM.3 Process Improvement

Table 12 — Organizational life cycle processes - PIM process group

The Reuse process group (REU) covers one process to systematically exploit reuse opportunities in organization's product portfolio.

REU.2 Management of Products for Reuse

Table 13 — Organizational life cycle processes - REU process group

3.2. Measurement framework

The measurement framework provides the necessary requirements and rules for the capability dimension. It defines a schema which enables an assessor to determine the capability level of a given process. These capability levels are defined as part of the measurement framework.

To enable the rating, the measurement framework provides process attributes defining a measurable property of process capability. Each process attribute is assigned to a specific capability level. The extent of achievement of a certain process attribute is represented by means of a rating based on a defined rating scale. The rules from which an assessor can derive a final capability level for a given process are represented by a process capability level model.

Automotive SPICE defines its own measurement framework.

Note: The Automotive SPICE measurement framework is an adaption of ISO/IEC 33020:2019. Text incorporated from ISO/IEC 33020 within this chapter is written in italic font and marked with a left side bar.

3.2.1. Process capability levels and process attributes

The process capability levels, and associated process attributes are described in detail in chapter 5.

Process attributes are features of a process that can be evaluated on a scale of achievement, providing a measurement of the capability of the process. They are applicable to all processes.

A capability level is a set of process attribute(s) that work together to provide a major enhancement in the capability to perform a process. Each attribute addresses a specific aspect of the capability level. The levels constitute a rational way of progressing through improvement of the capability of any process.

There are six capability levels as listed in table 13, incorporating nine process attributes:

<i>Level 0: Incomplete process</i>	<i>The process is not implemented or fails to achieve its process purpose.</i>
<i>Level 1: Performed process</i>	<i>The implemented process achieves its process purpose</i>
<i>Level 3: Established process</i>	<i>The previously described managed process is now implemented using a defined process that is capable of achieving its process outcomes.</i>
<i>Level 4: Predictable process</i>	<i>The previously described established process now operates predictively within defined limits to achieve its process outcomes. Quantitative management needs are identified, measurement data are collected and analyzed to identify assignable causes of variation. Corrective action is taken to address assignable causes of variation.</i>
<i>Level 5: Innovating process</i>	<i>The previously described predictable process is now continually improved to respond to organizational change.</i>

Table 14 — Process capability levels

Within this process assessment model, the determination of capability is based upon the nine process attributes (PA) as listed in Table 14.

Attribute ID	Process Attributes
<i>Level 0: Incomplete process</i>	
<i>Level 1: Performed process</i>	
PA 1.1	<i>Process performance process attribute</i>
<i>Level 2: Managed process</i>	
PA 2.1	<i>Performance management process attribute</i>
PA 2.2	<i>Work product management process attribute</i>
<i>Level 3: Established process</i>	
PA 3.1	<i>Process definition process attribute</i>
PA 3.2	<i>Process deployment process attribute</i>
<i>Level 4: Predictable process</i>	
PA 4.1	<i>Quantitative analysis process attribute</i>
PA 4.2	<i>Quantitative control process attribute</i>
<i>Level 5: Innovating process</i>	
PA 5.1	<i>Process innovation process attribute</i>
PA 5.2	<i>Process innovation implementation process attribute</i>

Table 15 — Process attributes

3.2.2. Process attribute rating

To support the rating of process attributes, the measurement framework provides a defined rating scale with an option for refinement, different rating methods and different aggregation methods depending on the class of the assessment (e.g. required for organizational maturity assessments).

3.2.2.1. Rating scale

Within this process measurement framework, a process attribute is a measurable property of process capability. A process attribute rating is a judgement of the degree of achievement of the process attribute for the assessed process.

The rating scale is shown in Table 15.

Note. The rating scale is identical to ISO/IEC 33020:2019

<i>N</i>	<i>Not achieved</i>	<i>There is little or no evidence of achievement of the defined process attribute in the assessed process.</i>
<i>P</i>	<i>Partially achieved</i>	<i>There is some evidence of an approach to, and some achievement of, the defined process attribute in the assessed process. Some aspects of achievement of the process attribute may be unpredictable.</i>

L	Largely achieved	There is evidence of a systematic approach to, and significant achievement of, the defined process attribute in the assessed process. Some weaknesses related to this process attribute may exist in the assessed process.
F	Fully achieved	There is evidence of a complete and systematic approach to, and full achievement of, the defined process attribute in the assessed process. No significant weaknesses related to this process attribute exist in the assessed process.

Table 16 — Rating scale

The ordinal scale defined above shall be understood in terms of percentage achievement of a process attribute. The corresponding percentages shall be:

N	Not achieved	0 to ≤ 15% achievement
P	Partially achieved	> 15% to ≤ 50% achievement
L	Largely achieved	> 50% to ≤ 85% achievement
F	Fully achieved	> 85% to ≤ 100% achievement

Table 17 — Rating scale percentage values

The ordinal scale may be further refined for the measures P and L as defined below.

P-	Partially achieved:	There is some evidence of an approach to, and some achievement of, the defined process attribute in the assessed process. Many aspects of achievement of the process attribute may be unpredictable.
P+	Partially achieved:	There is some evidence of an approach to, and some achievement of, the defined process attribute in the assessed process. Some aspects of achievement of the process attribute may be unpredictable.
L-	Largely achieved:	There is evidence of a systematic approach to, and significant achievement of, the defined process attribute in the assessed process. Many weaknesses related to this process attribute may exist in the assessed process.
L+	Largely achieved:	There is evidence of a systematic approach to, and significant achievement of, the defined process attribute in the assessed process. Some weaknesses related to this process attribute may exist in the assessed process.

Table 18 — Refinement of rating scale

The corresponding percentages shall be:

P-	Partially achieved -	> 15% to ≤ 32.5% achievement
P+	Partially achieved +	> 32.5 to ≤ 50% achievement
L-	Largely achieved -	> 50% to ≤ 67.5% achievement
L+	Largely achieved +	> 67.5% to ≤ 85% achievement

Table 19 — Refined rating scale percentage values

3.2.3. Rating and aggregation method

Rating and aggregation methods are taken from ISO/IEC 33020:2019, which provides the following definitions:

A process outcome is the observable result of successful achievement of the process purpose.

A process attribute outcome is the observable result of achievement of a specified process attribute.

Process outcomes and process attribute outcomes may be characterised as an intermediate step to providing a process attribute rating.

When performing rating, the rating method employed shall be specified relevant to the class of assessment. The following rating methods are defined.

The use of rating method may vary according to the class, scope and context of an assessment. The lead assessor shall decide which (if any) rating method to use. The selected rating method(s) shall be specified in the assessment input and referenced in the assessment report.

ISO/IEC 33020:2019 provides the following 3 rating methods:

Rating method R1

The approach to process attribute rating shall satisfy the following conditions:

- a) Each process outcome of each process within the scope of the assessment shall be characterized for each process instance, based on validated data;*
- b) Each process attribute outcome of each process attribute for each process within the scope of the assessment shall be characterized for each process instance, based on validated data;*
- c) Process outcome characterizations for all assessed process instances shall be aggregated to provide a process performance attribute achievement rating;*
- d) Process attribute outcome characterizations for all assessed process instances shall be aggregated to provide a process attribute achievement rating.*

Rating method R2

The approach to process attribute rating shall satisfy the following conditions:

- a) Each process attribute for each process within the scope of the assessment shall be characterized for each process instance, based on validated data;*
- b) Process attribute characterizations for all assessed process instances shall be aggregated to provide a process attribute achievement rating.*

Rating method R3

Process attribute rating across assessed process instances shall be made without aggregation.

In principle the three rating methods defined in ISO/IEC 33020:2019 depend on

- a) whether the rating is made only on process attribute level (Rating method 3 and 2) or – with more level of detail – both on process attribute and process attribute outcome level (Rating method 1); and
- b) the type of aggregation ratings across the assessed process instances for each process

If a rating is performed for both process attributes and process attribute outcomes (Rating method 1), the result will be a process performance attribute outcome rating on level 1 and a process attribute achievement rating on higher levels.

Depending on the class, scope and context of the assessment an aggregation within one process (one-dimensional, vertical aggregation), across multiple process instances (one-dimensional, horizontal aggregation) or both (two-dimensional, matrix aggregation) is performed.

ISO/IEC 33020:2019 provides the following examples:

When performing an assessment, ratings may be summarised across one or two dimensions.

For example, when rating a

- *process attribute for a given process, one may aggregate ratings of the associated process (attribute) outcomes – such an aggregation will be performed as a vertical aggregation (one dimension).*
- *process (attribute) outcome for a given process attribute across multiple process instances, one may aggregate the ratings of the associated process instances for the given process (attribute) outcome such an aggregation will be performed as a horizontal aggregation (one dimension)*
- *process attribute for a given process, one may aggregate the ratings of all the process (attribute) outcomes for all the processes instances – such an aggregation will be performed as a matrix aggregation across the full scope of ratings (two dimensions)*

The standard defines different methods for aggregation. Further information can be taken from ISO/IEC 33020:2019.

3.2.4. Process capability level model

The process capability level achieved by a process shall be derived from the process attribute ratings for that process according to the process capability level model defined in Table 19.

The process capability level model defines the rules how the achievement of each level depends on the rating of the process attributes for the assessed and all lower levels.

As a general rule the achievement of a given level requires a largely achievement of the corresponding process attributes and a full achievement of any lower lying process attribute.

Scale	Process attribute	Rating
Level 1	PA 1.1: Process performance process attribute	Largely
Level 2	PA 1.1: Process performance process attribute PA 2.1: Process performance management process attribute PA 2.2: Work product management process attribute	Fully Largely Largely
Level 3	PA 1.1: Process performance process attribute PA 2.1: Process performance management process attribute PA 2.2: Work product management process attribute PA 3.1: Process definition process attribute PA 3.2: Process deployment process attribute	Fully Fully Fully Largely Largely
Level 4	PA 1.1: Process performance process attribute PA 2.1: Process performance management process attribute PA 2.2: Work product management process attribute PA 3.1: Process definition process attribute PA 3.2: Process deployment process attribute PA 4.1: Quantitative analysis process attribute PA 4.2: Quantitative control process attribute	Fully Fully Fully Fully Fully Largely Largely
Level 5	PA 1.1: Process performance process attribute PA 2.1: Process performance management process attribute PA 2.2: Work product management process attribute PA 3.1: Process definition process attribute PA 3.2: Process deployment process attribute PA 4.1: Quantitative analysis process attribute PA 4.2: Quantitative control process attribute PA 5.1: Process innovation process attribute PA 5.2: Process innovation implementation process attribute	Fully Fully Fully Fully Fully Fully Fully Fully Largely Largely

Table 20 — Process capability level model

3.3. Process assessment model

The process assessment model offers indicators in order to identify whether the process outcomes and the process attribute outcomes (achievements) are present or absent in the instantiated processes of projects and organizational units. These indicators provide guidance for assessors in accumulating the necessary objective evidence to support judgments of capability. They are not intended to be regarded as a mandatory set of checklists to be followed.

3.3.1. Assessment indicators

According to ISO/IEC 33004, a process assessment model needs to define a set of assessment indicators:

Assessment Indicators

A process assessment model shall be based on a set of assessment indicators that:

- a) explicitly address the purpose and process outcomes, as defined in the selected process reference model, of each of the processes within the scope of the process assessment model;*
- b) demonstrate the achievement of the process attributes within the scope of the process assessment model;*
- c) demonstrate the achievement (where relevant) of the process quality levels within the scope of the process assessment model.*

The assessment indicators generally fall into three types:

- a) **practices** that support achievement of either the process purpose or the specific process attribute.*
- b) **information items** and their characteristics that demonstrate the respective achievements.*
- c) **resources and infrastructure** that support the respective achievements.*

[ISO/IEC 33004:2015, 6.3.1]

In this assessment model, only practices and information items are used.

Practices are representing activity-oriented indicators, where Information items are representing result-oriented indicators. Both practices and information items are used for judging objective evidence to be collected and accumulated in the performance of an assessment.

As a first type of assessment indicator, practices are provided, which can be divided into two types:

1. Base practices (BP), applying to capability level 1

They provide an indication of the extent of achievement of the process outcomes. Base practices relate to one or more process outcomes, thus being always process-specific and not generic.

2. Generic practices (GP), applying to capability levels 2 to 5

They provide an indication of the extent of process attribute achievement. Generic practices relate to one or more process attribute achievements, thus applying to any process.

As a second type of assessment indicators, **information items (II)** including their **characteristics (IIC)** are provided in Annex B.

These are meant to offer a good practice and state-of-the-art knowledge guide for the assessor. Therefore, information items including their characteristics are supposed to be a quickly accessible information source during an assessment.

Information item characteristics shall not be interpreted as a required structure of a corresponding work products, which is defined by the project and organization, respectively.

Please refer to chapter 3.3.2 for understanding the difference between information items and work products.

ISO 33004:2015 requires the mapping of assessment indicators to process attributes as shown in figure 3.

The capability of a process on level 1 is only characterized by the measure of the extent to which the process outcomes are achieved. According to ISO 33003:2015, a measurement framework requires each level to reveal a process attribute. Therefore, the only process performance attribute for capability Level 1 (PA.1.1) has a single generic practice (GP 1.1.1) pointing as an editorial reference to the respective process performance indicators (see figure 3 and chapter 4).

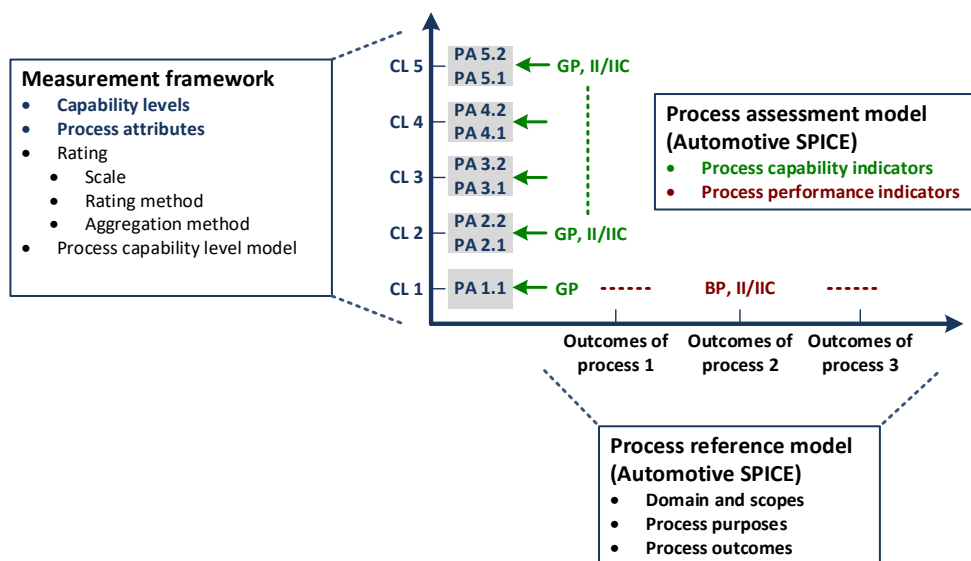


Figure 3 — Relationship between assessment indicators and process capability

The detailed mapping of base practices / indicators and generic practices / indicators to process outcomes and achievements, is provided in corresponding tables in chapter 4 and 5, respectively.

3.3.2. Understanding information items and work products

In order to judge the presence or absence of process outcomes and process achievements an assessment obtains objective evidence. All such evidence comes either from the examination of work products related to a specific output of the processes assessed, or from statements made by the performers and managers of the processes. Sources for such evidence is either repository content of the assessed processes, or testimony provided by the performers and managers of the assessed processes.

As described in chapter 3.3.1, this process assessment model provides information items serving as indicators to guide the assessor when judging a process attribute achievement.

3.3.2.1. Information items versus work products

ISO/IEC 33001 provides the following definition of the term “information item”:

information item

separately identifiable body of information that is produced, stored, and delivered for human use

Note 1 to entry: An information item can be produced in several versions during a system, software, or service life cycle. Syn: information product.

[ISO/IEC 33001:2015, 3.1.4]

Note: Human use includes the information stored, managed and processed by a tool.

One common definition of the term “work product” is:

work product

artifact resulting from the execution of a process

[Sevocab]

Both terms are used in different context in an assessment:

- Information items are defining relevant pieces of information used by the assessors to judge the achievement of process attributes.
- work product are produced by the organization assessed when performing, managing, establishing, analyzing and innovating processes.

Information items (together with their characteristics) are provided as guidance for “what to look for” when examining the work products available in the assessed organization. The extent of implementation of an information item (inline with its defined characteristics) in a related work product serves as objective evidence supporting the assessment of a particular process. A documented process and assessor judgment is needed to ensure that the process context (application domain, business purpose, development methodology, size of the organization, etc.) is considered when using this information.

Information items shall therefore not be mistaken for the work product generated by the assessed organization itself. There is no 1:1 relationship between an information item and the work product taken as sample evidence by the assessor when assessing the achievement of a process outcome. An output generated by a process may comprise multiple information item characteristics and multiple outputs may also contain the same information item characteristics.

Information item characteristics should be considered as indicators when considering whether, given the context, a work product is contributing to the intended purpose of the process.

3.3.2.2. Types of work products

A work product to be considered as evidence when rating a process attribute may not necessary be outputs from the processes assessed but can also be originated from other processes of the organization. Once such a work product is used in the performance of a process under assessment, it may be considered by the assessor as objective evidence.

In a lot of cases work products are comprising documentation aspects, such as specifications, reports, records, architectural designs, software code etc.

Examples of work products not comprising any documentation aspects are software binaries, raw data, or a physical electronic hardware.

3.3.3. Understanding the level of abstraction of a PAM

The term "process" can be understood at three levels of abstraction. Note that these levels of abstraction are not meant to define a strict black-or-white split, nor is it the aim to provide a scientific classification schema – the message here is to understand that, in practice, when it comes to the term "process" there are different abstraction levels, and that a PAM resides at the highest.

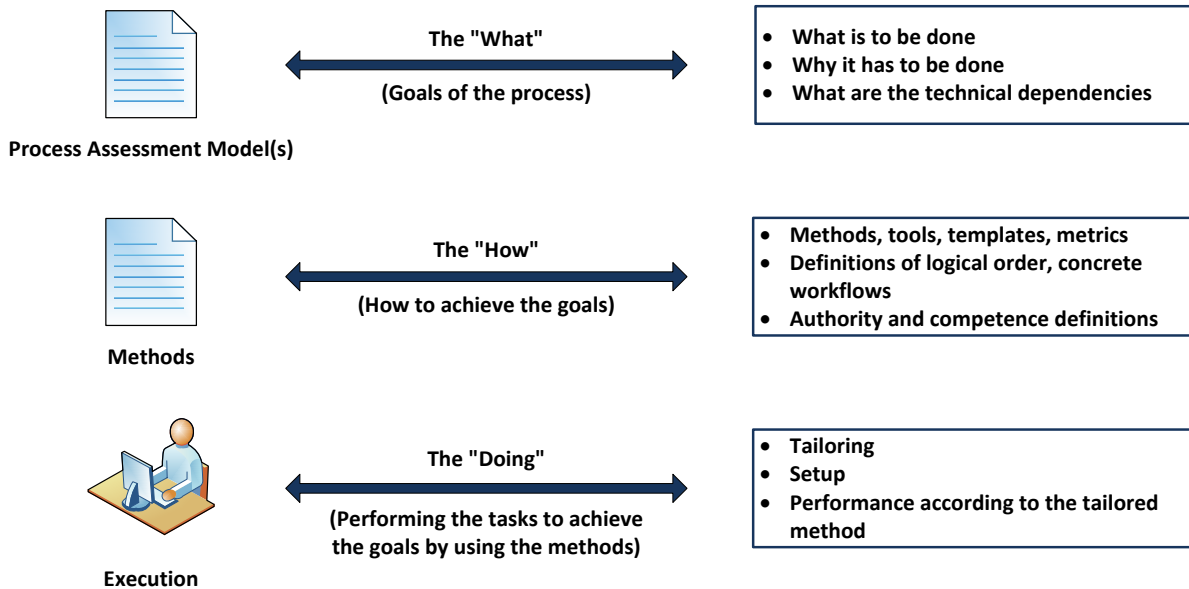


Figure 4 — Possible levels of abstraction for the term "process"

Capturing experience acquired during product development (i.e. at the DOING level) in order to share this experience with others means creating a HOW level. However, a HOW is always specific to a particular context such as a company, an organizational unit, or a product line. For example, the HOW of a project, organizational unit, or company A is potentially not applicable as is to a project, organizational unit, or company B. However, both might be expected to adhere the principles represented by PAM indicators for process outcomes and process attribute achievements. These indicators are at the WHAT level while deciding on solutions for concrete templates, proceedings, and tooling etc. is left to the HOW level.

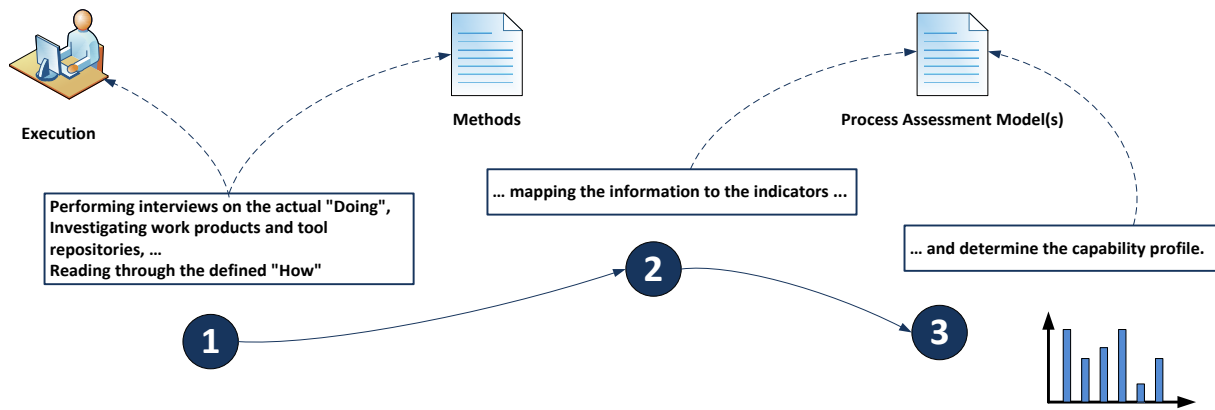


Figure 5 — Performing a process assessment for determining process capability

3.3.4. Why a PRM and PAM are not a lifecycle model

A lifecycle model defines phases and activities in a logical timely order, possibly including cycles or loops, and parallelization. For example, some standards such as ISO 26262 or ISO/SAE 21434 are centered around a lifecycle model (neither of these standards in fact represents a PRM according to ISO/IEC 33004). Companies, organizational units, or projects will interpret a lifecycle model and detail it out into roles, organizational interactions and interfaces, tools or tool chains, work instructions, and artifacts. Lifecycle models therefore are a concept at the HOW level (see Section 0).

In contrast, a PRM/PAM according to ISO/IEC 33004 (formerly ISO/IEC 15504-2) is at the level of the WHAT by abstracting from any HOW level, see Figure 4 in Section 0. In Automotive SPICE® v2.5, v3.0 and v3.1, this is indicated by the process MAN.3 Project Management requiring in BP2 "Define project life cycle". A PRM/PAM groups a set of coherent and related characteristics of a particular technical topic and calls it 'process'; in different terms, a process in a PRM represents a 'distinct conceptual silo'. In this respect, a PRM/PAM neither predefines, nor discourages, any order in which PRM processes, or Base Practices, are to be performed; ultimately, in Automotive SPICE consistency must be fulfilled as required by the traceability/consistency Base Practices in MAN.3 or SYS.x, SWE.x, and HWE.X.

As a consequence, it is the assessor's responsibility to perform a mapping of elements in such a HOW level to the Assessment Indicators in the PAM, see Figure 5 Section 0.

In this respect, a PRM or PAM further is not supposed to represent a product element hierarchy either.

4. Process reference model and performance indicators (Level 1)

The processes in the process dimension can be drawn from the Automotive SPICE process reference model, which is incorporated in the tables below indicated by a red bar at the left side.

Each table related to one process in the process dimension contains the process reference model (indicated by a red bar) and the process performance indicators necessary to define the process assessment model. The process performance indicators consist of base practices (indicated by a green bar) and output information items (indicated by a blue bar).

Process reference model	<p>Process ID</p> <p>Process name</p> <p>Process purpose</p> <p>Process outcomes</p>	<p>The individual processes are identified with a unique process identifier and a process name. A process purpose statement is provided, and process outcomes are defined to represent the process dimension of the Automotive SPICE process reference model. The background coloring of process ID's and names are indicating the assignment to the corresponding process group.</p>
	Process performance indicators	<p>Base practices</p>
<p>Output information items</p>		<p>The output information items that are relevant to accomplish the process purpose and fulfill the process outcomes summarized at the end of a process to demonstrate their relationship to the process outcomes.</p> <p><i>Note: Refer to Annex B for the characteristics of each information item.</i></p>

Table 21 — Template for the process description

4.1. Acquisition process group (ACQ)

4.1.1. ACQ.4 Supplier Monitoring

Process ID
ACQ.4
Process name
Supplier Monitoring
Process purpose
The purpose is to track and assess the performance of an external contract-based supplier company against agreed commitments.
Process outcomes
<ol style="list-style-type: none"> 1) joint activities, as agreed between the customer and the supplier, are performed 2) all information, agreed upon for exchange, is communicated regularly between the customer and the supplier 3) performance of the supplier is monitored against the agreements 4) changes to the agreement, if needed, are negotiated between the customer and the supplier and documented in the agreement
Base practices
ACQ.4.BP1: Agree on and maintain joint activities, joint interfaces, and information to be exchanged. Establish and maintain an agreement on information to be exchanged, on joint activities, joint interfaces, responsibilities, type and frequency of joint activities, communications, meetings, status reports, and reviews.
ACQ.4.BP2: Exchange all agreed information. Use the defined joint interfaces between customer and supplier for the exchange of all agreed information.
ACQ.4.BP3: Review development work products with the supplier. Review development work products with the supplier on the agreed regular basis, covering technical aspects, problems and risks. Track open measures. <i>Note 1: see SUP.9 for management of problems</i>
ACQ.4.BP4: Review progress of the supplier. Review progress of the supplier regarding schedule, quality, and cost on the agreed regular basis. Track open measures to closure and perform risk mitigation activities. <i>Note 2: see MAN.5 for management of risks</i>
ACQ.4.BP5: Act to correct deviations. Take action when agreed objectives are not achieved. Negotiate changes to objectives and document them in the agreements.

ACQ.4 – Supplier Monitoring	Outcome 1	Outcome 2	Outcome 3	Outcome 4
Output Information items				
02-01 Commitment/Agreement	X	X	X	X
13-01 Acceptance record			X	
13-52 Communication evidence	X	X		
13-09 Meeting support record	X	X		
13-14 Progress status record		X	X	
13-16 Change request				X
13-19 Review record		X		
14-02 Corrective action				X
15-01 Analysis report			X	
Base Practices				
BP1: Agree on and maintain joint processes, joint interfaces, and information to be exchanged	X	X		X
BP2: Exchange all agreed information	X	X	X	
BP3: Review development work products with the supplier	X		X	X
BP4: Review progress of the supplier	X		X	X
BP5: Act to correct deviations			X	X

4.2. Supply process group (SPL)

4.2.1. SPL.2 Product Release

Process ID
SPL.2
Process name
Product Release
Process purpose
The purpose is to control the release of a product to the intended customer.
Process outcomes
<ol style="list-style-type: none"> 1) The contents of the product releases are determined. 2) The release package is assembled from configured items. 3) The release documentation is defined and produced. 4) Release approval is performed against defined criteria. 5) The release package is made available to the intended customer.

Base practices
<p>SPL.2.BP1: Define the functional content of releases. Define the functionality to be included and the release criteria for each release.</p> <p><i>Note 1: This may include the hardware elements, software elements, and application parameters (influencing the identified system functionality) that are needed for which release.</i></p>
<p>SPL.2.BP2: Define release package. Define the release as well as supporting tools and information.</p> <p><i>Note 2: The release package may include also programming tools.</i></p>
<p>SPL.2.BP3: Ensure unique identification of releases. Ensure a unique identification of the release based upon the intended purpose and expectations of the release.</p> <p><i>Note 3: Unique identification may be realized by a classification and numbering scheme for product releases.</i></p>
<p>SPL.2.BP4: Build the release from items under configuration control. Build the release from items under configuration control to ensure integrity.</p> <p><i>Note 4: This practice may be supported by the SUP.8 Configuration Management Process.</i></p>
<p>SPL.2.BP5: Ensure release approval before delivery. Criteria for the release are satisfied before delivery takes place.</p>

SPL.2.BP6: Provide a release note. A release is accompanied by information detailing key characteristics of the release.

Note 5: The release note may include information about legal aspects like relevant target markets, legislation that is considered etc. See also VAL.1.

SPL.2.BP7: Communicate the type, service level and duration of support for a release. Identify and communicate the type, service level and duration of support for a release.

SPL.2.BP8: Deliver the release package to the intended customer. Deliver the release package to the intended customer.

Note 6: The intended customer may be an internal organizational unit or an external organization.

SPL.2 – Product Release	Outcome 1	Outcome 2	Outcome 3	Outcome 4	Outcome 5
Output Information items					
11-03 Release note	X		X	X	X
11-04 Product release package		X	X		
13-06 Delivery evidence			X		X
13-13 Product release approval				X	X
18-06 Product release criteria	X	X		X	
Base Practices					
BP1: Define the functional content of releases	X				
BP2: Define release package	X				
BP3: Establish a product release classification and numbering scheme			X		
BP4: Build the release from configured items		X			
BP5: Ensure product release approval before delivery				X	
BP6: Provide a release note			X		X
BP7: Communicate the type, service level and duration of support for a release			X		X
BP8: Deliver the release package to the intended customer					X

4.3. System engineering process group (SYS)

4.3.1. SYS.1 Requirements Elicitation

Process ID
SYS.1
Process name
Requirements Elicitation
Process purpose
The purpose is to gather, analyze, and track evolving stakeholder needs and requirements throughout the lifecycle of the product and/or service to establish a set of agreed requirements.
Process outcomes
<ol style="list-style-type: none"> 1) Continuing communication with the stakeholder is established. 2) Stakeholder expectations are understood and requirements are defined and agreed. 3) Stakeholder requirements changes arising from stakeholder needs are analyzed to enable associated risk assessment and impact management. 4) Determination of stakeholder requirements status is ensured for all affected parties.
Base practices
<p>SYS.1.BP1: Obtain stakeholder expectations and requests. Obtain and define stakeholder expectations and requests through direct solicitation of stakeholder input, and through review of stakeholder business proposals (where relevant) and other documents containing inputs to stakeholder requirements, and consideration of the target operating and hardware environment.</p> <p><i>Note 1: Documenting the stakeholder, or the source of a stakeholder requirement, supports stakeholder requirements agreement and change analysis (see BP3 and BP4).</i></p>
<p>SYS.1.BP2: Understand stakeholder expectations. Ensure that all affected parties understand each requirement in the same way.</p> <p><i>Note 2: Examples of affected parties are customers, suppliers, design partners, joint venture partners, or outsourcing parties.</i></p>
<p>SYS.1.BP3: Agree on requirements. Formalize the stakeholder’s expectations and requests into requirements and obtain an explicit agreement from all affected parties to work on these requirements.</p> <p><i>Note 3: The agreed stakeholder requirements may be based on feasibility studies and/or cost and schedule impact analysis.</i></p>

SYS.1.BP4: Analyze stakeholder requirements changes. Analyze and manage all changes made to the stakeholder requirements against the agreed stakeholder requirements. Assess the impact and risks and initiate appropriate change control and mitigation actions.

Note 4: Requirements changes may arise from different sources as for instance changing technology, stakeholder needs, or legal constraints.

Note 5: Refer to SUP.10 Change Request Management, if required.

SYS.1.BP5: Communicate requirements status. Ensure all affected parties can be aware of the status and disposition of their requirements including changes and can communicate necessary information and data.

SYS.1 Requirements Elicitation	Outcome 1	Outcome 2	Outcome 3	Outcome 4
Output Information Items				
15-51 Analysis Results			X	
13-52 Communication Evidence	X	X		
17-00 Requirement		X		
17-54 Requirement Attribute		X	X	X
Base Practices				
BP1: Obtain stakeholder expectations and requests	X			
BP2: Understand stakeholder expectations		X		
BP3: Agree on requirements		X		
BP4: Analyze stakeholder requirements changes			X	
BP5: Communicate requirements status	X			X

4.3.2. SYS.2 System Requirements Analysis

Process ID
SYS.2
Process name
System Requirements Analysis
Process purpose
The purpose is to establish a structured and analyzed set of system requirements consistent with the stakeholder requirements.
Process outcomes
<ol style="list-style-type: none"> 1) System requirements are specified 2) System requirements are structured and prioritized 3) System requirements are analyzed for correctness and technical feasibility 4) The impact of system requirements on the operating environment is analyzed
Base practices
<p>SYS.2.BP1: Specify system requirements. Use the stakeholder requirements to identify and document the functional and non-functional requirements for the system according to defined characteristics for requirements.</p> <p><i>Note 1: Characteristics of requirements are defined in standards such as ISO IEEE 29148, ISO 26262-8:2018, or the INCOSE Guide For Writing Requirements.</i></p> <p><i>Note 2: Examples for defined characteristics of requirements shared by technical standards are verifiability (i.e. verification criteria being inherent in the requirements formulation), unambiguity/comprehensibility, freedom from design and implementation, and not contradicting any other requirement.</i></p>
<p>SYS.2.BP2: Structure system requirements. Structure and prioritize the system requirements.</p> <p><i>Note 3: Examples for structuring criteria can be grouping (e.g. by functionality) or product variants identification.</i></p> <p><i>Note 4: Prioritization can be done according to project or stakeholder needs via e.g. definition of release scopes. Refer to SPL.2.BP1</i></p>
<p>SYS.2.BP3: Analyze system requirements. Analyze the specified system requirements including their interdependencies to ensure correctness, technical feasibility, and to support project management regarding project estimates.</p> <p><i>Note 5: See MAN.3.BP3 for project feasibility and MAN.3.BP5 for project estimates.</i></p> <p><i>Note 6: Technical feasibility can be evaluated based on e.g. platform or product line, or by means of prototype development or product demonstrators.</i></p>
<p>SYS.2.BP4: Analyze the impact on the system context. Analyze the impact that the system requirements will have on elements in the relevant system context.</p>

SYS.2.BP5: Ensure consistency and establish bidirectional traceability. Ensure consistency and establish bidirectional traceability between system requirements and stakeholder requirements.

Note 7: Bidirectional traceability supports consistency, facilitates impact analyses of change requests, and supports the demonstration of coverage of stakeholder requirements.

SYS.2.BP6: Communicate agreed system requirements and impact on the system context. Communicate the agreed system requirements, and results of the impact analysis on the system context, to all affected parties.

SYS.2 System Requirements Analysis	Outcome 1	Outcome 2	Outcome 3	Outcome 4	Outcome 5	Outcome 6
Output Information Items						
17-00 Requirement	X	X				
17-54 Requirement Attribute		X	X			
15-51 Analysis Results			X	X		
13-51 Consistency Evidence					X	
13-52 Communication Evidence						X
Base Practices						
BP1: Specify system requirements	X					
BP2: Structure system requirements		X				
BP3: Analyze system requirements			X			
BP4: Analyze the impact on the system context				X		
BP5: Ensure consistency and establish bidirectional traceability					X	
BP6: Communicate agreed system requirements and impact on the system context						X

4.3.3. SYS.3 System Architectural Design

Process ID
SYS.3
Process name
System Architectural Design
Process purpose
The purpose is to establish an analyzed system architecture consistent with the system requirements.
Process outcomes
<ol style="list-style-type: none"> 1) A system architecture is designed including a definition of the system elements with their behavior, their interfaces, their relationships, and their interactions. 2) The system architecture is analyzed against defined criteria, and special characteristics are identified. 3) Consistency and bidirectional traceability are established between system architecture and system requirements. 4) The agreed system architecture and the special characteristics are communicated to all affected parties.
Base practices
<p>SYS.3.BP1: Specify static aspects of the system architecture. Specify and document the static aspects of the system architecture with respect to the functional and nonfunctional system requirements, including external interfaces and a defined set of system elements with their interfaces and relationships.</p>
<p>SYS.3.BP2: Specify dynamic aspects of the system architecture. Specify and document the dynamic aspects of the system architecture with respect to the functional and nonfunctional system requirements including the behavior of the system elements and their interaction in different system modes.</p> <p><i>Note 1: Examples of interactions of system elements are timing diagrams reflecting inertia of mechanical components, processing times of ECUs, and signal propagation times of bus systems.</i></p>

SYS.3.BP3: Analyze system architecture. Analyze the system architecture regarding relevant technical design aspects related to the product lifecycle, and to support project management regarding project estimates, and derive Special Characteristics for hardware elements. Document a rationale for the system architectural design decision.

Note 2: See MAN.3.BP3 for project feasibility and MAN.3.BP5 for project estimates.

Note 3: Examples for product lifecycle phases are production, maintenance & repair, decommissioning.

Note 4: Examples for technical aspects are manufacturability for production, suitability of pre-existing system elements to be reused, or availability of system elements.

Note 5: Examples for methods being suitable for analyzing technical aspects are prototypes, simulations, and qualitative analyses (e.g. FMEA approaches)

Note 6: Examples of design rationales are proven-in-use, reuse of a product platform or product line), a make-or-buy decision, or found in an evolutionary way (e.g. set-based design).

SYS.3.BP4: Ensure consistency and establish bidirectional traceability. Ensure consistency and establish bidirectional traceability between the elements of the system architecture and the system requirements that represent properties or characteristics of the physical end product.

Note 7: Bidirectional traceability further supports consistency, and facilitates impact analysis of change requests, and demonstration of verification coverage.

Note 8: There may be non-functional system requirements that the system architectural design does not trace to. Examples are development process requirements. Such requirements are still subject to verification.

SYS.3.BP5: Communicate agreed system architecture. Communicate the agreed system architecture, including the special characteristics, to all affected parties.

SYS.3 System Architectural Design	Outcome 1	Outcome 2	Outcome 3	Outcome 4
Output Information Items				
04-06 System Architecture	X			
13-51 Consistency Evidence			X	
13-52 Communication Evidence				X
15-51 Analysis Results		X		
17-57 Special Characteristics		X		
Base Practices				
BP1: Specify static aspects of system architecture	X			
BP2: Specify dynamic aspects of system architecture	X			
BP3: Analyze the system architecture		X		
BP4: Ensure consistency and establish bidirectional traceability			X	
BP5: Communicate agreed system architecture				X

4.3.4. SYS.4 System Integration and Integration Verification

Process ID
SYS.4
Process name
System Integration and Integration Verification
Process purpose
The purpose is to integrate systems elements and verify that the integrated system elements are consistent with the system architecture.
Process outcomes
<ol style="list-style-type: none"> 1) Verification measures are specified for system integration verification of the integrated system elements based on the system architecture, including the interfaces of, and interactions between, system elements. 2) System elements are integrated up to a complete integrated system consistent with the release scope. 3) Verification measures are selected according to the release scope considering criteria for regression verification. 4) Integrated system elements are verified using the selected verification measures, and the results of the system integration verification are recorded. 5) Consistency and bidirectional traceability are established between verification measures and the elements of the system architecture. 6) Bidirectional traceability between verification results and verification measures is established. 7) Results of the system integration and integration verification are summarized and communicated to all affected parties.

Base practices

SYS.4.BP1: Specify verification measures for system integration. Specify the verification measures, based on a defined order and preconditions for the integration of system elements against the system static and dynamic aspects of the system architecture, including

- techniques for the verification measures;
- pass/fail criteria for verification measures;
- a definition of entry and exit criteria for the verification measures;
- the required verification infrastructure and environment setup.

Note 1: Examples on what a verification measure may focus are the timeliness and timing dependencies of the correct signal flow between interfacing system elements, or interactions between hardware and software.

The system integration test cases may focus on

- *the correct signal flow between system items*
- *the timeliness and timing dependencies of signal flow between system items*
- *the correct interpretation of signals by all system items using an interface*
- *the dynamic interaction between system items*

SYS.4.BP2: Select verification measures. Document the selection of verification measures for each integration step considering selection criteria including criteria for regression verification. The documented selection of verification measures shall have sufficient coverage according to the release scope.

Note 2: Examples for selection criteria can be prioritization of requirements, the need for regression verification (due to e.g. changes to the system architectural design or to system components), or the intended use of the delivered product release (e.g. test bench, test track, public road etc.)

SYS.4.BP3: Integrate system elements and perform integration verification. Integrate the system elements until the system is fully integrated according to the specified interfaces and interactions between the system elements, and according to the defined order and defined preconditions. Perform the selected system integration verification measures. Record the verification measure data including pass/fail status and corresponding verification measure data.

Note 3: Examples for preconditions for starting system integration can be successful system element verification or qualification of pre-existing system elements.

Note 4: See SUP.9 for handling verification results that deviate from expected results

SYS.4.BP4: Ensure consistency and establish bidirectional traceability. Ensure consistency and establish bidirectional traceability between verification measures and the system architecture. Establish bidirectional traceability between verification results and verification measures.

Note 5: Bidirectional traceability supports consistency, and facilitates impact analysis of change requests, and demonstration of verification coverage.

SYS.4.BP5: Summarize and communicate results. Summarize the system integration and integration verification results and communicate them to all affected parties.

Note 6: Providing all necessary information from the test case execution in a summary enables other parties to judge the consequences.

SYS.4 System Integration and Integration Verification	Outcome 1	Outcome 2	Outcome 3	Outcome 4	Outcome 5	Outcome 6	Outcome 7
Output Information Items							
08-60 Verification Measure	X						
06-50 Integration Sequence Instruction		X					
08-58 Verification Measure Selection Set			X				
15-52 Verification Results				X			
13-51 Consistency Evidence					X	X	
13-52 Communication Evidence							X
11-06 Integrated System		X					
Base Practices							
BP1: Specify verification measures for system integration	X						
BP2: Select verification measures			X				
BP3: Integrate system elements and perform integration verification.		X		X			
BP4: Ensure consistency and establish bidirectional traceability					X	X	
BP5: Summarize and communicate results							X

4.3.5. SYS.5 System Verification

Process ID
SYS.5
Process name
System Verification
Process purpose
The purpose is to ensure that the system is verified to be consistent with the system requirements.
Process outcomes
<ol style="list-style-type: none"> 1) Verification measures are specified for system verification of the system based on the system requirements. 2) Verification measures are selected according to the release scope considering criteria for regression verification. 3) The integrated system is verified using the selected verification measures and the results of system verification are recorded. 4) Consistency and bidirectional traceability are established between verification measures and system requirements. 5) Bidirectional traceability is established between verification results and verification measures. 6) Verification results are summarized and communicated to all affected parties.
Base practices
<p>SYS.5.BP1: Specify verification measures for system verification. Specify the verification measures for system verification suitable to provide evidence for compliance with the functional and non-functional information in the system requirements, including</p> <ul style="list-style-type: none"> • techniques for the verification measures; • pass/fail criteria for verification measures; • a definition of entry and exit criteria for the verification measures; • necessary sequence of verification measures; • the required verification infrastructure and environment setup. <p><i>Note 1: The system verification measures may cover aspects such as thermal, environmental, robustness/lifetime, and EMC.</i></p>
<p>SYS.5.BP2: Select verification measures. Document the selection of verification measures considering selection criteria including criteria for regression verification. The selection of verification measures shall have sufficient coverage according to the release scope.</p> <p><i>Note 2: Examples for criteria for selection can be prioritization of requirements, the need for regression verification (due to e.g. changes to the system requirements), the intended use of the delivered product release (test bench, test track, public road etc.)</i></p>

SYS.5.BP3: Perform verification of the integrated system. Perform the verification of the integrated system using the selected verification measures. Record the verification results including pass/fail status and corresponding verification measure data.

Note 3: See SUP.9 for handling verification results that deviate from expected results

SYS.5.BP4: Ensure consistency and establish bidirectional traceability. Ensure consistency and establish bidirectional traceability between verification measures and system requirements. Establish bidirectional traceability between verification results and verification measures.

Note 4: Bidirectional traceability supports consistency, and facilitates impact analysis of change requests, and demonstration of verification coverage

SYS.5.BP5: Summarize and communicate results. Summarize the system verification results and communicate them to all affected parties.

Note 5: Providing all necessary information from the test case execution in a summary enables other parties to judge the consequences.

SYS.5 System Verification	Outcome 1	Outcome 2	Outcome 3	Outcome 4	Outcome 5	Outcome 6
Output Information item						
08-60 Verification Measure	X					
08-58 Verification Measure Selection Set		X				
15-52 Verification Results			X			
13-51 Consistency Evidence				X	X	
13-52 Communication Evidence						X
Base Practices						
BP1: Specify verification measures for system verification	X					
BP2: Select verification measures		X				
BP3: Perform verification of the integrated system			X			
BP4: Ensure consistency and establish bidirectional traceability.				X	X	
BP5: Summarize and communicate results						X

4.4. Software engineering process group (SWE)

4.4.1. SWE.1 Software Requirements Analysis

Process ID
SWE.1
Process name
Software Requirements Analysis
Process purpose
The purpose is to establish a structured and analyzed set of software requirements consistent with the system requirements, and the system architecture.
Process outcomes
<ol style="list-style-type: none"> 1) Software requirements are specified. 2) Software requirements are structured and prioritized. 3) Software requirements are analyzed for correctness and technical feasibility. 4) The impact of software requirements on the operating environment is analyzed. 5) Consistency and bidirectional traceability are established between software requirements and system requirements. 6) Consistency and bidirectional traceability are established between software requirements and system architecture. 7) The software requirements are agreed and communicated to all affected parties.
Base practices
<p>SWE.1.BP1: Specify software requirements. Use the system requirements and the system architecture to identify and document the functional and non-functional requirements for the software according to defined characteristics for requirements.</p> <p><i>NOTE 1: Characteristics of requirements are defined in standards such as ISO IEEE 29148, ISO 26262-8:2018, or the INCOSE Guide for Writing Requirements.</i></p> <p><i>Note 2: Examples for defined characteristics of requirements shared by technical standards are verifiability (i.e. verification criteria being inherent in the requirements formulation), unambiguity/comprehensibility, freedom from design and implementation, and not contradicting any other requirement.</i></p> <p><i>Note 3: In case of software-only development, the system requirements and the system architecture refer to a given operating environment. In that case, stakeholder requirements can be used as the basis for identifying the required functions and capabilities of the software.</i></p> <p><i>Note 4: The hardware-software-interface (HSI) definition puts in context hardware and therefore is an interface decision at the system design level (see SYS.3). If such a HSI exists, then it may provide input to software requirements.</i></p>

SWE.1.BP2: Structure software requirements. Structure and prioritize the software requirements.

Note 5: Examples for structuring criteria can be grouping (e.g. by functionality) or variants identification.

Note 6: Prioritization can be done according to project or stakeholder needs via e.g. definition of release scopes. Refer to SPL.2.BP1.

SWE.1.BP3: Analyze software requirements. Analyze the specified software requirements including their interdependencies to ensure correctness, technical feasibility, and to support project management regarding project estimates.

Note 7: See MAN.3.BP3 for project feasibility and MAN.3.BP5 for project estimates.

Note 8: Technical feasibility can be evaluated based on e.g. platform or product line, or by prototyping.

SWE.1.BP4: Analyze the impact on the operating environment. Analyze the impact that the software requirements will have on elements in the operating environment.

SWE.1.BP5: Ensure consistency and establish bidirectional traceability. Ensure consistency and establish bidirectional traceability between software requirements and system architecture. Ensure consistency and establish bidirectional traceability between software requirements and system requirements.

Note 9: Redundant traceability is not intended.

Note 10: There may be non-functional system requirements that the software requirements do not trace to. Examples are process requirements or requirements related to later software product lifecycle phases such as incident handling. Such requirements are still subject to verification.

Note 11: Bidirectional traceability supports consistency, and facilitates impact analysis of change requests, and demonstration of verification coverage

Note 12: In case of software development only, the system requirements and system architecture refer to a given operating environment. In that case, consistency and bidirectional traceability can be ensured between stakeholder requirements and software requirements.

SWE.1.BP6: Communicate agreed software requirements and impact on the operating environment. Communicate the agreed software requirements, and the results of the analysis of impact on the operating environment, to all affected parties.

SWE.1 Software Requirements Analysis	Outcome 1	Outcome 2	Outcome 3	Outcome 4	Outcome 5	Outcome 6	Outcome 7
Output Information Items							
17-00 Requirement	X	X					
17-54 Requirement Attribute		X					
15-51 Analysis Results			X	X			
13-51 Consistency Evidence					X	X	
13-52 Communication Evidence							X
Base Practices							
BP1: Specify software requirements	X						

BP2: Structure software requirements		X					
BP3: Analyze software requirements			X				
BP4: Analyze the impact on the operating environment				X			
BP5: Ensure consistency and establish bidirectional traceability					X	X	
BP6: Communicate agreed software requirements and impact on the operating environment							X

4.4.2. SWE.2 Software Architectural Design

Process ID
SWE.2
Process name
Software Architectural Design
Process purpose
The purpose is to establish an analyzed software architecture consistent with the software requirements.
Process outcomes
<ol style="list-style-type: none"> 1) A software architecture is designed including static and dynamic aspects. 2) The software architecture is analyzed against defined criteria. 3) Consistency and bidirectional traceability are established between software architecture and software requirements. 4) The software architecture is agreed and communicated to all affected parties.
Base practices
<p>SWE.2.BP1: Specify static aspects of the software architecture. Specify and document the static aspects of the software architecture with respect to the functional and non-functional software requirements, including external interfaces and a defined set of components with their interfaces and relationships.</p> <p><i>Note 1: The hardware-software-interface (HIS) definition puts in context the hardware design and therefore is an aspect of system design (SYS.3).</i></p>
<p>SWE.2.BP2: Specify dynamic aspects of the software architecture. Specify and document the dynamic aspects of the software architecture with respect to the functional and non-functional software requirements, including the behavior of the components and their interaction in different system modes, and concurrency aspects.</p> <p><i>Note 2: Examples for concurrency aspects are application-relevant interrupt handling, preemptive processing, multi-threading.</i></p> <p><i>Note 3: Examples for behavioral descriptions are natural language or semi-formal notation (e.g. SysML, UML).</i></p>

SWE.2.BP3: Analyze software architecture. Analyze the software architecture regarding relevant technical design aspects and to support project management regarding project estimates. Document a rationale for the software architectural design decision.

Note 4: See MAN.3.BP3 for project feasibility and MAN.3.BP5 for project estimates.

Note 5: The analysis may include the suitability of pre-existing software components for the current application.

Note 6: Examples of methods suitable for analyzing technical aspects are prototypes, simulations, qualitative analyses.

Note 7: Examples of technical aspects are functionality, timings, and resource consumption (e.g. (ROM, RAM, external / internal EEPROM or Data Flash or CPU load).

Note 8: Design rationales can include arguments such as proven-in-use, reuse of a software framework or software product line, a make-or-buy decision, or found in an evolutionary way (e.g. set-based design).

SWE.2.BP4: Ensure consistency and establish bidirectional traceability. Ensure consistency and establish bidirectional traceability between the software architecture and the software requirements.

Note 9: There may be non-functional software requirements that the software architectural design does not trace to. Examples are development process requirements. Such requirements are still subject to verification.

Note 10: Bidirectional traceability supports consistency, and facilitates impact analysis of change requests, and demonstration of verification coverage

SWE.2.BP5: Communicate agreed software architecture. Communicate the agreed software architecture to all affected parties.

SWE.2 Software Architectural Design	Outcome 1	Outcome 2	Outcome 3	Outcome 4
Output Information items				
04-04 Software Architecture	X			
13-51 Consistency Evidence			X	
13-52 Communication Evidence				X
15-51 Analysis Results		X		
05-01 Resource Consumption Targets		X		
Base Practices				
BP1: Specify static aspects of software architecture	X			
BP2: Specify dynamic aspects of software architecture	X			
BP3: Analyze software architecture		X		
BP4: Ensure consistency and establish bidirectional traceability			X	
BP5: Communicate agreed software architecture				X

4.4.3. SWE.3 Software Detailed Design and Unit Construction

Process ID
SWE.3
Process name
Software Detailed Design and Unit Construction
Process purpose
The purpose is to establish a software detailed design the software architecture, and to construct software units consistent with the software detailed design.
Process outcomes
<ol style="list-style-type: none"> 1) A detailed design is specified including static and dynamic aspects 2) Software units defined by the software detailed design are produced 3) Consistency and bidirectional traceability are established between software detailed design and software architecture; and consistency and bidirectional traceability are established between software units and software detailed design 4) The software detailed design is agreed and communicated to all affected parties

Base practices
<p>SWE.3.BP1: Specify the static aspects of the detailed design. For each software component specify the behavior of its software units, their static structure and relationships, their interfaces including</p> <ul style="list-style-type: none"> • valid data value ranges for inputs and outputs (from the application domain perspective) • physical or measurement units applicable to inputs and outputs (from the application domain perspective) <p><i>Note 1: The boundary of a software unit is independent from the software unit's representation in the source code, code file structure, or model-based implementation, respectively. It is rather driven by the semantics of the application domain perspective. Therefore, a software unit may be, at the code level, represented by a single subroutine or a set of subroutines.</i></p> <p><i>Note 2: Examples of valid data value ranges with applicable physical units from the application domain perspective are '0..200 [m/s]', '0.. 3.8 [A]' or '1..100 [N]'. For mapping such application domain value ranges to programming language-level data types (such as unsigned Integer with a value range of 0..65535) refer to BP2.</i></p> <p><i>Note 3: Examples of a measurement unit are '%' or '‰'.</i></p> <p><i>Note 4: A counter is an example of a parameter, or a return value, to which neither a physical nor a measurement unit is applicable.</i></p> <p><i>Note 5: The hardware-software-interface (HSI) definition puts in context the hardware design and therefore is an aspect of system design (SYS.3).</i></p>

SWE.3.BP2: Specify the dynamic aspects of the detailed design. Specify and document the dynamic aspects of the detailed design with respect to the software architecture, including the interactions between relevant software units to fulfill the component's dynamic behavior.

Note 6: Examples for behavioral descriptions are natural language or semi-formal notation (e.g. SysML, UML).

SWE.3.BP3: Develop software units. Develop and document the software units consistent with the detailed design, and according to coding principles.

Note 7: Examples for coding principles at Capability Level 1 are not to use implicit type conversions, only one entry and one exit point in subroutines, and range checks (design-by-contract, defensive programming). Further examples see e.g. ISO 26262-6 clause 8.4.5 together with table 6.

SWE3.BP4: Ensure consistency and establish bidirectional traceability. Ensure consistency and establish bidirectional traceability between the software detailed design and the software architecture. Ensure consistency and establish bidirectional traceability between the developed software units and the software detailed design. Ensure consistency and establish traceability between the software detailed design and the software requirements.

Note 8: Redundancy should be avoided by establishing a combination of these approaches.

Note 9: Examples for tracing a software unit to a software requirement directly are communication matrices or basis software aspects such as list of diagnosis identifiers inherent in an Autosar configuration.

Note 10: Bidirectional traceability supports consistency, and facilitates impact analysis of change requests, and demonstration of verification coverage.

SWE.3.BP5: Communicate agreed software detailed design and software units. Communicate the agreed software detailed design and software units to all affected parties.

SWE.3 Software Detailed Design and Unit Construction	Outcome 1	Outcome 2	Outcome 3	Outcome 4
Output Information Items				
04-05 Software Detailed Design	X			
11-05 Software Unit	X	X		
13-51 Consistency Evidence			X	
13-52 Communication Evidence				X
Base Practices				
BP1: Specify the static aspects of the detailed design	X			
BP2: Specify the dynamic aspects of the detailed design	X			
BP3: Develop software units		X		
BP4: Ensure consistency and establish bidirectional traceability			X	
BP5: Communicate agreed software detailed design and software units				X

4.4.4. SWE.4 Software Unit Verification

Process ID
SWE.4
Process name
Software Unit Verification
Process purpose
The purpose is to verify that software units are consistent with the software detailed design.
Process outcomes
<ol style="list-style-type: none"> 1) Verification measures for software unit verification are specified 2) Software units verification measures are selected according to the release scope considering criteria for regression verification 3) Software units are verified using the selected verification measures, and results are recorded 4) Consistency and bidirectional traceability are established between verification measures and software units; and consistency and bidirectional traceability are established between verification results and verification measures 5) Results of the software unit verification are summarized and communicated to all affected parties

Base practices
<p>SWE.4.BP1: Specify software unit verification measures. Specify verification measures for each software unit defined in the software detailed design. Define pass/fail criteria for verification measures and a suitable verification environment, including a definition of entry and exit criteria for verification measures.</p> <p><i>Note 1: Examples for unit verification measures are static analysis, code reviews, and unit testing.</i></p> <p><i>Note 2: Examples for criteria to decide on the necessity of unit testing as a verification measure are thresholds of code metrics.</i></p> <p><i>Note 3: Static analysis can be done based on MISRA rulesets and other coding standards.</i></p>
<p>SWE.4.BP2: Select software unit verification measures. Document the selection of verification measures considering selection criteria including criteria for regression verification. The documented selection of verification measures shall have sufficient coverage according to the release scope.</p>
<p>SWE.4.BP3: Perform software unit verification. Perform software unit verification using the selected verification measures. Record the verification results including pass/fail status and corresponding verification measure data.</p> <p><i>Note 4: See SUP.9 for handling of verification results that deviate from expected results.</i></p>

SWE.4.BP4: Ensure consistency and establish bidirectional traceability. Ensure consistency and establish bidirectional traceability between verification measures and the software units defined in the detailed design. Establish bidirectional traceability between the verification results and the verification measures.

Note 5: Bidirectional traceability supports consistency, and facilitates impact analysis of change requests, and demonstration of verification coverage.

SWE.4.BP5: Summarize and communicate results. Summarize the results of software unit verification and communicate them to all affected parties.

Note 6: Providing all necessary information from the test case execution in a summary enables other parties to judge the consequences.

SWE.4 Software Unit Verification	Outcome 1	Outcome 2	Outcome 3	Outcome 4	Outcome 5
Output Information Items					
08-60 Verification Measure	X				
08-58 Verification Measure Selection Set		X			
15-52 Verification Results			X		
13-51 Consistency Evidence				X	
13-52 Communication Evidence					X
Base Practices					
BP1: Specify software unit verification measures	X				
BP2: Select software unit verification measures		X			
BP3: Verify software units			X		
BP4: Ensure consistency and establish bidirectional traceability for software unit verification				X	
BP5: Summarize and communicate results					X

4.4.5. SWE.5 Software Component Verification and Integration Verification

Process ID
SWE.5
Process name
Software Component Verification and Integration Verification
Process purpose
The purpose is to verify that software components are consistent with the software architectural design, and to integrate software elements and verify that the integrated software elements are consistent with the software architecture and software detailed design.

Process outcomes

- 1) Verification measures are specified for software integration verification of the integrated software elements based on the software architecture and detailed design, including the interfaces of, and interactions between, the software components;
- 2) Verification measures for software components are specified to provide evidence for compliance of the software components with the software components' behavior and interfaces;
- 3) Software elements are integrated up to a complete integrated software;
- 4) Verification measures are selected according to the release scope considering criteria for regression verification;
- 5) Software components are verified using the selected verification measures, and the results of the integration verification are recorded;
- 6) Integrated software elements are verified using the selected verification measures, and the results of the integration verification are recorded;
- 7) Consistency and bidirectional traceability are established between verification measures and the software architecture and detailed design; and consistency and bidirectional traceability are established between verification results and verification measures;
- 8) The results of software component verification and software elements integration verification are summarized and communicated to all affected parties.

Base practices

SWE.5.BP1: Specify software integration verification measures. Specify verification measures, based on a defined order and preconditions for the integration of software elements, against the defined static and dynamic aspects of the software architecture, including

- techniques for the verification measures;
- pass/fail criteria for verification measures;
- entry and exit criteria for verification measures;
- the required verification infrastructure and environment setup.

Note 1: Examples on which the software integration verification measures may focus on are the correct dataflow and dynamic interaction between software components together with their timing dependencies, the correct interpretation of data by all software components using an interface, and the compliance to resource consumption objectives.

Note 2: The software integration verification measure may be supported by using hardware debug interfaces or simulation environments (e.g. Software-in-the-Loop-Simulation).

SWE.5.BP2: Specify verification measures for verifying software component behavior.

Specify verification measures for software component verification against the defined software components' behavior and their interfaces in the software architecture, including

- techniques for the verification measures;
- entry and exit criteria for verification measures;
- pass/fail criteria for verification measures;
- the required verification infrastructure and environment setup.

Note 3: Verification measures are related to software components but not to the software units since software unit verification is addressed in the process SWE.4 Software Unit Verification.

SWE.5.BP3: Select verification measures. Document the selection of integration verification measures for each integration step considering selection criteria including criteria for regression verification. The documented selection of verification measures shall have sufficient coverage according to the release scope.

Note 4: Examples for selection criteria can be the need for continuous integration /continuous development regression verification (due to e.g. changes to the software architectural or detailed design), or the intended use of the delivered product release (e.g. test bench, test track, public road etc.).

SWE.5.BP4: Integrate software elements and perform integration verification. Integrate the software elements until the software is fully integrated according to the specified interfaces and interactions between the Software elements, and according to the defined order and defined preconditions. Perform the selected integration verification measures. Record the verification measure data including pass/fail status and corresponding verification measure data.

Note 5: Examples for preconditions for starting software integration are qualification of pre-existing software components, off-the-shelf software components, open-source-software, or auto-code generated software.

Note 6: Defined preconditions may allow e.g. big-bang-integration of all software components, continuous integration, as well as step-wise integration (e.g. across software units and/or software components up to the fully integrated software) with accompanying verification measures.

Note 7: See SUP.9 for handling deviations of verification results deviate expected results.

SWE.5.BP5: Perform software component verification. Perform the selected verification measures for verifying software component behavior. Record the verification results including pass/fail status and corresponding verification measure data.

Note 8: See SUP.9 for handling verification results that deviate from expected results.

SWE.5.BP6: Ensure consistency and establish bidirectional traceability. Ensure consistency and establish bidirectional traceability between verification measures and the static and dynamic aspects of the software architecture and detailed design. Ensure consistency and establish bidirectional traceability between verification results and verification measures.

Note 9: Bidirectional traceability supports consistency, and facilitates impact analysis of changerequests, and demonstration of verification coverage

SWE.5.BP7: Summarize and communicate results. Summarize the software component verification and the software integration verification results and communicate them to all affected parties.

Note 10: Providing all necessary information from the test case execution in a summary enables other parties to judge the consequences.

SWE.5 Software Component Verification and Integration Verification	Outcome 1	Outcome 2	Outcome 3	Outcome 4	Outcome 5	Outcome 6	Outcome 7	Outcome 8
Output Information Items								
08-60 Verification Measure	X	X						
06-50 Integration Sequence Instruction			X					
08-58 Verification Measure Selection Set				X				
15-52 Verification Results					X	X		
13-51 Consistency Evidence							X	
13-52 Communication Evidence								X
01-03 Software Component			X					
01-50 Integrated Software			X					
Base Practices								
BP1: Specify software integration verification measures	X							
BP2: Specify verification measures for verifying software component behavior		X						
BP3: Select verification measures				X				
BP4: Integrate software elements and perform integration verification			X			X		
BP5: Perform software component verification					X			
BP6: Ensure consistency and establish bidirectional traceability							X	
BP7: Summarize and communicate results								X

4.4.6. SWE.6 Software Verification

Process ID
SWE.6
Process name
Software Verification
Process purpose
The purpose of the Software Verification process is to ensure that the integrated software is verified to be consistent with the software requirements.
Process outcomes
<ol style="list-style-type: none"> 1) Verification measures are specified for software verification of the software based on the software requirements; 2) Verification measures are selected according to the release scope considering criteria for regression verification; 3) The integrated software is verified using the selected verification measures and the results of software verification are recorded; 4) Consistency and bidirectional traceability are established between verification measures and software requirements; and consistency and bidirectional traceability are established between verification results and verification measures; 5) Results of the software verification are summarized and communicated to all affected parties.
Base practices
<p>SWE.6.BP1: Specify verification measures for software verification. Specify the verification measures for software verification suitable to provide evidence for compliance of the integrated software with the functional and non-functional information in the software requirements, including</p> <ul style="list-style-type: none"> • techniques for the verification measures; • pass/fail criteria for verification measures; • a definition of entry and exit criteria for the verification measures; • necessary sequence of verification measures; • the required verification infrastructure and environment setup. <p><i>Note 1: The selection of appropriate techniques for verification measures may depend on the content of the respective software requirement (e.g. boundary values and equivalence classes for data range-oriented requirements, positive/sunny-day-test vs. negative testing such as fault injection), or on requirements-based testing vs. “error guessing based on knowledge or experience”.</i></p>

SWE.6.BP2: Select verification measures. Document the selection of verification measures considering selection criteria including criteria for regression verification. The documented selection of verification measures shall have sufficient coverage according to the release scope.

Note 2: Examples for selection criteria can be prioritization of requirements, continuous development, the need for regression verification (due to e.g. changes to the software requirements), or the intended use of the delivered product release (test bench, test track, public road etc.)

SWE.6.BP3: Verify the integrated software. Perform the selected verification measures. Record the verification results including pass/fail status and corresponding verification measure data.

Note 3: See SUP.9 for handling verification results that deviate from expected results.

SWE.6.BP4: Ensure consistency and establish bidirectional traceability. Ensure consistency and establish bidirectional traceability between verification measures and software requirements. Ensure consistency and establish bidirectional traceability between verification results and verification measures.

Note 4: Bidirectional traceability supports consistency, and facilitates impact analysis of change requests, and demonstration of verification coverage.

SWE.6.BP5: Summarize and communicate results. Summarize the software verification results and communicate them to all affected parties.

Note 5: Providing all necessary information from the test case execution in a summary enables other parties to judge the consequences.

SWE.6 Software Verification	Outcome 1	Outcome 2	Outcome 3	Outcome 4	Outcome 5
Output Information Items					
08-60 Verification Measure	X				
08-58 Verification Measure Selection Set		X			
15-52 Verification Results			X		
13-51 Consistency Evidence				X	
13-52 Communication Evidence					X
Base Practices					
BP1: Specify verification measures for software verification	X				
BP2: Select verification measures		X			
BP3: Verify the integrated software			X		
BP4: Ensure consistency and establish bidirectional traceability.				X	
BP5: Summarize and communicate results					X

4.5. Validation process group (VAL)

4.5.1. VAL.1 Validation

Process ID
VAL.1
Process name
Validation
Process purpose
The purpose is to provide evidence that the end product, allowing direct end user interaction, satisfies the intended use expectations in its operational target environment.
Process outcomes
<p><i>Note 1: Validation measures are specified that are suitable to provide evidence for satisfaction of the intended use expectations in the operational target environment;</i></p> <ol style="list-style-type: none"> 1) Validation measures are selected considering criteria for regression verification; 2) The product is validated using the selected validation measures and the results of validation are recorded; 3) Consistency and unidirectional traceability are established between validation measures and stakeholder requirements; and consistency and bidirectional traceability is established between validation results and validation measures; 4) Results of the validation are summarized and communicated to all affected parties.
Base practices
<p>VAL.1.BP1: Specify validation measures for product validation. Specify the validation measures for the end product based on the stakeholder requirements to provide evidence that it fulfills its intended use expectations in its operational target environment, and</p> <ul style="list-style-type: none"> • techniques for the validation measures; • pass/fail criteria for validation measures; • a definition of entry and exit criteria for the validation measures; • necessary sequence of validation measures; • the required validation infrastructure and environment setup. <p><i>Note 1: An example for validation-relevant stakeholder requirements are homologation or legal type approval requirements. Further examples of sources of intended use expectations are technical risks (see MAN.5, SYS.3.BP4, SWE.2.BP3, HWE.2.BP6).</i></p> <p><i>Note 2: Where stakeholder requirements cannot be specified comprehensively or change frequently, repeated validation of (often rapidly developed) increments in product evolution may be employed to refine stakeholder requirements, and to mitigate risks in the correct identification of needs.</i></p> <p><i>Note 3: Validation may also be conducted to confirm that the product also satisfies the often less formally expressed, but sometimes overriding, attitudes, experience, and subjective tests that comprise stakeholder or end user satisfaction.</i></p>

VAL.1.BP2: Select validation measures. Document the selection of validation measures considering selection criteria including criteria for regression validation. The documented selection of validation measures shall have sufficient coverage according to the release scope.

Note 4: Examples for criteria for selection can be the release purpose of the delivered product (such as test bench, test track, validation on public roads, field use by end users), homologation/ type approval, confirmation of requirements, or the need for regression due to e.g. changes to stakeholder requirements and needs.

VAL.1.BP3: Perform validation and evaluate results. Perform the validation of the integrated system using the selected validation measures. Record the validation results including pass/fail status. Evaluate the validation results.

Note 5: Validation results can be used as a means for identifying stakeholder or system requirements e.g. in the case of mock-ups or concept studies.

Note 6: See SUP.9 for handling of non-conformances and SUP.10 for raising change requests.

VAL.1.BP4: Ensure consistency and establish traceability. Ensure consistency and establish bidirectional traceability from validation measures to the stakeholder requirements from which they are derived. Establish bidirectional traceability between validation results and validation measures.

Note 7: Examples of sources of validation measures from which they can be derived are legal requirements, homologation requirements, results of technical risk analyses, or stakeholder and system requirements (see SYS.1 and SYS.2).

Note 8: If sources of validation measures source are e.g. legal or homologation requirements, then direct bidirectional traceability from those sources to the validation measures are not possible. In such a case, unidirectional traceability is sufficient.

Note 9: Bidirectional traceability supports consistency, and facilitates impact analyses of change requests, and demonstration of verification coverage.

VAL.1.BP5: Summarize and communicate results. Summarize the validation results and inform all affected parties.

Note 10: Providing all necessary information from the test case execution in a summary enables other parties to judge the consequences.

VAL.1 Validation	Outcome 1	Outcome 2	Outcome 3	Outcome 4	Outcome 5	Outcome 6
Output Information Items						
08-59 Validation Measure	X					
08-58 Validation Measure Selection Set		X				
13-24 Validation Results			X			
13-51 Consistency Evidence				X	X	
13-52 Communication Evidence						X
Base Practices						

BP1: Specify validation measures	X					
BP2: Select validation measures		X				
BP3: Perform validation and evaluate results			X			
BP4: Ensure consistency and establish traceability.				X	X	
BP5: Summarize and communicate results						X

4.6. Machine Learning Engineering process group (MLE)

4.6.1. MLE.1 Machine Learning Requirements Analysis

Process ID
MLE.1
Process name
Machine Learning Requirements Analysis
Process purpose
The purpose is to refine the machine learning-related software requirements into a set of ML requirements.
Process outcomes
<ol style="list-style-type: none"> 1) the ML requirements, including ML data requirements, are identified and specified based on the software requirements and the components of the software architecture 2) ML requirements are structured and prioritized 3) ML requirements are analyzed for correctness and verifiability 4) the impact of ML requirements on the ML operating environment is analyzed 5) consistency and bidirectional traceability are established between ML requirements and software requirements, and between ML requirements and software architecture 6) the ML requirements are agreed and communicated to all affected parties
Base practices
<p>MLE.1.BP1: Specify ML requirements. Use the software requirements and the software architecture to identify and specify functional and non-functional ML requirements, as well as ML data requirements specifying data characteristics and their expected distributions.</p> <p><i>Note 1: Non-functional requirements may include relevant characteristics of the ODD and KPIs as robustness, performance and level of trustworthiness.</i></p> <p><i>Note 2: The ML data requirements are input for SUP.11 Machine Learning Data Management but also for other MLE processes.</i></p> <p><i>Note 3: In case of ML development only, stakeholder requirements represent the software requirements.</i></p>
<p>MLE.1.BP2: Structure ML requirements. Structure and prioritize the ML requirements.</p> <p><i>Note 4: Examples for structuring criteria can be grouping (e.g. by functionality) or variants identification.</i></p> <p><i>Note 5: Prioritization can be done according to project or stakeholder needs via e.g. definition of release scopes. Refer to SPL.2.BP1.</i></p>
<p>MLE.1.BP3: Analyze ML requirements. Analyze the specified ML requirements including their interdependencies to ensure correctness, technical feasibility, and ability for machine learning model testing, and to support project management regarding project estimates.</p> <p><i>Note 6: See MAN.3.BP3 for project feasibility and MAN.3.BP5 for project estimates.</i></p>

MLE.1.BP4: Analyze the impact on the ML operating environment. Analyze the impact that the ML requirements will have on interfaces of software components and the ML operating environment.

Note 7: The ML operating environment is defined as the infrastructure and information which both the trained ML model and the deployed ML model need for execution.

MLE.1.BP5: Ensure consistency and establish bidirectional traceability. Ensure consistency and establish bidirectional traceability between ML requirements and software requirements and between ML requirements and the software architecture.

Note 8: Bidirectional traceability supports consistency, facilitates impact analyses of change requests, and verification coverage demonstration.

Note 9: Redundant traceability is not intended, but at least one out of the given traceability paths.

MLE.1.BP6: Communicate agreed ML requirements. Communicate the agreed ML requirements, and the results of the impact analysis on the ML operating environment to all affected parties.

MLE.1 – Machine Learning Requirements Analysis	Outcome 1	Outcome 2	Outcome 3	Outcome 4	Outcome 5	Outcome 6
Output Information Items						
17-00 Requirement	X	X				
17-54 Requirement attribute		X	X			
13-52 Communication evidence						X
13-51 Consistency evidence					X	
15-51 Analysis results			X	X		
Base Practices						
BP1: Specify ML requirements	X					
BP2: Structure ML requirements		X				
BP3: Analyze ML requirements			X			
BP4: Analyze the impact on the ML operating environment				X		
BP5: Ensure consistency and establish bidirectional traceability					X	
BP6: Communicate agreed ML requirements						X

4.6.2. MLE.2 Machine Learning Architecture

Process ID
MLE.2
Process name
Machine Learning Architecture
Process purpose
The purpose is to establish an ML architecture supporting training and deployment, consistent with the ML requirements, and to evaluate the ML architecture against defined criteria.
Process outcomes
<ol style="list-style-type: none"> 1) an ML architecture is developed 2) hyperparameter ranges and initial values are determined as a basis for the training 3) evaluation of ML architectural elements is conducted 4) interfaces of the ML architectural elements are defined 5) resource consumption objectives for the ML architectural elements are defined 6) consistency and bidirectional traceability are established between the ML architectural elements and the ML requirements 7) the ML architecture is agreed and communicated to all affected parties
Base practices
<p>MLE.2.BP1: Develop ML architecture. Develop and document the ML architecture that specifies ML architectural elements including details of the ML model, pre- and postprocessing, and hyperparameters which are required to create, train, test, and deploy the ML model.</p> <p><i>Note 1: Necessary details of the ML model may include layers, activation functions, and backpropagation. The level of detail of the ML model may not need to cover aspects like single neurons.</i></p> <p><i>Note 2: The details of the ML model may differ between the ML model used during training and the deployed ML model.</i></p>
<p>MLE.2.BP2: Determine hyperparameter ranges and initial values. Determine and document the hyperparameter ranges and the initial values as a basis for the training.</p>
<p>MLE.2.BP3: Evaluate ML architectural elements. Define evaluation criteria for the ML architectural elements. Evaluate ML architectural elements according to the defined criteria.</p> <p><i>Note 3: Trustworthiness and explainability might be criteria for the evaluation of the ML architectural elements.</i></p>
<p>MLE.2.BP4: Define interfaces of the ML architectural elements. Determine and document the internal and external interfaces of each ML architectural element including its interfaces to related software components.</p>
<p>MLE.2.BP5: Define resource consumption objectives for the ML architectural elements. Determine and document the resource consumption objectives for all relevant ML architectural elements during training and deployment.</p>

MLE.2.BP6: Ensure consistency and establish bidirectional traceability. Ensure consistency and establish bidirectional traceability between the ML architectural elements and the ML requirements.

Note 4: Bidirectional traceability supports consistency, and facilitates impact analyses of change requests, and verification coverage demonstration.

Note 5: The bidirectional traceability should be established on a reasonable level of abstraction to the ML architectural elements.

MLE.2.BP7: Communicate agreed ML architecture. Inform all affected parties about the agreed ML architecture including the details of the ML model and the initial hyperparameter values.

MLE.2 – Machine Learning Architecture	Outcome 1	Outcome 2	Outcome 3	Outcome 4	Outcome 5	Outcome 6	Outcome 7
Output Information Items							
04-51 ML architecture	X	X	X	X	X		
13-52 Communication evidence							X
13-51 Consistency evidence						X	
01-54 Hyperparameter	X	X					
15-51 Analysis results	X		X				
Base Practices							
BP1: Develop ML architecture	X						
BP2: Determine hyperparameter ranges and initial values.		X					
BP3: Evaluate ML architectural elements			X				
BP4: Define interfaces of the ML architectural elements				X			
BP5: Define resource consumption objectives for the ML architectural elements					X		
BP6: Ensure consistency and establish bidirectional traceability						X	
BP7: Communicate agreed ML architecture							X

4.6.3. MLE.3 Machine Learning Training

Process ID
MLE.3
Process name
Machine Learning Training
Process purpose
The purpose is to optimize the ML model to meet the defined ML requirements.
Process outcomes
<ol style="list-style-type: none"> 1) an ML training and validation approach is specified 2) the data set for ML training and ML validation is created 3) the ML model, including hyperparameter values, is optimized to meet the defined ML requirements 4) consistency and bidirectional traceability are established between the ML training and validation data set and the ML data requirements 5) results of optimization are summarized, and the trained ML model is agreed and communicated to all affected parties

Base practices
<p>MLE.3.BP1: Specify ML training and validation approach. Specify an approach which supports the training and validation of the ML model to meet the defined ML requirements. The ML training and validation approach includes</p> <ul style="list-style-type: none"> • entry and exit criteria of the training and validation • approaches for hyperparameter tuning / optimization • approach for data set creation and modification • training and validation environment <p><i>Note 1: The ML training and validation approach may include random dropout and other robustification methods.</i></p> <p><i>Note 2: ML validation is the optimization of the hyperparameters during Machine Learning Training (MLE.3). The term “validation” has a different meaning than VAL.1.</i></p> <p><i>Note 3: The training environment should reflect the environment of the deployed model.</i></p>
<p>MLE.3.BP2: Create ML training and validation data set. Select data from the ML data collection provided by SUP.11 and assign them to the data set for training and validation of the ML model according to the specified ML training and validation approach.</p> <p><i>Note 4: The ML training and validation data set may include corner cases, unexpected cases, and normal cases depending on the ML requirements.</i></p> <p><i>Note 5: A separated data set for training and validation might not be required in some cases (e.g., k-fold cross validation, no optimization of hyperparameters).</i></p>
<p>MLE.3.BP3: Create and optimize ML model. Create the ML model according to the ML architecture and train it, using the identified ML training and validation data set according to the</p>

ML training and validation approach to meet the defined ML requirements, and training and validation exit criteria.

MLE.3.BP4: Ensure consistency and establish bidirectional traceability. Ensure consistency and establish bidirectional traceability between the ML training and validation data set and the ML data requirements.

Note 6: Bidirectional traceability supports consistency and facilitates impact analyses of change requests.

MLE.3.BP5: Summarize and communicate agreed trained ML model. Summarize the results of the optimization and inform all affected parties about the agreed trained ML model.

MLE.3 – Machine Learning Training	Outcome 1	Outcome 2	Outcome 3	Outcome 4	Outcome 5
Output Information Items					
08-65 ML training and validation approach	X				
03-51 ML data set		X			
01-53 Trained ML model			X		
01-54 Hyperparameter			X		
13-51 Consistency evidence				X	
13-52 Communication evidence					X
Base Practices					
BP1: Specify ML training and validation approach	X				
BP2: Create ML training and validation data set		X			
BP3: Create and optimize ML model			X		
BP4: Ensure consistency and establish bidirectional traceability				X	
BP5: Summarize and communicate agreed trained ML model					X

4.6.4. MLE.4 Machine Learning Model Testing

Process ID
MLE.4
Process name
Machine Learning Model Testing
Process purpose
The purpose is to ensure compliance of the trained ML model and the deployed ML model with the ML requirements.
Process outcomes
<ol style="list-style-type: none"> 1) an ML test approach is defined 2) an ML test data set is created 3) the trained ML model is tested 4) the deployed ML model is derived from the trained ML model and tested 5) consistency and bidirectional traceability are established between the ML test approach and the ML requirements, and the ML test data set and the ML data requirements, and the ML test approach and ML test results 6) results of the ML model testing are summarized and communicated with the deployed ML model to all affected parties

Base practices
<p>MLE.4.BP1: Specify an ML test approach. Specify an ML test approach suitable to provide evidence for compliance of the trained ML model and the deployed ML model with the ML requirements. The ML test approach includes</p> <p>ML test scenarios with distribution of data characteristics (e.g., gender, weather conditions, street conditions within the ODD) defined by ML requirements;</p> <ul style="list-style-type: none"> • distribution and frequency of each ML test scenario inside the ML test data set • expected test result per test datum • entry and exit criteria of the testing • approach for data set creation and modification • the required testing infrastructure and environment setup <p><i>Note 1: Expected test result per test datum might require labeling of test data to support comparison of output of the ML model with the expected output.</i></p> <p><i>Note 2: Test datum is the smallest amount of data which is processed by the ML model into only one output. E.g., one image in photo processing or an audio sequence in voice recognition.</i></p>

MLE.4.BP2: Create ML test data set. Create the ML test data set needed for testing of the trained ML model and testing of the deployed ML model from the ML data collection provided by SUP.11 considering the ML test approach. The ML test data set shall not be used for training.

Note 3: The ML test data set for the trained ML model might differ from the test data set of the deployed ML model.

Note 4: Additional data sets might be used for special purposes like assurance of safety, fairness, robustness.

MLE.4.BP3: Test trained ML model. Test the trained ML model according to the ML test approach using the created ML test data set. Record and evaluate the test results and logs.

Note 5: Evaluation of test logs might include pattern analysis of failed test data to support e.g., trustworthiness.

MLE.4.BP4: Derive deployed ML model. Derive the deployed ML model from the trained ML model according to the ML architecture. The deployed ML model shall be used for testing and delivery to software integration.

Note 6: The deployed ML model will be integrated into the target system and may differ from the trained ML model which often requires powerful hardware and uses interpretative languages.

MLE.4.BP5: Test deployed ML model. Test the deployed ML model according to the ML test approach using the created ML test data set. Record and evaluate the test results and logs.

MLE.4.BP6: Ensure consistency and establish bidirectional traceability. Ensure consistency and establish bidirectional traceability between the ML test approach and the ML requirements, and the ML test data set and the ML data requirements, and the ML test approach and ML test results.

Note 7: Bidirectional traceability supports consistency, and facilitates impact analyses of change requests, and verification coverage demonstration.

MLE.4.BP7: Summarize and communicate results. Summarize the test results of the ML model. Inform all affected parties about the agreed results and the deployed ML model.

MLE.4 – Machine Learning Model Testing	Outcome 1	Outcome 2	Outcome 3	Outcome 4	Outcome 5	Outcome 6
Output Information Items						
08-64 ML test approach	X					
03-51 ML data set		X				
13-50 Test result			X	X		
11-50 Deployed ML model				X		
13-51 Consistency evidence					X	
13-52 Communication evidence						X

Base Practices						
BP1: Specify an ML test approach	X					
BP2: Create ML test data set		X				
BP3: Test trained ML model			X			
BP4: Derive deployed ML model				X		
BP5: Test deployed ML model				X		
BP6: Ensure consistency and establish bidirectional traceability					X	
BP7: Summarize and communicate results						X

4.7. Hardware Engineering process group (HWE)

4.7.1. HWE.1 Hardware Requirements Analysis

Process ID
HWE.1
Process name
Hardware Requirements Analysis
Process purpose
The purpose is to establish a structured and analyzed set of hardware requirements consistent with the system requirements, and the system architectural design.
Process outcomes
<ol style="list-style-type: none"> 1) Hardware requirements are specified; 2) Hardware requirements are structured and prioritized; 3) Hardware requirements are analyzed for correctness and technical feasibility; 4) The impact of hardware requirements on the operating environment is analyzed; 5) Consistency and bidirectional traceability are established between hardware requirements and system requirements. 6) Consistency and bidirectional traceability are established between hardware requirements and system architectural design; 7) The hardware requirements are agreed and communicated to all affected parties.

Base practices
<p>HWE.1.BP1: Specify hardware requirements. Use the system requirements, and the system architecture including interface definitions, to identify and document the functional and non-functional requirements of the hardware according to defined characteristics for requirements.</p> <p><i>Note 1: Characteristics of requirements are defined in standards such as ISO IEEE 29148, ISO/IEC IEEE 24765, ISO 26262-8:2018, or the INCOSE Guide For Writing Requirements.</i></p> <p><i>Note 2: Examples for defined characteristics of requirements shared by the above-mentioned standards are verifiability (i.e. verification criteria being inherent in the requirements sentence), unambiguity/comprehensibility, freedom from design and implementation, and not contradicting any other requirement.</i></p> <p><i>Note 3: In case of hardware development only, the system requirements and the system architecture refer to a given operating environment. In that case, stakeholder requirements can be used as the basis for identifying the required functions and capabilities of the hardware.</i></p> <p><i>Note 4: The hardware-software-interface (HSI) definition puts in context hardware and therefore is an interface decision at the system design level (see SYS.3). If such a HSI exists, then it may provide input to hardware requirements.</i></p>

HWE.1.BP2: Structure hardware requirements. Structure and prioritize the hardware requirements.

Note 5: Examples for structuring criteria can be grouping (e.g. by functionality) or variants identification.

Note 6: Prioritization can be done according to project or stakeholder needs via e.g. definition of release scopes. Refer to SPL.2.BP1.

HWE.1.BP3: Analyze hardware requirements. Analyze the specified hardware requirements including their interdependencies to ensure correctness, technical feasibility, and to support project management regarding project estimates.

Note 7: See MAN.3.BP3 for project feasibility and MAN.3.BP5 for project estimates.

Note 8: The analysis of technical feasibility can be done based on a given hardware design (e.g., platform) or by prototype development.

HWE.1.BP4: Analyze the impact on the operating environment. Identify the interfaces between the specified hardware and other elements of the operating environment. Analyze the impact that the hardware requirements will have on these interfaces and the operating environment.

HWE.1.BP5: Ensure consistency and establish bidirectional traceability. Ensure consistency and establish traceability between hardware requirements and the system architecture. Ensure consistency and establish traceability between hardware requirements and system requirements.

Note 9: Redundant traceability is not intended. concern.

Note 10: There may be non-functional hardware requirements that the hardware design does not trace to. Examples are development process requirements. Such requirements are still subject to verification.

Note 11: Bidirectional traceability supports consistency, and facilitates impact analysis of change requests, and demonstration of verification coverage.

Note 12: In case of hardware development only, the system requirements and system architecture refer to a given operating environment. In that case, consistency and bidirectional traceability can be ensured between stakeholder requirements and hardware requirements.

HWE.1.BP6: Communicate agreed hardware requirements and impact on the operating environment. Communicate the agreed hardware requirements and results of the analysis of impact on the operating environment to all relevant parties.

HWE.1 Hardware Requirements Analysis	Outcome 1	Outcome 2	Outcome 3	Outcome 4	Outcome 5	Outcome 6	Outcome 7
Output Information Items							
13-52 Communication Evidence							X
13-51 Consistency Evidence					X	X	
17-58 Hardware Requirement	X	X	X				

17-54 Requirement Attribute		X					
15-51 Analysis Results			X	X			
Base practices							
BP1: Specify hardware requirements	X						
BP2: Structure hardware requirements		X					
BP3: Analyze hardware requirements			X				
BP4: Analyze the impact on the operating environment				X			
BP5: Ensure consistency and establish bidirectional traceability					X	X	
BP6: Communicate agreed hardware requirements							X

4.7.2. HWE.2 Hardware Design

Process ID
HWE.2
Process name
Hardware Design
Process purpose
The purpose is to provide an analyzed design based on the hardware requirements, that is suitable for manufacturing, and to derive production-relevant data.
Process outcomes
<ol style="list-style-type: none"> 1) A hardware architecture and hardware detailed design is designed that identifies the elements of the hardware and describes their behavior as well as their interfaces, and the dynamic interactions of the hardware elements; 2) The hardware architecture and the hardware detailed design is analyzed and special characteristics are identified; 3) Consistency and bidirectional traceability are established between hardware requirements and hardware design; 4) Hardware production data is derived from the HW detailed design and communicated to all affected parties; 5) Information for production test is derived from the HW detailed design and communicated to all affected parties; 6) The hardware architecture and hardware detailed design and the special characteristics are agreed and communicated to all affected parties.

Base practices
<p>HWE.2.BP1: Develop hardware architecture. Develop the hardware architecture that identifies the hardware components. Document the rationale for the defined hardware architecture.</p> <p><i>Note 1: Examples for aspects reflected in the hardware architecture are ground concept, supply concept, EMC concept.</i></p> <p><i>Note 2: Examples for a design rationale can be implied by the reuse of a standard hardware, platform, or product line, respectively, or by a make-or-buy decision, or found in an evolutionary way.</i></p>
<p>HWE.2.BP2: Develop hardware detailed design. Based on components identified in the hardware architecture, develop the detailed design description and the schematics for the intended hardware variants, including the interfaces between the hardware elements. Derive the layout, the hardware bill of materials, and the production data.</p> <p><i>Note 3: The identification of hardware parts and their suppliers in the hardware bill of materials may be subject to a pre-defined repository (see also IATF 16949:2016, clause 8.4.1.2.).</i></p> <p><i>Note 4: Hardware detailed design may be subject to constraints such as availability of hardware parts on the market, hardware design rules, layout rules, creepage and clearance distances, compliance of HW parts with industry standards such as AEC-Q, REACH.</i></p>

HWE.2.BP3: Specify dynamic aspects. Evaluate and document the dynamic behaviour of the relevant hardware elements and the interaction between them.

Note 5: Not all hardware elements have dynamic behaviour that needs to be described.

HWE.2.BP4: Analyze the hardware architecture and the hardware detailed design. Analyze the hardware architecture and hardware detailed design regarding relevant technical aspects, and support project management regarding project estimates. Identify special characteristics.

Note 6: Examples for technical aspects are manufacturability for production, suitability of pre-existing hardware components to be reused, or availability of hardware elements.

Note 7: Examples of methods suitable for analyzing technical aspects are prototype testing, simulations, calculations, quantitative or qualitative analyses such as FMEA.

HWE.2.BP5: Ensure consistency and establish bidirectional traceability. Ensure consistency and establish traceability between hardware elements and hardware requirements. Ensure consistency and establish traceability between the hardware detailed design and components of the hardware architecture.

Note 8: There may be non-functional hardware requirements that the hardware design does not trace to. Examples are development process requirements. Such requirements are still subject to verification.

Note 9: Bidirectional traceability further supports consistency, and facilitates impact analysis of change requests, and demonstration of verification coverage.

HWE.2.BP6: Communicate agreed hardware architecture and hardware detailed design. Communicate the agreed hardware architecture and the hardware detailed design, including the special characteristics and relevant production data, to all affected parties.

HWE.2 Hardware Design	Outcome 1	Outcome 2	Outcome 3	Outcome 4	Outcome 5	Outcome 6
Output Information Items						
04-52 Hardware Architecture	X					
04-53 Hardware Detailed Design	X					
15-51 Analysis Results		X				
13-51 Consistency Evidence			X			
17-57 Special Characteristics		X				
13-52 Communication Evidence						X
04-54 Hardware Schematics	X			X	X	
14-54 Hardware Bill of Materials	X			X	X	
04-55 Hardware Layout	X			X	X	
03-54 Hardware Production Data	X			X	X	
04-56 Hardware Element Interface	X					

Base practices						
BP1: Develop hardware architecture	X			X	X	
BP2: Develop hardware design	X			X	X	
BP3: Specify dynamic aspects	X					
BP4: Analyze the hardware architecture and the hardware detailed design		X				
BP5: Ensure consistency and establish bidirectional traceability			X			
BP7: Communicate agreed hardware architecture and hardware detailed design				X	X	X

4.7.3. HWE.3 Verification against Hardware Design

Process ID
HWE.3
Process name
Verification against Hardware Design
Process purpose
The purpose is to ensure that the production data compliant hardware is verified to provide evidence for compliance with the hardware design.
Process outcomes
<ol style="list-style-type: none"> 1) Verification measures are specified for verification of the hardware against the hardware design, including the interfaces between hardware elements and the dynamic aspects; 2) Verification measures are selected according to the release scope-considering criteria for regression verification; 3) Verification is performed, if applicable on production data compliant samples, using the selected verification measures, and verification results are recorded; 4) Consistency and bidirectional traceability are established between hardware elements and verification measures; 5) Bidirectional traceability is established between verification measures and verification results; 6) Verification results are summarised and communicated to all affected parties.

Base practices
<p>HWE.3.BP1: Specify verification measures for the verification against hardware design. Specify the verification measures suitable to provide evidence for compliance of the hardware with the hardware design and its dynamic aspects, including includes</p> <ul style="list-style-type: none"> • techniques for the verification measures • pass/fail criteria for verification measures • a definition of entry and exit criteria for the verification measures • necessary sequence of verification measures • the required verification infrastructure and environment setup <p><i>Note 1: Examples on what a verification measure may focus on are the timeliness and timing dependencies of the correct signal flow between interfacing hardware elements, interactions between hardware components.</i></p> <p><i>Note 2: Measuring points can be used for stepwise testing of hardware elements.</i></p>
<p>HWE.3.BP2: Ensure use of compliant samples. Ensure that the samples used for verification against hardware design are compliant with the corresponding production data, including special characteristics.</p> <p><i>Note 3: Examples of compliance are sample reports, record of visual inspection, ICT report.</i></p>

HWE.3.BP3: Select verification measures. Document the selection of verification measures considering selection criteria including regression criteria. The documented selection of verification measures shall have sufficient coverage according to the release scope.

Note 4: Examples for selection criteria can be prioritization of requirements, the need for regression due to changes to the hardware design, or the intended use of the delivered hardware release (e.g. test bench, test track, public road etc.)

HWE.3.BP4: Verify hardware design. Verify the hardware design using the selected verification measures. Record the verification results including pass/fail status and corresponding verification measure output data.

Note 5: See SUP.9 for handling of non-conformances.

HWE.3.BP5: Ensure consistency and establish bidirectional traceability. Ensure consistency and establish bidirectional traceability between hardware elements and the verification measures. Establish bidirectional traceability between the verification measures and verification results.

Note 6: Bidirectional traceability supports consistency, and facilitates impact analysis of change requests, and demonstration of verification coverage.

HWE.3.BP6: Summarize and communicate results. Summarise the verification results and communicate them to all affected parties.

Note 7: Providing all necessary information from the test case execution in a summary enables other parties to judge the consequences.

HWE.3 Verification against Hardware Design	Outcome 1	Outcome 2	Outcome 3	Outcome 4	Outcome 5	Outcome 6
Output Information Items						
08-60 Verification Measure	X					
08-58 Verification Measure Selection Set		X				
15-52 Verification Results			X			
13-51 Consistency Evidence				X	X	
13-52 Communication Evidence						X
Base practices						
BP1: Specify verification measures for the verification against hardware design	X					
BP2: Ensure use of compliant samples			X			
BP3: Select verification measures		X				
BP4: Verify hardware design			X			
BP5: Ensure consistency and establish bidirectional traceability				X	X	
BP6: Summarize and communicate results						X

4.7.4. HWE.4 Verification against Hardware Requirements

Process ID
HWE.4
Process name
Verification against Hardware Requirements
Process purpose
The purpose is to ensure that the complete hardware is verified to be consistent with the hardware requirements.
Process outcomes
<ol style="list-style-type: none"> 1) Verification measures are specified for verification of the hardware against the hardware requirements; 2) Verification measures are selected considering criteria for regression verification; 3) Verification is performed, if applicable on production data compliant samples, using the selected verification measures, and verification results are recorded; 4) Consistency and bidirectional traceability are established between verification measures and hardware requirements; 5) Bidirectional traceability is established between verification measures and verification results; 6) Verification results are summarized and communicated to all affected parties.
Base practices
<p>HWE.4.BP1: Specify verification measures for the verification against hardware requirements. Specify the verification measure to provide evidence for compliance with the hardware requirements. This includes</p> <ul style="list-style-type: none"> • techniques for the verification measures • pass/fail criteria for verification measures • a definition of entry and exit criteria for the verification measures • necessary sequence of verification measures • the required verification infrastructure and environment setup <p><i>Note 1: The system verification measures may cover aspects such as thermal, environmental, robustness/lifetime, and EMC.</i></p>
<p>HWE.4.BP2: Ensure use of compliant samples. Ensure that the samples used for the verification against hardware requirements are compliant with the corresponding production data, including special characteristics, provided by hardware design.</p> <p><i>Note 2: Examples of compliance are sample reports, record of visual inspection, ICT report.</i></p>

HWE.4.BP3: Select verification measures. Document the selection of verification measures considering selection criteria including regression criteria. The documented selection of verification measures shall have sufficient coverage according to the release scope.

Note 3: Examples for selection criteria can be prioritization of requirements, the need for regression due to changes to the hardware requirements, or the intended use of the delivered hardware release (e.g. test bench, test track, public road etc.).

HWE.4.BP4: Verify the compliant hardware samples. Verify the compliant hardware samples using the selected verification measures. Record the verification results including pass/fail status and corresponding verification measure output data.

Note 4: See SUP.9 for handling of non-conformances.

HWE.4.BP5: Ensure consistency and establish bidirectional traceability. Ensure consistency between hardware requirements and verification measures. Establish bidirectional traceability between hardware requirements and verification measures. Establish bidirectional traceability between verification measures and verification results.

Note 5: Bidirectional traceability supports consistency, and facilitates impact analysis of change requests, and demonstration of verification coverage.

HWE.4.BP6: Summarize and communicate results. Summarize the verification results and communicate them to all affected parties.

Note 6: Providing all necessary information from the test case execution in a summary enables other parties to judge the consequences.

HWE.4 Verification against Hardware Requirements	Outcome 1	Outcome 2	Outcome 3	Outcome 4	Outcome 5	Outcome 6
Output Information Items						
08-60 Verification Measure	X					
08-58 Verification Measure Selection Set		X				
15-52 Verification Results			X			
13-51 Consistency Evidence				X	X	
13-52 Communication Evidence						X
Base practices						
BP1: Specify verification measures for the verification against hardware requirements	X					
BP2: Ensure use of compliant samples			X			
BP3: Select verification measures		X				
BP4: Verify hardware			X			
BP5: Ensure consistency and establish bidirectional traceability				X	X	
BP6: Summarize and communicate results						X

4.8. Supporting process group (SUP)

4.8.1. SUP.1 Quality Assurance

Process ID
SUP.1
Process name
Quality Assurance
Process purpose
The purpose of the Quality Assurance Process is to provide independent and objective assurance that work products and processes comply with defined criteria and that non-conformances are resolved and further prevented.
Process outcomes
<ol style="list-style-type: none"> 1) Quality assurance is performed independently and objectively without conflicts of interest. 2) Criteria for the quality of work products and process performance are defined. 3) Conformance of work products and process performance with the defined criteria and targets is verified, documented and communicated to the relevant parties. 4) Non-conformances are tracked, resolved, and further prevented. 5) Non-conformances are escalated to appropriate levels of management. 6) Management ensures that escalated non-conformances are resolved.
Base practices
<p>SUP.1.BP1: Ensure independence of quality assurance. Ensure that quality assurance is performed independently and objectively without conflicts of interest.</p> <p><i>Note 1: Possible inputs for evaluating the independence may be assignment to financial and/or organizational structure as well as responsibility for processes that are subject to quality assurance (no self-monitoring).</i></p>
<p>SUP.1.BP2: Define criteria for quality assurance. Define quality criteria for work products as well as for process tasks and their performance.</p> <p><i>Note 2: Quality criteria may consider internal and external inputs such as customer requirements, standards, milestones, etc.</i></p>
<p>SUP.1.BP3: Assure quality of work products. Identify work products subject to quality assurance according to the quality criteria. Perform appropriate activities to evaluate the work products against the defined quality criteria and document the results.</p> <p><i>Note 3: Quality assurance activities may include reviews, problem analysis and lessons learned that improve the work products for further use.</i></p>

SUP.1.BP4: Assure quality of process activities. Identify processes subject to quality assurance according to the quality criteria. Perform appropriate activities to evaluate the processes against their defined quality criteria and associated target values and document the results.

NOTE 4: Quality assurance activities may include process assessments, problem analysis, regular check of methods, tools and the adherence to defined processes, and consideration of lessons learned.

SUP.1.BP5: Summarize and communicate quality assurance activities and results. Regularly report performance, non-conformances, and trends of quality assurance activities to all affected parties.

SUP.1.BP6: Ensure resolution of non-conformances. Analyze, track, correct, resolve and further prevent non-conformances found in quality assurance activities.

NOTE 5: Non-conformances detected in work products may be entered into the problem resolution management process (SUP.9).

NOTE 6: Non-conformances detected in the process definition or implementation may be entered into a process improvement process (PIM.3).

SUP.1.BP7: Escalate non-conformances. Escalate relevant non-conformances to appropriate levels of management and other relevant stakeholders to facilitate their resolution.

NOTE 7: The decision whether to escalate non-conformances may be based on criteria such as delay of resolution, urgency, and risk.

SUP.1 Quality Assurance	Outcome 1	Outcome 2	Outcome 3	Outcome 4	Outcome 5	Outcome 6
Output Information Items						
16-50 Organization structure	X				X	
18-52 Escalation path					X	X
18-07 Quality criteria		X	X	X		
13-52 Communication evidence			X	X	X	
13-18 Quality record			X	X		
13-19 Review record			X	X		
14-02 Corrective action				X		X
Base practices						
BP1: Ensure independence of quality assurance.	X					
BP2: Define criteria for quality assurance.		X				
BP3: Assure quality of document information.			X	X		
BP4: Assure quality of process activities.			X	X		

BP5: Summarize and communicate quality assurance activities and results.			X	X	X	
BP6: Ensure resolution of non-conformances.				X		X
BP7: Escalate non-conformances.					X	X

4.8.2. SUP.8 Configuration Management

Process ID
SUP.8
Process name
Configuration Management
Process purpose
The purpose of the Configuration Management Process is to establish and maintain the integrity of relevant work products and make them available to affected parties.
Process outcomes
<ol style="list-style-type: none"> 1) Selection criteria for configuration items are defined and applied. 2) Configuration item properties are defined. 3) Configuration management is established. 4) Modifications are controlled. 5) Baselining is applied. 6) The status of the configuration items is recorded and reported. 7) The completeness and consistency of the baselines is ensured. 8) The availability of backup and recovery mechanisms is verified.
Base practices
<p>SUP.8.BP1: Identify configuration items. Define selection criteria for identifying relevant work products to be subject to configuration management. Identify and document configuration items according to the defined selection criteria.</p> <p><i>NOTE 1: Configuration items are representing work products or group of work products which are designated for configuration management and treated as a single entity in the configuration management process.</i></p> <p><i>NOTE 2: Configuration items may vary widely in complexity, size and type, ranging from an entire system including all hardware, software and documentation, to a single module or a minor hardware component.</i></p> <p><i>NOTE 3: The selection criteria may be applied to single work products or a group of work products.</i></p>

SUP.8.BP2: Define configuration item properties. Define the necessary properties needed for the modification and control of configuration items.

NOTE 4: The configuration item properties may be defined for single configuration items or a group of items.

NOTE 5: Configuration item properties may include a status model (e.g. under work, checked in, tested, released, etc.), storage location, access rights, etc.

NOTE 6: The application of properties may be implemented by attributes of the configuration items in the configuration item list.

SUP.8.BP3: Establish configuration management. Establish configuration management mechanisms for control of identified configuration items including the configuration item properties, including mechanisms to control parallel modifications of configuration items.

NOTE 7: This may include specific mechanisms for different configuration item types, such as branch and merge management or checkout control for software or drawing revisioning for mechanic and hardware

SUP.8.BP4: Control modifications. Control modifications using the configuration management mechanisms.

NOTE 8: This may include the application of a defined status model.

SUP.8.BP5: Establish baselines. Establish baselines for internal purposes and for external delivery for all relevant configuration items according to the configuration item properties.

SUP.8.BP6: Summarize and communicate configuration status. Record, summarize and communicate the status of configuration items and established baselines to affected parties to support the monitoring of the current work product progress and status.

NOTE 9: Regular communication of the configuration status, e.g. based on a defined status model supports project management and quality activities and dedicated project phases like software integration.

SUP.8.BP7: Ensure completeness and consistency. Ensure that the information about configuration items is correct and complete with respect to the necessary configuration item properties. Ensure the completeness and consistency of baselines.

NOTE 10: Completeness and consistency of a baseline means that all required configuration items are included within the baseline, and are consistent to each other, and have the required status. This can be used to support gate approval.

SUP.8.BP8: Verify backup and recovery mechanisms' availability. Verify the availability of appropriate backup and recovery mechanism for the configuration management including the controlled work products. Initiate measures in case of insufficient backup and recovery mechanisms.

NOTE 11: Backup and recovery mechanisms may be defined and implemented by organizational units outside the project team. This may include references to corresponding procedures or regulations.

SUP.8 – Configuration Management	Outcome 1	Outcome 2	Outcome 3	Outcome 4	Outcome 5	Outcome 6	Outcome 7	Outcome 8
Output Information items								
18-53 Configuration item selection criteria	X							
01-52 Configuration item list	X	X					X	
13-10 Configuration management record		X	X	X	X	X		
16-03 Configuration management system			X	X	X			
13-08 Baseline					X		X	
14-01 Change history			X	X		X		
15-56 Configuration status summary						X		
13-51 Consistency Evidence							X	
06-52 Backup and recovery mechanism information								X
Base Practices								
BP1: Identify configuration items	X							
BP2: Define configuration item properties		X						
BP3: Establish configuration management			X					
BP4: Control modifications				X				
BP5: Establish baselines					X			
BP6: Summarize and communicate configuration status						X		
BP7: Ensure completeness and consistency							X	
BP8: Verify backup and recovery mechanisms availability								X

4.8.3. SUP.9 Problem Resolution Management

Process ID
SUP.9
Process name
Problem Resolution Management
Process purpose
The purpose of the Problem Resolution Management Process is to ensure that problems are identified, recorded, analyzed, and their resolution is managed and controlled.
Process outcomes
<ol style="list-style-type: none"> 1) Problems are uniquely identified, recorded and classified 2) Problems are analyzed and assessed to determine an appropriate solution 3) Problem resolution is initiated 4) Problems are tracked to closure 5) The status of problems including trends identified are reported to stakeholders
Base practices
<p>SUP.9.BP1: Identify and record the problem. Each problem is uniquely identified, described and recorded. A status is assigned to each problem to facilitate tracking. Supporting information is provided to reproduce and diagnose the problem.</p> <p><i>NOTE 1: Problems may relate to e.g., product, resources, or methods.</i></p> <p><i>NOTE 2: Example values for the problem status are “new”, “solved”, “closed”, etc.</i></p> <p><i>NOTE 3: Supporting information may include, e.g. the origin of the problem, how it can be reproduced, environmental information, by whom it has been detected.</i></p> <p><i>NOTE 4: Unique identification supports traceability to changes made as needed by the change request management process (SUP.10).</i></p>
<p>SUP.9.BP2: Determine the cause and the impact of the problem. Analyze the problem, determine its cause and impact. Categorize the problem.</p> <p><i>NOTE 5: Problem categorization (e.g., light, medium, severe) may be based on severity, criticality, urgency, etc.</i></p>
<p>SUP.9.BP3: Authorize urgent resolution action. Obtain authorization for immediate action if a problem requires an urgent resolution according to the categorization.</p>
<p>SUP.9.BP4: Raise alert notifications. If according to the categorization the problem has a high impact on other systems or other affected parties, an alert notification needs to be raised accordingly.</p>
<p>SUP.9.BP5: Initiate problem resolution. Initiate appropriate actions according to the categorization to resolve the problem including review of those actions or initiate a change request.</p>

SUP.9.BP6: Track problems to closure. Track the status of problems to closure including all related change requests. The closure of problems is accepted by relevant stakeholders.

SUP.9.BP7: Report the status of problem resolution activities. Collect and analyze problem resolution management data, identify trends, and initiate related actions. Regularly report the results of data analysis, the identified trends and the status of problem resolution activities to relevant stakeholders.

NOTE 6: Collected data may contain information about where the problems occurred, how and when they were found, what were their impacts, etc.

SUP.9 – Problem Resolution Management	Outcome 1	Outcome 2	Outcome 3	Outcome 4	Outcome 5
Output Information item					
13-07 Problem record	X	X	X	X	
15-55 Problem analysis evidence		X			
15-12 Problem status report					X
Base Practices					
BP1: Identify and record the problem	X			X	
BP2: Determine the cause and the impact of the problem	X	X			
BP3: Authorize urgent resolution action			X		
BP4: Raise alert notifications			X		
BP5: Initiate problem resolution			X		
BP6: Track problems to closure				X	X
BP7: Report the status of problem resolution activities					X

4.8.4. SUP.10 Change Request Management

Process ID
SUP.10
Process name
Change Request Management
Process purpose
The purpose of the Change Request Management Process is to ensure that change requests are managed, tracked and implemented.
Process outcomes
<ol style="list-style-type: none"> 1) Requests for changes are recorded and identified 2) Change requests are analyzed, dependencies and relationships to other change requests are identified, and the impact is estimated 3) Change requests are approved before implementation and prioritized accordingly 4) Bidirectional traceability is established between change requests and affected work products 5) Implementation of change requests is confirmed 6) Change requests are tracked to closure and status of change requests is communicated to affected parties

Base practices
<p>SUP.10.BP1: Identify and record the change requests. The scope for application of change requests is identified. Each change request is uniquely identified, described, and recorded, including the initiator and reason of the change request. A status is assigned to each change request to facilitate tracking.</p> <p><i>NOTE 1: Change requests may be used for changes related to e.g., product, process, methods.</i></p> <p><i>NOTE 2: Example values for the change request status are “open”, “under investigation”, “implemented”, etc.</i></p> <p><i>NOTE 3: The change request handling may differ across the product life cycle e.g., during prototype construction and series development</i></p>
<p>SUP.10.BP2: Analyze and assess change requests. Change requests are analyzed by relevant parties according to analysis criteria. Work products affected by the change request and dependencies to other change requests are determined. The impact of the change requests is assessed.</p> <p><i>NOTE 4: Examples for analysis criteria are: resource requirements, scheduling issues, risks, benefits, etc.</i></p>
<p>SUP.10.BP3: Approve change requests before implementation. Change requests are prioritized and approved for implementation based on analysis results and availability of resources.</p> <p><i>NOTE 5: A Change Control Board (CCB) is an example mechanism used to approve change requests.</i></p> <p><i>NOTE 6: Prioritization of change requests may be done by allocation to releases.</i></p>

SUP.10.BP4: Establish bidirectional traceability. Establish bidirectional traceability between change requests and work products affected by the change requests. In case that the change request is initiated by a problem, establish bidirectional traceability between change requests and the corresponding problem reports.

SUP.10.BP5: Confirm the implementation of change requests. The implementation of change requests is confirmed before closure by relevant stakeholders.

SUP.10.BP6: Track change requests to closure. Change requests are tracked to closure. The status of change requests is communicated to all affected parties.

NOTE 7: Examples for informing affected parties can be daily standup meetings or tool-supported workflows.

SUP.10 – Change Request Management	Outcome 1	Outcome 2	Outcome 3	Outcome 4	Outcome 5	Outcome 6
Output Information item						
18-57 Change analysis criteria		X				
13-16 Change request	X	X	X		X	X
13-22 Traceability record				X		
Base Practices						
BP1: Identify and record the change requests	X					
BP2: Analyze and assess change requests		X				
BP3: Approve change requests before implementation			X			
BP4: Establish bidirectional traceability				X		
BP5: Confirm the implementation of change requests					X	
BP6: Track change requests to closure						X

4.8.5. SUP.11 Machine Learning Data Management

Process ID
SUP.11
Process name
Machine Learning Data Management
Process purpose
The purpose is to define and align ML data with ML data requirements, maintain the integrity and quality of the ML data, and make them available to affected parties.
Process outcomes

- 1) an ML data management system including a ML data lifecycle is established
- 2) an ML data quality approach is developed including ML data quality criteria
- 3) collected ML data are processed for consistency with ML data requirements
- 4) ML data are verified against defined ML data quality criteria and updated as needed
- 5) ML data are agreed and communicated to all affected parties

Base practices

SUP.11.BP1: Establish an ML data management system. Establish an ML data management system which supports

- ML data management activities;
- relevant sources of ML data;
- ML data life cycle including a status model;
- interfaces to affected parties.

Note 1: Supported ML data management activities may include data collection, labeling/annotation, and structuring.

SUP.11.BP2: Develop an ML data quality approach. Develop an approach to ensure that the quality of ML data is analyzed based on defined ML data quality criteria and activities are performed to support avoidance of biases of data.

Note 2: Examples of ML data quality criteria are relevant data sources, reliability and consistency of labelling, completeness against ML data requirements.

Note 3: The ML data management system should support the quality criteria and activities of the ML data quality approach.

Note 4: Biases to avoid may include sampling bias (e.g., gender, age) and feedback loop bias.

Note 5: For creation of ML data sets see MLE.3.BP2 and MLE.4.BP2.

SUP.11.BP3: Collect ML data. Relevant sources for raw data are identified and continuously monitored for changes. The raw data is collected according to the ML data requirements.

Note 6: The identification and collection of ML data might be an organizational responsibility.

Note 7: Continuous monitoring should include the ODD and may lead to changes of the ML requirements.

SUP.11.BP4: Process ML data. The raw data are processed (annotated, analyzed, and structured) according to the ML data requirements.

SUP.11.BP5: Assure quality of ML data. Perform the activities according to the ML data quality approach to ensure that the ML data meets the defined ML data quality criteria.

Note 8: These activities may include sample-based reviews or statistical methods.

SUP.11.BP6: Communicate agreed processed ML data. Inform all affected parties about the agreed processed ML data and provide them to the affected parties.

SUP.11 – Machine Learning Data Management	Outcome 1	Outcome 2	Outcome 3	Outcome 4	Outcome 5
Output Information Items					
16-52 ML data management system	X				
19-50 ML data quality approach		X			
03-53 ML data			X	X	
13-52 Communication evidence					X
Base Practices					
BP1: Establish an ML data management system	X				
BP2: Develop an ML data quality approach		X			
BP3: Collect ML data			X		
BP4: Process ML data			X		
BP5: Assure quality of ML data				X	
BP6: Communicate agreed processed ML data					X

4.9. Management process group (MAN)

4.9.1. MAN.3 Project Management

Process ID
MAN.3
Process name
Project Management
Process purpose
The purpose is to identify and control the activities, and establish resources necessary for a project to develop a product, in the context of the project's requirements and constraints.
Process outcomes
<ol style="list-style-type: none"> 1) The scope of the work for the project is defined. 2) The feasibility of achieving the goals of the project with available resources and constraints is evaluated. 3) The activities and resources necessary to complete the work are sized and estimated. 4) Interfaces within the project, and with other projects and organizational units, are identified and monitored. 5) Plans for the execution of the project are developed, implemented and maintained. 6) Progress of the project is monitored and reported. 7) Adjustment is performed when project goals are not achieved.
Base practices
MAN.3.BP1: Define the scope of work. Identify the project's goals, motivation and boundaries.
<p>MAN.3.BP2: Define project life cycle. Define the life cycle for the project, which is appropriate to the scope, context, and complexity of the project. Define a release scope for relevant milestones.</p> <p><i>Note 1: This may include the alignment of the project life cycle with the customer's development process.</i></p>
<p>MAN.3.BP3: Evaluate feasibility of the project. Evaluate the feasibility of achieving the goals of the project with respect to time, project estimates, and available resources.</p> <p><i>Note 2: The evaluation of feasibility may consider technical constraints of the project.</i></p>
<p>MAN.3.BP4: Define and monitor work packages. Define and monitor work packages and their dependencies according to defined project life cycle and estimations.</p> <p><i>Note 3: The structure and the size of the work packages support an adequate progress monitoring.</i></p> <p><i>Note 4: Work packages may be organized in a work breakdown structure.</i></p>

MAN.3.BP5: Define and monitor project estimates and resources. Define and monitor project estimates of effort and resources based on project's goals, project risks, motivation and boundaries.

Note 5: Examples of necessary resources are budget, people or infrastructure

Note 6: Project risks (using MAN.5) and quality criteria (using SUP.1) may be considered.

Note 7: Estimations and resources may include engineering, management and supporting processes.

MAN.3.BP6: Define and monitor required skills, knowledge, and experience. Identify and monitor the required skills, knowledge, and experience for the project in line with the estimates and work packages.

Note 8: Training, mentoring or coaching of individuals may be applied to resolve deviations from required skills and knowledge.

MAN.3.BP7: Define and monitor project interfaces and agreed commitments. Identify and agree interfaces of the project with affected stakeholders and monitor agreed commitments. Define an escalation mechanism for commitments that are not fulfilled.

Note 9: Affected stakeholders may include other projects, organizational units, sub-contractors, and service providers.

MAN.3.BP8: Define and monitor project schedule. Allocate resources to work packages and schedule each activity of the project. Monitor the performance of activities against schedule.

MAN.3.BP9: Ensure consistency. Regularly adjust estimates, resources, skills, work packages and their dependencies, schedules, plans, interfaces, and commitments for the project to ensure consistency with the scope of work.

Note 10: This may include the consideration of critical dependencies, that are an input for risk management.

MAN.3.BP10: Review and report progress of the project. Regularly review and report the status of the project and the fulfillment of work packages against estimated effort and duration to all affected parties. Prevent recurrence of identified problems.

Note 11: Project reviews may be executed at regular intervals by the management. At the end of a project, a project review may contribute to identify best practices and lessons learned.

Note 12: Refer to SUP.9 for resolution of problems

MAN.3 Project Management	Outcome 1	Outcome 2	Outcome 3	Outcome 4	Outcome 5	Outcome 6	Outcome 7
Output Information item							
08-53 Scope of work	X						
08-54 Feasibility analysis							
14-10 Work package			X	X	X		
13-52 Communication evidence		X	X				
13-16 Change request							X
13-51 Consistency evidence		X					X
14-02 Corrective action						X	X
18-52 Escalation path				X		X	X
08-56 Schedule			X		X		X
14-50 Stakeholder groups list				X			
15-06 Project status report				X		X	
Base Practices							
BP1: Define the scope of work	X						
BP2: Define project life cycle	X	X					
BP3: Evaluate feasibility of the project		X					
BP4: Define and monitor work packages			X	X	X		X
BP5: Define and monitor project estimates and resources		X	X				X
BP6: Define and monitor required skills, knowledge, and experience			X				X
BP7: Define and monitor project interfaces and agreed commitments			X		X		X
BP8: Define and monitor project schedule						X	X
BP9: Ensure consistency			X	X	X		X
BP10: Review and report progress of the project						X	X

4.9.2. MAN.5 Risk Management

Process ID
MAN.5
Process name
Risk Management
Process purpose
The purpose is to Regularly identify, analyze, treat and monitor process related risks and product related risks.
Process outcomes
<ol style="list-style-type: none"> 1) The sources of risks are identified and regularly updated. 2) Potential undesirable events are identified as they develop during the conduct of the project. 3) Risks are analyzed and the priority in which to apply resources to treatment of these risks is determined. 4) Risk measures are defined, applied, and assessed to determine changes in the status of risk and the progress of the risk treatment activities. 5) Appropriate treatment is taken to correct or avoid the impact of risk based on its priority, probability, and consequence or other defined risk threshold.
Base practices
<p>MAN.5.BP1: Identify sources of risks. Identify and regularly update the sources of risks with affected parties.</p> <p><i>Note 1: Risks may include technical, economic and timing risks.</i></p> <p><i>Note 2: Risks may include the suppliers deliverables and services.</i></p> <p><i>Note 3: See MAN.7 in Automotive SPICE for Cybersecurity for management of cybersecurity risks.</i></p> <p><i>Note 4: The risk sources may vary across the entire project life cycle.</i></p>
<p>MAN.5.BP2: Identify potential undesirable events. Identify potential undesirable events within the scope of the risk management for the project.</p>
<p>MAN.5.BP3: Determine risks. Determine the probability and severity of the undesirable events to support priorities for mitigating the risks.</p> <p><i>Note 5: Different methods may be used to analyze technical risks of a system, for example, functional analysis, simulation, FMEA, FTA etc.</i></p>
<p>MAN.5.BP4: Define risk treatment options. For each risk select a treatment option to accept, mitigate, avoid or share (transfer) the risk.</p>
<p>MAN.5.BP5: Define and perform risk treatment activities. Define and perform risk activities for risk treatment options.</p>

MAN.5.BP6: Monitor risks. Regularly re-evaluate the risk related to the identified potential undesirable events to determine changes in the status of a risk and to evaluate the progress of the of risk treatment activities.

Note 6: Risks of high priority may need to be communicated to and monitored by higher levels of management.

MAN.5.BP7: Take corrective action. When risk treatment activities are not effective, take appropriate corrective action.

Note 7: Corrective actions may involve reevaluation of risks, developing and implementing new mitigation concepts or adjusting the existing concepts.

MAN.5 Risk Management	Outcome 1	Outcome 2	Outcome 3	Outcome 4	Outcome 5
Output Information item					
08-55 Risk measure				X	X
13-20 Risk action request	X				X
14-02 Corrective action				X	X
15-08 Risk analysis		X	X		X
15-09 Risk status			X		X
Base Practices					
BP1: Identify sources of risks	X				
BP2: Identify potential undesirable events		X			
BP3: Determine risks			X		
BP4: Define risk treatment options				X	X
BP5: Define and perform risk treatment activities.				X	X
BP6: Monitor risks				X	
BP7: Take corrective action					X

4.9.3. MAN.6 Measurement

Process ID
MAN.6
Process name
Measurement
Process purpose
The purpose is to Collect and analyze data relating to the development results and processes implemented within the organization and its projects, to support effective management of the processes.
Process outcomes
<ol style="list-style-type: none"> 1) The measurement information needs that are necessary to evaluate the achievement of process objectives and the achievement of desired documented information are identified. 2) An appropriate set of metrics, driven by the information needs are identified and/or developed. 3) Measurement activities are identified and performed. 4) The required metrics are collected, stored, analyzed, and the results interpreted. 5) Metrics are used to support decisions and provide an objective basis for communication.
Base practices
<p>MAN.6.BP1: Identify information needs. Identify the measurement information needs that are necessary to evaluate the achievement of process objectives and documented information.</p> <p><i>Information needs may change over time. Therefore, the measurement process may be used in an iterative way.</i></p>
<p>MAN.6.BP2: Specify metrics. Identify and develop an appropriate set of metrics based on measurement information needs.</p> <p><i>Metrics may be related to processes or development results.</i></p>
<p>MAN.6.BP3: Collect and store metrics. Collect and store both base and derived metrics, including any context information necessary to verify and understand the metrics.</p> <p><i>Base metrics in the context of this process are directly gathered metrics like “number of defects found” or “number of lines of code”, where derived metrics are two or more metrics that are brought in relation to each other like “number of defects found per line of code”.</i></p>
<p>MAN.6.BP4: Analyze collected metrics. Analyze and interpret measured values to support decision-making.</p>
<p>MAN.6.BP5: Communicate analysis results. Communicate analysis results to all affected parties.</p>
<p>MAN.6.BP6: Use metrics for decision-making. Make accessible and use information from collected metrics and analysis results for any decision-making process for which it is relevant.</p>

MAN.6 Measurement	Outcome 1	Outcome 2	Outcome 3	Outcome 4	Outcome 5
Output Information item					
03-03 Benchmarking data				X	X
03-04 Customer satisfaction data				X	X
03-06 Process performance information				X	X
07-01 Customer satisfaction survey		X	X	X	X
07-04 Process metric		X	X	X	X
07-05 Project metric		X	X	X	X
07-06 Quality metric		X	X	X	X
07-08 Service level metric		X	X	X	X
15-51 Analysis results	X			X	X
Base Practices					
BP1: Identify information needs	X				
BP2: Specify metrics		X	X		
BP3: Collect and store metrics			X	X	
BP4: Analyse collected metrics				X	X
BP5: Communicate measurement information					X
BP6: Use metrics for decision-making					X

4.10. Process improvement process group (PIM)

4.10.1.PIM.3 Process Improvement

Process ID
PIM.3
Process name
Process Improvement
Process purpose
The purpose is to continually improve the organization’s effectiveness and efficiency through the processes used and ensure alignment of the processes with the business needs.
Process outcomes
1) commitment is established to provide resources to sustain improvement measures

- 2) issues arising from the organization's internal or external environment are identified as improvement opportunities and justified as reasons for change
- 3) analysis of the current status of the existing process is performed
- 4) improvement goals are identified and prioritized, and consequent changes to the process are defined, documented and implemented
- 5) the effects of process implementation are monitored, measured and confirmed against the identified improvement goals
- 6) knowledge gained from the improvement is communicated within the organization

Base practices

PIM.3.BP1: Establish commitment. Establish commitment to support the process improvement staff, to provide resources and further enablers to sustain improvement actions.

Note 1: The process improvement process is a generic process, which can be used at all levels (e.g. organizational level, process level, project level, etc.) and which can be used to improve all processes.

Note 2: Commitment at all levels of management may support process improvement.

Note 3: Enablers for improvement measures may include trainings, methods, infrastructure, etc.

PIM.3.BP2: Identify improvement measures. Identify issues from the analysis of process performance and derive improvement opportunities with justified reasons for change.

Note 4: Analysis may include problem report trend analysis (see SUP.9), analysis from Quality Assurance and Verification results and records (see SUP.1), validation results and records, and product quality metrics like defect rate.

Note 5: Issues and improvement suggestions may be addressed by the customer

Note 6: Sources for identification of issues may include: process assessment results, audits, customer's satisfaction reports, measurements of organizational effectiveness/efficiency, costs of quality.

PIM.3.BP3: Establish process improvement goals. Analyze the current status of the existing processes and establish improvement goals.

Note 7: The current status of processes may be determined by process assessment.

PIM.3.BP4: Prioritize improvements. Prioritize the improvement goals and improvement measures.

PIM.3.BP5: Define process improvement measures. Process improvement measures are defined.

Note 8: Improvements may be documented in incremental steps.

PIM.3.BP6: Implement process improvement measures. Implemented and apply the improvements to the processes. Update the Process documentation and train people as needed.

Note 8: Process application can be supported by establishing policies, adequate process infrastructure, process training, process coaching and tailoring processes to local needs.

Note 9: Improvements may be piloted before roll out within the organization.

PIM.3.BP7: Confirm process improvement. The effects of process implementation are monitored and measured, and the achievement of defined improvement goals is confirmed.

PIM.3.BP8: Communicate results of improvement. Knowledge gained from the improvements and progress of the improvement implementation is communicated to affected parties.

PIM.3 Process Improvement	Outcome 1	Outcome 2	Outcome 3	Outcome 4	Outcome 5	Outcome 6
Output Information item						
02-01 Commitment/agreement	X					
05-00 Goals				X		
06-04 Training material				X		X
07-04 Process metric					X	X
10-00 Process description				X		
13-52 Communication evidence						X
13-16 Change request		X				
15-51 Analysis result		X	X	X	X	
15-13 Assessment/audit report			X		X	
15-16 Improvement opportunity		X	X	X		
16-06 Process repository				X		
Base Practices						
BP1: Establish commitment	X					
BP2: Identify improvement measures		X	X			
BP3: Establish process improvement goals				X		
BP4: Prioritize improvements				X		
BP5: Plan process improvement measures				X		
BP6: Implement process improvement measures				X		
BP7: Confirm process improvement					X	
BP8: Communicate results of improvement						X

4.11. Reuse process group (REU)

4.11.1.REU.2 Management of Products for Reuse

Process ID
REU.2
Process name
Management of Products for Reuse
Process purpose
The purpose is to ensure that reused work products are analyzed, verified, and approved for their target context.
Process outcomes
<ol style="list-style-type: none"> 1) products for reuse are selected using defined criteria 2) products for reuse are analyzed for portability and interoperability 3) limitations for reuse are defined and communicated 4) products for reuse are verified 5) products for reuse are provided to affected parties 6) communication mechanism is established with the reuse product provider

Base practices
<p>REU.2.BP1: Select products for reuse. Select the products to be reused using defined criteria.</p> <p><i>Note 1: Products for reuse may be systems, hardware or software components, third party components or legacy components.</i></p>
<p>REU.2.BP2: Analyze the reuse capability of the product. Analyze the designated target architecture and the product to be reused to determine its applicability in the target architecture according to relevant criteria.</p> <p><i>Note 2: Examples for criteria can be requirements compliance, verifiability of the product to be reused in the target architecture, or portability/interoperability.</i></p>
<p>REU.2.BP3: Define limitations for reuse. Define and communicate limitations for the products to be reused.</p> <p><i>Note 3: Limitations may address parameters of operational environment.</i></p>
<p>REU.2.BP4: Ensure qualification of products for reuse. Provide evidence that the product for reuse is qualified for the intended use of the deliverable.</p> <p><i>Note 4: Qualification may be demonstrated by verification evidence.</i></p> <p><i>Note 5: Verification may include the appropriateness of documentation.</i></p>

REU.2.BP5: Provide products for reuse. Make available the product to be reused to affected parties.

Note 6: Refer to HWE.3, SWE.5 or SYS.4 for more information on integration of hardware, software, or system components.

REU.2.BP6: Communicate information about effectiveness of reuse activities. Establish communication and notification mechanism about experiences and technical outcomes to the provider of reused products.

Note 7: The communication with the provider of a reused product may depend on whether the product is under development or not.

REU.2 Management of Products for Reuse	Outcome 1	Outcome 2	Outcome 3	Outcome 4	Outcome 5	Outcome 6
Output Information item						
04-02 Domain architecture		X	X			
12-03 Reuse candidate	X				X	
13-52 Communication evidence						X
15-07 Reuse analysis evidence		X	X			
13-53 Qualification evidence				X		
Base Practices						
BP1: Select products for reuse	X					
BP2: Analyze the reuse capability of the product		X				
BP3: Define limitations for reuse			X			
BP4: Ensure qualification of products for reuse				X		
BP5: Provide products for reuse					X	
BP6: Communicate information about effectiveness of reuse activities						X

5. Process capability levels and process attributes

The definition of process capability indicators for each process attribute is an integral part of a measurement framework. Process capability indicators such as generic practices and information items are the means to support the judgment of the degree of achievement of the associated process attribute.

This chapter defines the generic practices and information items and their mapping to the process attributes for each capability level defined in the measurement framework [3.2].

Note: Due to lack of a defined process attribute for process capability level 0, no generic practices and information items are defined.

Process capability level	<p>Process attribute ID</p> <p>Process attribute name</p> <p>Process attribute scope</p> <p>Process achievements</p>	<p>Each process attribute is identified with a unique identifier and name. A process attribute scope statement is provided, and process achievements are defined.</p>
	<p>Generic practices</p>	<p>A set of generic practices for the process attribute providing a definition of the activities to be performed to accomplish the process attribute scope and fulfill the process achievements.</p> <p>The generic practice headers are summarized at the end of a process to demonstrate their relationship to the process attribute achievements.</p>
Process attribute achievement indicators	<p>Output information items</p>	<p>The output information items that are relevant to accomplish the process attribute scope and fulfill the process achievements are summarized at the end of a process attribute section to demonstrate their relationship to the process achievements.</p> <p><i>Note: Refer to Annex B for the characteristics of each information item.</i></p>

Table 22 — Template for the process description

5.1. Process capability level 0: Incomplete process

The process is not implemented or fails to achieve its process purpose. At this level there is little or no evidence of any systematic achievement of the process purpose.

5.2. Process capability Level 1: Performed process

The implemented process achieves its process purpose. The following process attribute demonstrates the achievement of this level.

5.2.1. PA 1.1 Process performance process attribute

Process attribute ID	
PA 1.1	
Process attribute name	
Process performance process attribute	
Process attribute scope	
The process performance process attribute is a measure of the extent to which the process purpose is achieved.	
Process attribute achievements	
1) The process achieves its defined outcomes	
Generic practices	
GP 1.1.1 Achieve the process outcomes Achieve the intent of the base practices. Produce work products that evidence the process outcomes.	
PA 1.1 Process performance process attribute	Achievement a
Information Items	
Process specific information items, as described in chapter 4	X
Generic practices	
GP 1.1.1 Achieve the process outcomes	X

5.3. Process capability Level 2: Managed process

The following process attributes, together with the previously defined process attribute, demonstrate the achievement of this level.

5.3.1. PA 2.1 Process performance management process attribute

Process attribute ID
PA 2.1
Process attribute name
Process performance management process attribute
Process attribute scope
The performance management process attribute is a measure of the extent to which the performance of the process is managed.
Process attribute achievements
<ol style="list-style-type: none"> 1) Strategy for the performance of the process is defined based on identified objectives 2) Performance of the process is planned 3) Performance of the process is monitored and adjusted to meet the planning 4) Needs for human resources including responsibilities and authorities for performing the process are determined 5) Needs for physical and material resources are determined 6) Personnel performing the process are prepared for executing their responsibilities 7) Physical and material resources for performing the process are identified, made available, allocated and used 8) Interfaces between the involved parties are managed to ensure both effective communication and the assignment of responsibilities

Generic practices
<p>GP 2.1.1: Identify the objectives and define a strategy for the performance of the process.</p> <p>The scope of the process activities including the management of process performance and the management of work products are determined.</p> <p>Corresponding results to be achieved are determined.</p> <p>Process performance objectives and associated criteria are identified.</p> <p><i>Note 1: Budget targets and delivery dates to the customer, targets for test coverage and process lead time are examples for process performance objectives.</i></p> <p><i>Note 2: Performance objectives are the basis for planning and monitoring.</i></p> <p>Assumptions and constraints are considered when identifying the performance objectives.</p> <p>Approach and methodology for the process performance is determined.</p> <p><i>Note 3: A process performance strategy may not necessarily be document-ed specifically for each process. Elements applicable for multiple processes may be documented jointly, e.g. as part of a common project handbook or in a joint test strategy.</i></p>

GP 2.1.2: Plan the performance of the process.

The planning for the performance of the process is established according to the defined objectives, criteria, and strategy.

Process activities and work packages are defined.

Estimates for work packages are identified using appropriate methods.

Note 4: Schedule and milestones are defined.

GP 2.1.3: Determine resource needs.

The required amount of human resources and the needs for process performance experience, knowledge and skills are determined.

The needs for physical and material resources are determined.

Note 5: Physical and material resources may include equipment, laboratories, materials, tools, licenses etc.

Required responsibilities and authorities to perform the process, and to manage the corresponding work products are determined.

Note 6: The definition of responsibilities and authorities does not necessarily require formal role descriptions.

GP 2.1.4: Identify and make available resources.

The individuals performing and managing the process are identified and allocated according to the determined needs.

The individuals performing and managing the process are being qualified to execute their responsibilities.

Note 7: Qualification of individuals may include training, mentoring, or coaching.

The other resources, necessary for performing the process are identified, made available, allocated and used according to the determined needs.

GP 2.1.5: Monitor and adjust the performance of the process.

Process performance is monitored to identify deviations from the planning.

Appropriate actions in case of deviations from the planning are taken.

The planning is adjusted as necessary.

GP 2.1.6: Manage the interfaces between involved parties.

The individuals and groups including required external parties involved in the process performance are determined.

Responsibilities are assigned to the relevant individuals or parties.

Communication mechanisms between the involved parties are determined.

Effective communication between the involved parties is established and maintained.

PA 2.1 - Process Performance Management	Achievement 1	Achievement 2	Achievement 3	Achievement 4	Achievement 5	Achievement 6	Achievement 7	Achievement 8
Output Information Item								
19-01 Process performance strategy	X							
18-58 Process performance quality criteria	X							
14-10 Work package		X						
08-56 Schedule		X	X					
13-14 Progress status record			X					
17-55 Resource needs				X	X			
08-61 Resource allocation						X	X	
08-62 Communication matrix								X
13-52 Communication evidence								X
Generic Practices								
GP 2.1.1: Identify the objectives and define a strategy for the performance of the process	X							
GP 2.1.2: Plan the performance of the process		X						
GP 2.1.3: Determine resource needs				X	X			
GP 2.1.4: Identify and make available resources						X	X	
GP 2.1.5: Monitor and adjust the performance of the process			X					
GP 2.1.6: Manage the interfaces between involved parties								X

5.3.2. PA 2.2 Work product management process attribute

Process attribute ID
PA 2.2
Process attribute name
Work product management process attribute
Process attribute scope
The work product management process attribute is a measure of the extent to which the work products produced by the process are appropriately managed.
Process attribute achievements
<ol style="list-style-type: none"> 1) Requirements for the work products of the process are defined 2) Requirements for storage and control of the work products are defined 3) The work products are appropriately identified, stored, and controlled 4) The work products are reviewed and adjusted as necessary to meet requirements

Generic practices
<p>GP 2.2.1 Define the requirements for the work products.</p> <p>The requirements for the content and structure of the work products to be produced are defined. Quality criteria for the work products are identified. Appropriate review and approval criteria for the work products are defined.</p> <p><i>Note 1: Possible sources of documentation requirements may be best practices or lessons learned from other projects, standards, organization requirements, customer requirements, etc.</i></p> <p><i>Note 2: There may be types of work products where no review or approval is required, thus no need for the corresponding criteria is given.</i></p>
<p>GP 2.2.2 Define the requirements for storage and control of the work products.</p> <p>Requirements for the storage and control of the work products are defined, including their identification and distribution.</p> <p><i>Note 1: Possible sources for the identification of requirements for storage and control may be:</i></p> <ul style="list-style-type: none"> • Legal requirements, data policies • Best practices from other projects • Tool related requirements, etc. <p><i>Note 2: Examples for work product storage are files in a file system, ticket in a tool, Wiki entry, paper documents etc.</i></p> <p><i>Note 3: Where a status attribute was required for control of work products in the base practices, this should now usually be managed with a defined status model.</i></p>

GP 2.2.3 Identify, store and control the work products.

The work products to be controlled are identified.
 The work products are stored and controlled in accordance with the requirements.
 Change control is established for work products.
 Versioning and baselining of the work products is performed in accordance with the requirements.
 The work products including the revision status are made available through appropriate mechanisms.

GP 2.2.4 Review and adjust work products.

The work products are reviewed against the defined criteria.
 Resolution of issues arising from work products reviews is ensured.

PA 2.2 - Work product management process attribute	Achievement 1	Achievement 2	Achievement 3	Achievement 4
Output Information Item				
17-05 Requirements for work products	X	X		
18-59 Review and approval criteria for work products	X			
18-60 Quality criteria for work products	X			
13-19 Review record				X
13-08 Baseline			X	
16-00 Repository			X	
Generic Practices				
GP 2.2.1 Define the requirements for the work products	X			
GP 2.2.2 Define the requirements for storage and control of the work products.		X		
GP 2.2.3 Identify, store and control the work products.			X	
GP 2.2.4 Review and adjust work products.				X

5.4. Process capability Level 3: Established process

The following process attributes, together with the previously defined process attributes, demonstrate the achievement of this level.

5.4.1. PA 3.1 Process definition process attribute

Process attribute ID
PA 3.1
Process attribute name
Process definition process attribute
Process attribute scope
The process definition process attribute is a measure of the extent to which a standard process is maintained to support the deployment of the defined process.
Process attribute achievements
<ol style="list-style-type: none"> 1) A standard process is developed, established, and maintained that describes the fundamental elements that must be incorporated into a defined process 2) the required inputs and the expected outputs for the standard process are defined 3) roles, responsibilities, authorities, and required competences for performing the standard process are defined 4) tailoring guidelines for deriving the defined process from the standard process are defined 5) Required infrastructure and work environment for performing the process are identified as part of the standard process 6) Suitable methods and required activities for monitoring the effectiveness, suitability and adequacy of the process are determined

Generic practices
<p>GP 3.1.1 Establish and maintain the standard process.</p> <p>A standard process is developed including required activities and their interactions. Inputs and outputs of the standard process are defined including the corresponding entry and exit criteria to determine the interactions and sequence with other processes.</p> <p>Process performance roles are identified and assigned to the standard process activities including their level of involvement, responsibilities, and authorities.</p> <p><i>Note 1: An example for describing the involvement of the process roles in the activities is a RASI/RASIC representation.</i></p> <p>Guidance, procedures, and templates are provided to support execution of the process as needed.</p> <p><i>Note 2: Procedures may also include description of specific methods to be used.</i></p> <p>Appropriate tailoring guidelines including corresponding criteria are defined based on identified deployment needs and context of the standard process.</p> <p>The standard process is maintained according to corresponding feedback from monitoring the deployed process.</p> <p><i>Note 3: For guidance on how to perform process improvements see the Process Improvement process (PIM.3).</i></p>

GP 3.1.2 Determine the required competencies.

Required competencies, skills, and experience for performing the process are determined for the identified roles.

Appropriate qualification methods to acquire the necessary competencies and skills are determined, maintained and available for the identified roles

Note 4: Qualification methods are e.g., trainings, mentoring, self-study. Preparation includes e.g., identification or definition of trainings, mentoring concepts, self-learning material.

GP 3.1.3 Determine the required resources.

Required physical and material resources and process infrastructure needs are determined.

Note 5: This may include e.g., facilities, tools, licenses, networks, services, and samples supporting the establishment of the required work environment.

GP 3.1.4 Determine suitable methods to monitor the standard process.

Methods and required activities for monitoring the effectiveness, suitability and adequacy of the standard process are determined.

Note 6: Methods and activities to gather feedback regarding the standard process may be lessons learned, process compliance checks, internal audits, management reviews, change requests, etc.

Appropriate criteria and information needed to monitor the standard process are defined.

Note 7: Information about process performance may be of qualitative or quantitative nature.

	Achievement 1	Achievement 2	Achievement 3	Achievement 4	Achievement 5	Achievement 6
PA 3.1 - Process definition process attribute						
Output Information Item						
06-51 Tailoring guideline				X		
08-63 Process monitoring method						X
10-00 Process description	X	X				
10-50 Role description			X			
10-51 Qualification method description			X			
10-52 Process resource and infrastructure description					X	
Generic Practices						
GP 3.1.1 Establish and maintain the standard process	X	X	X	X		
GP 3.1.2 Determine the required competencies			X			
GP 3.1.3 Determine the required infrastructure					X	
GP 3.1.4 Determine suitable methods to monitor the standard process						X

5.4.2. PA 3.2 Process deployment process attribute

Process attribute ID
PA 3.2
Process attribute name
Process deployment process attribute
Process attribute scope
The process deployment process attribute is a measure of the extent to which the standard process is deployed as a defined process to achieve its process outcomes.
Process attribute achievements
<ol style="list-style-type: none"> 1) A defined process is deployed based upon an appropriately selected and/or tailored standard process 2) Assignment of persons necessary for performing the defined process to roles is performed and communicated including their responsibilities and authorities 3) Required education, training and experience is ensured and monitored for the person(s) assigned to the roles 4) Required resources for performing the defined process are made available, allocated, and maintained 5) Appropriate information is collected and analysed as a basis for understanding the behaviour of the process

Generic practices
<p>GP 3.2.1 Deploy a defined process that satisfies the context specific requirements of the use of the standard process.</p> <p>The defined process is appropriately selected and/or tailored from the standard process. Conformance of defined process with standard process requirements and tailoring criteria is verified.</p> <p>The defined process is used as managed process to achieve the process outcomes.</p> <p><i>Note 1: Changes in the standard process may require updates of the defined process.</i></p>
<p>GP 3.2.2 Ensure required competencies for the defined roles.</p> <p>Persons are allocated to the defined roles according to the required competencies and skills. Assignment of persons to roles and corresponding responsibilities and authorities for performing the defined process are communicated.</p> <p>Gaps in competencies and skills are identified, and corresponding qualification measures are initiated and monitored.</p> <p>Availability and usage of the project staff are measured and monitored.</p>
<p>GP 3.2.3 Ensure required resources to support the performance of the defined process.</p>

Required information to perform the process is made available, allocated and used.
 Required physical and material resources, process infrastructure and work environment are made available, allocated and used.
 Availability and usage of resources are measured and monitored.

GP 3.2.4 Monitor the performance of the defined process.

Information is collected and analyzed to understand the behavior of the process, and to evaluate the suitability, effectiveness, and adequacy of the defined process.

Results of the analysis are used to identify where continual improvement of the standard and/or defined process can be made.

Note 2: For guidance on how to perform process improvements see the Process Improvement process (PIM.3).

	Achievement 1	Achievement 2	Achievement 3	Achievement 4	Achievement 5
PA 3.2 - Process deployment process attribute					
Output Information Item					
10-00 Process description	X				
15-54 Tailoring documentation	X				
14-53 Role assignment		X	X		
13-55 Process resource and infrastructure documentation				X	
03-06 Process performance information					X
Generic Practices					
GP 3.2.1 Deploy a defined process	X				
GP 3.2.2 Ensure required competencies		X	X		
GP 3.2.3 Ensure required resources				X	
GP 3.2.4 Monitor the performance of the defined process					X

5.5. Process capability Level 4: Predictable process

The following process attributes, together with the previously defined process attributes, demonstrate the achievement of this level.

5.5.1. PA 4.1 Quantitative analysis process attribute

Process attribute ID
PA 4.1
Process attribute name
Quantitative analysis process attribute
Process attribute scope
The quantitative analysis process attribute is a measure of the extent to which information needs are defined, relationships between process elements are identified and data are collected.
Process attribute achievements
<ol style="list-style-type: none"> 1) Process information needs in support of relevant defined quantitative business goals are established. 2) Measurable relationships between process elements that contribute to the process performance, and data collection techniques and data collection frequency, are identified. 3) Process measurement objectives are derived from process information needs. 4) Techniques for analyzing the collected data are selected. 5) Quantitative control limits for process performance in support of relevant business goals are established. 6) Results of measurement are collected, validated and reported in order to monitor the extent to which the quantitative targets/objectives for process performance are met. <p><i>Note: Information needs typically reflect management, technical, project, process or product needs.</i></p>

Generic practices
<p>GP 4.1.1 Identify business goals. Business goals are identified that are supported by the quantitatively measured process.</p>
<p>GP 4.1.2 Establish process information needs. Stakeholders of the identified business goals and the quantitatively measured process are identified, and their information needs are defined and agreed.</p>
<p>GP 4.1.3 Identify measurable relationships between process elements. Identify the relationships between process elements, or sets of process elements, which contribute to the process information needs.</p> <p><i>Note 1: Examples of process elements are documented information items, activities, tasks.</i></p>

GP 4.1.4 Derive process measurement approach and select analysis techniques.

Based on the measurable relationships of process elements, or set of process elements, the process measurement metrics are derived to satisfy the established process information needs.

Frequency of data collection is defined.

Select analysis techniques, appropriate to collected data.

Algorithms and methods to create derived measurement results from base measures are defined, as appropriate.

Verification mechanism for base and derived measures is defined.

Note 2: Typically, the standard process definition is extended to include the collection of data for process measurement.

GP 4.1.5 Establish quantitative control limits.

Establish quantitative control limits for the derived metrics. Agreement with process stakeholders is established.

GP 4.1.6 Collect product and process measurement results through performing the defined process.

Data collection mechanisms are created for all identified metrics.

Required data is collected across process instances of within the defined frequency and recorded.

Measurement results are analyzed and reported to the identified stakeholders.

Note 3: A product measure can contribute to a process measure, e.g., the productivity of testing characterized by the number of defects found in a given timeframe in relation to the product defect rate in the field.

PA 4.1 - Quantitative analysis process attribute	Achievement 1	Achievement 2	Achievement 3	Achievement 4	Achievement 5	Achievement 6
Output Information Items						
IIC's will be added with the final version of Automotive SPICE 4.0						
Generic Practices						
GP 4.1.1 Identify business goals	X					
GP 4.1.2 Establish process information needs	X					
GP 4.1.3 Identify measurable relationships between process elements		X				

GP 4.1.4 Derive process measurement approach and select analysis techniques			X	X		
GP 4.1.5 Establish quantitative control limits					X	
GP 4.1.6 Collect product and process measurement results through performing the de-fined process						X

5.5.2. PA 4.2 Quantitative control process attribute

Process attribute ID
PA 4.2
Process attribute name
Quantitative control process attribute
Process attribute scope
The quantitative control process attribute is a measure of the extent to which objective data are used to manage process performance that is predictable.
Process attribute achievements
<ol style="list-style-type: none"> 1) Variations in process performance are identified. 2) Assignable causes of process variation are determined through analysis of the collected quantitative data. 3) Distributions that characterize the performance of the process are established. 4) Corrective actions are taken to address assignable causes of variation.
Generic practices
<p>GP 4.2.1 Identify variations in process performance.</p> <p>Deviations in the performance of process instances from the established quantitative control limits are determined based on the collected quantitative measurement data.</p>
<p>GP 4.2.2 Identify causes of variation.</p> <p>The determined deviations in process performance are analyzed to identify potential cause(s) of variation using the defined analysis techniques.</p> <p>Distributions are used to quantitatively understand the variation of process performance under the influence of potential causes of variation.</p> <p>Consequences of process variation are analyzed.</p>
<p>GP 4.2.3 Identify and implement corrective actions to address assignable causes.</p> <p>Results are provided to those responsible for taking action.</p> <p>Corrective actions are determined and implemented to address each assignable cause of variation.</p> <p>Corrective action results are monitored and evaluated to determine their effectiveness.</p> <p><i>Note 1: Assignable cause may indicate a possible problem in the defined process.</i></p>

PA 4.2 - Quantitative control process attribute	Achievement a	Achievement b	Achievement c	Achievement d
Output Information Items				
<div style="border: 1px solid black; padding: 2px; color: red;">IIC's will be added with the final version of Automotive SPICE 4.0</div>				
Generic Practices				
GP 4.2.1 Identify variations in process performance	X			
GP 4.2.2 Identify causes of variation		X	X	
GP 4.2.3 Identify and implement corrective actions to address assignable causes				X

5.6. Process capability Level 5: Innovating process

The following process attributes, together with the previously defined process attributes, demonstrate the achievement of this level.

5.6.1. PA 5.1 Process innovation process attribute

Process attribute ID
PA 5.1
Process attribute name
Process innovation process attribute
Process attribute scope
The process innovation process attribute is a measure of the extent to which changes to the process are identified from investigations of innovative approaches to the definition and deployment of the process.
Process attribute achievements
<ol style="list-style-type: none"> 1) Process innovation objectives are defined that support the relevant business goals. 2) Quantitative data are analyzed to identify opportunities for innovation. 3) Innovation opportunities derived from new technologies and process concepts are identified.
Generic practices
<p>GP 5.1.1 Define the process innovation objectives for the process that support the relevant business goals.</p> <p>New business visions and goals are analyzed to give guidance for new process objectives and potential areas of process innovation.</p> <p>Quantitative and qualitative process innovation objectives are defined and documented.</p>
<p>GP 5.1.2 Analyze quantitative data of the process.</p> <p>Common causes of variation in process performance are identified and analyzed to get a quantitative understanding of their impact.</p>
<p>GP 5.1.3 Identify innovation opportunities.</p> <p>Identify opportunities for innovation based on the quantitative understanding of the analyzed data.</p> <p>Industry best practices, new technologies and process concepts are identified and evaluated. Feedback on opportunities for innovation is actively sought.</p> <p>Emergent risks are considered in evaluating improvement opportunities.</p>

PA 5.1 - Process innovation process attribute	Achievement 1	Achievement 2	Achievement 3
Information Items			
<div style="border: 1px solid black; padding: 5px; color: red;">IIC's will be added with the final version of Automotive SPICE 4.0</div>			
Generic Practices			
GP 5.1.1 Define the process innovation objectives for the process that support the relevant business goals	X		
GP 5.1.2 Analyze quantitative data of the process		X	
GP 5.1.3 Identify innovation opportunities			X

5.6.2. PA 5.2 Process innovation implementation process attribute

Process attribute ID
PA 5.2
Process attribute name
Process innovation implementation process attribute
Process attribute scope
The process innovation process implementation attribute is a measure of the extent to which changes to the definition, management and performance of the process achieves the relevant process innovation objectives.
Process attribute achievements
<ol style="list-style-type: none"> 1) Impact of all proposed changes is assessed against the objectives of the defined process and standard process. 2) Implementation of all agreed changes is managed to ensure that any disruption to the process performance is understood and acted upon. 3) Effectiveness of process change on the basis of quantitative performance and innovation feedback.
Generic practices
<p>GP 5.2.1 Define, and assess the impact of, proposed changes.</p> <p>Specified changes are assessed against product quality and process performance requirements and goals.</p> <p>Impact of changes to other defined and standard processes is considered.</p> <p>Objective priorities for process innovation are established.</p> <p>Commitment to innovation is demonstrated by organizational management including other relevant stakeholders.</p>
<p>GP 5.2.2. Implement agreed process changes.</p> <p>A mechanism is established for incorporating accepted changes into the defined and standard process(es) effectively and completely.</p> <p>Process changes are implemented and effectively communicated to all affected parties.</p>

GP 5.2.3 Evaluate the effectiveness of process change.

Performance and capability of the changed process are measured and compared with historical data.

Performance and capability of the changed process are analyzed to determine whether the process performance has improved with respect to common causes of variations.

Other feedback is recorded, such as opportunities for further innovation of the standard process.

A mechanism is available for documenting and reporting analysis results to stakeholders of standard and defined process.

PA 5.2 - Process innovation implementation attribute	Achievement a	Achievement b	Achievement c
Information Items			
<div style="border: 1px solid black; padding: 5px; color: red;">IIC's will be added with the final version of Automotive SPICE 4.0</div>			
Generic Practices			
GP 5.2.1 Define, and assess the impact of, proposed changes	X		
GP 5.2.2. Implement agreed process changes		X	
GP 5.2.3 Evaluate the effectiveness of process change			X

Annex A Conformity statements

Annex A.1 Introduction

The Automotive SPICE process assessment and process reference model are meeting the requirements for conformance defined in ISO/IEC 33004. The process assessment model can be used in the performance of assessments that meet the requirements of ISO/IEC 33002.

This clause serves as the statement of conformance of the process assessment and process reference models to the requirements defined in ISO/IEC 33004.

| [ISO/IEC 33004, 5.5 and 6.4]

Due to copyright reasons each requirement is only referred by its number. The full text of the requirements can be drawn from ISO/IEC 33004.

Annex A.2 Conformance to the requirements for process reference models

Clause 5.3, "Requirements for process reference models"

The following information is provided in chapter 1 and 3 of this document:

- the declaration of the domain of this process reference model;
- the description of the relationship between this process reference model and its intended context of use; and
- the description of the relationship between the processes defined within this process reference model.

The descriptions of the processes within the scope of this process reference model meeting the requirements of ISO/IEC 33004 clause 5.4 is provided in chapter 4 of this document.

| [ISO/IEC 33004, 5.3.1]

The relevant communities of interest and their mode of use and the consensus achieved for this process reference model is documented in the copyright notice and the scope of this document.

| [ISO/IEC 33004, 5.3.2]

The process descriptions are unique. The identification is provided by unique names and by the identifier of each process of this document.

| [ISO/IEC 33004, 5.3.3]

Clause 5.4, "Process descriptions"

These requirements are met by the process descriptions in chapter 4 of this document.

| [ISO/IEC 33004, 5.4]

Annex A.3 Conformance to the requirements for process assessment models

Clause 6.1, "Introduction"

The purpose of this process assessment model is to support assessment of process capability within the automotive domain using the process measurement framework defined in ISO/IEC 33020:2015.

| [ISO/IEC 33004, 6.1]

Clause 6.2, "Process assessment model scope"

The process scope of this process assessment model is defined in the process reference model included in chapter 3.1 of this document. The Automotive SPICE process reference model is satisfying the requirements of ISO/IEC 33004, clause 5 as described in Annex A.2.

The process capability scope of this process assessment model is defined in the process measurement framework specified in ISO/IEC 33020:2015, which defines a process measurement framework for process capability satisfying the requirements of ISO/IEC 33003.

| [ISO/IEC 33004, 6.2]

Clause 6.3, "Requirements for process assessment models"

The Automotive SPICE process assessment model is related to process capability.

| [ISO/IEC 33004, 6.3.1]

This process assessment model incorporates the process measurement framework specified in ISO/IEC 33020:2015, which satisfies the requirements of ISO/IEC 33003.

| [ISO/IEC 33004, 6.3.2]

This process assessment model is based on the Automotive SPICE Reference Model included in this document.

This process assessment model is based on the Measurement Framework defined in ISO/IEC 33020:2015.

| [ISO/IEC 33004, 6.3.3]

The processes included in this process assessment model are identical to those specified in the Process Reference Model

| [ISO/IEC 33004, 6.3.4]

For all processes in this process assessment model all levels defined in the process measurement framework from ISO/IEC 33020:2015 are addressed.

| [ISO/IEC 33004, 6.3.5]

This process assessment model defines

- the selected process quality characteristic;
- the selected process measurement framework;
- the selected process reference model(s);
- the selected processes from the process reference model(s)

in chapter 3 of this document.

| [ISO/IEC 33004, 6.3.5 a-d]

In the capability dimension, this process assessment model addresses all of the process attributes and capability levels defined in the process measurement framework in ISO/IEC 33020.

| [ISO/IEC 33004, 6.3.5 e]

Clause 6.3.1, "Assessment indicators"

Note: Due to an error in numbering in the published version of ISO/IEC 33004 the following reference numbers are redundant to those stated above. To refer to the correct clauses from ISO/IEC 33004, the text of clause heading is additionally specified for the following three requirements.

The Automotive SPICE process assessment model provides a two-dimensional view of process capability for the processes in the process reference model, through the inclusion of assessment indicators as defined in chapter 3.3. The assessment indicators used are:

- Base practices and information items

| [ISO/IEC 33004, 6.3.1 a, "Assessment indicators"]

- Generic practices and Generic resources

| [ISO/IEC 33004, 6.3.1 b, "Assessment indicators"]

Clause 6.3.2, "Mapping process assessment models to process reference models"

The mapping of the assessment indicators to the purpose and process outcomes of the processes in the process reference model is included in each description of the base practices in chapter 4.

The mapping of the assessment indicators to the process attributes in the process measurement framework including all of the process attribute achievements is included in each description of the generic practices in chapter 5.

Each mapping is indicated by a reference in square brackets.

| [ISO/IEC 33004, 6.3.2, "Mapping process assessment models"]

Clause 6.3.3, "Expression of assessment results"

The process attributes and the process attribute ratings in this process assessment model are identical to those defined in the measurement framework. As a consequence, results of assessments based upon this process assessment model are expressed directly as a set of process attribute ratings for each process within the scope of the assessment. No form of translation or conversion is required.

| [ISO/IEC 33004, 6.3.3, "Expression of assessment results"]

Annex A.4 Conformance to the requirements for measurement frameworks

This chapter will be updated with the final version of this document

Annex B Information item characteristics

Information item characteristics listed in this Annex can be used when reviewing potential outputs of process implementation. The characteristics are provided as guidance for the attributes to look for, in a particular sample work product, to provide objective evidence supporting the assessment of a particular process.

A documented process and assessor judgment is needed to ensure that the process context (application domain, business purpose, development methodology, size of the organization, etc.) is considered when using this information.

Information items are defined using the schema in table B.1. Information items and their characteristics should be considered as a starting point for considering whether, given the context, they are contributing to the intended purpose of the process, not as a check-list of what every organization must have.

Information item identifier	An identifier number for the Information item which is used to reference the Information item.
Information item name	Provides an example of a typical name associated with the Information item characteristics. This name is provided as an identifier of the type of Information item the practice or process might produce. Organizations may call these Information items by different names. The name of the Information item in the organization is not significant. Similarly, organizations may have several equivalent Information items which contain the characteristics defined in one Information item type. The formats for the Information items can vary. It is up to the assessor and the organizational unit coordinator to map the actual Information items produced in their organization to the examples given here.
Information item characteristics	Provides examples of the potential characteristics associated with the Information item types. The assessor may look for these in the samples provided by the organizational unit.

Table B.1 — Structure of IIC tables

Information items (with the ID NN-00) are sets of characteristics that would be expected to be evident in Information items of generic types as a result of achievement of an attribute. The generic Information items form the basis for the classification of specific Information items defined as process performance indicators.

Specific Information item types are typically created by process owners and applied by process deployers in order to satisfy an outcome of a particular process purpose.

*Note: The generic Information items denoted with * are not used in the Automotive SPICE process assessment model but are included for completeness.*

ID	Name	Characteristics
01-00	Configuration item	<ul style="list-style-type: none"> • Item which is maintained under configuration control: <ul style="list-style-type: none"> - may include components, subsystems, libraries, test cases, compilers, data, documentation, physical media, and external interfaces - Version identification is maintained • Description of the item is available including the: <ul style="list-style-type: none"> - type of item - associated configuration management library, file, system - responsible owner - date when placed under configuration control - status information (i.e., development, baselined, released) - relationship to lower level configured items - identification of the change control records - identification of change history
01-03	Software component	<ul style="list-style-type: none"> • Software element in the software architecture above the software unit level. • Represented by a design model element or executable code such as libs or scripts and a configuration description, if applicable.
01-50	Integrated software	<ul style="list-style-type: none"> • Software executable (e.g. simulator with stubbing, debug-able, object code) including <ul style="list-style-type: none"> - application parameters values - all configured software elements
01-51	Application parameter	<ul style="list-style-type: none"> • Name • Description • Value domain, threshold values, characteristic curves • Owner • Means of data application (e.g. flashing interfaces) • If necessary a grouping/a categorization: <ul style="list-style-type: none"> - name of the category/group/file name - description • Actual value or characteristic curve applied
01-52	Configuration item list	<ul style="list-style-type: none"> • Identifies items under configuration control • Identifies the name of work product and an associated reference (to file, to tool artifact) • Identifies required attributes
01-53	Trained ML model	<ul style="list-style-type: none"> • The trained ML model is the output of the training process. It consists of the software representing the ML architecture, the set of weights which were optimized during the training, and the final set of hyperparameters.
01-54	Hyperparameter	<ul style="list-style-type: none"> • Hyperparameters are a subset of application parameters (see 01-51) and are used to control the ML model which has to be trained, e.g. <ul style="list-style-type: none"> - Learn rate of training - Scaling of network (number of layers or neurons per layer) - Loss function

ID	Name	Characteristics
02-00	Contract	<ul style="list-style-type: none"> • Defines what is to be purchased or delivered • Identifies time frame for delivery or contracted service dates • Identifies any statutory requirements • Identifies monetary considerations • Identifies any warranty information • Identifies any copyright and licensing information • Identifies any customer service requirements • Identifies service level requirements • References to any performance and quality expectations/constraints/monitoring • Standards and procedures to be used • Evidence of review and approval • As appropriate to the contract the following are considered: <ul style="list-style-type: none"> - references to any acceptance criteria - references to any special customer needs (i.e., confidentiality requirements, security, hardware, etc.) - references to any change management and problem resolution procedures - identification of any interfaces to independent agents and subcontractors - identification of customer's role in the development and maintenance process - identification of resources to be provided by the customer
02-01	Commitment / agreement	<ul style="list-style-type: none"> • Signed off by all parties involved in the commitment/agreement • Establishes what the commitment is for • Establishes the resources required to fulfill the commitment, such as: <ul style="list-style-type: none"> - time - people - budget - equipment - facilities
03-00	Data	<ul style="list-style-type: none"> • A reinterpretable representation of information in a formalized manner suitable for communication, interpretation, or communication, or processing
03-03	Benchmarking data	<ul style="list-style-type: none"> • Results of measurement of current performance that allow comparison against historical or target values • Relates to key goals/process/product/market need criteria and information to be benchmarked
03-04	Customer satisfaction data	<ul style="list-style-type: none"> • Determines levels of customer satisfaction with products and services • Mechanism to collect data on customer satisfaction: <ul style="list-style-type: none"> - results of field performance data - results of customer satisfaction survey - interview notes - meeting minutes from customer meetings

ID	Name	Characteristics
03-06	Process Performance information	<ul style="list-style-type: none"> • Data comparing process performance against expected levels • Defined input and output work products available • Meeting minutes • Change records • Task completion criteria met • Quality criteria met • Resource allocation and tracking
03-50	Verification Measure data	<ul style="list-style-type: none"> • Verification measure data are data recorded during the execution of a verification measure, e.g. <ul style="list-style-type: none"> - for test cases: raw data, logs - measurements: values - calculations: values - simulations: protocol - reviews such as optical inspections à findings record - analyses: values
03-51	ML data set	<ul style="list-style-type: none"> • Selection of ML Data for e.g., ML model training (ML Training and Validation Data Set) or test of the trained and deployed model (ML Test Data Set).
03-53	ML data	<ul style="list-style-type: none"> • Datum to be used for Machine Learning. The datum has to be attributed by metadata, e.g., unique ID and data characteristics. Examples: <ul style="list-style-type: none"> - Visual data like a photo or videos (but a video could also be considered as sequence of photos depending on the intended use) - Audio recording - Sensor data - Data created by an algorithm - Data might be processed to create additional data. E.g., processing could add noise, change colors or merge pictures.
03-54	Hardware production data	<ul style="list-style-type: none"> • Consists of bill of materials • Consists of layout e.g. GERBER data • Specifies requirements for EOL test e.g. <ul style="list-style-type: none"> - Test type (AOI, ICT, boundary scan) - Test coverage - Electrical loads • Acceptance criteria • In case of semiconductor development: mask data (GDS2)
04-00	Design	<ul style="list-style-type: none"> • Describes the overall product/system structure • Identifies the required product/system elements • Identifies the relationship between the elements • Consideration is given to: <ul style="list-style-type: none"> - any required performance characteristics - any required interfaces - any required security characteristics

ID	Name	Characteristics
04-02	Domain architecture	<ul style="list-style-type: none"> • Identified domain model(s) tailored from overall architecture ?? • Identified product and work product specifications • Definition of boundaries, dependencies and relationships with other domains (Domain Interface Specification) • Identification of domain vocabulary • Identification of the domain representation standard • Provides an overview of the functions, features, and concepts in the domains
04-03	Domain model	<ul style="list-style-type: none"> • Must provide a clear explanation and description, on usage and properties, for reuse purposes • Identification of the management and structures used in the model
04-04	Software architecture	<ul style="list-style-type: none"> • A justifying rationale for the chosen architecture. • Behavior of software elements: <ul style="list-style-type: none"> - Allocation of requirements to software elements - Reusable components - Describes the individual behavior of an element - Describes the relationship between software elements - Settings for system parameters (such as application parameters) • Interface Definitions: <ul style="list-style-type: none"> - Technical characteristics of interfaces for relationships between software elements: <ul style="list-style-type: none"> ○ Synchronization of Processes and tasks ○ Programming language call ○ APIs ○ Specifications of SW libraries - Method definitions in an object- oriented class definitions or UML/SysML interface classes - Callback functions, “hooks” • Dynamics of software elements and software states: <ul style="list-style-type: none"> - Operating modes (e.g. start-up, shutdown, normal mode, calibration, diagnosis, etc.) - intercommunication (processes, tasks, threads) and priority - time slices and cycle time - interrupts with their priorities - interactions between software elements
04-05	Software detailed design	<ul style="list-style-type: none"> • Elements of a software detailed design: <ul style="list-style-type: none"> - Control flow definition - Provides format of input/output data - Algorithms - Defined data structures - Justified global variables • Examples for expression languages, depending on the complexity or criticality of a software unit: <ul style="list-style-type: none"> - natural language or informal languages - semi-formal languages (e.g. UML, SysML) - formal languages (e.g. model-based approach)

<p>04-06</p>	<p>System architecture</p>	<ul style="list-style-type: none"> • A justifying rationale for the chosen architecture. • Behavior of system elements: <ul style="list-style-type: none"> - Mapping of requirements to system elements - Reusable components - Describes the individual behaviors of an element - Describes the interrelationship between system elements - Manual/human control actions, e.g., according to STPA - Settings for system parameters (such as application parameters) • Interface Definitions: <ul style="list-style-type: none"> - Technical characteristics of interfaces for relationships between two system elements: • Interfaces for System e.g. <ul style="list-style-type: none"> - bus interfaces (CAN, MOST, LIN, Flexray etc.) - thermal influences - hardware-software-interfaces (HSI), see below - electromagnetic interfaces - optical interfaces - hardware-mechanical-interfaces (e.g., a cable satisfying both mechanical and electrical requirements, housing interface to a PCB) - hardware-mechanical interconnection technology such as connectors, pressfit - creepage and clearance distances • Fixations such as adhesive joints, screw bolts/fitting, riveted bolts, welding • System interfaces related to EE Hardware e.g. <ul style="list-style-type: none"> - analogue or digital interfaces (PWM, I/O) and their pin configurations - SPI bus, I2C bus, electrical interconnections - placement, e.g., thermal interfaces between hardware elements (heat dissipation) - soldering - creepage and clearance distances • Interfaces for mechanical engineering e.g. <ul style="list-style-type: none"> - friction - thermal influences - tolerances - clutches - fixations such as adhesive joints, screw bolts/fitting, riveted bolts, welding - forces (as a result of e.g., vibrations or friction) - placement - shape - A hardware-software interface, e.g. <ul style="list-style-type: none"> - connector pin configurations and floating IOs for μCs/MOSFETs - signal scaling & resolution to be reflected by the application software • Mechanical-hardware interfaces e.g. <ul style="list-style-type: none"> - such as mechanical dimensioning - positioning of connectors - positioning of e.g. hall sensors in relation to the bus-bar - tolerances • Dynamics of system elements and system states:
---------------------	----------------------------	--

ID	Name	Characteristics
		<ul style="list-style-type: none"> - Description of the system states and operation modes (startup, shutdown, sleep mode, diagnosis/calibration mode, production mode, degradation, emergency such as “limp-home”, etc.) - Description of the dependencies among the system components regarding the operation modes - Interactions between system elements such as inertia of mechanical components to be reflected by the ECU, signal propagation and processing time through the hardware and software and e.g. bus systems
04-50	Build list	<ul style="list-style-type: none"> • Identification of aggregates of the software application system • Identification of required system elements (application parameter settings, macro libraries, data bases, job control languages, etc.) • Necessary sequence ordering identified for compiling the software release • Input and output source libraries identified
04-51	ML architecture	<ul style="list-style-type: none"> • An ML architecture is basically a special part of a software architecture (see 04-04). Additionally <ul style="list-style-type: none"> - ML architecture describes the overall structure of the ML-based software element - ML architecture specifies ML architectural elements including an ML model and other ML architectural elements, provided to train, deploy, and test the ML model. - describes interfaces within the ML-based software element and to other software elements - ML architecture describes details of the ML model like used layers, activation functions, loss function, and backpropagation - ML architecture contains defined hyperparameter ranges and initial values for training start - resource consumption objectives are defined - ML architecture contains allocated ML requirements
04-52	Hardware architecture	<ul style="list-style-type: none"> • Describes the overall hardware structure • Identifies the required hardware components • Includes the rationale for chosen options of hardware architecture • Identifies own developed and supplied hardware components • Identifies the required internal and external hardware component interfaces • Specifies the interfaces of the hardware components • Specifies the dynamic behaviour • Identifies the relationship and dependency between hardware components • Describes all hardware variants to be developed • Describes power supply, thermal and grounding concepts

ID	Name	Characteristics
04-53	Hardware detailed design description	<ul style="list-style-type: none"> • Describes the interconnections between the hardware parts • Specifies the interfaces of the hardware parts • Specifies the dynamic behavior (Examples are: transitions between electrical states of hardware parts, power-up and power-down sequences, frequencies, modulations, signal delays, debounce times, filters, short circuit behavior, self-protection) • Describes the conclusions and decisions based on e.g. analysis reports, datasheets, application notes • Describes the constraints for layout
04-54	Hardware Schematics	<ul style="list-style-type: none"> • Identifies the hardware parts • Specifies the connections of the hardware parts • Specifies the unique identification of all hardware parts • Specifies unique variant identification
04-55	Hardware Layout	<ul style="list-style-type: none"> • Specifies the placement of the hardware parts and labels • Specifies manufacturing data e.g. circuit paths (width, routing), vias, testing points, number of layers, drillings, material of the PCB, shape, soldering resist mask, PCB coating • Specifies a unique layout identification
04-56	Hardware element interface	<ul style="list-style-type: none"> • is defined by output, input, type, and electrical characteristics including signal tolerances. • Examples of interfaces are <ul style="list-style-type: none"> - high level interfaces like SPI, I2C, CAN, LIN, Ethernet - electrical interconnections - thermal interfaces between hardware elements (heat dissipation) •
05-00	Goals	<ul style="list-style-type: none"> • Identifies the objective to be achieved • Identifies who is expected to achieve the goal • Identifies any incremental supporting goals • Identifies any conditions/constraints • Identifies the timeframe for achievement • Are reasonable and achievable within the resources allocated • Are current, established for current project, organization • Are optimized to support known performance criteria and plans
05-01	Resource Consumption Targets	<ul style="list-style-type: none"> • Resource consumption targets for software architectural elements referring to resources like memory (ROM, RAM, external / internal EEPROM or Data Flash), CPU load
06-00	User documentation	<ul style="list-style-type: none"> • Identifies: <ul style="list-style-type: none"> - external documents - internal documents - current site distribution and maintenance list maintained • Documentation kept synchronized with latest product release • Addresses technical issues

ID	Name	Characteristics
06-02	Handling and Storage Guide	<ul style="list-style-type: none"> • Defines the tasks to perform in handling and storing products including: <ul style="list-style-type: none"> - providing for master copies of code and documentation - disaster recovery - addressing appropriate critical safety and security issues • Provides a description of how to store the product including: <ul style="list-style-type: none"> - storage environment required - the protection media to use - packing materials required - what items need to be stored - assessments to be done on stored product • Provides retrieval instructions
06-04	Training material	<ul style="list-style-type: none"> • Updated and available for new releases • Coverage of system, application, operations, maintenance as appropriate to the application • Course listings and availability
06-50	Integration sequence instruction	<ul style="list-style-type: none"> • Identification of required physical elements (e.g. hardware, mechanical, wiring elements), and software executables and application parameter settings • necessary sequence or ordering of integration • preconditions for starting system integration
06-51	Tailoring guideline	<ul style="list-style-type: none"> • Criteria for tailoring, • Proceeding of tailoring describing how to derive the defined process from the standard process • Requirements for the defined process to ensure integrity and consistency of the defined process • Subset of process assets that is essential for the defined process
06-52	Backup and recovery mechanism information	<ul style="list-style-type: none"> • Description / confirmation of existing backup and recovery mechanisms • References to corresponding procedures or regulations
07-00	Measure	<ul style="list-style-type: none"> • Available to those with a need to know • Understood by those expected to use them • Provides value to the organization/project • Non-disruptive to the workflow • Appropriate to the process, life cycle model, organization: <ul style="list-style-type: none"> - is accurate - source data is validated - results are validated to ensure accuracy • Has appropriate analysis and commentary to allow meaningful interpretation by users
07-01	Customer satisfaction survey	<ul style="list-style-type: none"> • Identification of customer and customer information • Date requested • Target date for responses • Identification of associated hardware/software/product configuration • Ability to record feedback

ID	Name	Characteristics
07-02	Field metric	<ul style="list-style-type: none"> • Measures attributes of the performance of system's operation at field locations, such as: <ul style="list-style-type: none"> - field defects - performance against defined service level measures - system ability to meet defined customer requirements - support time required - user complaints (may be third party users) - customer requests for help - performance trends - problem reports - enhancements requested
07-04	Process metric	<ul style="list-style-type: none"> • Measurements about the process' performance: <ul style="list-style-type: none"> - ability to produce sufficient work products - adherence to the process - time it takes to perform process - defects related to the process • Measures the impact of process change • Measures the efficiency of the process
07-05	Project metric	<ul style="list-style-type: none"> • Monitors key processes and critical tasks, provides status information to the project on: <ul style="list-style-type: none"> - project performance against established plan - resource utilization against established plan - time schedule against established plan - process quality against quality expectations and/or criteria - product quality against quality expectations and/or criteria - highlight product performance problems, trends • Measures the results of project activities: <ul style="list-style-type: none"> - tasks are performed on schedule - product's development is within the resource commitments allocated • References any goals established
07-06	Quality metric	<ul style="list-style-type: none"> • Measures quality attributes of the work products defined: <ul style="list-style-type: none"> - functionality - reliability - usability - efficiency - maintainability - portability • Measures quality attributes of the "end customer" quality perception <p><i>Note: Refer ISO/IEC 25010 for detailed information on measurement of product quality.</i></p>

ID	Name	Characteristics
07-08	Service level metric	<ul style="list-style-type: none"> • Real time metrics taken while a system is operational, it measures the system's performance or expected service level • Identifies aspects such as: <ul style="list-style-type: none"> - capacity - throughput - operational performance - operational service - service outage time - up time - job run time
07-50	Process performance metric	<ul style="list-style-type: none"> • Monitors key activities and critical tasks, provides status information to the project on: <ul style="list-style-type: none"> - task progress against established plan - resource utilization against established schedule - process performance quality against defined criteria - highlight process performance problems, trends
08-00	Plan	<ul style="list-style-type: none"> • As appropriate to the application and purpose: <ul style="list-style-type: none"> - Identifies what objectives or goals there are to be satisfied - Establishes the options and approach for satisfying the objectives, or goals • Identification of the plan owner • Includes: <ul style="list-style-type: none"> - the objective and scope of what is to be accomplished - assumptions made - constraints - risks - tasks to be accomplished - schedules, milestones and target dates - critical dependencies - maintenance disposition for the plan • Method/approach to accomplish plan • Identifies: <ul style="list-style-type: none"> - task ownership, including tasks performed by other parties (e.g. supplier, customer) - quality criteria - required work products • Includes resources to accomplish plan objectives: <ul style="list-style-type: none"> - time - staff (key roles and authorities e.g. sponsor) - materials/equipment - budget • Includes contingency plan for non-completed tasks • Plan is approved

ID	Name	Characteristics
08-04	Configuration management plan	<ul style="list-style-type: none"> • Defines or references the procedures to control changes to configuration items • Defines measurements used to determine the status of the configuration management activities • Defines configuration management audit criteria • Approved by the configuration management function • Identifies configuration library tools or mechanism • Includes management records and status reports that show the status and history of controlled items • Specifies the location and access mechanisms for the configuration management library • Storage, handling and delivery (including archival and retrieval) mechanisms specified
08-12	Project plan	<ul style="list-style-type: none"> • Contains strategies to achieve the project objectives • Defines responsibilities for performance of activities • Defines internal and external stakeholders and their commitments • Defines internal and external communication • Defines an escalation mechanism • Defines: <ul style="list-style-type: none"> - work products to be developed - life cycle model and methodology to be used - customer requirements related to project management - project resources • Milestones and target dates: <ul style="list-style-type: none"> - estimates - quality criteria - processes and methods to employ - contingency actions
08-13	Quality plan	<ul style="list-style-type: none"> • Objectives/goal for quality • Defines the activities tasks required to ensure quality • References related work products • References any regulatory requirements, standards, customer requirements • Identifies the expected quality criteria • Specifies the monitoring and quality checkpoints for the defined life cycle and associated activities planned • Defines the methods of assuring quality • Identifies the quality criteria for work products and process tasks • Specifies the threshold/tolerance level allowed prior to requiring corrective actions • Defines quality measurements and timing of the collection • Specifies mechanism to feed collected quality record back into process impacted by poor quality • Defines the approach to guaranteeing objectivity • Approved by the quality responsible organization/function: <ul style="list-style-type: none"> - identifies escalations opportunities and channels - defines the cooperation with customer and supplier QA

ID	Name	Characteristics
08-14	Recovery plan	<ul style="list-style-type: none"> • Identifies what is to be recovered: <ul style="list-style-type: none"> - procedures/methods to perform the recovery - schedule for recovery - time required for the recovery - critical dependencies - resources required for the recovery - list of backups maintained - staff responsible for recovery and roles assigned - special materials required - required work products - required equipment - required documentation - locations and storage of backups - contact information on who to notify about the recovery - verification procedures - cost estimation for recovery
08-16	Release plan	<ul style="list-style-type: none"> - Identifies the functionality to be included in each release - Identifies the associated elements required (i.e., hardware, software, documentation etc.) - Mapping of the customer requests, requirements satisfied to particular releases of the product
08-17	Reuse plan	<ul style="list-style-type: none"> • Defines the policy about what items to be reused • Defines standards for construction of reusable objects: <ul style="list-style-type: none"> - defines the attributes of reusable components - quality/reliability expectations - standard naming conventions • Defines the reuse repository (library, CASE tool, file, data base, etc.) • Identifies reusable components: <ul style="list-style-type: none"> - directory of component - description of components - applicability of their use - method to retrieve and use them - restrictions for modifications and usage • Method for using reusable components • Establishes goal for reusable components
08-18	Review plan	<ul style="list-style-type: none"> • Defines: <ul style="list-style-type: none"> - what to be reviewed - roles and responsibilities of reviewers - criteria for review (check-lists, requirements, standards) - expected preparation time - schedule for reviews • Identification of: <ul style="list-style-type: none"> - procedures for conducting review - review inputs and outputs - expertise expected at each review - review records to keep - review measurements to keep - resources, tools allocated to the review

ID	Name	Characteristics
08-19	Risk management plan	<ul style="list-style-type: none"> • Project risks identified and prioritized • Mechanism to track the risk • Threshold criteria to identify when corrective action required • Proposed ways to mitigate risks: <ul style="list-style-type: none"> - risk mitigator - work around - corrective actions activities/tasks - monitoring criteria - mechanisms to measure risk
08-20	Risk mitigation plan	<ul style="list-style-type: none"> • Planned risk treatment activities and tasks: <ul style="list-style-type: none"> - describes the specifics of the risk treatment selected for a risk or combination of risks found to be unacceptable - describes any difficulties that may be found in implementing the treatment • Treatment schedule • Treatment resources and their allocation • Responsibilities and authority: <ul style="list-style-type: none"> - describes who is responsible for ensuring that the treatment is being implemented and their authority • Treatment control measures: <ul style="list-style-type: none"> - defines the measures that will be used to evaluate the effectiveness of the risk treatment • Treatment cost • Interfaces among parties involved: <ul style="list-style-type: none"> - describes any coordination among stakeholders or with the project's master plan that must occur for the treatment to be properly implemented - Environment/infrastructure: <ul style="list-style-type: none"> - describes any environmental or infrastructure requirements or impacts (e.g., safety or security impacts that the treatment may have) • Risk treatment plan change procedures and history
08-27	Problem management plan	<ul style="list-style-type: none"> • Defines problem resolution activities including identification, recording, description and classification • Problem resolution approach: evaluation and correction of the problem • Defines problem tracking • Mechanism to collect and distribute problem resolutions
08-28	Change management plan	<ul style="list-style-type: none"> • Defines change management activities including identification, recording, description, analysis and implementation • Defines approach to track status of change requests • Defines verification and validation activities • Change approval and implication review
08-29	Improvement plan	<ul style="list-style-type: none"> • Improvement objectives derived from organizational business goals • Organizational scope • Process scope, the processes to be improved • Key roles and responsibilities • Appropriate milestones, review points and reporting mechanisms • Activities to be performed to keep all those affected by the improvement program informed of progress

ID	Name	Characteristics
08-50	Test specification	<ul style="list-style-type: none"> • Test Design Specification • Test Case Specification • Test Procedure Specification • Identification of test cases for regression testing • Additionally, for system integration: <ul style="list-style-type: none"> - identification of required system elements (hardware elements, wiring elements, settings for parameters (such as application parameters or global variables) , data bases, etc.) - necessary sequence or ordering identified for integrating the system elements
08-52	Test plan	<ul style="list-style-type: none"> • Test Plan according to ISO29119-3 • Context: <ul style="list-style-type: none"> - project/Test sub-process - test item(s) - test scope - assumptions and constraints - stakeholder - testing communication • Test strategy <ul style="list-style-type: none"> - identifies what needs there are to be satisfied - establishes the options and approach for satisfying the needs (black-box and/or white-box-testing, boundary class test determination, regression testing strategy, etc.) - establishes the evaluation criteria against which the strategic options are evaluated - identifies any constraints/risks and how these will be addressed - test design techniques - test completion criteria - test ending criteria - test start, abort and re-start criteria - metrics to be collected - test data requirements - retesting and regression testing - suspension and resumption criteria - deviations from the Organizational Test Strategy - Test data requirements - Test environment requirements - Test sub-processes - Test deliverables - Testing activities and estimates
08-53	Scope of work	<ul style="list-style-type: none"> • Summary of deliverables for a project • Intended use for the deliverables • Main functions to be realized • Target delivery date and major milestones • Work products and activities that are not in scope of the project as needed • Target markets • Applicable standards and legal requirements • Relevant customer requirements for performance of activities • Reuse options • Integration of third party deliveries

ID	Name	Characteristics
08-54	Feasibility analysis	<ul style="list-style-type: none"> • Statement about the ability of the project to achieve the project objectives with available resources
08-55	Risk measure	<ul style="list-style-type: none"> • Identifies the probability of risk occurring • Identifies the impact of risk occurring • Establishes measures for each risk defined • Measures the change in the risk state
08-56	Schedule	<ul style="list-style-type: none"> • Identifies the activities to be performed • Identifies the releases of products • Identifies the expected and actual start and completion date for required activities against progress/completion of activities • Identifies dependencies between activities and critical paths • Has a mapping to scheduled resource data • Identifies resource allocation, resource workload and critical resources <p><i>Note: A schedule is consistent with the defined work packages, see 14-10</i></p>
08-57	Validation Measure Selection Set	<ul style="list-style-type: none"> • Include criteria for re-validation in the case of changes (regression). • Identification of validation measures, also for regression
08-58	Verification Measure Selection Set	<ul style="list-style-type: none"> • Include criteria for re-verification in the case of changes (regression). • Identification of test cases, also for regression testing
08-59	Validation Measure	<ul style="list-style-type: none"> • A validation measure can be a test case, a measurement, a simulation, an emulation, or an end user survey • The specification of a validation measure includes <ul style="list-style-type: none"> - pass/fail criteria for validation measures (completion and end criteria) - a definition of entry and exit criteria for the validation measures, and abort and re-start criteria • Techniques • Necessary validation environment & infrastructure • Necessary sequence or ordering
08-60	Verification Measure	<ul style="list-style-type: none"> • A verification measure can be a test case, a measurement, a calculation, a simulation, a review, an optical inspection, or an analysis • The specification of a verification measure includes <ul style="list-style-type: none"> - pass/fail criteria for verification measures (test completion and ending criteria) - a definition of entry and exit criteria for the verification measures, and abort and re-start criteria • Techniques (e.g. black-box and/or white-box-testing, equivalence classes and boundary values, fault injection for Functional Safety, penetration testing for Cybersecurity, back-to-back testing for model-based development, ICT) • Necessary Verification environment & infrastructure • Necessary sequence or ordering

ID	Name	Characteristics
08-61	Resource allocation	<ul style="list-style-type: none"> • Detailed / named resources are allocated to process tasks • Overall resource workload is considered (e.g., allocation of resources to multiple projects) <p><i>Note: Work breakdown structure may be used to refine the detailed resource allocation</i></p> <p><i>Note: A resource allocation may be integrated into/part of the schedule, see 14-06</i></p>
08-62	Communication matrix	<ul style="list-style-type: none"> • List of relevant process internal / external stakeholders • Roles and contact information of the parties involved • Definition of required interfaces between stakeholders • Communication subject • Communication means and frequency • Documentation needs of the communication (e.g., type of communication record)
08-63	Process Monitoring Method	<ul style="list-style-type: none"> • Measures including criteria for monitoring effectiveness, suitability, and adequacy of the standard process • Method for collecting and analyzing the monitoring measures
08-64	ML test approach	<ul style="list-style-type: none"> • The ML test approach describes <ul style="list-style-type: none"> - test scenarios with distribution of data characteristics (e.g., sex, weather conditions, street conditions within the ODD) defined by ML requirements - quantity of each test scenario inside the test data set - expected test result per test datum - pass/fail criteria for the ML testing - entry and exit criteria for the ML testing - the required ML testing infrastructure and environment configuration
08-65	ML training and validation approach	<ul style="list-style-type: none"> • The ML Training and Validation approach describes at least: <ul style="list-style-type: none"> - entry and exit criteria of the ML training - approaches for hyperparameter tuning / optimization to be used in the training - approach for data set creation and modification - training environment, including required training hardware (e.g., GPU, or supercomputer to be used) - interface adapter for provision of input data and storage of output data - if required, actions to organize the data set and training environment • The ML training and validation approach may additionally include methods for robustification methods like random dropout
09-00	Policy	<ul style="list-style-type: none"> • Authorized • Available to all personnel impacted by the policy • Establishes practices/rules to be adhered to

ID	Name	Characteristics
09-03	Reuse policy	<ul style="list-style-type: none"> • Identification of reuse requirements • Establishes the rules of reuse • Documents the reuse adoption strategy including goals and objectives • Identification of the reuse program • Identification of the name of the reuse sponsor • Identification of the reuse program participants • Identification of the reuse steering function • Identification of reuse program support functions
10-00	Process description	<ul style="list-style-type: none"> • Process description of a standard or defined process (e.g., after tailoring), including: <ul style="list-style-type: none"> - Scope and the intended use of the process - Process activities including description and dependencies - Inputs including information needed and expected outputs - Entry and exit criteria - Roles assigned to process activities (e.g., as RASIC) - Guidelines - Templates - Description of specific methods
10-50	Role description	<ul style="list-style-type: none"> • Name/identifier (Unique within the organization) • Assigned activities (e.g., as RASIC) • Responsibilities and authorities • Required competencies • Required skills • Required experience and trainings
10-51	Qualification method description	<ul style="list-style-type: none"> • Training courses • Training material • Mentoring concepts • Self-learning material
10-52	Process resource and infrastructure description	<ul style="list-style-type: none"> • Required facilities • Required tools and corresponding licenses • Required networks • Required services • Required samples
11-00	Product	<ul style="list-style-type: none"> • Is a result/deliverable of the execution of a process, includes services, systems (software and hardware) and processed materials • Has elements that satisfy one or more aspects of a process purpose • May be represented on various media (tangible and intangible)

ID	Name	Characteristics
11-03	Release note	<ul style="list-style-type: none"> • Coverage for key elements (as appropriate to the application): • Description of what is new or changed (including features removed) • System information and requirements • Identification of conversion programs and instructions • Release numbering implementation may include: <ul style="list-style-type: none"> - the major release number - the feature release number - the defect repair number - the alpha or beta release; and the iteration within the alpha or beta release • Identification of the component list (version identification included): <ul style="list-style-type: none"> - hardware / software / product elements, libraries, etc. - associated documentation list • New/changed parameter information (e.g. for application parameters or global variables) and/or commands • Backup and recovery information • List of open known problems, faults, warning information, etc. • Identification of verification and diagnostic procedures • Technical support information • Copyright and license information • The release note may include an introduction, the environmental requirements, installation procedures, product invocation, new feature identification and a list of defect resolutions, known defects and workarounds
11-04	Product release package	<ul style="list-style-type: none"> • Includes the hardware/software/product • Includes and associated release elements such as: <ul style="list-style-type: none"> - system hardware/software/product elements - associated customer documentation - application parameter definitions defined - command language defined - installation instructions - release letter
11-05	Software Unit	<ul style="list-style-type: none"> • Software design model element at the lowest level, or commented source code including <ul style="list-style-type: none"> - parameter and return value information - variables and data structures defined - data types defined - algorithms defined - configuration files - or executable code - or commented scripts - or auto-coded code represented by model elements
11-06	Integrated System	<ul style="list-style-type: none"> • Integrated product • Application parameters values • All configured elements for the product release are included
11-50	Deployed ML model	<ul style="list-style-type: none"> • The deployed model is a ML specific software item. • It is derived from the trained ML model (see 01-53) and is to be integrated into the target system. • It may differ from the trained ML model which often requires powerful hardware and uses interpretative languages.

ID	Name	Characteristics
12-01	Request for quotation	<ul style="list-style-type: none"> • Reference to the requirements specifications • Identifies supplier selection criteria • Identifies desired characteristics, such as: <ul style="list-style-type: none"> - system architecture, configuration requirements or the requirements for service (consultants, maintenance, etc.) - quality criteria or requirements - project schedule requirements - expected delivery/service dates - cost/price expectations - regulatory standards/requirements • Identifies submission constraints: <ul style="list-style-type: none"> - date for resubmission of the response - requirements with regard to the format of response
12-03	Reuse candidate	<ul style="list-style-type: none"> • Identifies the product to be reused • Identifies the responsible person for the products to be reused • Identifies the reuse goals and objectives • Identifies the list of reuse assets • Identifies the issues/risks of reusing the component including specific requirements (hardware, software, resource and other reuse components) • Identifies the person who will be qualifying the reuse candidate
13-00	Record	<ul style="list-style-type: none"> • Work product stating results achieved or provides evidence of activities performed in a process • An item that is part of a set of identifiable and retrievable data
13-01	Acceptance record	<ul style="list-style-type: none"> • Record of the receipt of the delivery • Identification of the date received • Identification of the delivered components • Records the verification of any customer acceptance criteria defined • Signed by receiving customer
13-04	Communication evidence	<ul style="list-style-type: none"> • All forms of interpersonal communication e.g.: <ul style="list-style-type: none"> - letters - e-mails - blog - forum - live chat - wikis - photo protocol - meeting support record
13-06	Delivery evidence	<ul style="list-style-type: none"> • Evidence of items shipped/delivered electronically to customer • Identification of: <ul style="list-style-type: none"> - to whom it was sent - address, where delivered - delivery date - receipt of delivered product

ID	Name	Characteristics
13-07	Problem record	<ul style="list-style-type: none"> • Identifies the submitter of the problem • Identifies the group/person(s) responsible for providing problem resolution • Includes a description of the problem • Identifies classification of the problem (criticality, urgency, relevance etc.) • Identifies the status of the problem <ul style="list-style-type: none"> - States such as “open”, “in review”, “in implementation”, “closed”, “rejected”, “cancelled”, ... - Transitions between states with conditions and authorities • Identifies the expected closure date
13-08	Baseline	<ul style="list-style-type: none"> • Identifies a state of one or a set of work products and artifacts which are consistent and complete • Basis for next process steps and/or delivery • Is unique and may not be changed <p><i>Note: This should be established before a release to identify consistent and complete delivery</i></p>
13-09	Meeting support record	<ul style="list-style-type: none"> • Agenda and minutes that are records that define: <ul style="list-style-type: none"> - purpose of meeting - attendees - date, place held - reference to previous minutes - what was accomplished - identifies issues raised - any open issues - next meeting if any
13-10	Configuration management record	<ul style="list-style-type: none"> • Status of the configuration items and modifications • Identifies activities performed e.g. backup, storage, archiving, handling and delivery of configuration items • Supports consistency of the product, e.g. <ul style="list-style-type: none"> - work product revision - when updated - work product status - when approved - reference to approval source
13-13	Product release approval	<ul style="list-style-type: none"> • Content information of what is to be shipped or delivered • Identification of: <ul style="list-style-type: none"> - for whom it is intended - the address where to deliver - the date released - Evidence of supplier approval

ID	Name	Characteristics
13-14	Progress status record	<ul style="list-style-type: none"> • Record of the status of a plan(s) (actual against planned) such as: <ul style="list-style-type: none"> - status of actual activities/work packages against planned activities/work package - status of actual results against established objectives/goals - status of actual resources allocation against planned resources - status of actual cost against budget estimates - status of actual time against planned schedule - status of actual quality against planned quality • Record of any deviations from planned activities and reason why
13-15	Proposal review record	<ul style="list-style-type: none"> • Scope of proposal and requirements • Possible contingencies or risks • Alignment of the proposal with the strategic business plan of the organization • Protection of proprietary information • Requirements which differ from those in the original documentation • Capability to meet contractual requirements • Responsibility for subcontracted work • Terminology • Supplier ability to meet obligations • Approved
13-16	Change request	<ul style="list-style-type: none"> • Identifies purpose of change • Identifies requester contact information • Impacted system(s) • Impact to operations of existing system(s) defined • Impact to associated documentation defined • Criticality of the request, due date • Information supporting the tracking of change requests to closure <ul style="list-style-type: none"> - progress status attribute (e.g., open, allocated, implemented, closed) - time stamp of status change - person who changed a status - rationale for changing a status
13-18	Quality record	<ul style="list-style-type: none"> • Identifies what tasks/activities/process produce the information • Identifies when the data was collected • Identifies source of any associated data • Identifies the associated quality criteria • Identifies any associated measurements using the information • Identifies any requirements to be adhered to create the record, or satisfied by the record

ID	Name	Characteristics
13-19	Review record	<ul style="list-style-type: none"> • Provides the context information about the review: <ul style="list-style-type: none"> - what was reviewed - lists reviewers who attended - status of the review • Provides information about the coverage of the review: <ul style="list-style-type: none"> - check-lists - review criteria - requirements - compliance to standards • Records information about: <ul style="list-style-type: none"> - the readiness for the review - preparation time spent for the review - time spent in the review - reviewers, roles and expertise • Review findings: <ul style="list-style-type: none"> - non-conformances - improvement suggestions • Identifies the required corrective actions: <ul style="list-style-type: none"> - risk identification - prioritized list of deviations and problems discovered - the actions, tasks to be performed to fix the problem - ownership for corrective action - status and target closure dates for identified problems

ID	Name	Characteristics
13-20	Risk action request	<ul style="list-style-type: none"> • Date of initiation • Scope • Subject • Request originator • Risk management process context: <ul style="list-style-type: none"> - this section may be provided once, and then referenced in subsequent action requests if no changes have occurred - process scope - stakeholder perspective - risk categories - risk thresholds - project objectives - project assumptions - project constraints • Risks: <ul style="list-style-type: none"> - this section may cover one risk or many, as the user chooses - where all the information above applies to the whole set of risks, one action request may suffice - where the information varies, each request may cover the risk or risks that share common information - risk description(s) - risk probability - risk consequences - expected timing of risk • Risk treatment alternatives: <ul style="list-style-type: none"> - treatment option selected- avoid/reduce/transfer - alternative descriptions - recommended alternative(s) - justifications • Risk action request disposition: <ul style="list-style-type: none"> - each request should be annotated as to whether it is accepted, rejected, or modified, and the rationale provided for whichever decision is taken
13-21	Change control record	<ul style="list-style-type: none"> • Used as a mechanism to control change to baselined products/products in official project release libraries • Record of the change requested and made to a baselined product (work products, software, customer documentation, etc.): <ul style="list-style-type: none"> - identification of system, documents impacted with change - identification of change requester - identification of party responsible for the change - identification of status of the change • Linkage to associated customer requests, internal change requests, etc. • Appropriate approvals • Duplicate requests are identified and grouped

ID	Name	Characteristics
13-22	Traceability Record	<ul style="list-style-type: none"> • All requirements (customer and internal) are to be traced • Identifies a mapping of requirement to life cycle work products • Provides the linkage of requirements to work product decomposition (i.e., requirement → design → code → test → deliverables, etc.) • Provides forward and backwards mapping of requirements to associated work products throughout all phases of the life cycle <p><i>Note: this may be included as a function of another defined work product (example: A CASE tool for design decomposition may have a mapping ability as part of its features)</i></p>
13-24	Validation Results	<ul style="list-style-type: none"> • Validation data, logs, feedback, or documentation • Validation measure passed • Validation measure not passed • Validation measure not executed • Information about the validation execution (date, participants etc.) • Abstraction or summary of validation results
13-25	Verification results	<ul style="list-style-type: none"> • Verification check-list • Passed items of verification • Failed items of verification • Pending items of verification • Problems identified during verification • Risk analysis • Recommendation of actions • Conclusions of verification • Signature of verification
13-50	Test result	<ul style="list-style-type: none"> • Level Test Log • Anomaly Report • Level Test Report (Summary) <ul style="list-style-type: none"> - test cases not passed - test cases not executed - information about the test execution (date, tester name etc.) <p>Additionally where necessary:</p> <ul style="list-style-type: none"> • Level Interim Test Status Report • Master Test Report (Summary)

ID	Name	Characteristics
13-51	Consistency Evidence	<ul style="list-style-type: none"> • Demonstrates bidirectional traceability between artifacts or information in artifacts, throughout all phases of the life cycle, by e.g. <ul style="list-style-type: none"> - tool links - hyperlinks - editorial references - naming conventions • Evidence that the content of the referenced or mapped information coheres semantically along the traceability chain, e.g. by <ul style="list-style-type: none"> - performing pair working or group work - performing by peers, e.g. spot checks - maintaining revision histories in documents - providing change commenting (via e.g. meta-information) of database or repository entries <p><i>Note: verification measures and verification results are, by definition, consistent, however still need to be traceable.</i></p> <p>This evidence can be accompanied by e.g. Definition of Done (DoD) approaches.</p>
13-52	Communication Evidence	<ul style="list-style-type: none"> • All forms of interpersonal communication such as <ul style="list-style-type: none"> - e-mails - tool-supported workflows - podcast - blog - videos - forum - live chat - wikis - photo protocol - meeting, orally or via meeting minutes
13-53	Qualification evidence	<ul style="list-style-type: none"> • Qualification check-list • Passed items of Qualification • Failed items of Qualification • Pending items of Qualification • Problems identified during Qualification • Risk analysis • Recommendation of actions • Conclusions of Qualification • Approval
13-54	Configuration management record	<ul style="list-style-type: none"> • Status of the configuration item and modifications • Identifies activities performed e.g. backup, storage, archiving, handling and delivery for the configuration item • Supports consistency of the product, e.g. <ul style="list-style-type: none"> - work product revision - when updated - work product status - when approved - reference to approval source • Befüllter Status des jeweiligen Eintrags in der CI Liste. Wird laufend aktualisiert. Dient u.A. zur Statusermittlung.

ID	Name	Characteristics
13-55	Process resource and infrastructure documentation	<ul style="list-style-type: none"> • Information on availability, allocation, and usage of <ul style="list-style-type: none"> - Facilities - Tools and corresponding licenses - Networks - Services - Samples • for non-standard and critical resources and infrastructure. •
14-00 *	Register	<ul style="list-style-type: none"> • A register is a compilation of data or information captured in a defined sequence to enable: <ul style="list-style-type: none"> - an overall view of evidence of activities that have taken place - monitoring and analyzes - provides evidence of performance of a process over time
14-01	Change history	<ul style="list-style-type: none"> • Historical records of all changes made to an object (document, file, software component, etc.): <ul style="list-style-type: none"> - description of change - version information about changed object - date of change - change requester information - change control record information
14-02	Corrective action	<ul style="list-style-type: none"> • Identifies the initial problem • Identifies the ownership for completion of defined action • Defines a solution (series of actions to fix problem) • Identifies the open date and target closure date • Contains a status indicator • Indicates follow up audit actions
14-08	Tracking system	<ul style="list-style-type: none"> • Ability to record customer and process owner information • Ability to record related system configuration information • Ability to record information about problem or action needed: <ul style="list-style-type: none"> - date opened and target closure date - severity/criticality of item - status of any problem or actions needed - information about the problem or action owner - priority of problem resolution • Ability to record proposed resolution or action plan • Ability to provide management status information • Information is available to all with a need to know • Integrated change control system(s)/records

ID	Name	Characteristics
14-09	Work breakdown structure	<ul style="list-style-type: none"> • Defines tasks to be performed, and their amendments • Documents ownership for tasks • Documents critical dependencies between tasks • Documents inputs and output work products • Documents the critical dependencies between defined work products • Method to assign tasks for support processes, management processes and primary life cycle processes • Information needed to perform these tasks • Major work packages • External and internal dependencies <p><i>Note: A work breakdown structure may be integrated into/part of the schedule, see 14-06</i></p>
14-10	Work package	<ul style="list-style-type: none"> • Defines activities to be performed • Documents ownership for activities e.g., by domains • Documents critical dependencies to other work packages • Documents input and output work products • Documents the critical dependencies between defined work products • Information needed to perform these activities <p><i>Note: The work package descriptions may be integrated into/part of the schedule, see 08-56</i></p>
14-50	Stakeholder groups list	<ul style="list-style-type: none"> • Identifies: <ul style="list-style-type: none"> - relevant stakeholder groups // affected or involved parties?? - weight/importance of each stakeholder group - representative(s) for each stakeholder group - information needs of each stakeholder group
14-51	Cybersecurity scenario register	<ul style="list-style-type: none"> • Identifies: <ul style="list-style-type: none"> - damage scenarios - ID - title - description - impact category: <ul style="list-style-type: none"> ○ safety ○ financial ○ operational ○ privacy ○ quality • Threat scenarios <ul style="list-style-type: none"> - ID - asset concerned - security property: <ul style="list-style-type: none"> ○ confidentiality ○ integrity ○ availability • Attack feasibility (high/medium/low/very low)

ID	Name	Characteristics
14-52	Asset Library	<ul style="list-style-type: none"> • Identifies: <ul style="list-style-type: none"> - title - description - security property: <ul style="list-style-type: none"> ○ confidentiality ○ integrity ○ availability - stakeholders related to the asset
14-53	Role Assignment	<ul style="list-style-type: none"> • Assignment of person(s) to roles <ul style="list-style-type: none"> - required competencies vs existing competencies - required skills vs existing skills - required experience and trainings based on identified competencies / skills gap
14-54	Hardware Bill of materials	<ul style="list-style-type: none"> • Uniquely identifies type, supplier, and amount of the complete set of all hardware parts of the hardware
15-00 *	Report	<ul style="list-style-type: none"> • A work product describing a situation that: <ul style="list-style-type: none"> - includes results and status - identifies applicable/associated information - identifies considerations/constraints - provides evidence/verification
15-01	Analysis report	<ul style="list-style-type: none"> • What was analyzed? • Who did the analysis? • The analysis criteria used: <ul style="list-style-type: none"> - selection criteria or prioritization scheme used - decision criteria - quality criteria • Records the results: <ul style="list-style-type: none"> - what was decided/selected - reason for the selection - assumptions made - potential risks • Aspects of correctness to analyze include: <ul style="list-style-type: none"> - completeness - understandability - testability - verifiability - feasibility - validity - consistency - adequacy of content
15-03	Configuration status report	<ul style="list-style-type: none"> • Identification of the number of items under configuration management • Identification of risks associated to configuration management • Identification of the number of configuration management items lost and reason for their loss • Identification of problem and issues related to configuration management • Identification of receiving parties • Identification of baselines made

ID	Name	Characteristics
15-05	Evaluation report	<ul style="list-style-type: none"> • States the purpose of evaluation • Method used for evaluation • Requirements used for the evaluation • Assumptions and limitations • Identifies the context and scope information required: <ul style="list-style-type: none"> - date of evaluation - parties involved - context details - evaluation instrument (check-list, tool) used • Records the result: <ul style="list-style-type: none"> - data - identifies the required corrective and preventive actions - improvement opportunities, as appropriate
15-06	Project status report	<ul style="list-style-type: none"> • A report of the current status of the project • Schedule: <ul style="list-style-type: none"> - planned progress (established objectives/goals) or completion (dates/deadlines) of tasks against - actual progress of tasks - reasons for variance from planned progress - threats to continued progress - contingency plans to maintain progress • Resources (human resources, infrastructure, hardware/materials, budget): <ul style="list-style-type: none"> - planned expenditure against actual expenditure - reasons for variance between planned and actual expenditure - expected future expenditure - contingency plans to achieve budget goals • Quality goals: <ul style="list-style-type: none"> - actual quality measures - reasons for variance from planned quality measures - contingency plans to achieve quality goals • Project issues: <ul style="list-style-type: none"> - issues which may affect the ability of the project to achieve its goals. - contingency plans to overcome threats to project goals
15-07	Reuse analysis evidence	<ul style="list-style-type: none"> • Identification of reuse opportunities • Identification of constraints for reuse • Identification of regression test cases • Identification of reuse infrastructure • Identification of known defects

ID	Name	Characteristics
15-08	Risk analysis	<ul style="list-style-type: none"> • Identifies the risks analyzed • ID • Impact scenario (e.g., damage scenario) • Records the results of the analysis: <ul style="list-style-type: none"> - potential ways to mitigate the risk - selected risk treatment option (e.g. risk acceptance as cybersecurity claim or risk reduction) - assumptions made - probability of occurrence (e.g., attack feasibility) - risk value - constraints
15-09	Risk status	<ul style="list-style-type: none"> • Identifies the status of an identified risk: <ul style="list-style-type: none"> - related project or activity - risk statement - condition - consequence - changes in priority - duration of mitigation, when started - risk mitigation activities in progress - responsibility - constraints
15-12	Problem status report	<ul style="list-style-type: none"> • Presents a summary of problem records: <ul style="list-style-type: none"> - by problem categories/classification - by status of problem resolution - by other relevant categories - indicates progress of problem resolution
15-13	Assessment/audit report	<ul style="list-style-type: none"> • States the purpose of assessment • Method used for assessment • Requirements used for the assessment • Assumptions and limitations • Identifies the context and scope information required: <ul style="list-style-type: none"> - date of assessment - organizational unit assessed - sponsor information - assessment team - attendees - scope/coverage - assesses and information - assessment Instrument (check-list, tool) used • Records the result: <ul style="list-style-type: none"> - data - identifies the required corrective actions - improvement opportunities
15-16	Improvement opportunity	<ul style="list-style-type: none"> • Identifies what the problem is • Identifies what the cause of a problem is • Suggest what could be done to fix the problem • Identifies the value (expected benefit) in performing the improvement • Identifies the penalty for not making the improvement

ID	Name	Characteristics
15-18	Process performance report	No requirements additional to Evaluation report (Generic)
15-50	Vulnerability analysis report	<ul style="list-style-type: none"> • Identifies <ul style="list-style-type: none"> - ID - description - attack path concerned - attack feasibility (e.g., CVSS (Common Vulnerability Scoring System) rating)
15-51	Analysis Results	<p>Identification of the object under analysis.</p> <p>The analysis criteria used, e.g.</p> <ul style="list-style-type: none"> • selection criteria or prioritization scheme used • decision criteria • quality criteria <p>The analysis results, e.g.</p> <ul style="list-style-type: none"> • what was decided/selected • reason for the selection • assumptions made • potential risks <p>Aspects of the analysis may include</p> <ul style="list-style-type: none"> • correctness • understandability • verifiability • feasibility • validity
15-52	Verification Results	<ul style="list-style-type: none"> • Verification data and logs <ul style="list-style-type: none"> - Verification measure passed - Verification measure not passed - Verification measure not executed - information about the test execution (date, tester name etc.) - Abstraction or summary of verification results
15-53	Configuration status report	<ul style="list-style-type: none"> • Summary of configuration management records including relevant status • Analysis of the configuration management overall state, such as <ul style="list-style-type: none"> - identification of risks associated to configuration management - identification of the number of configuration management items lost and reason for their loss - identification of problem and issues related to configuration management - identification of receiving parties - identification of baselines made
15-54	Tailoring documentation	<ul style="list-style-type: none"> • Applied criteria for tailoring, • Evidence that the defined process is tailored from the standard process according to the defined criteria

ID	Name	Characteristics
15-55	Problem analysis evidence	<ul style="list-style-type: none"> • Author and involved parties • Date of the analysis • Context and root cause of the problem • Analysis result may include <ul style="list-style-type: none"> - Impact - Potential risks - Affected parties - Potential solution (if known)
15-56	Configuration status summary	<ul style="list-style-type: none"> • Summary of configuration management records including relevant status • Analysis of the configuration management overall state • Identification of baselines made
16-00 *	Repository	<ul style="list-style-type: none"> • Repository for components • Storage and retrieval capabilities • Ability to browse content • Listing of contents with description of attributes • Sharing and transfer of components between affected groups • Effective controls over access • Maintain component descriptions • Recovery of archive versions of components • Ability to report component status • Changes to components are tracked to change/user requests
16-03	Configuration management system	<ul style="list-style-type: none"> • Supports the configuration management for the scope of the configuration item list contents • Correct configuration of products • Can recreate any release or test configuration • Ability to report configuration status • Has to cover all relevant tools
16-06	Process repository	<ul style="list-style-type: none"> • Contains process descriptions • Supports multiple presentations of process assets
16-50	Organization structure	<ul style="list-style-type: none"> • Disciplinary reporting line • Organizational units and sub-units, if applicable
16-51	Process Infrastructure	<ul style="list-style-type: none"> • Description of methods and guidelines needed to perform the process • Templates or at least detailed requirements for the documented information of the process • Tools or work environment needed to perform the process
16-52	ML data management system	<ul style="list-style-type: none"> • The ML data management system is part of the configuration management system (see 16-03) and • Supports data management activities like data collection, description, ingestion, exploration, profiling, labeling/annotation, selection, structuring and cleansing • Provides the data for different purposes, e.g., training, testing • Supports the relevant sources of ML data

ID	Name	Characteristics
17-00	Requirement	<ul style="list-style-type: none"> • An expectation of functions and capabilities (e.g. nonfunctional requirements), or one of its interfaces <ul style="list-style-type: none"> - from a black-box perspective - that is verifiable, does not imply a design or implementation decision, is unambiguous, and does not introduce contradictions to other requirements. • A requirements statement that implies, or represents, a design or implementation decision is called “Design Constraint”. • Examples for requirements aspects at the system level are thermal characteristics such as <ul style="list-style-type: none"> - heat dissipation - dimensions - weight - materials • Examples of aspects related to requirements about system interfaces are <ul style="list-style-type: none"> - connectors - cables - housing • Examples for requirements at the hardware level are <ul style="list-style-type: none"> - lifetime and mission profile, lifetime robustness - maximum price - storage and transportation requirements - functional behavior of analog or digital circuits and logic - quiescent current, voltage impulse responsiveness to crank, start-stop, drop-out, load dump - temperature, maximum hardware heat dissipation - power consumption depending on the operating state such as sleep-mode, start-up, reset conditions - frequencies, modulation, signal delays, filters, control loops - power-up and power-down sequences, accuracy and precision of signal acquisition or signal processing time - computing resources such as memory space and CPU clock tolerances - maximum abrasive wear and shearing forces for e.g. pins or soldering joints - requirements resulting from lessons learned - safety related requirements derived from the technical safety concept

ID	Name	Characteristics
17-03	Stakeholder requirements	<ul style="list-style-type: none"> • Purpose/objectives defined • Includes issues/requirements from (contract) reviews • Identifies any: <ul style="list-style-type: none"> - time schedule/constraints - required feature and functional characteristics - necessary performance considerations/constraints - necessary internal/external interface considerations/constraints - required system characteristics/constraints - human engineering considerations/constraints - security considerations/constraints - environmental considerations/constraints - operational considerations/constraints - maintenance considerations/constraints - installation considerations/constraints - support considerations/constraints - design constraints - safety/reliability considerations/constraints - quality requirements/expectations
17-05	Requirements for documented information	<ul style="list-style-type: none"> • Requirements for content and structure, storage and control <ul style="list-style-type: none"> - Identifies documentation specific meta data, such as date, author information, review and approval status or others - Identifies requirements on documentation structure, e.g. table of content or figures or other formal aspects - May be provided by documentation templates - May be based on tool specific templates - Defines the storage location such as data repository, tool, versioning system - Requirements for versioning - Requirements for baselining - Distribution of the documents - Maintenance and disposal of the documents - May be specific for certain types of documents
17-08	Interface requirements specification	<ul style="list-style-type: none"> • Defines relationships between two products, process or process tasks • Defines criteria and format for what is common to both • Defines critical timing dependencies or sequence ordering • Description of the physical interfaces of each system component like: <ul style="list-style-type: none"> - bus interfaces (CAN, MOST, LIN, Flexray etc.) - transceiver (type, manufacturer, etc.) - analogue interfaces - digital interfaces (PWM, I/O) - additional interfaces (IEEE, ISO, Bluetooth, USB, etc.) • Identification of the software interfaces of software components and other software item in terms of: <ul style="list-style-type: none"> - inter-process communication mechanisms - bus communication mechanisms

ID	Name	Characteristics
17-11	Software requirements specification	<ul style="list-style-type: none"> • Identifies standards to be used • Identifies any software structure considerations/constraints • Identifies the required software elements • Identifies the relationship between software elements • Consideration is given to: <ul style="list-style-type: none"> - any required software performance characteristics - any required software interfaces - any required security characteristics required - any database design requirements - any required error handling and recovery attributes - any required resource consumption characteristics
17-12	System requirements specification	<ul style="list-style-type: none"> • System requirements include: functions and capabilities of the system; business, organizational and user requirements; safety, security, human-factors engineering (ergonomics), interface, operations, and maintenance requirements; design constraints and qualification requirements. • Identifies the required system overview • Identifies any interrelationship considerations/constraints between system elements • Identifies any relationship considerations/constraints between the system elements and the software • Identifies any design considerations/constraints for each required system element, including: <ul style="list-style-type: none"> - memory/capacity requirements - hardware interface requirements - user interface requirements - external system interface requirements - performance requirements - command structures - security/data protection characteristics - application parameter settings - manual operations - reusable components • Describes the operation capabilities • Describes environmental capabilities • Documentation requirements • Reliability requirements • Logistical Requirements • Describes security requirements • Diagnosis requirements
17-54	Requirement Attribute	<ul style="list-style-type: none"> • Meta-attributes that support structuring and definition of release scopes of requirements. • Can be realized by means of tools. <p><i>Note: usage of requirements attributes may further support analysis of requirements.</i></p>

ID	Name	Characteristics
17-55	Resource needs	<ul style="list-style-type: none"> • Identification of required resources for process performance • Staff including competencies, skills and authorities needs • Material, equipment, and infrastructure • Time and budget <p><i>Note: Needs are derived from Work Breakdown structure and schedule</i></p>
17-57	Special Characteristics	<ul style="list-style-type: none"> • Special Characteristics in terms of relevant standards such as IATF 16949, VDA 6.x Guidelines, ISO 26262. • Special Characteristics according to IATF 16949:2016-10 [15], Chapters 8.3.3.3, are product characteristics or production process parameters that may have an impact on safety or compliance with official regulations, the fit, the function, the performance or further processing of the product. • Special characteristics shall be verifiable according to VDA vol. 1 <p><i>Note: A typical method for identifying and rate special characteristics is an FMEA.</i></p>
17-58	Hardware requirement	<ul style="list-style-type: none"> • Specifies <ul style="list-style-type: none"> - an expectation of hardware functions and capabilities (e.g. nonfunctional requirements), or one of its interfaces - from a black-box perspective - that is verifiable, does not imply a design or implementation decision, is unambiguous, and does not introduce contradictions to other requirements. • A requirement that implies, or represents, a design or implementation decision is called "Design Constraint". • Hardware requirements specify particular desired characteristics of the hardware and can include <ul style="list-style-type: none"> - lifetime and mission profile, lifetime robustness - maximum price - storage and transportation requirements - functional behaviour of analog or digital circuits and logic - quiescent current, voltage impulse responsiveness to crank, start-stop, drop-out, load dump - temperature, maximum hardware heat dissipation - power consumption depending on the operating state such as sleep-mode, start-up, reset conditions - frequencies, modulation, signal delays, filters, control loops - power-up and power-down sequences, accuracy and precision of signal acquisition or signal processing time - computing resources such as memory space and CPU clock tolerances - maximum abrasive wear and shearing forces for e.g. pins or soldering joints - requirements resulting from lessons learned - safety related requirements derived from the technical safety concept

ID	Name	Characteristics
18-00 *	Standard	<ul style="list-style-type: none"> • Identification of to whom/what they apply • Expectations for conformance are identified • Conformance to requirements can be demonstrated • Provisions for tailoring or exception to the requirements are included
18-06	Product release criteria	<ul style="list-style-type: none"> • Defines expectations for product release: <ul style="list-style-type: none"> - release type and status - required elements of the release - product completeness including documentation - adequacy and coverage of testing - limit for open defects - change control status
18-07	Quality criteria	<ul style="list-style-type: none"> • Defines the expectations for documented information and process performance • Including thresholds/tolerance levels, required measurements, required checkpoints • Defines what is an adequate documented information (required elements, completeness expected, accuracy, etc.) • Defines what constitutes the completeness of the defined tasks • Defines what constitutes the performance of the defined tasks • Establishes expected performance attributes
18-51	Verification criteria	<ul style="list-style-type: none"> • Each requirement is verifiable or can be assessed • Verification criteria define the qualitative and quantitative criteria for verification of a requirement. • Verification criteria demonstrate that a requirement can be verified within agreed constraints. (Additional Requirement to 17-00 Requirements specification)
18-52	Escalation path	<ul style="list-style-type: none"> • Defined mechanisms to report and confirm escalation relevant issues • Identifies stakeholders to be included in the escalation path • Identifies levels of escalation
18-53	Configuration item selection criteria	<ul style="list-style-type: none"> • Identify types of items to be subject to configuration control
18-54	Problem status model	<ul style="list-style-type: none"> • Defines approach to track status of problems, including <ul style="list-style-type: none"> - states such as “open”, “in review”, “in implementation”, “closed”, “rejected”, “cancelled”, ... - transitions between states with conditions and authorities
18-55	Problem classification	<ul style="list-style-type: none"> • Defines criteria for alert notification to be initiated and urgent problem resolution activities to be authorized. <ul style="list-style-type: none"> - criteria may be based on priority, impact, urgency, etc. - identifies associated alert notifications, and urgent problem resolution authorization definitions
18-56	Change request status model	<ul style="list-style-type: none"> • Defines approach to track status of change requests <ul style="list-style-type: none"> - states such as “open”, “in review”, “in implementation”, “closed”, “rejected”, “cancelled” ... - transitions between states with conditions and authorities, e.g for approval

ID	Name	Characteristics
18-57	Change analysis criteria	<ul style="list-style-type: none"> • Defines analysis criteria, such as <ul style="list-style-type: none"> - resource requirements - scheduling issues - risks - benefits
18-58	Process performance quality criteria	<ul style="list-style-type: none"> • Established performance attributes and evaluation criteria used to evaluate the achievement of strategic process performance goals • Quality criteria include e.g., <ul style="list-style-type: none"> - activity/work package completeness criteria (e.g., definition of done), - lifecycle criteria (entry/exit), - frequency of tasks performance, - process achievement goals for project milestones • Related measurements for process performance monitoring
18-59	Review and approval criteria for documented information	<ul style="list-style-type: none"> • Specifies for each type of documented information review and approval needs <ul style="list-style-type: none"> - If and when a review is required - Who shall review it - Who shall approve it - Criteria for approval •
18-60	Quality criteria for documented information	<ul style="list-style-type: none"> • Establishes what is an adequate work product: <ul style="list-style-type: none"> - required elements - completeness expected - accuracy - etc.
19-00	Strategy	<ul style="list-style-type: none"> • Identifies what needs and objectives or goals there are to be satisfied • Establishes the options and approach for satisfying the needs, objectives, or goals • Establishes the evaluation criteria against which the strategic options are evaluated • Identifies any constraints/risks and how these will be addressed
19-01	Process performance strategy	<ul style="list-style-type: none"> • Process scope • Needs, objectives, and goals to be satisfied • Options, approach, and methods to perform the process activities • Assumptions and constraints (given implicitly by e.g., budget, resources) • References to relevant regulatory requirements, standards and customer requirements • Risks of the chosen strategy/approach and appropriate mitigation measures • Deliverables (relevant input to / outputs of affected parties, supplier and customer)

ID	Name	Characteristics
19-05	Reuse strategy	<ul style="list-style-type: none"> • Identify the goals for reuse • Identify the commitment for creating reusable components • Determine which product lines and type of artifacts should be supported with reuse • Identify system and hardware/software/product elements which can be reused within the organization • Identify the reuse repository and tools
19-50	ML data quality approach	<ul style="list-style-type: none"> • The ML data quality approach • Defines Quality criteria (see 18-07) e.g., the relevant data sources, reliability and consistency of labelling, completeness against ML data requirements • Describes analysis activities of the data • Describes activities to ensure the quality of the data to avoid issues e.g., data bias, bad labeling
20-00	Template	<ul style="list-style-type: none"> • Defines the attributes associated with a work product to be created as a consequence of a process execution • Identifies technical elements typically associated with this product type • Defines expected form and style
21-00	Work product	<ul style="list-style-type: none"> • Defines the attributes associated with an artifact from a process execution: <ul style="list-style-type: none"> - key elements to be represented in the work product

Table B.2 — Information item characteristics

Annex C Key concepts and guidance

This section will be updated with the final version of Automotive SPICE 4.0

Annex D Reference standards

Annex E provides a list of reference standards and guidelines that support implementation of the Automotive SPICE PAM / PRM.

ISO/IEC 33001:2015	Information technology -- Process assessment – Concepts and terminology
ISO/IEC 33002:2015	Information technology -- Process assessment – Requirements for performing process assessment
ISO/IEC 33003:2015	Information technology -- Process assessment – Requirements for process measurement frameworks
ISO/IEC 33004:2019	Information technology -- Process assessment – Requirements for process reference, process assessment and maturity models
ISO/IEC 33020:2015	Information technology -- Process assessment – Process measurement framework for assessment of process capability
ISO/IEC/IEEE 24765:2017	Systems and software engineering -- Vocabulary
ISO/IEC IEEE 29148:2018	Systems and software engineering
INCOSE Guide for Writing Requirements	https://www.incose.org/
PAS 1883:2020	Operational design domain (ODD) taxonomy for an automated driving system (ADS)

Table E.1 — Reference standards