KNÜVENERMACKERT
ACCEPTED PROCESSES FOLLOW EASY RULES

SPICE BOOKLET

- INTRODUCTION TO AUTOMOTIVE SPICE®
- WHAT IS NEW IN ASPICE® 4.0
- AUTOMOTIVE SPICE® 4.0 WITH RATING CONSISTANCY DIAGRAMS OF GUIDELINE 2.0
- MECHANICAL ENGINEERING SPICE® 2.0
- AUTOMOTIVE SPICE® FOR CYBERSECURITY 1.0
- AGILE SPICE® 1.3
- ORGANIZATION SPICE® 3.0
- ASSESSMENT GUIDE

8th Edition 2024

Table of content

Introduction to Automotive SPICE®

What is new in ASPICE® 4.0

Automotive SPICE® 4.0

Extentions

Assessment Guide

Information Item Characteristics

For further information about Automotive SPICE® visit *http://vda-qmc.de/en/automotive-spice/*

Photo: Adobe Stock, Layout: Serife Kittelberger – www.lavora.de

Knüvener Mackert GmbH provides certified Automotive SPICE® trainings according to the International Assessor Certification Scheme (intacs®).

Download the KNÜVENER MACKERT SPICE Booklet free of charge from ***https://knuevenermackert.com/SPICEbooklet*** or order a printed copy from ***info@knuevenermackert.com***

ISBN 978-3-00-061926-7

INTRODUCTION TO AUTOMOTIVE SPICE®

Why process quality?
Automotive SPICE® supports the quality of your daily processes

✓ **Increase quality**

- Work products (WPs) are based on qualified input
- WPs are verified and validated based on criteria
- WPs are produced as planned and scheduled
- Organization learns based on feedback

✓ **Reduce cost**

- Early identification and correction of defects
- Proven processes and templates; experienced team
- Transparent and smooth progress
- Do it right the first time
- Less duplicated work, re-work and extra work
- Productivity increase

✓ **Manage risks and complexity**

- Manage risks effectively and in time
- Develop increasing functionality in reduced time

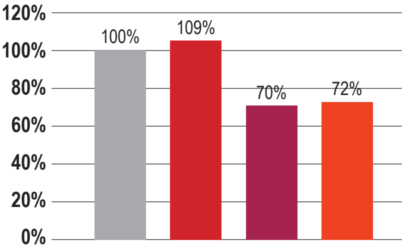
✓ **Meet customers expectation – current and future business**

- Avoid penalty (payments and/or ‘high’ awareness)
- Win quotations (positive supplier ranking, flexibility)

✓ **For your own sake**

- Less priority hopping
- Clear responsibilities
- Pride in one's own work
- Less discussions
- No double work
- <please add your personal points here>
- …

How to invest in quality and cost saving



- ⬜ 100% = Development without Automotive SPICE
- 🟥 With Automotive SPICE: 9% more initial development costs
- 🟪 With Automotive SPICE: 30% less bugs
- 🟧 With Automotive SPICE: 28% less effort for maintenance

Source: Paolo Panaroni, Luca Fogli, Why Automotive SPICE?, IntecsSolutions

Provided example:
Realization of one complex function

- 280 errors of A or B priority in a C sample costs $ 2.184.000.
  Saving by Automotive SPICE (30%): $ 655.000
- 80 open of A or B priority in SOP costs $ 6.760.000
  Saving by Automotive SPICE (30%): $ 2.028.000

Source: Frank Lenkeit (K-GQX-S/2), Volkswagen AG,
17 Jahre Automotive SPICE ohne Fortschritt? Gate4SPICE, Berlin

Cost of error correction

Concept phase:	$1.300,00	SOP:	$ 84.500,00
A Sample:	$4.550,00	Production:	$104.000,00
B Sample :	$5.200,00	Ahead of customer:	$117.000,00
C Sample:	$7.800,00		

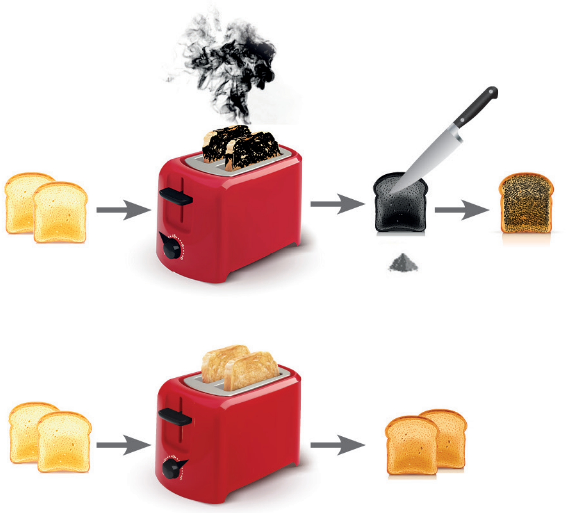
"Source: Study Audi, BMW, Daimler, Porsche and Volkswagen; Seidler, Southworth, ASPICE Made Easy-Case Studies and Lessons Learned, IBM Rational Automotive Engineering Symposium"

WHAT MATCHES BEST?

Reduction of quality cost ▪ Fewer quality issues and lower warranty cost ▪ Early error identification and correction ▪ Global learning and prevention	□	or	□	▪ Increasing number of quality issues ▪ Poor rectification of root causes ▪ Checks are late or incomplete ▪ Poor / unknown product component mature
Managed risks ▪ Early risk identification ▪ Systemic risk tracking and mitigation ▪ Certifications are easy to achieve	□	or	□	▪ Problems appear ‚suddenly' ▪ Reputation drops ▪ Certifications are missing or at risk ▪ New bids are hard to win (e.g. for Safety)
Increase of productivity ▪ People concentrate on their tasks efficiently ▪ Templates and tools are aligned to standards ▪ Limited maintenance cost for standard tools	□	or	□	▪ Fire fighting ▪ Unclear responsibilities ▪ Priority hopping ▪ Poor tool alignment to specific ways of working
Customer Satisfaction ▪ Flexibility within distributed development ▪ Detailed insight into progress and status ▪ Easy adaption of products, standards, and tools to project and culture needs	□	or	□	▪ Internal and external deliverables are late, incomplete, or of poor quality ▪ Deliveries are difficult to integrate ▪ Needed flexibility results in extraordinary cost

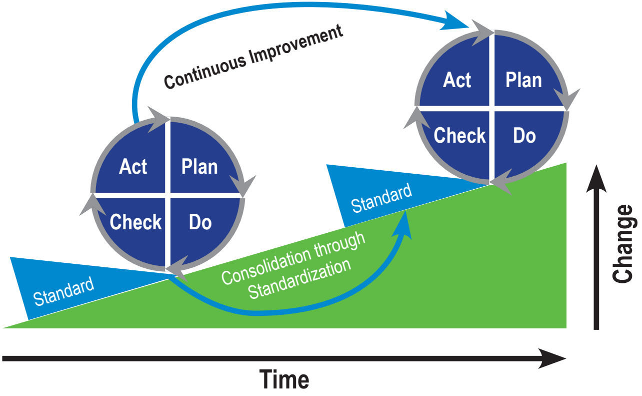
How to benefit from Automotive SPICE®

- Identification of development risks and capabilities associated with suppliers of mechatronic systems
- Identification of risks and capabilities in your own development
- Benchmarking for strengths and potentials of development processes of a project or an organizational unit
- Evaluation of implemented process changes
- Improve transparency, quality and productivity by clarifying and tracking the responsibilities within the development



HIGH QUALITY PROCESSES FOR HIGH QUALITY PRODUCTS

Organizations learn only by improving the standard



By Johannes Vietze - Own work, CC BY-SA 3.0, wikimedia,...

Plan:

- Inform about Automotive SPICE and define goal
- Analyze where you are
- Plan the roadmap for improvement
- Enable persons and infrastructure for change

Do:

- Make the management commitment continuously visible
- Define and agree on process interfaces
- Develop the process solution steps with the people

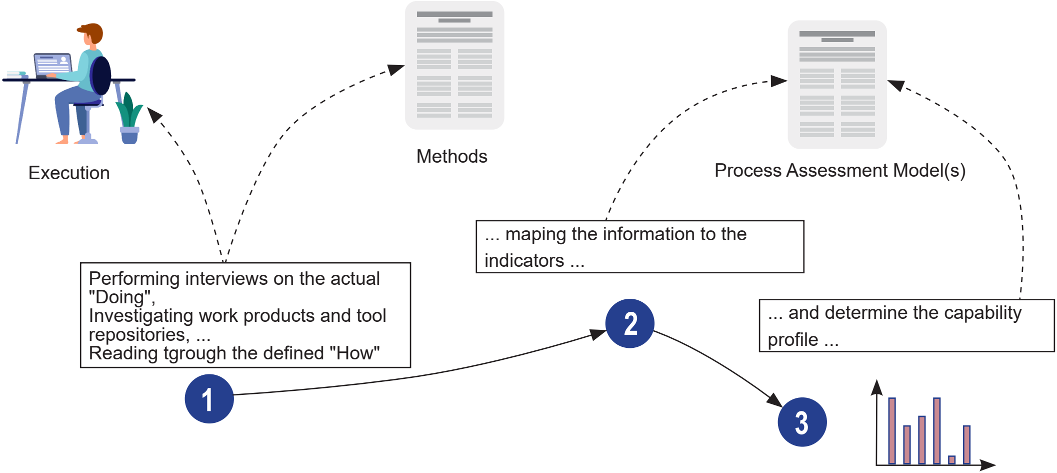
Check:

- Try the new process solution step-by-step
- Check and improve the process and templates

Act:

- Plan and execute the process trainings and roll-out

Performing a process assessment for determining process capability



Performing interviews on the actual "Doing",
Investigating work products and tool repositories, ...
Reading tgrough the defined "How"

... maping the information to the indicators ...

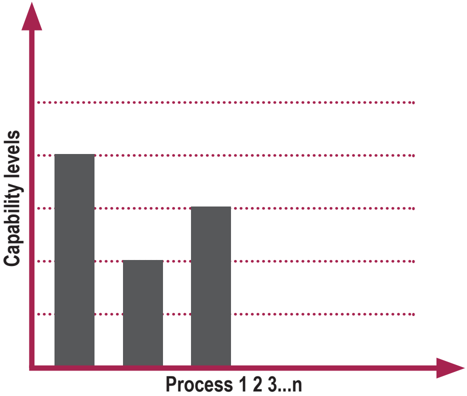
... and determine the capability profile ...

2 Dimensions of Automotive SPICE®

The concept of process capability determination by using the Automotive SPICE® assessment model is based on a two-dimensional framework. The framework consists of a process dimension and a capability dimension.

Capability dimension

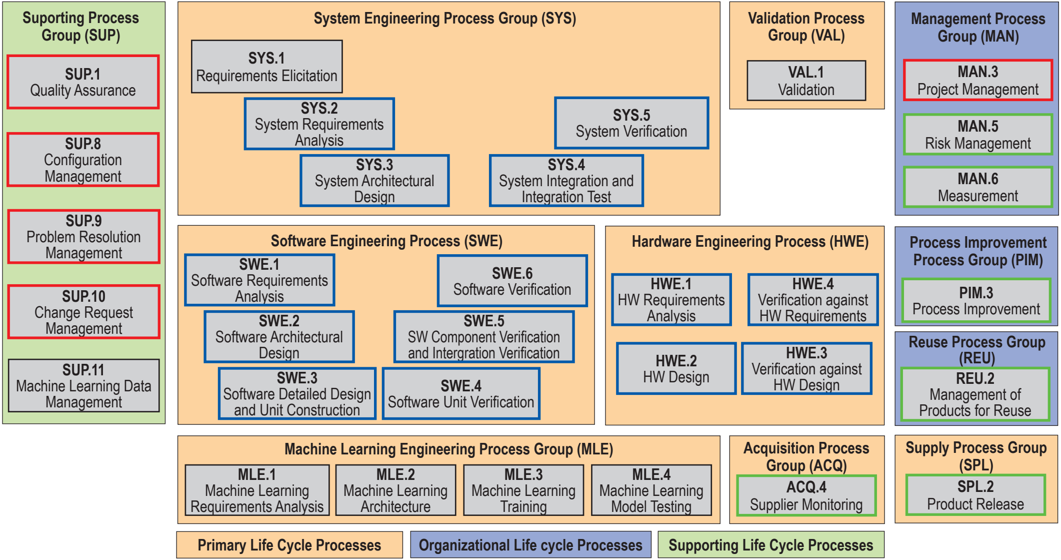
- Capability levels
- Process attributes
- Rating
  - Scale
  - Rating method
  - Aggregation method
- Process capability level model



Process dimension

- Domain and scope
- Processes with purpose and outcomes

Automotive SPICE® 4.0 – Overview of the Process Groups and Processes



Suporting Process Group (SUP)

SUP.1 Quality Assurance

SUP.8 Configuration Management

SUP.9 Problem Resolution Management

SUP.10 Change Request Management

SUP.11 Machine Learning Data Management

System Engineering Process Group (SYS)

SYS.1 Requirements Elicitation

SYS.2 System Requirements Analysis

SYS.3 System Architectural Design

SYS.4 System Integration and Integration Test

SYS.5 System Verification

Validation Process Group (VAL)

VAL.1 Validation

Management Process Group (MAN)

MAN.3 Project Management

MAN.5 Risk Management

MAN.6 Measurement

Software Engineering Process (SWE)

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 SW Component Verification and Intergration Verification

SWE.6 Software Verification

Hardware Engineering Process (HWE)

HWE.1 HW Requirements Analysis

HWE.2 HW Design

HWE.3 Verification against HW Design

HWE.4 Verification against HW Requirements

Process Improvement Process Group (PIM)

PIM.3 Process Improvement

Reuse Process Group (REU)

REU.2 Management of Products for Reuse

Machine Learning Engineering Process Group (MLE)

MLE.1 Machine Learning Requirements Analysis

MLE.2 Machine Learning Architecture

MLE.3 Machine Learning Training

MLE.4 Machine Learning Model Testing

Acquisition Process Group (ACQ)

ACQ.4 Supplier Monitoring

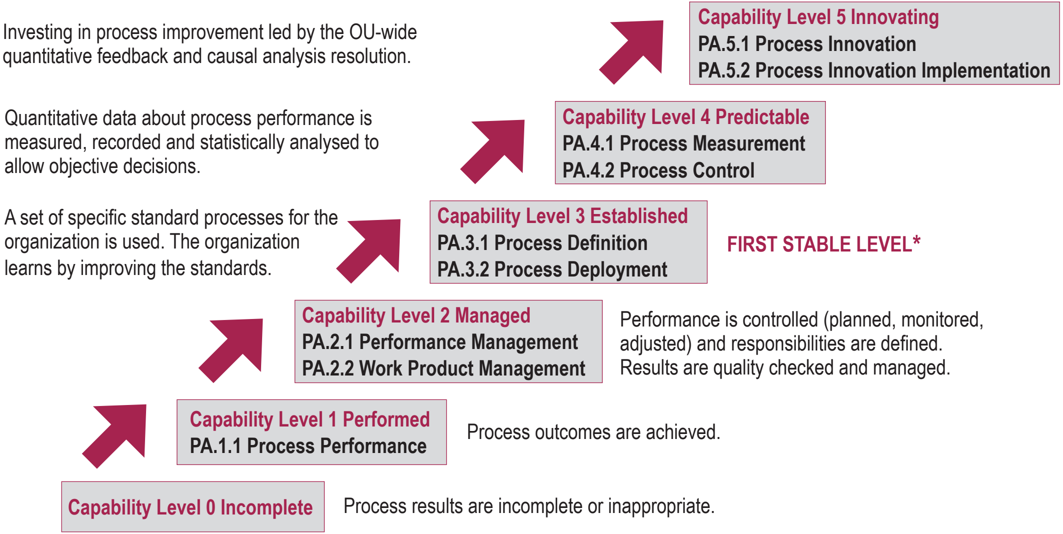
Supply Process Group (SPL)

SPL.2 Product Release

Primary Life Cycle Processes | **Organizational Life cycle Processes** | **Supporting Life Cycle Processes**

The **Recommended VDA Scope** contains the **BASIS** processes and the relevant **PLUG-IN** processes. The **FLEX** processes are optional.

Automotive SPICE® Capability Dimension



Capability Level 5 Innovating
PA.5.1 Process Innovation
PA.5.2 Process Innovation Implementation

Investing in process improvement led by the OU-wide quantitative feedback and causal analysis resolution.

Capability Level 4 Predictable
PA.4.1 Process Measurement
PA.4.2 Process Control

Quantitative data about process performance is measured, recorded and statistically analysed to allow objective decisions.

Capability Level 3 Established
PA.3.1 Process Definition
PA.3.2 Process Deployment

A set of specific standard processes for the organization is used. The organization learns by improving the standards.

FIRST STABLE LEVEL*

Capability Level 2 Managed
PA.2.1 Performance Management
PA.2.2 Work Product Management

Performance is controlled (planned, monitored, adjusted) and responsibilities are defined. Results are quality checked and managed.

Capability Level 1 Performed
PA.1.1 Process Performance

Process outcomes are achieved.

Capability Level 0 Incomplete

Process results are incomplete or inappropriate.

* By experience, lower Capability Levels are not stable i.e. either increase or decrease over a period of about 18 months.

4 Key Concepts of Automotive SPICE®

1. Use qualified input to aim qualified output

Each expert shall perform the work using **qualified input** and shall provide **qualified output** to the next one in the value chain. *Hints:*

- Divide the work into small tasks (e.g. < 40h)
- Get the tasks 'done' continuously one after another
- Qualify and approve the work products continuously
- Use clear criteria and efficient methods to qualify

2. Agree and summarize

Engineering processes:

- **Agree** on requirements and design
- **Summarize** results of step-by-step verification

Management and support processes:

- **Agree** on strategies, plans and schedules
- **Summarize** the results and report to relevant parties

3. Divide and control

On system, domain, sub-domain and component level:

1. **Specify** and **design** the solution.
2. **Delegate to lower level** OR
   implement solution on unit level.
3. **Integrate** and **verify integration** against the design before **qualifying** the solution against the specification.

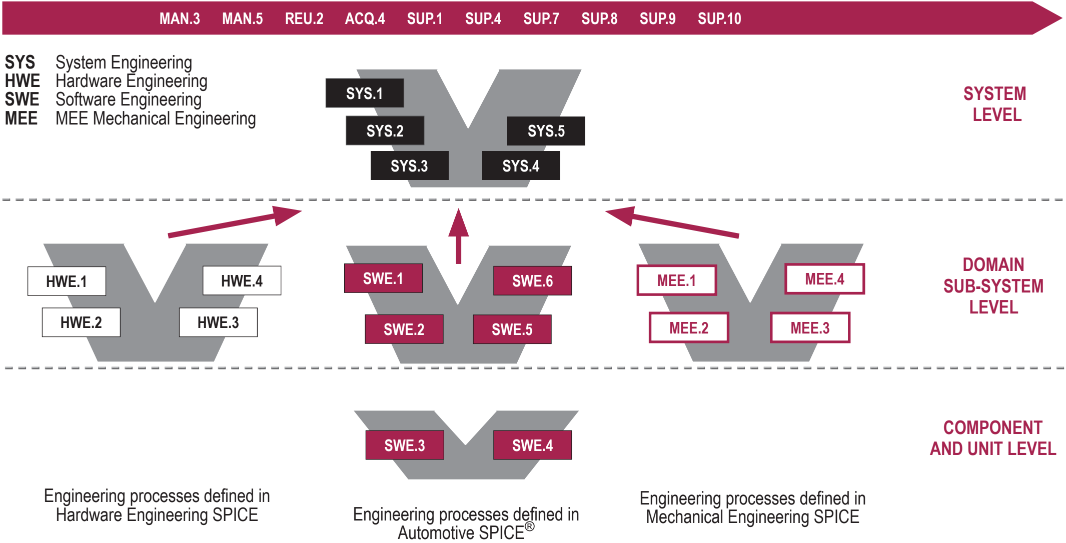
4. Traceability

Each item (requirement, design element, implementation, test case / result, finding, scheduled activity, …) has to have a **reference to its source and to its verification.**

The traceability is used ...
… to check for **consistency**,
… to analyze its **impact** and
… to show **completeness**.

Automotive SPICE® Scope and Plug-In Concept

Automotive SPICE® is a standard used for improving and evaluating development processes of mechatronic systems. It is a framework which applies to traditional or agile developments. It supports the engineering of products which are critical according to safety or security. With the “Plug-in concept” of Automotive SPICE® version 3.x the processes for development of mechanical and EE parts are more and more in focus.

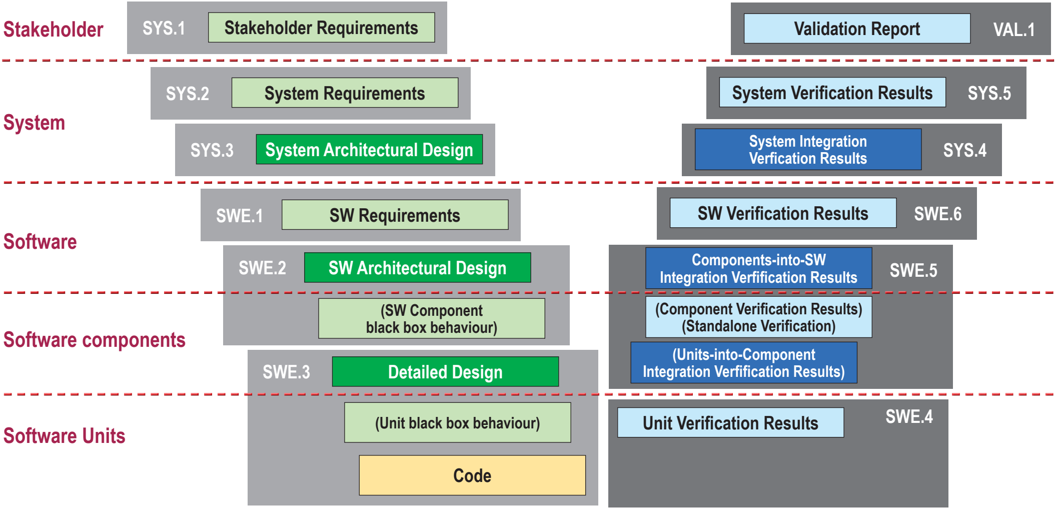


Qualification Verfication *Versus* Integration Verfication

Some processes have similar purpose, but differ in their level of detail:

Left: In requirements processes the problem is specified; In design processes the planned solutions, their structure elements, interfaces and dynamic behavior are specified

Right: Tests verify the test object either versus the related specification (light green) or versus the related design (dark green)



Stakeholder
SYS.1
Stakeholder Requirements
Validation Report
VAL.1
System
SYS.2
System Requirements
System Verification Results
SYS.5
SYS.3
System Architectural Design
System Integration Verfication Results
SYS.4
Software
SWE.1
SW Requirements
SW Verification Results
SWE.6
SWE.2
SW Architectural Design
Components-into-SW Integration Verfification Results
SWE.5
Software components
(SW Component black box behaviour)
(Component Verification Results) (Standalone Verification)
SWE.3
Detailed Design
(Units-into-Component Integration Verfification Results)
Software Units
(Unit black box behaviour)
Unit Verification Results
SWE.4
Code

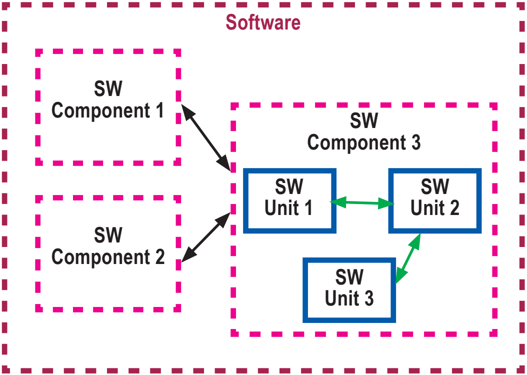
SWE.4: Verification of a single software unit (blue)
against the detailed design.

SWE.5: Integration, and integration verification of software units into their component (green)
against the detailed design.

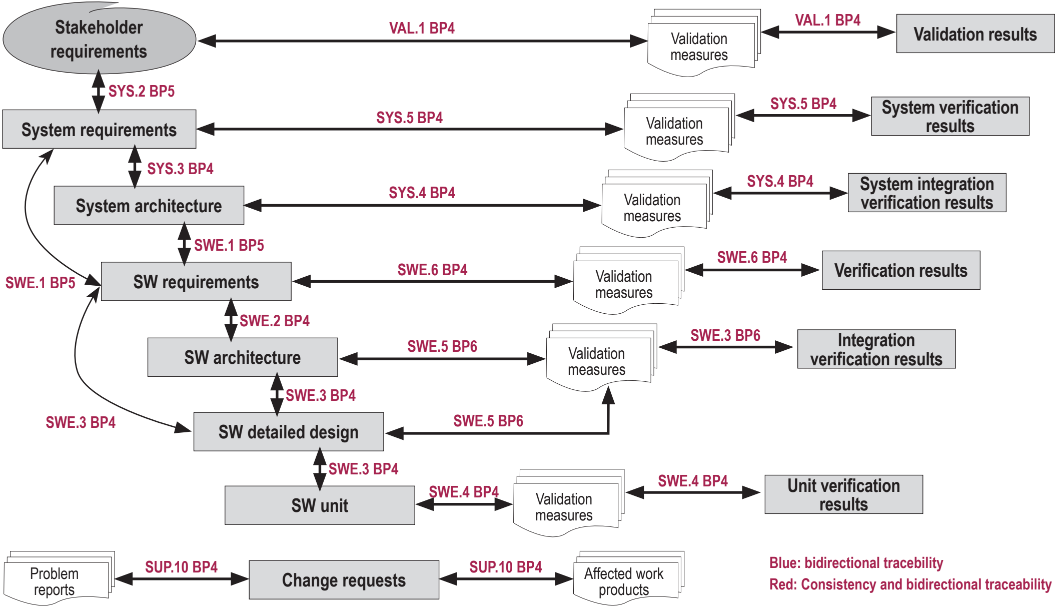
SWE.5 Verification of a single software component (prior to integration with other components) (pink)
against the detailed design.

SWE.5: Integration and verification of software components (black)
against the software architectual design.

SWE.6: Verification of the integrated software (red)
against the software requirements.



Consistency and traceability between system and software work products



Stakeholder requirements — VAL.1 BP4 — Validation measures — VAL.1 BP4 — Validation results

SYS.2 BP5

System requirements — SYS.5 BP4 — Validation measures — SYS.5 BP4 — System verification results

SYS.3 BP4

System architecture — SYS.4 BP4 — Validation measures — SYS.4 BP4 — System integration verification results

SWE.1 BP5

SWE.1 BP5

SW requirements — SWE.6 BP4 — Validation measures — SWE.6 BP4 — Verification results

SWE.2 BP4

SW architecture — SWE.5 BP6 — Validation measures — SWE.3 BP6 — Integration verification results

SWE.3 BP4

SWE.3 BP4

SW detailed design — SWE.5 BP6

SWE.3 BP4

SW unit — SWE.4 BP4 — Validation measures — SWE.4 BP4 — Unit verification results

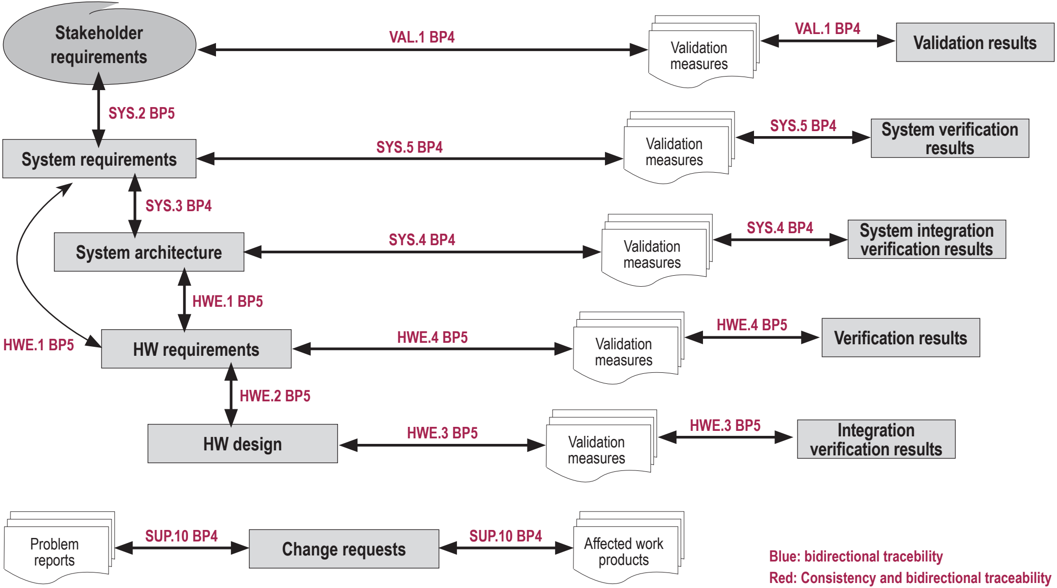
Problem reports — SUP.10 BP4 — Change requests — SUP.10 BP4 — Affected work products

Blue: bidirectional tracebility

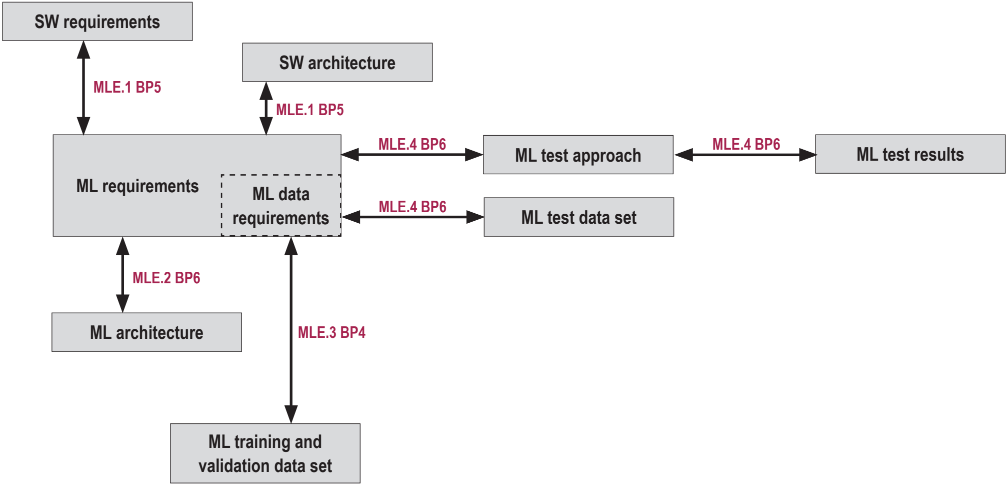
Red: Consistency and bidirectional traceability

Bidirectional traceability supports coverage, consistency and impact analysis

Consistency and traceability between system and hardware work products



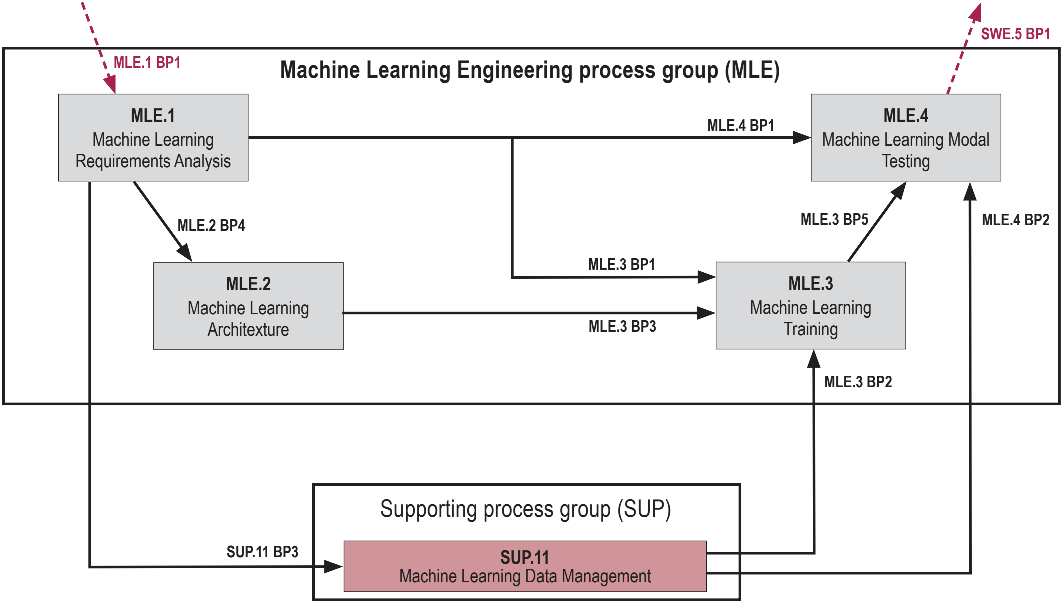
Consistency and traceability between ML work products



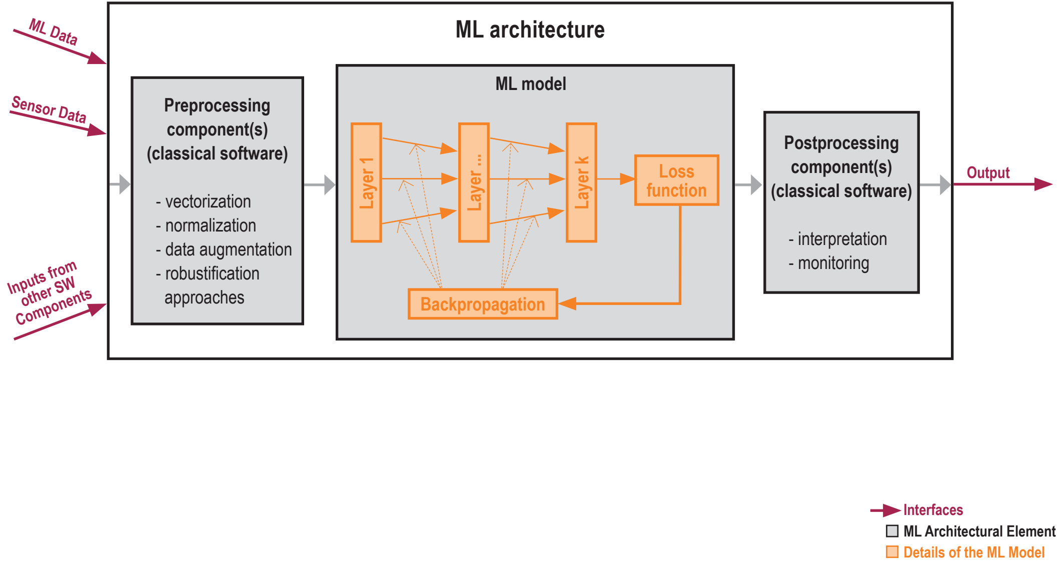
Blue: bidirectional tracebility

Red: Consistency and bidirectional traceability

Interdependencies within MLE and SUP.11


MLE.1 BP1
SWE.5 BP1
Machine Learning Engineering process group (MLE)
MLE.1
Machine Learning Requirements Analysis
MLE.4 BP1
MLE.4
Machine Learning Modal Testing
MLE.2 BP4
MLE.3 BP5
MLE.4 BP2
MLE.3 BP1
MLE.2
Machine Learning Architexture
MLE.3
Machine Learning Training
MLE.3 BP3
MLE.3 BP2
Supporting process group (SUP)
SUP.11 BP3
SUP.11
Machine Learning Data Management

Example of an ML Architecture



ML Data

Sensor Data

Inputs from other SW Components

ML architecture

Preprocessing component(s) (classical software)

- vectorization
- normalization
- data augmentation
- robustification approaches

ML model

Layer 1

Layer ...

Layer k

Loss function

Backpropagation

Postprocessing component(s) (classical software)

- interpretation
- monitoring

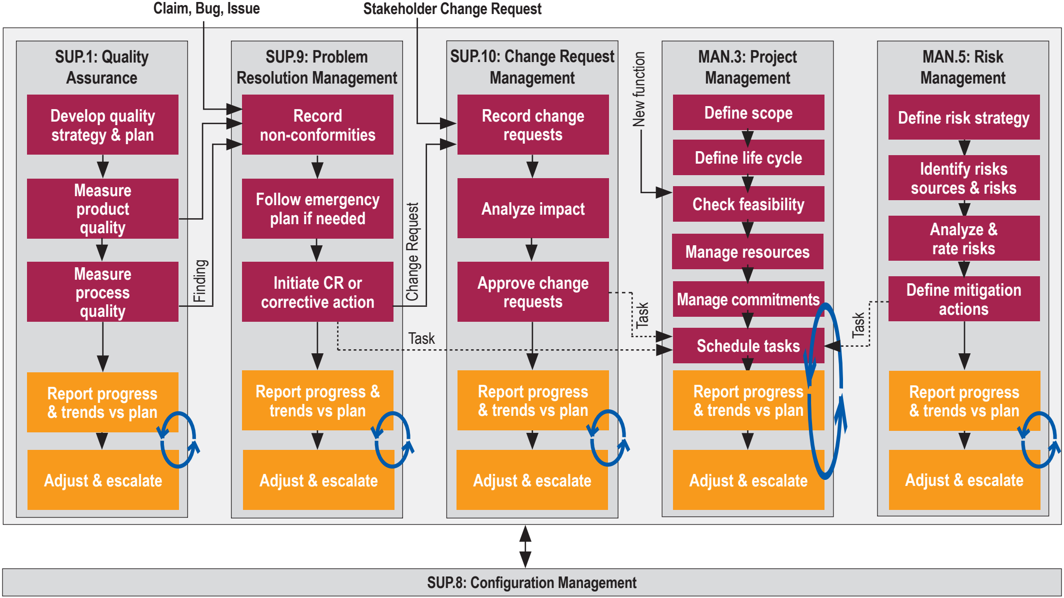
Output

→ Interfaces

□ ML Architectural Element

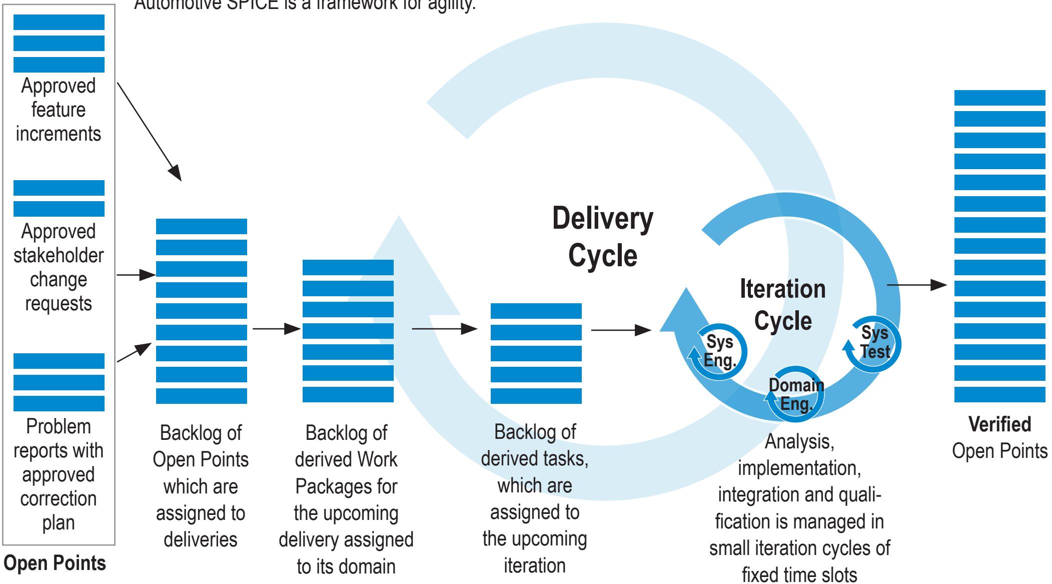
🟧 Details of the ML Model

An interpretation of the main Automotive SPICE® management and support processes


Claim, Bug, Issue
Stakeholder Change Request
SUP.1: Quality Assurance
Develop quality strategy & plan
Measure product quality
Measure process quality
Report progress & trends vs plan
Adjust & escalate
Finding
SUP.9: Problem Resolution Management
Record non-conformities
Follow emergency plan if needed
Initiate CR or corrective action
Report progress & trends vs plan
Adjust & escalate
Change Request
Task
SUP.10: Change Request Management
Record change requests
Analyze impact
Approve change requests
Report progress & trends vs plan
Adjust & escalate
New function
Task
MAN.3: Project Management
Define scope
Define life cycle
Check feasibility
Manage resources
Manage commitments
Schedule tasks
Report progress & trends vs plan
Adjust & escalate
Task
MAN.5: Risk Management
Define risk strategy
Identify risks sources & risks
Analyze & rate risks
Define mitigation actions
Report progress & trends vs plan
Adjust & escalate
SUP.8: Configuration Management

Automotive SPICE® and Agility

Automotive SPICE applications can benefit by agile methods, e.g. in project management.
Automotive SPICE is a framework for agility.



Approved feature increments

Approved stakeholder change requests

Problem reports with approved correction plan

Open Points

Backlog of Open Points which are assigned to deliveries

Backlog of derived Work Packages for the upcoming delivery assigned to its domain

Delivery Cycle

Backlog of derived tasks, which are assigned to the upcoming iteration

Iteration Cycle

Sys Eng.

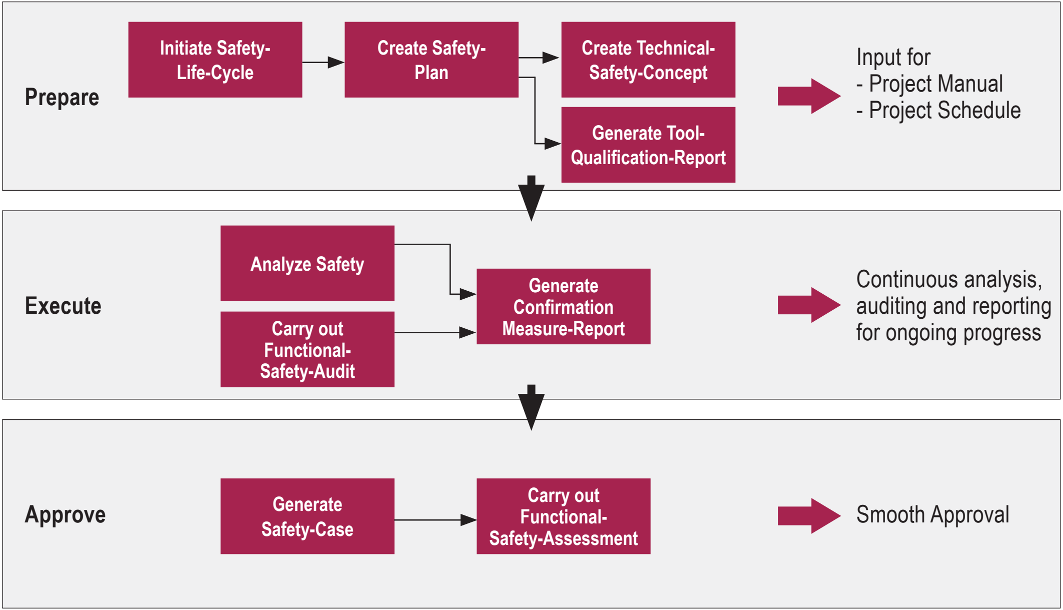
Domain Eng.

Sys Test

Analysis, implementation, integration and qualification is managed in small iteration cycles of fixed time slots

Verified Open Points

An Example for A Functional Safety Implementation



Prepare

Initiate Safety-Life-Cycle

Create Safety-Plan

Create Technical-Safety-Concept

Generate Tool-Qualification-Report

Input for
- Project Manual
- Project Schedule

Execute

Analyze Safety

Carry out Functional-Safety-Audit

Generate Confirmation Measure-Report

Continuous analysis, auditing and reporting for ongoing progress

Approve

Generate Safety-Case

Carry out Functional-Safety-Assessment

Smooth Approval

Automotive SPICE® and Functional Safety

Automotive SPICE			ISO 26262
MAN.3	Project Management	++	Safety management during the concept phase and the product development
		+++	Item definition (top level)
		++	Initiation of the safety lifecycle
		++	Initiation of product development at the system level
		++	Initiation of product development at the hardware level
		++	Initiation of product development at the software level
ACQ.4	Supplier Monitoring	++	Interfaces within distributed developments
SUP.1	Quality Assurance	++	Safety management during the concept phase and the product development
		+++	Functional safety assessment
SUP.2	Verification	+++	Verification
SUP.7	Documentation	+++	Documentation
SUP.8	Configuration Management	++	Configuration Management
SUP.10	Change Request Management	++	Change Management
SPL.2	Product Release	+++	Release for production

A successful application of Automotive SPICE supports the compliance to ISO 26262.

The related Automotive SPICE process provides weak / medium / strong (+/++/+++) support to the related chapter in ISO 26262.

Automotive SPICE® and Functional Safety

Automotive SPICE			ISO 26262
SYS.1	Requirements Elicitation	+++	Item definition (detailed level)
SYS.2	System Requirements Analysis	+	Functional safety concept
		+	Specification of the technical safety requirements
		++	Specification and management of safety requirements
SYS.3	System Architectural Design	++	System design
SYS.4	System Integration and Integration Test	++	Item integration and testing
SWE.1	Software Requirements Analysis	++	Specificaiton of software safety requirements
SWE.2	Software Architectural Design	++	Software architectural design
SWE.3	Software Detailed Design and Unit Construction	++	Software unit design and implementation
SWE.4	Software Unit Verification	++	Software unit testing
SWE.5	Software Integration and Integration Tests	++	Software integration and testing
SWE.6	Software Qualification Testing	++	Verification of software safety requirements

If compliance to ISO 26262 is required, the related chapters shall be considered during application of Automotive SPICE.

What is new in ASPICE® 4.0

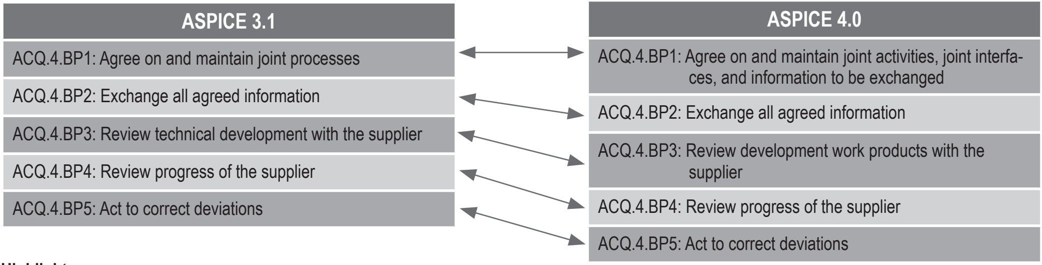
What is new in ASPICE 4 - Overview

	Deleted	Added
Processes	Processes of low relevance deleted: - ACQ.3,11-15; - SUP.2,4,7; SPL.1	• Hardware Engineering: HWE.1-4; • Machine Learning: MLE.1,4 • Validation: VAL.1
Practices	Many BPs and GPs restructured and summarized to less practices but (nearly) same content. Therefore, much less BPs and GPs per Process or Process Attribute in ASPICE 4.0	
Terms	Output Work Product	Information Item
	see Authenticity and Integrity	Verification
	Specification	Requirements
Training and Certification	Guideline Training	New initial certification level: Process Expert (PE) New training concept in Oct. 2024: PE–PA–CA Certification cards with model extensions (VDA only)
Guideline	Version 1 with a lot of additional information relevant for process definition / improvement	Version 2 with rating information only
VDA Scope	The new “**Recommended VDA Scope**“ contains the processes in the Base Process Group and at least one Plug-In of the Engineering Process Group, but none of the Flex / Optional Process Group	

ACQ.4 Supplier Monitoring

The **purpose** of the Supplier Monitoring Process is to track and assess the performance of the supplier against agreed requirements.

The **purpose** is to track and assess the performance of an external contract-based supplier company against agreed commitments.



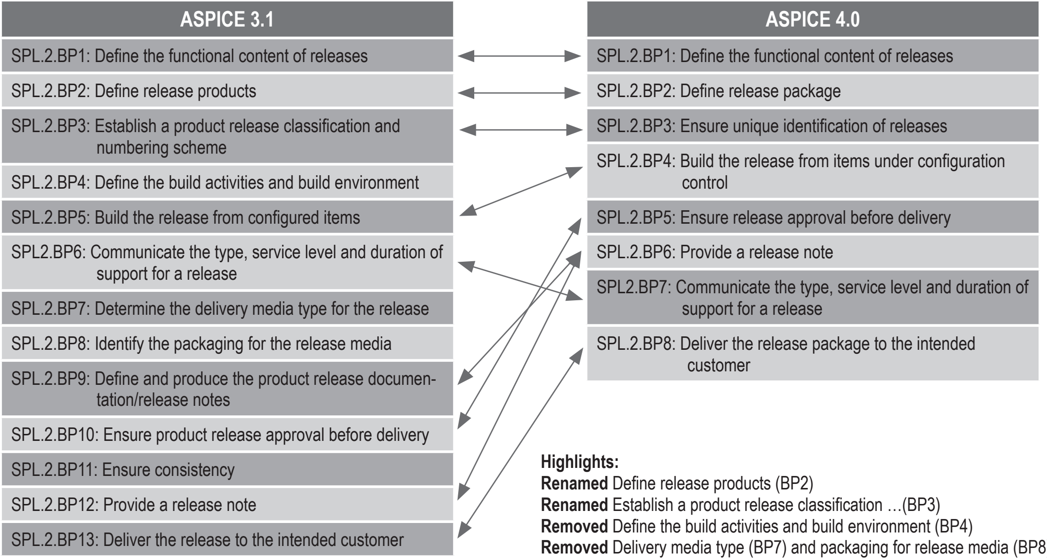
ASPICE 3.1	ASPICE 4.0
ACQ.4.BP1: Agree on and maintain joint processes	ACQ.4.BP1: Agree on and maintain joint activities, joint interfaces, and information to be exchanged
ACQ.4.BP2: Exchange all agreed information	ACQ.4.BP2: Exchange all agreed information
ACQ.4.BP3: Review technical development with the supplier	ACQ.4.BP3: Review development work products with the supplier
ACQ.4.BP4: Review progress of the supplier	ACQ.4.BP4: Review progress of the supplier
ACQ.4.BP5: Act to correct deviations	ACQ.4.BP5: Act to correct deviations

Highlights:
Renamed Agree on and maintain joint processes (BP1)
Renamed Review technical development with the supplier (BP3)

SPL.2 Product Release

The **purpose** of the Product Release Process is to control the release of a product to the intended customer

The **purpose** is to control the release of a product to the intended customer.



ASPICE 3.1

- SPL.2.BP1: Define the functional content of releases
- SPL.2.BP2: Define release products
- SPL.2.BP3: Establish a product release classification and numbering scheme
- SPL.2.BP4: Define the build activities and build environment
- SPL.2.BP5: Build the release from configured items
- SPL2.BP6: Communicate the type, service level and duration of support for a release
- SPL.2.BP7: Determine the delivery media type for the release
- SPL.2.BP8: Identify the packaging for the release media
- SPL.2.BP9: Define and produce the product release documentation/release notes
- SPL.2.BP10: Ensure product release approval before delivery
- SPL.2.BP11: Ensure consistency
- SPL.2.BP12: Provide a release note
- SPL.2.BP13: Deliver the release to the intended customer

ASPICE 4.0

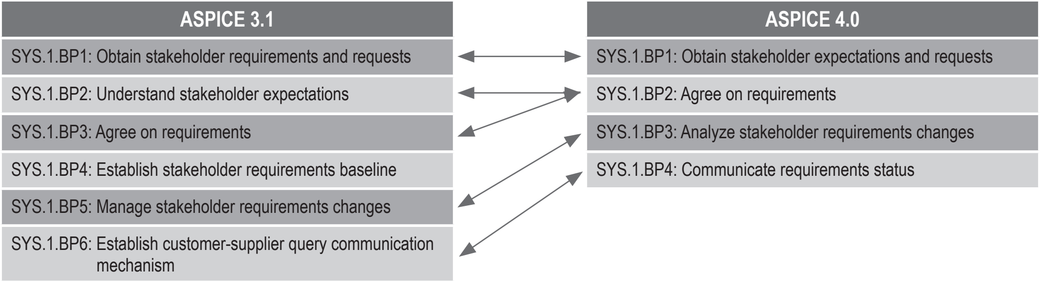
- SPL.2.BP1: Define the functional content of releases
- SPL.2.BP2: Define release package
- SPL.2.BP3: Ensure unique identification of releases
- SPL.2.BP4: Build the release from items under configuration control
- SPL.2.BP5: Ensure release approval before delivery
- SPL.2.BP6: Provide a release note
- SPL2.BP7: Communicate the type, service level and duration of support for a release
- SPL.2.BP8: Deliver the release package to the intended customer

Highlights:
Renamed Define release products (BP2)
Renamed Establish a product release classification …(BP3)
Removed Define the build activities and build environment (BP4)
Removed Delivery media type (BP7) and packaging for release media (BP8)
Removed Ensure consistency (BP11)

SYS.1 Requirements Elicitation

The **purpose** of the Requirements Elicitation Process is to gather, process, and track evolving stakeholder needs and requirements throughout the lifecycle of the product and/or service so as to establish a requirements baseline that serves as the basis for defining the needed work products.

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.



ASPICE 3.1

SYS.1.BP1: Obtain stakeholder requirements and requests

SYS.1.BP2: Understand stakeholder expectations

SYS.1.BP3: Agree on requirements

SYS.1.BP4: Establish stakeholder requirements baseline

SYS.1.BP5: Manage stakeholder requirements changes

SYS.1.BP6: Establish customer-supplier query communication mechanism

ASPICE 4.0

SYS.1.BP1: Obtain stakeholder expectations and requests

SYS.1.BP2: Agree on requirements

SYS.1.BP3: Analyze stakeholder requirements changes

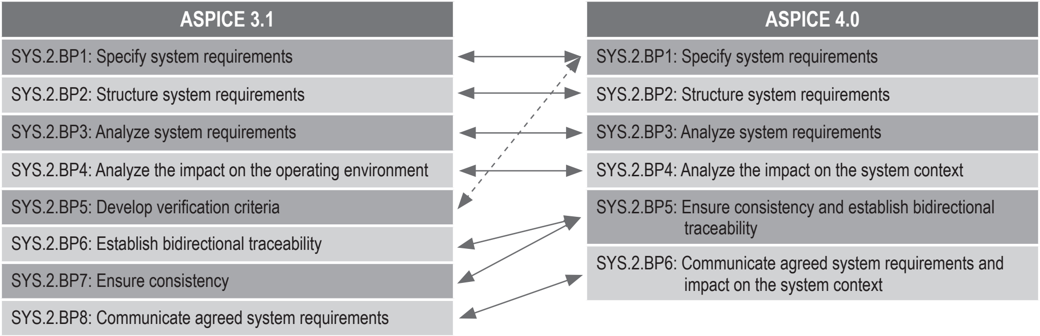
SYS.1.BP4: Communicate requirements status

Highlights:
Combined Understand stakeholder expectations (BP2) and Agree on requirements (BP3)
Removed Establish stakeholder requirements baseline (BP4)
Renamed Manage stakeholder requirements changes (BP5)
Renamed Establish customer-supplier query communication mechanism (BP6)

SYS.2 System Requirements Analysis

The **purpose** of the System Requirements Analysis Process is to transform the defined stakeholder requirements into a set of system requirements that will guide the design of the system.

The **purpose** is to establish a structured and analyzed set of system requirements consistent with the stakeholder requirements.



ASPICE 3.1

SYS.2.BP1: Specify system requirements

SYS.2.BP2: Structure system requirements

SYS.2.BP3: Analyze system requirements

SYS.2.BP4: Analyze the impact on the operating environment

SYS.2.BP5: Develop verification criteria

SYS.2.BP6: Establish bidirectional traceability

SYS.2.BP7: Ensure consistency

SYS.2.BP8: Communicate agreed system requirements

ASPICE 4.0

SYS.2.BP1: Specify system requirements

SYS.2.BP2: Structure system requirements

SYS.2.BP3: Analyze system requirements

SYS.2.BP4: Analyze the impact on the system context

SYS.2.BP5: Ensure consistency and establish bidirectional traceability

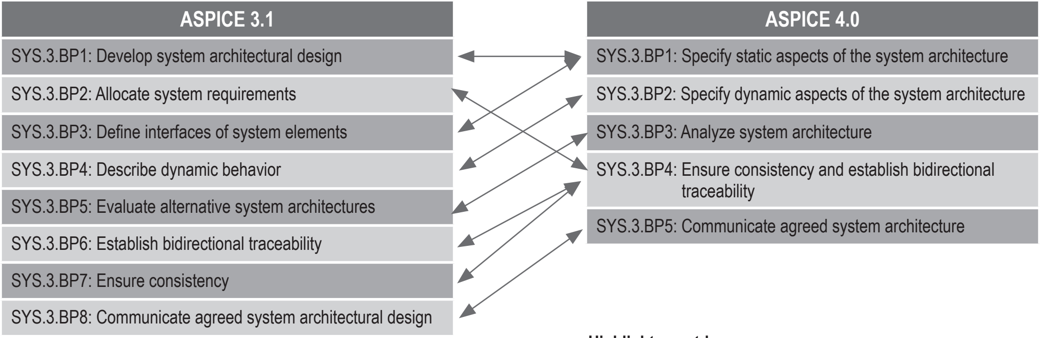
SYS.2.BP6: Communicate agreed system requirements and impact on the system context

Highlights:
Removed Develop verification criteria (BP5) but covered in Note section **[BP1]**
Renamed operating environment (BP4) to system context **[BP4]**
Combined traceability (BP6) and consistency (BP7)

SYS.3 System Architectural Design

The **purpose** of the System Architectural Design Process is to establish a system architectural design and identify which system requirements are to be allocated to which elements of the system, and to evaluate the system architectural design against defined criteria.

The **purpose** is to establish an analyzed system architecture, comprising static and dynamic aspects, consistent with the system requirements.



ASPICE 3.1
SYS.3.BP1: Develop system architectural design
SYS.3.BP2: Allocate system requirements
SYS.3.BP3: Define interfaces of system elements
SYS.3.BP4: Describe dynamic behavior
SYS.3.BP5: Evaluate alternative system architectures
SYS.3.BP6: Establish bidirectional traceability
SYS.3.BP7: Ensure consistency
SYS.3.BP8: Communicate agreed system architectural design

ASPICE 4.0
SYS.3.BP1: Specify static aspects of the system architecture
SYS.3.BP2: Specify dynamic aspects of the system architecture
SYS.3.BP3: Analyze system architecture
SYS.3.BP4: Ensure consistency and establish bidirectional traceability
SYS.3.BP5: Communicate agreed system architecture

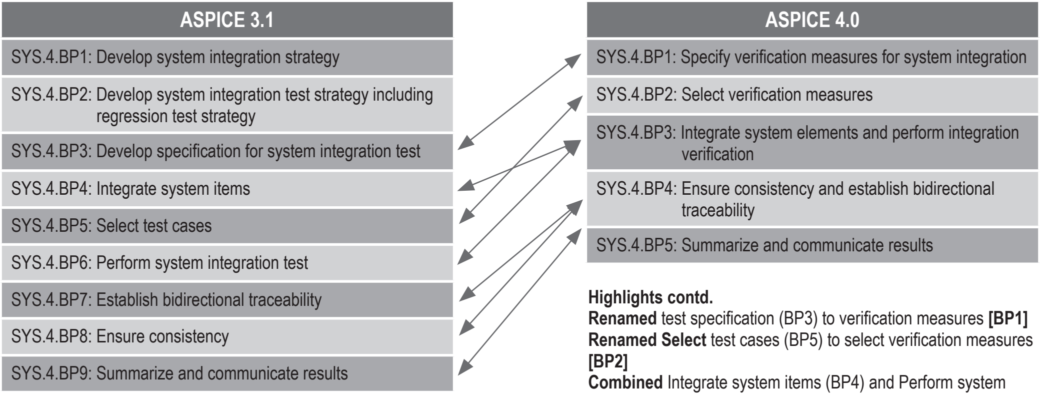
Highlights:
Combined Develop system architectural design (BP1) and Define interfaces of system elements (BP3)
Removed Allocate system requirements (BP2)
Renamed Describe dynamic behavior (BP4)

Highlights contd.
Removed Evaluate alternative system architectures (BP5) but softened the expectations in **[BP3]**
Added Analyze system architecture **[BP3]**
Emphasis on special characteristics of hardware element to cover functional safety aspects (e.g., FMEA, reuse, proven-in use) **[BP3]**
Combined traceability (BP6) and consistency (BP7)

SYS.4 System Integration and Integration Test (Verification)

The **purpose** of the System Integration and Integration Test Process is to integrate the system items to produce an integrated system consistent with the system architectural design and to ensure that the system items are tested to provide evidence for compliance of the integrated system items with the system architectural design, including the interfaces between system items.

The **purpose** is to integrate systems elements and verify that the integrated system elements are consistent with the system architecture.



ASPICE 3.1
SYS.4.BP1: Develop system integration strategy
SYS.4.BP2: Develop system integration test strategy including regression test strategy
SYS.4.BP3: Develop specification for system integration test
SYS.4.BP4: Integrate system items
SYS.4.BP5: Select test cases
SYS.4.BP6: Perform system integration test
SYS.4.BP7: Establish bidirectional traceability
SYS.4.BP8: Ensure consistency
SYS.4.BP9: Summarize and communicate results

ASPICE 4.0
SYS.4.BP1: Specify verification measures for system integration
SYS.4.BP2: Select verification measures
SYS.4.BP3: Integrate system elements and perform integration verification
SYS.4.BP4: Ensure consistency and establish bidirectional traceability
SYS.4.BP5: Summarize and communicate results

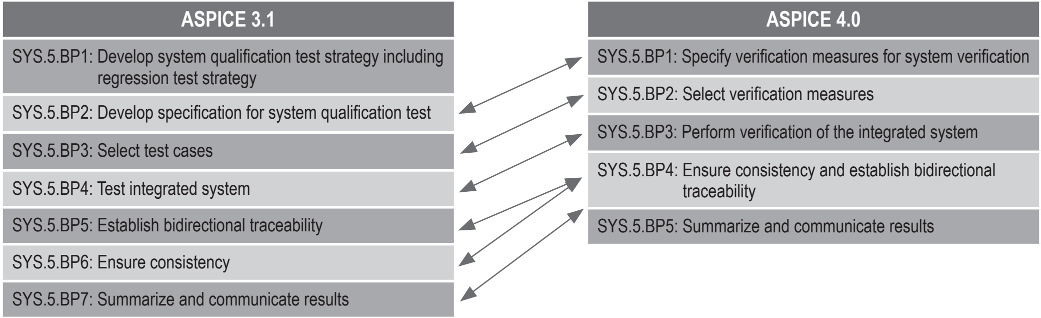
Highlights:
Rename from “Test” To “Verification”
Removed Develop system integration strategy (BP1) and system integration test strategy but expected in GP 2.1.1

Highlights contd.
Renamed test specification (BP3) to verification measures **[BP1]**
Renamed Select test cases (BP5) to select verification measures **[BP2]**
Combined Integrate system items (BP4) and Perform system integration test (BP6)
Combined traceability (BP7) and consistency (BP8)

SYS.5 System Qualification Test (System Verification)

The **purpose** of the System Qualification Test Process is to ensure that the integrated system is tested to provide evidence for compliance with the system requirements and that the system is ready for delivery.

The **purpose** is to ensure that the system is verified to be consistent with the system requirements.



ASPICE 3.1

SYS.5.BP1: Develop system qualification test strategy including regression test strategy

SYS.5.BP2: Develop specification for system qualification test

SYS.5.BP3: Select test cases

SYS.5.BP4: Test integrated system

SYS.5.BP5: Establish bidirectional traceability

SYS.5.BP6: Ensure consistency

SYS.5.BP7: Summarize and communicate results

ASPICE 4.0

SYS.5.BP1: Specify verification measures for system verification

SYS.5.BP2: Select verification measures

SYS.5.BP3: Perform verification of the integrated system

SYS.5.BP4: Ensure consistency and establish bidirectional traceability

SYS.5.BP5: Summarize and communicate results

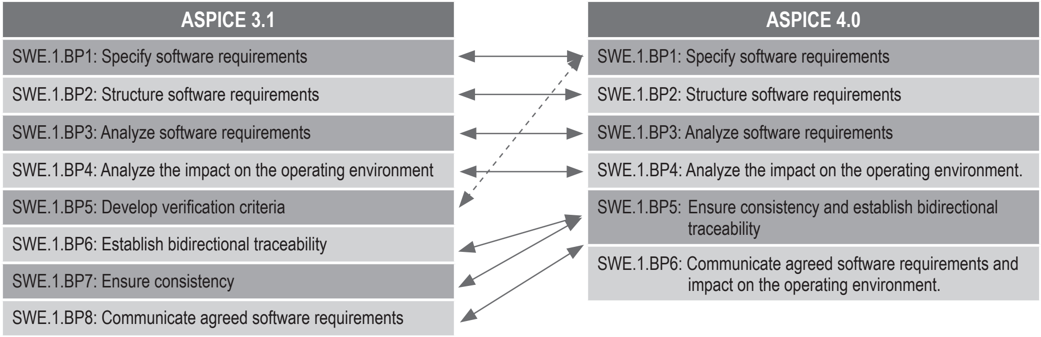
Highlights:
Renamed system qualification test to system verification
Removed Develop system qualification test strategy (BP1) but expected in GP 2.1.1
Renamed test specification (BP2) to verification measures **[BP1]**

Highlights contd.
Renamed Select test cases (BP3) to select verification measures **[BP2]**
Renamed Test integrated system (BP4) to Perform verification of the integrated system **[BP3]**
Combined traceability (BP5) and consistency (BP6)

SWE.1 Software Requirements Analysis

The **purpose** of the Software Requirements Analysis Process is to transform the software related parts of the system requirements into a set of software requirements.

The **purpose** is to establish a structured and analyzed set of software requirements consistent with the system requirements, and the system architecture.



ASPICE 3.1

- SWE.1.BP1: Specify software requirements
- SWE.1.BP2: Structure software requirements
- SWE.1.BP3: Analyze software requirements
- SWE.1.BP4: Analyze the impact on the operating environment
- SWE.1.BP5: Develop verification criteria
- SWE.1.BP6: Establish bidirectional traceability
- SWE.1.BP7: Ensure consistency
- SWE.1.BP8: Communicate agreed software requirements

ASPICE 4.0

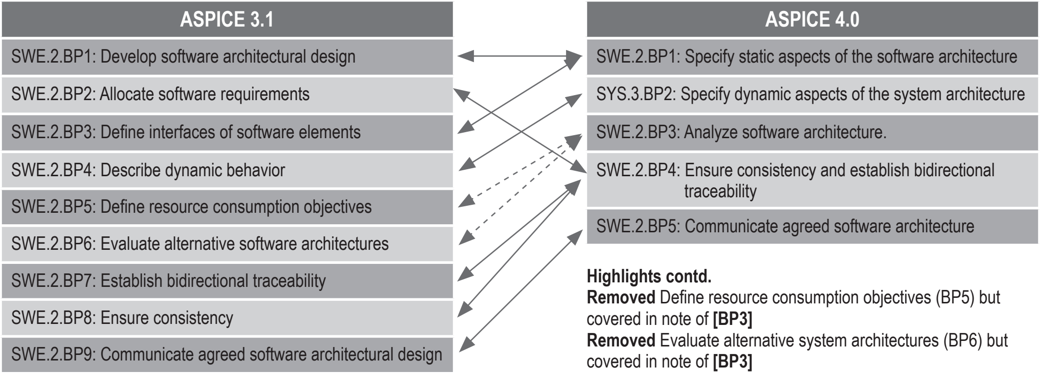
- SWE.1.BP1: Specify software requirements
- SWE.1.BP2: Structure software requirements
- SWE.1.BP3: Analyze software requirements
- SWE.1.BP4: Analyze the impact on the operating environment.
- SWE.1.BP5: Ensure consistency and establish bidirectional traceability
- SWE.1.BP6: Communicate agreed software requirements and impact on the operating environment.

Highlights:
Removed application parameter (BP1) but covered in Information item (work product) characteristics
Removed Develop verification criteria (BP5) but covered in Note section **[BP1] Combined** traceability (BP6) and consistency (BP7)

SWE.2 Software Architectural Design

The **purpose** of the System Architectural Design Process is to establish a system architectural design and identify which system requirements are to be allocated to which elements of the system, and to evaluate the system architectural design against defined criteria.

The **purpose** is to establish an analyzed software architecture, comprising static and dynamic aspects, consistent with the software requirements.



ASPICE 3.1

SWE.2.BP1: Develop software architectural design

SWE.2.BP2: Allocate software requirements

SWE.2.BP3: Define interfaces of software elements

SWE.2.BP4: Describe dynamic behavior

SWE.2.BP5: Define resource consumption objectives

SWE.2.BP6: Evaluate alternative software architectures

SWE.2.BP7: Establish bidirectional traceability

SWE.2.BP8: Ensure consistency

SWE.2.BP9: Communicate agreed software architectural design

ASPICE 4.0

SWE.2.BP1: Specify static aspects of the software architecture

SYS.3.BP2: Specify dynamic aspects of the system architecture

SWE.2.BP3: Analyze software architecture.

SWE.2.BP4: Ensure consistency and establish bidirectional traceability

SWE.2.BP5: Communicate agreed software architecture

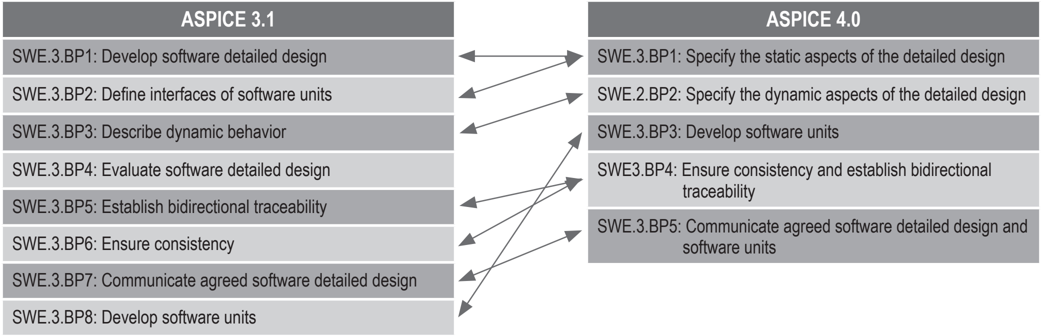
Highlights:
Combined Develop software architectural design (BP1) and Define interfaces of software elements (BP3)
Removed Allocate system requirements (BP2)
Renamed Describe dynamic behavior (BP4)

Highlights contd.
Removed Define resource consumption objectives (BP5) but covered in note of **[BP3]**
Removed Evaluate alternative system architectures (BP6) but covered in note of **[BP3]**
Added Analyze system architecture **[BP3]**
Emphasis on design decisions/rationale to consider reuse, provenin use and make vs. buy) **[BP3]**
Combined traceability (BP6) and consistency (BP7)

SWE.3 Software Detailed Design and Unit Construction

The **purpose** of the Software Detailed Design and Unit Construction Process is to provide an evaluated detailed design for the software components and to specify and to produce the software units.

The **purpose** is to establish a software detailed design, comprising static and dynamic aspects, consistent with the software architecture, and to construct software units consistent with the software detailed design.



ASPICE 3.1

SWE.3.BP1: Develop software detailed design

SWE.3.BP2: Define interfaces of software units

SWE.3.BP3: Describe dynamic behavior

SWE.3.BP4: Evaluate software detailed design

SWE.3.BP5: Establish bidirectional traceability

SWE.3.BP6: Ensure consistency

SWE.3.BP7: Communicate agreed software detailed design

SWE.3.BP8: Develop software units

ASPICE 4.0

SWE.3.BP1: Specify the static aspects of the detailed design

SWE.2.BP2: Specify the dynamic aspects of the detailed design

SWE.3.BP3: Develop software units

SWE3.BP4: Ensure consistency and establish bidirectional traceability

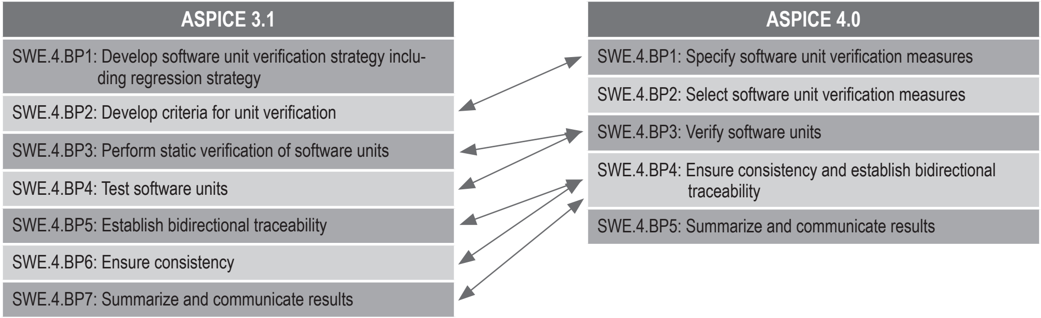
SWE.3.BP5: Communicate agreed software detailed design and software units

Highlights:
Combined Develop software detailed design (BP1) and Define interfaces of software units (BP2)
Removed Evaluate software detailed design (BP4)
Combined traceability (BP5) and consistency (BP6)
Emphasised coding principles from ISO 26262 Clause 6 in note of **[BP3]**

SWE.4 Software Unit Verification

The **purpose** of the Software Unit Verification Process is to verify software units to provide evidence for compliance of the software units with the software detailed design and with the non-functional software requirements.

The **purpose** is to verify that software units are consistent with the software detailed design.



ASPICE 3.1
SWE.4.BP1: Develop software unit verification strategy including regression strategy
SWE.4.BP2: Develop criteria for unit verification
SWE.4.BP3: Perform static verification of software units
SWE.4.BP4: Test software units
SWE.4.BP5: Establish bidirectional traceability
SWE.4.BP6: Ensure consistency
SWE.4.BP7: Summarize and communicate results

ASPICE 4.0
SWE.4.BP1: Specify software unit verification measures
SWE.4.BP2: Select software unit verification measures
SWE.4.BP3: Verify software units
SWE.4.BP4: Ensure consistency and establish bidirectional traceability
SWE.4.BP5: Summarize and communicate results

Highlights:
Renamed “Test” To “Verification”
Removed Develop software unit verification strategy (BP1) but expected in GP 2.1.1
Renamed “criteria” (BP2) to verification measures **[BP1]**

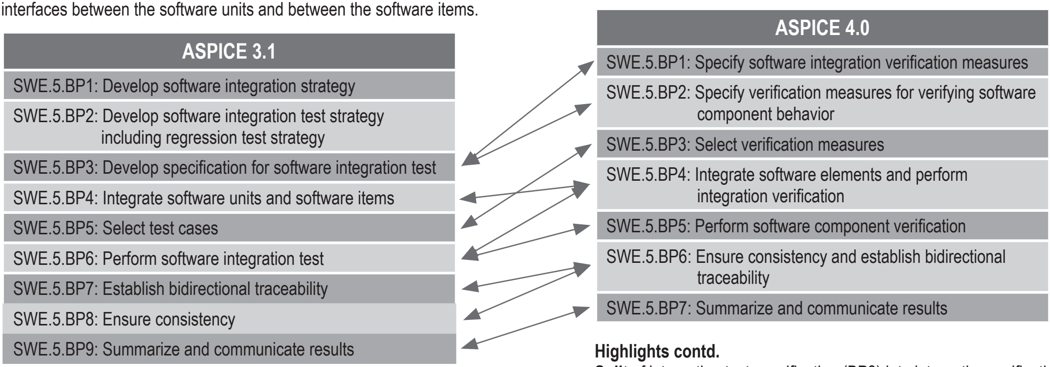
Highlights contd.
Added select of unit verification measures **[BP2]**
Combined static verification (BP3) and unit test (BP4)
Combined traceability (BP5) and consistency (BP6)

SWE.5 Software Integration and Integration Test

(Component Verification and Integration Verification)

The **purpose** of the Software Integration and Integration Test Process is to integrate the software units into larger software items up to a complete integrated software consistent with the software architectural design and to ensure that the software items are tested to provide evidence for compliance of the integrated software items with the software architectural design, including the interfaces between the software units and between the software items.

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.



ASPICE 3.1
SWE.5.BP1: Develop software integration strategy
SWE.5.BP2: Develop software integration test strategy including regression test strategy
SWE.5.BP3: Develop specification for software integration test
SWE.5.BP4: Integrate software units and software items
SWE.5.BP5: Select test cases
SWE.5.BP6: Perform software integration test
SWE.5.BP7: Establish bidirectional traceability
SWE.5.BP8: Ensure consistency
SWE.5.BP9: Summarize and communicate results

ASPICE 4.0
SWE.5.BP1: Specify software integration verification measures
SWE.5.BP2: Specify verification measures for verifying software component behavior
SWE.5.BP3: Select verification measures
SWE.5.BP4: Integrate software elements and perform integration verification
SWE.5.BP5: Perform software component verification
SWE.5.BP6: Ensure consistency and establish bidirectional traceability
SWE.5.BP7: Summarize and communicate results

Highlights:

Renamed process name Software Integration and Integration Test to Component Verification and Integration Verification

Removed Develop software integration strategy (BP1) and software integration test strategy (BP2) but expected in GP 2.1.1

Renamed Select test cases (BP5) to select verification measures **[BP3]**

Highlights contd.

Split of integration test specification (BP3) into integration verification **[BP1]** and component behaviour verification **[BP2]**

Combined Integrate software units and software items (BP4) and Perform software integration test (BP6)

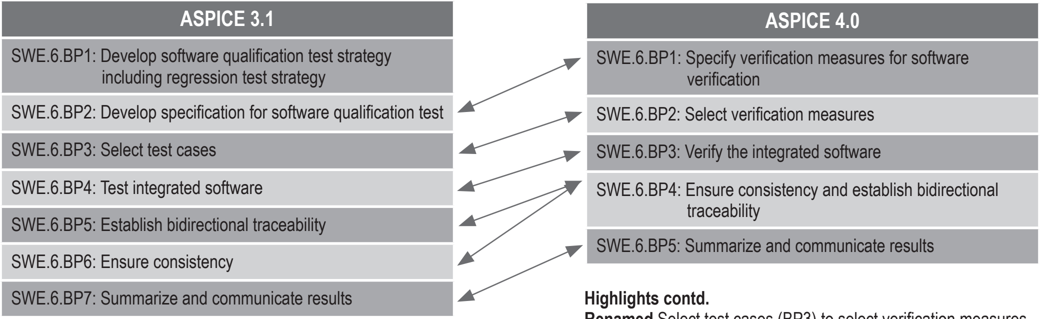
Split of Perform software integration test (BP6) into integration verification **[BP4]** and component verification **[BP5]**

Combined traceability (BP7) and consistency (BP8)

SWE.6 Software Qualification Test (Software Verification)

The **purpose** of the Software Qualification Test Process is to ensure that the integrated software is tested to provide evidence for compliance with the software requirements.

The **purpose** of the Software Verification process is to ensure that the integrated software is verified to be consistent with the software requirements.



ASPICE 3.1
SWE.6.BP1: Develop software qualification test strategy including regression test strategy
SWE.6.BP2: Develop specification for software qualification test
SWE.6.BP3: Select test cases
SWE.6.BP4: Test integrated software
SWE.6.BP5: Establish bidirectional traceability
SWE.6.BP6: Ensure consistency
SWE.6.BP7: Summarize and communicate results

ASPICE 4.0
SWE.6.BP1: Specify verification measures for software verification
SWE.6.BP2: Select verification measures
SWE.6.BP3: Verify the integrated software
SWE.6.BP4: Ensure consistency and establish bidirectional traceability
SWE.6.BP5: Summarize and communicate results

Highlights:
Renamed system qualification test to software verification
Removed Develop software qualification test strategy (BP1) but expected in GP 2.1.1
Renamed test specification (BP2) to verification measures **[BP1]**

Highlights contd.
Renamed Select test cases (BP3) to select verification measures **[BP2]**
Renamed Test integrated software (BP4) to Verify the integrated software **[BP3]**
Combined traceability (BP5) and consistency (BP6)

VAL.1 Validation

Highlights:

New process that focuses on fulfilment of intended use and target operating environment.
Approaches towards homologation / legal type approval requirements (e.g., specific safety and emissions regulations, traction battery approval – UNECE R100, SUMS – UNECE R156).
Test selection includes test track, public road tests, field use by end- users.
Traceability is between stakeholder requirements (SYS.1) and validation measures (VAL.1).

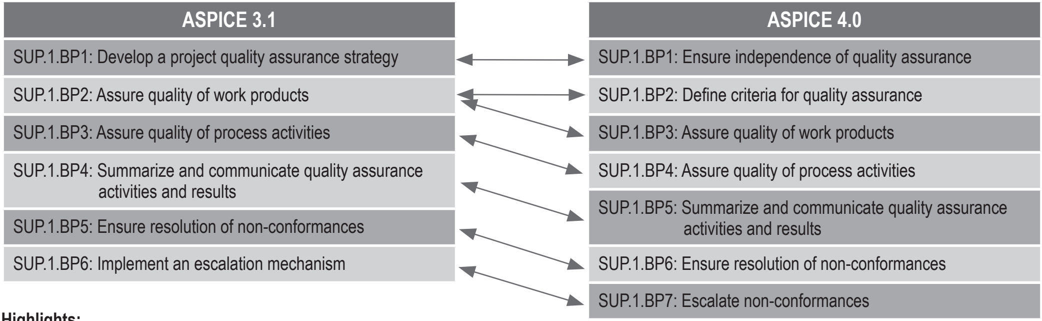
The **purpose** is to provide evidence that the end product, allowing direct end user interaction, satisfies the intended use expectations in its operational target environment.

ASPICE 4.0
VAL.1.BP1: Specify validation measures for product validation
VAL.1.BP2: Select validation measures
VAL.1.BP3: Perform validation and evaluate results
VAL.1.BP4: Ensure consistency and establish bidirectional traceability
VAL.1.BP5: Summarize and communicate results

SUP.1 Quality Assurance

The **purpose** of the Quality Assurance Process is to provide independent and objective assurance that work products and processes comply with predefined provisions and plans and that non-conformances are resolved and further prevented.

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.



ASPICE 3.1

SUP.1.BP1: Develop a project quality assurance strategy

SUP.1.BP2: Assure quality of work products

SUP.1.BP3: Assure quality of process activities

SUP.1.BP4: Summarize and communicate quality assurance activities and results

SUP.1.BP5: Ensure resolution of non-conformances

SUP.1.BP6: Implement an escalation mechanism

ASPICE 4.0

SUP.1.BP1: Ensure independence of quality assurance

SUP.1.BP2: Define criteria for quality assurance

SUP.1.BP3: Assure quality of work products

SUP.1.BP4: Assure quality of process activities

SUP.1.BP5: Summarize and communicate quality assurance activities and results

SUP.1.BP6: Ensure resolution of non-conformances

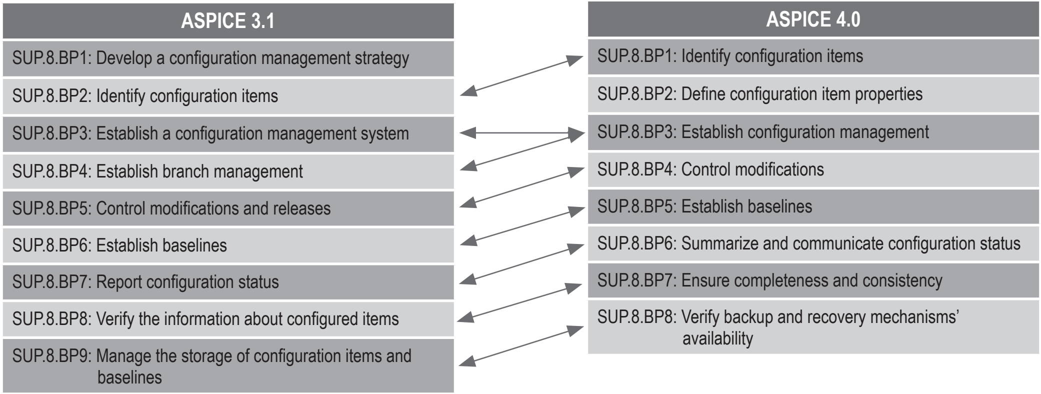
SUP.1.BP7: Escalate non-conformances

Highlights:
Removed Develop a project quality assurance strategy (BP1) but expected in GP 2.1.1 and partly covered by BP1 and BP2
Added Ensure independence of quality assurance **[BP1]**
Added Define criteria for quality assurance **[BP2]**

SUP.8 Configuration Management

The **purpose** of the Configuration Management Process is to establish and maintain the integrity of all work products of a process or project and make them available to affected parties.

The **purpose** of the Configuration Management Process is to establish and maintain the integrity of relevant configuration items and baselines, and make them available to affected parties.



ASPICE 3.1
SUP.8.BP1: Develop a configuration management strategy
SUP.8.BP2: Identify configuration items
SUP.8.BP3: Establish a configuration management system
SUP.8.BP4: Establish branch management
SUP.8.BP5: Control modifications and releases
SUP.8.BP6: Establish baselines
SUP.8.BP7: Report configuration status
SUP.8.BP8: Verify the information about configured items
SUP.8.BP9: Manage the storage of configuration items and baselines

ASPICE 4.0
SUP.8.BP1: Identify configuration items
SUP.8.BP2: Define configuration item properties
SUP.8.BP3: Establish configuration management
SUP.8.BP4: Control modifications
SUP.8.BP5: Establish baselines
SUP.8.BP6: Summarize and communicate configuration status
SUP.8.BP7: Ensure completeness and consistency
SUP.8.BP8: Verify backup and recovery mechanisms' availability

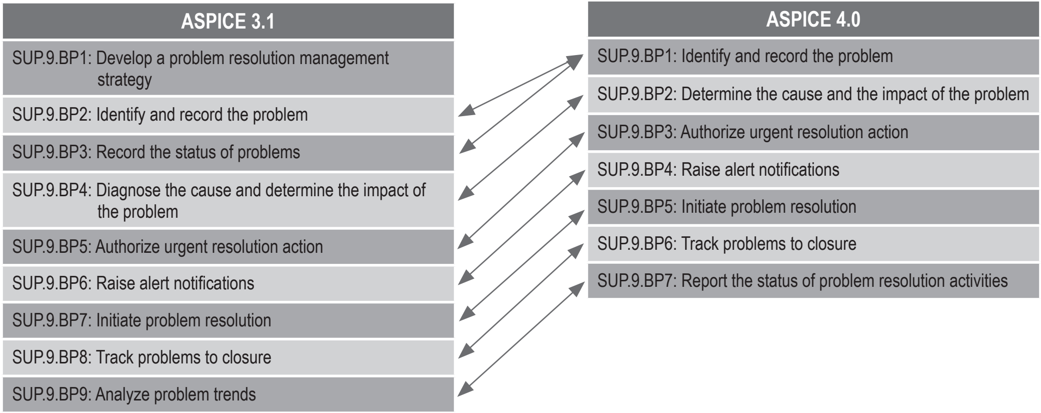
Highlights:
Removed Develop a configuration management strategy (BP1) but expected in GP 2.1.1
Added Define configuration item properties and **Emphasis** on defining selection criteria for configuration items **[BP1]**
Combined Establish a configuration management system (BP3) and Establish branch management (BP4)

Highlights contd.
Renamed Control modifications and releases (BP5), Report configuration status (BP7), Verify the information about configured items (BP8)
Renamed Manage the storage of configuration items and baselines (BP9)
Removed archiving (long-term storage) expectations

SUP.9 Problem Resolution Management

The **purpose** of the Problem Resolution Management Process is to ensure that problems are identified, analyzed, managed and controlled to resolution.

The **purpose** of the Problem Resolution Management Process is to ensure that problems are identified, recorded, analyzed, and their resolution is managed and controlled.



ASPICE 3.1

SUP.9.BP1: Develop a problem resolution management strategy

SUP.9.BP2: Identify and record the problem

SUP.9.BP3: Record the status of problems

SUP.9.BP4: Diagnose the cause and determine the impact of the problem

SUP.9.BP5: Authorize urgent resolution action

SUP.9.BP6: Raise alert notifications

SUP.9.BP7: Initiate problem resolution

SUP.9.BP8: Track problems to closure

SUP.9.BP9: Analyze problem trends

ASPICE 4.0

SUP.9.BP1: Identify and record the problem

SUP.9.BP2: Determine the cause and the impact of the problem

SUP.9.BP3: Authorize urgent resolution action

SUP.9.BP4: Raise alert notifications

SUP.9.BP5: Initiate problem resolution

SUP.9.BP6: Track problems to closure

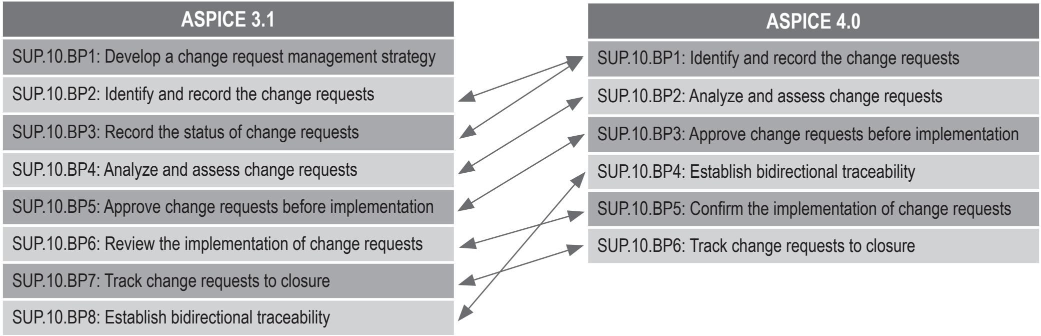
SUP.9.BP7: Report the status of problem resolution activities

Highlights:
Removed of problem resolution management strategy (BP1) but expected in GP 2.1.1
Combined Identify and record the problem (BP2) and Record the status of problems (BP3)
Renamed Diagnose the cause and determine the impact of the problem (BP4)
Renamed Analyze problem trends (BP9) and
Emphasis on reporting the status of problem resolution **[BP7]**

SUP.10 Change Request Management

The **purpose** of the Change Request Management Process is to ensure that change requests are managed, tracked and implemented.

The **purpose** of the Change Request Management Process is to ensure that change requests are recorded, analyzed, tracked, approved, and implemented.



ASPICE 3.1
SUP.10.BP1: Develop a change request management strategy
SUP.10.BP2: Identify and record the change requests
SUP.10.BP3: Record the status of change requests
SUP.10.BP4: Analyze and assess change requests
SUP.10.BP5: Approve change requests before implementation
SUP.10.BP6: Review the implementation of change requests
SUP.10.BP7: Track change requests to closure
SUP.10.BP8: Establish bidirectional traceability

ASPICE 4.0
SUP.10.BP1: Identify and record the change requests
SUP.10.BP2: Analyze and assess change requests
SUP.10.BP3: Approve change requests before implementation
SUP.10.BP4: Establish bidirectional traceability
SUP.10.BP5: Confirm the implementation of change requests
SUP.10.BP6: Track change requests to closure

Highlights:
Removal of change request management strategy but expected in GP 2.1.1
Combined Identify and record the change requests (BP2) and Record the status of change requests (BP3)
Renamed Review the implementation of change requests (BP6)

MAN.3 Project Management

The **purpose** of the Project Management Process is to identify, establish, and control the activities and resources necessary for a project to produce a product, in the context of the project's requirements and constraints.

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.

ASPICE 3.1	ASPICE 4.0
MAN.3.BP1: Define the scope of work	MAN.3.BP1: Define the scope of work
MAN.3.BP2: Define project life cycle	MAN.3.BP2: Define project life cycle
MAN.3.BP3: Evaluate feasibility of the project	MAN.3.BP3: Evaluate feasibility of the project
MAN.3.BP4: Define, monitor and adjust project activities	MAN.3.BP4: Define and monitor work packages
MAN.3.BP5: Define, monitor and adjust project estimates and resources	MAN.3.BP5: Define and monitor project estimates and resources
MAN.3.BP6: Ensure required skills, knowledge, and experience	MAN.3.BP6: Define and monitor required skills, knowledge, and experience
MAN.3.BP7: Identify, monitor and adjust project interfaces and agreed commitments	MAN.3.BP7: Define and monitor project interfaces and agreed commitments
MAN.3.BP8: Define, monitor and adjust project schedule	MAN.3.BP8: Define and monitor project schedule
MAN.3.BP9: Ensure consistency	MAN.3.BP9: Ensure consistency
MAN.3.BP10: Review and report progress of the project	MAN.3.BP10: Review and report progress of the project

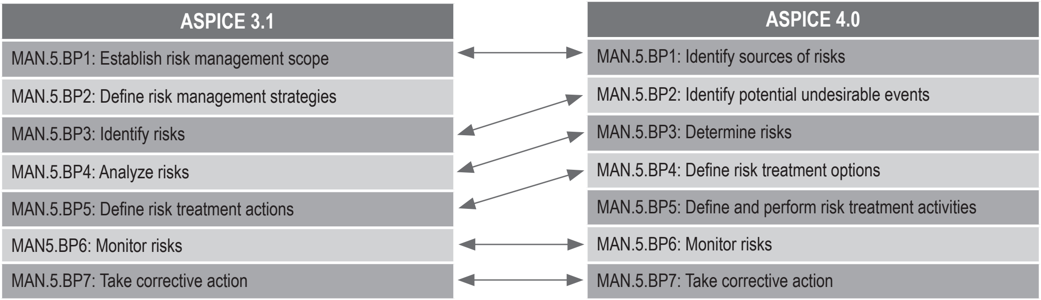
Highlights:
Renamed project activities (BP4) to work packages **[BP4]**
Renamed Define, monitor and adjust to Define and monitor **[BP4]**, **[BP5]**, **[BP7]** and **[BP8]**

Highlights contd.
Renamed Ensure required skills, knowledge, and experience (BP6)
Added escalation mechanism in **[BP7]**

MAN.5 Risk Management

The **purpose** of the Risk Management Process is to identify, analyze, treat and monitor the risks continuously.

The **purpose** is to Regularly identify, analyze, treat and monitor process related risks and product related risks.



ASPICE 3.1

MAN.5.BP1: Establish risk management scope

MAN.5.BP2: Define risk management strategies

MAN.5.BP3: Identify risks

MAN.5.BP4: Analyze risks

MAN.5.BP5: Define risk treatment actions

MAN5.BP6: Monitor risks

MAN.5.BP7: Take corrective action

ASPICE 4.0

MAN.5.BP1: Identify sources of risks

MAN.5.BP2: Identify potential undesirable events

MAN.5.BP3: Determine risks

MAN.5.BP4: Define risk treatment options

MAN.5.BP5: Define and perform risk treatment activities

MAN.5.BP6: Monitor risks

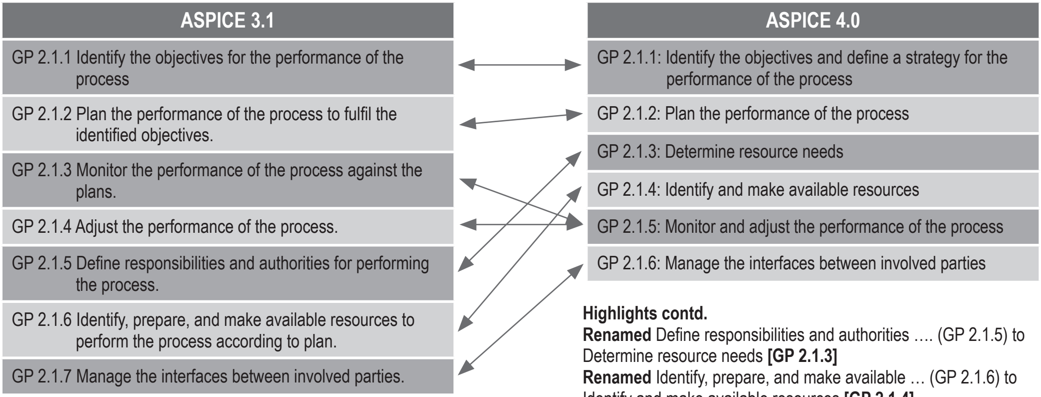
MAN.5.BP7: Take corrective action

Highlights:
Removed risk management strategies (BP2)
Renamed Identify risks (BP3) to Identify potential undesirable events **[BP2]**
Renamed Analyze risks (BP4) to Determine risks **[BP3]**
Added Define and perform risk treatment activities **[BP5]**

CL2 – PA 2.1 – Performance management (Process performance management in v4.0)

Scope: The performance management process attribute is a measure of the extent to which the performance of the process is managed

Scope: The performance management process attribute is a measure of the extent to which the performance of the process is managed.



ASPICE 3.1
GP 2.1.1 Identify the objectives for the performance of the process
GP 2.1.2 Plan the performance of the process to fulfil the identified objectives.
GP 2.1.3 Monitor the performance of the process against the plans.
GP 2.1.4 Adjust the performance of the process.
GP 2.1.5 Define responsibilities and authorities for performing the process.
GP 2.1.6 Identify, prepare, and make available resources to perform the process according to plan.
GP 2.1.7 Manage the interfaces between involved parties.

ASPICE 4.0
GP 2.1.1: Identify the objectives and define a strategy for the performance of the process
GP 2.1.2: Plan the performance of the process
GP 2.1.3: Determine resource needs
GP 2.1.4: Identify and make available resources
GP 2.1.5: Monitor and adjust the performance of the process
GP 2.1.6: Manage the interfaces between involved parties

Highlights:
Added “strategy definition” – removed from Level 1 BPs e.g., Test strategies, SUP.x strategies **[GP 2.1.1]**
Merged monitor (GP 2.1.3) and adjust (GP 2.1.4) GPs into **[GP 2.1.5]**

Highlights contd.
Renamed Define responsibilities and authorities …. (GP 2.1.5) to Determine resource needs **[GP 2.1.3]**
Renamed Identify, prepare, and make available … (GP 2.1.6) to Identify and make available resources **[GP 2.1.4]**

CL2 – PA 2.2 – Work product management

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.

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.

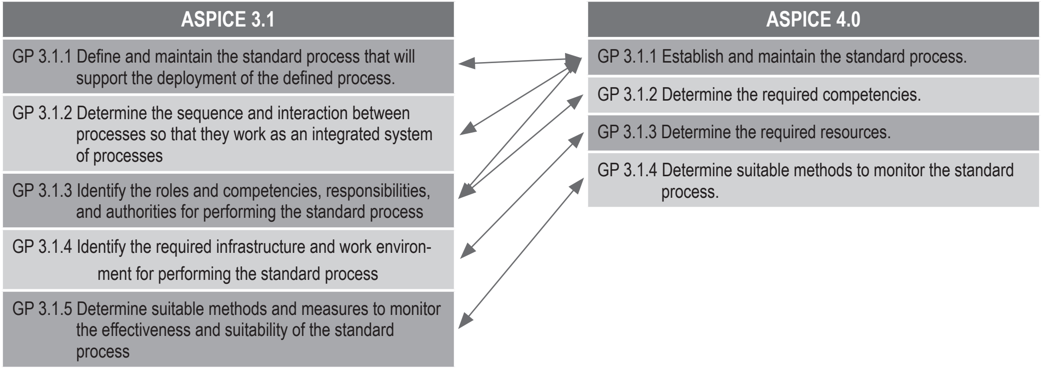
ASPICE 3.1		ASPICE 4.0
GP 2.2.1 Define the requirements for the work products	↔	GP 2.2.1 Define the requirements for the work products
GP 2.2.2 Define the requirements for documentation and control of the work products	↔	GP 2.2.2 Define the requirements for storage and control of the work products
GP 2.2.3 Identify, document and control the work products	↔	GP 2.2.3 Identify, store and control the work products
GP 2.2.4 Review and adjust work products to meet the defined requirements	↔	GP 2.2.4 Review and adjust work products

Highlights:
Renamed Identify, document and control the work products (GP 2.2.3) to Identify, store and control the work products **[GP 2.2.3]**
Renamed Review and adjust work products to meet the defined requirements (GP 2.2.4) to Review and adjust work products **[GP 2.2.4]**

CL3 – PA 3.1 – Process definition

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.

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.



ASPICE 3.1
GP 3.1.1 Define and maintain the standard process that will support the deployment of the defined process.
GP 3.1.2 Determine the sequence and interaction between processes so that they work as an integrated system of processes
GP 3.1.3 Identify the roles and competencies, responsibilities, and authorities for performing the standard process
GP 3.1.4 Identify the required infrastructure and work environment for performing the standard process
GP 3.1.5 Determine suitable methods and measures to monitor the effectiveness and suitability of the standard process

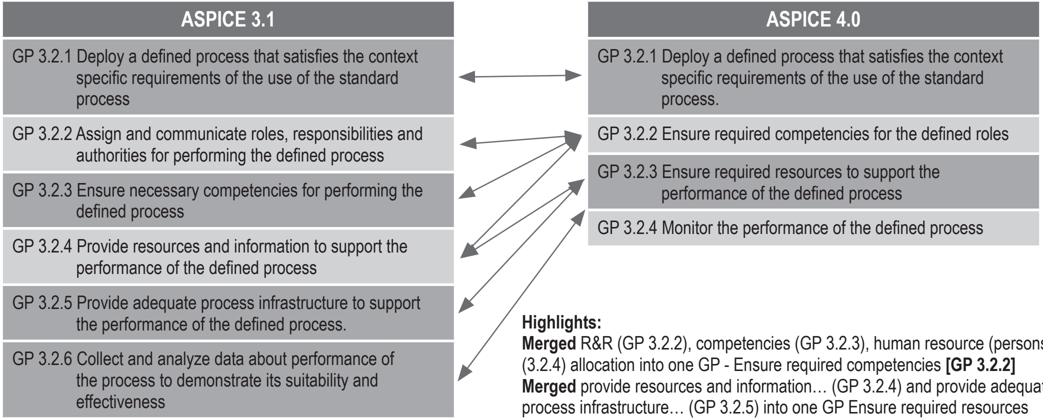
ASPICE 4.0
GP 3.1.1 Establish and maintain the standard process.
GP 3.1.2 Determine the required competencies.
GP 3.1.3 Determine the required resources.
GP 3.1.4 Determine suitable methods to monitor the standard process.

Highlights:
Merged standard process (GP 3.1.1) and sequence and interaction (GP 3.1.2) to **[GP 3.1.1]**
Split of R&R and competencies (GP 3.1.3) **into [GP 3.1.1]** and **[GP 3.1.2]**
Renamed identify the required infrastructure… (GP 3.1.4) and Determine suitable methods… (GP 3.1.5)

CL3 – PA 3.2 – Process deployment

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

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.



ASPICE 3.1
GP 3.2.1 Deploy a defined process that satisfies the context specific requirements of the use of the standard process
GP 3.2.2 Assign and communicate roles, responsibilities and authorities for performing the defined process
GP 3.2.3 Ensure necessary competencies for performing the defined process
GP 3.2.4 Provide resources and information to support the performance of the defined process
GP 3.2.5 Provide adequate process infrastructure to support the performance of the defined process.
GP 3.2.6 Collect and analyze data about performance of the process to demonstrate its suitability and effectiveness

ASPICE 4.0
GP 3.2.1 Deploy a defined process that satisfies the context specific requirements of the use of the standard process.
GP 3.2.2 Ensure required competencies for the defined roles
GP 3.2.3 Ensure required resources to support the performance of the defined process
GP 3.2.4 Monitor the performance of the defined process

Highlights:
Merged R&R (GP 3.2.2), competencies (GP 3.2.3), human resource (persons) (3.2.4) allocation into one GP - Ensure required competencies **[GP 3.2.2]**
Merged provide resources and information… (GP 3.2.4) and provide adequate process infrastructure… (GP 3.2.5) into one GP Ensure required resources **[GP 3.2.3]**
Renamed all GPs in ASPICE PAM 3.1 for simplification

AUTOMOTIVE SPICE® 4.0 with RATING CONSISTANCY DIAGRAMS of GUIDELINE 2.0

ACQ.4 Supplier Monitoring

The Purpose track and assess the performance of an external contract-based supplier company against agreed commitments.

Process outcomes

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.

ACQ.4 with 5 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.

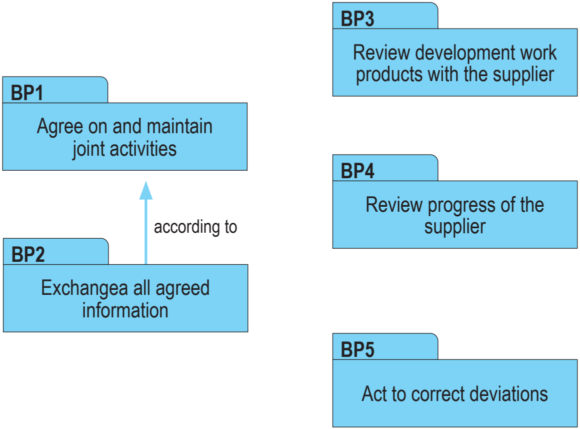
Note 1: see SUP.9 for management of problems

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.

Note 2: see MAN.5 for management of risks

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
13-52 Communication evidence	x	x	x	
13-09 Meeting support evidence	x	x		
13-14 Progress status		x	x	
13-16 Change request				x
13-19 Review evidence		x		
14-02 Corrective action				x
15-51 Analysis results			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


BP1
Agree on and maintain joint activities
according to
BP2
Exchangea all agreed information
BP3
Review development work products with the supplier
BP4
Review progress of the supplier
BP5
Act to correct deviations

SPL.2 Product Release

The purpose is to control the release of a product to the intended customer.

Process outcomes

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.

SPL.2 with 8 Base practices

SPL.2.BP1: Define the functional content of releases. Define the functionality to be included and the release criteria for each release.

Note 1: This may include the hardware elements, software elements, and extra application parameter files (influencing the identified system functionality) that are needed for the release.

SPL.2.BP2: Define release package. Define the release as well as supporting tools and information.

Note 2: The release package may include also programming tools.

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.

Note 3: Unique identification may be realized by a classification and numbering scheme for product releases.

SPL.2.BP4: Build the release from items under configuration control. Build the release from items under configuration control to ensure integrity.

Note 4: This practice may be supported by the SUP.8 Configuration Management Process.

SPL.2.BP5: Ensure release approval before delivery. Criteria for the release are satisfied before delivery takes place.

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 Validation.

SPL2.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 with 8 Base practices

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

SYS.1 Requirements Elicitation

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

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.

SYS.1 with 4 Base practices

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.

Note 1: Documenting the stakeholder, or the source of a stakeholder requirement, supports stakeholder requirements agreement and change analysis (see BP2 and BP3).

SYS.1.BP2: Agree on requirements. Formalize the stakeholder's expectations and requests into requirements. Reach a common understanding of the set of stakeholder requirements among affected parties by obtaining an explicit agreement from all affected parties.

Note 2: Examples of affected parties are customers, suppliers, design partners, joint venture partners, or outsourcing parties.

Note 3: The agreed stakeholder requirements may be based on feasibility studies and/or cost and schedule impact analysis.

SYS.1.BP3: Analyze stakeholder requirements changes. Analyze 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.BP4: 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: Agree on requirements		**x**		
BP3: Analyze stakeholder requirements changes			**x**	
BP4: Communicate requirements status	**x**			**x**

SYS.2 System Requirements Analysis

The purpose is to establish a structured and analyzed set of system requirements consistent with the stakeholder requirements.

Process outcomes

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.
5. Consistency and bidirectional traceability are established between system requirements and stakeholder requirements.
6. The system requirements are agreed and communicated to all affected parties.

SYS.2 with 6 Base practices

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.

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.

Note 2: Examples for defined characteristics of requirements shared by technical standards are verifiability (i.e., verification criteria being inherent in the requirements text), unambiguity/comprehensibility, freedom from design and implementation, and not contradicting any other requirement).

SYS.2.BP2: Structure system requirements. Structure and prioritize the system requirements.

Note 3: Examples for structuring criteria can be grouping (e.g., by functionality) or product variants identification.

Note 4: Prioritization can be done according to project or stakeholder needs via e.g., definition of release scopes. Please refer to SPL.2.BP1.

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.

Note 5: See MAN.3.BP3 for project feasibility and MAN.3.BP5 for project estimates.

Note 6: Technical feasibility can be evaluated based on e.g., platform or product line, or by means of prototype development or product demonstrators.

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.

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. Traceability alone, e.g., the existence of links, does not necessarily mean that the information is consistent with each other.

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

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

SYS.3 System Architectural Design

The purpose is to establish an analyzed system architecture, comprising static and dynamic aspects, consistent with the system requirements.

Process outcomes

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.

SYS.3 with 5 Base practices

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 non-functional system requirements, including external interfaces and a defined set of system elements with their interfaces and relationships.

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 non-functional system requirements including the behavior of the system elements and their interaction in different system modes.

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.

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 non-software system elements. Document a rationale for the system architectural design decisions.

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. Traceability alone, e.g., the existence of links, does not necessarily mean that the information is consistent with each other.

Note 8: There may be non-functional requirements that the system architectural design does not trace to. Examples are do not address, or represent, direct properties or characteristics of the physical end product. 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 with 5 Base practices

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**

SYS.4 System Integration and Integration Verification

The purpose is to integrate systems elements and verify that the integrated system elements are consistent with the system architecture.

Process outcomes

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, including 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.

SYS.4 with 5 Base practices

SYS.4.BP1: Specify verification measures for system integration. Specify the verification measures, based on a defined sequence 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, and
- the required verification infrastructure and environment setup.

Note 1: Examples on what a verification measure may focus are the timing dependencies of the correct signal flow between interfacing system elements, or interactions between hardware and software, as specified in the system architecture. 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, and/or*
- *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 sequence 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. Traceability alone, e.g., the existence of links, does not necessarily mean that the information is consistent with each other.

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 with 5 Base practices

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**					
03-50 Verification Measure Data				**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**

SYS.5 System Verification

The purpose is to ensure that the system is verified to be consistent with the system requirements.

Process outcomes

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, including 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.

SYS.5 with 5 Base practices

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

- 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, and
- the required verification infrastructure and environment setup.

Note 1: The system verification measures may cover aspects such as thermal, environmental, robustness/lifetime, and EMC.

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.

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.)*

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. Traceability alone, e.g., the existence of links, does not necessarily mean that the information is consistent with each other.

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 Items						
08-60 Verification Measure	**x**					
03-50 Verification Measure Data			**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**

SWE.1 Software Requirements Analysis

The purpose is to establish a structured and analyzed set of software requirements consistent with the system requirements, and the system architecture.

Process outcomes

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.

SWE.1 with 6 Base practices

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.

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.*

Note 2: *Examples for defined characteristics of requirements shared by technical standards are verifiability (i.e., verification criteria being inherent in the requirements text), unambiguity/comprehensibility, freedom from design and implementation, and not contradicting any other requirement).*

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.

Note 4: The hardware-software-interface (HSI) definition puts in context hardware and therefore it is an interface decision at the system design level. If such a HSI exists, then it may provide input to software requirements.

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 expressing product variants.

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. Traceability alone, e.g., the existence of links, does not necessarily mean that the information is consistent with each other.

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

SWE.2 Software Architectural Design

The purpose is to establish an analyzed software architecture, comprising static and dynamic aspects, consistent with the software requirements.

Process outcomes

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.

SWE.2 with 5 Base practices

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 software components with their interfaces and relationships.

Note 1: The hardware-software-interface (HSI) definition puts in context the hardware design and therefore is an aspect of system design (SYS.3).

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 software components and their interaction in different software modes, and concurrency aspects.

Note 2: Examples for concurrency aspects are application-relevant interrupt handling, preemptive processing, multi-threading.

Note 3: Examples for behavioral descriptions are natural language or semi-formal notation (e.g, SysML, UML).

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 EE-PROM 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. Traceability alone, e.g, the existence of links, does not necessarily mean that the information is consistent with each other.

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**		
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**

SWE.3 Software Detailed Design and Unit Construction

The purpose is to establish a software detailed design, comprising static and dynamic aspects, consistent with the software architecture, and to construct software units consistent with the software detailed design.

Process outcomes

1. A detailed design is specified including static and dynamic aspects.
2. Software units as specified in 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 source code and software detailed design; and consistency and bidirectional traceability are established between the software detailed design and the software requirements.
4. The source code and the agreed software detailed design are communicated to all affected parties.

SWE.3 with 5 Base practices

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

- valid data value ranges for inputs and outputs (from the application domain perspective), and
- physical or measurement units applicable to inputs and outputs (from the application domain perspective).

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.

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.

Note 3: Examples of a measurement unit are '%' or '‰'.

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.

Note 5: The hardware-software-interface (HSI) definition puts in context the hardware design and therefore is an aspect of system design (SYS.3).

SWE.3.BP2: Specify 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.*

SWE.3.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 in the detailed design to a software requirement directly are communication matrices or basis software aspects such as a 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. Traceability alone, e.g., the existence of links, does not necessarily mean that the information is consistent with each other.

SWE.3.BP5: Communicate agreed software detailed design and developed software units. Communicate the agreed software detailed design and developed 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 developed software units				x

SWE.4 Software Unit Verification

The purpose is to verify that software units are consistent with the software detailed design.

Process outcomes

1. Verification measures for software unit verification are specified.
2. Software unit verification measures are selected according to the release scope, including 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 bidirectional traceability is established between verification results and verification measures.
5. Results of the software unit verification are summarized and communicated to all affected parties.

SWE.4 with 5 Base practices

SWE.4.BP1: Specify software unit verification measures. Specify verification measures for each software unit defined in the software detailed design, including

- pass/fail criteria for verification measures,
- entry and exit criteria for verification measures, and
- the required verification infrastructure.

Note 1: Examples for unit verification measures are static analysis, code reviews, and unit testing.

Note 2: Static analysis can be done based on MISRA rulesets and other coding standards.

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.

SWE.4.BP3: Verify software units. Perform software unit verification 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 of verification results that deviate from expected results.

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 4: Bidirectional traceability supports consistency, and facilitates impact analysis of change requests, and demonstration of verification coverage. Traceability alone, e.g., the existence of links, does not necessarily mean that the information is consistent with each other.

SWE.4.BP5: Summarize and communicate results. Summarize the results of software unit verification 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.4 Software Unit Verification	Outcome 1	Outcome 2	Outcome 3	Outcome 4	Outcome 5
Output Information Items					
08-60 Verification Measure	x				
03-50 Verification Measure Data			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

SWE.5 Software Component Verification and Integration Verification

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, including 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 bidirectional traceability is 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

SWE.5.BP1: Specify software integration verification measures. Specify verification measures, based on a defined sequence 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, and
- 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, and
- 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 sequence 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 stepwise 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 with 7 Base practices

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. Establish bidirectional traceability between verification results and verification measures.

Note 9: Bidirectional traceability supports consistency, and facilitates impact analysis of change requests, and demonstration of verification coverage. Traceability alone, e.g., the existence of links, does not necessarily mean that the information is consistent with each other.

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					
03-50 Verification Measure Data					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

SWE.6 Software Verification

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

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, including 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 bidirectional traceability is established between verification results and verification measures.
5. Results of the software verification are summarized and communicated to all affected parties.

SWE.6 with 5 Base practices

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

- 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, and
- the required verification infrastructure and environment setup.

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".

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 verification of the integrated software 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.

SWE.6.BP4: Ensure consistency and establish bidirectional traceability. Ensure consistency and establish bidirectional traceability between verification measures and software 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. Traceability alone, e.g., the existence of links, does not necessarily mean that the information is consistent with each other.

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				
03-50 Verification Measure Data			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

VAL.1 Validation

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

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.

VAL.1 with 5 Base practices

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

- 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, and
- the required validation infrastructure and environment setup.

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).

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.

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.

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 end product 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 verification results that deviate from expected results

VAL.1.BP4: Ensure consistency and establish bidirectional 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 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. Traceability alone, e.g., the existence of links, does not necessarily mean that the information is consistent with each

VAL.1.BP5: Summarize and communicate results. Summarize the validation 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.

VAL.1 Validation	Outcome 1	Outcome 2	Outcome 3	Outcome 4
Output Information Items				
08-59 Validation Measure	**x**			
08-57 Validation Measure Selection Set	**x**			
13-24 Validation Results		**x**		
13-51 Consistency Evidence			**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**	
BP5: Summarize and communicate results				**x**

MLE.1 Machine Learning Requirements Analysis

The purpose is to refine the machine learning-related software requirements into a set of ML requirements.

Process outcomes

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.

MLE.1 with 6 Base practices

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 (e.g., gender, weather conditions, street conditions within the ODD) and their expected distributions.

Note 1: Non-functional requirements may include relevant characteristics of the ODD and KPIs as robustness, performance, and level of trustworthiness.

Note 2: The ML data requirements are input for SUP.11 Machine Learning Data Management but also for other MLE processes.

Note 3: In case of ML development only, stakeholder requirements represent the software requirements.

MLE.1.BP2: Structure ML requirements. Structure and prioritize the ML requirements.

Note 4: Examples for structuring criteria can be grouping (e.g., by functionality) or variants identification.

Note 5: Prioritization can be done according to project or stakeholder needs via e.g., definition of release scopes. Refer to SPL.2.BP1.

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.

Note 6: See MAN.3.BP3 for project feasibility and MAN.3.BP5 for project estimates.

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. Traceability alone, e.g., the existence of links, does not necessarily mean that the information is consistent with each other.

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

MLE.1.BP6: Communicate agreed ML requirements and impact on the operating environment. 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

MLE.2 Machine Learning Architecture

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

1. A 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.

MLE.2 with 7 Base practices

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.

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.

Note 2: The details of the ML model may differ between the ML model used during training and the deployed ML model.

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.

MLE.2.BP3: Analyze ML architectural elements. Define criteria for analysis of the ML architectural elements. Analyze ML architectural elements according to the defined criteria.

Note 3: Trustworthiness and explainability might be criteria for the analysis of the ML architectural elements.

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.

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.

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. Traceability alone, e.g., the existence of links, does not necessarily mean that the information is consistent with each other.

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

MLE.3 Machine Learning Training

The purpose is to optimize the ML model to meet the defined ML requirements.

Process outcomes

1. A 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.

MLE.3 with 5 Base practices

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
- entry and exit criteria of the training and validation,
- approaches for hyperparameter tuning / optimization,
- approach for data set creation and modification, and
- training and validation environment

Note 1: The ML training and validation approach may include random dropout and other robustification methods.

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.

Note 3: The training environment should reflect the environment of the deployed model.

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.

Note 4: The ML training and validation data set may include corner cases, unexpected cases, and normal cases depending on the ML requirements.

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).

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 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. Traceability alone, e.g., the existence of links, does not necessarily mean that the information is consistent with each other.

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

MLE.4 Machine Learning Model Testing

The purpose is to ensure compliance of the trained ML model and the deployed ML model with the ML requirements..

Process outcomes

1. A ML test approach is defined.
2. A 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 bidirectional traceability is established between 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.

MLE.4 with 7 Base practices

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

- ML test scenarios with distribution of data characteristics (e.g., gender, weather conditions, street conditions within the ODD) defined by ML requirements,
- 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, and
- the required testing infrastructure and environment setup.

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.

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.

Note 3: Data characteristic is one property of the data that may have different expressions in the ODD. E.g., weather condition may contain expressions like sunny, foggy or rainy.

Note 4: An ML test scenario is a combination of expressions of all defined data characteristics e.g., weather conditions = sunny, street conditions = gravel road.

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

Note 6: 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 ML test results.

Note 7: 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 8: 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.

MLE4.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 ML test results.

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 bidirectional traceability is established between the ML test approach and ML test results.

Note 9: Bidirectional traceability supports consistency, and facilitates impact analyses of change requests, and verification coverage demonstration. Traceability alone, e.g., the existence of links, does not necessarily mean that the information is consistent with each other.

MLE.4.BP7: Summarize and communicate results. Summarize the ML 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 5
Output Information Items						
08-64 ML test approach	**x**					
03-51 ML data set		**x**				
13-50 ML test results			**x**	**x**		
11-50 Deployed ML model				**x**		
13-51 Consistency evidence					**x**	
13-52 Communication evidence						**x**

MLE.4 Machine Learning Model Testing	Outcome 1	Outcome 2	Outcome 3	Outcome 4	Outcome 5	Outcome 6
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

HWE.1 Hardware Requirements Analysis

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

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.

HWE.1 with 6 Base practices

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.

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.

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 text), unambiguity/comprehensibility, freedom from design and implementation, and not contradicting any other requirement).

Note 3: In case of hardware-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 hardware.

Note 4: The hardware-software-interface (HSI) definition puts in context software and therefore is an interface decision at the system design level. If such a HSI exists, then it may provide input to hardware requirements.

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 analyses 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.

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. Traceability alone, e.g., the existence of links, does not necessarily mean that the information is consistent with each other.

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 affected 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	
17-00 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

HWE.2 Hardware Design

The purpose is to provide an analyzed design, including dynamic aspects, that is consistent with the hardware requirements and suitable for manufacturing, and to derive production-relevant data.

Process outcomes

1. A hardware architecture and hardware detailed design is developed 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 hardware detailed design and communicated to all affected parties.
5. Information for production test is derived from the hardware 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.

HWE.2 with 6 Base practices

HWE.2.BP1: Specify the hardware architecture. Develop the hardware architecture that identifies the hardware components. Document the rationale for the defined hardware architecture.

Note 1: Examples for aspects reflected in the hardware architecture are ground concept, supply concept, EMC concept.

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.

HWE.2.BP2: Specify the hardware detailed design. Based on components identified in the hardware architecture, specify the detailed design description and the schematics for the intended hardware variants, including the interfaces between the hardware elements. Derive the hardware layout, the hardware bill of materials, and the production data.

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.).

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.

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

Note 5: Not all hardware elements have dynamic behavior 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 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. Traceability alone, e.g, the existence of links, does not necessarily mean that the information is consistent with each other.

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: Specify the hardware architecture	x			x	x	
BP2: Specify the hardware detailed 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			
BP6: Communicate agreed hardware architecture and hardware detailed design				x	x	x

HWE.3 Verification against Hardware Design

The purpose is to ensure that the production data compliant hardware is verified to provide evidence for compliance with the hardware design.

Process outcomes

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, including criteria for regression verification.
3. Verification is performed 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 summarized and communicated to all affected parties.

HWE.3 with 6 Base practices

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. This includes

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

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.

Note 2: Measuring points can be used for stepwise testing of hardware elements.

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. Ensure that deviations are documented and that they do not alter verification results.

Note 3: Examples of compliance are sample reports, record of visual inspection, ICT report.

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. Traceability alone, e.g., the existence of links, does not necessarily mean that the information is consistent with each other.

HWE.3.BP6: Summarize and communicate results. Summarize 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					
03-50 Verification Measure Data			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

HWE.4 Verification against Hardware Requirements

The purpose is to ensure that the complete hardware is verified to be consistent with the hardware requirements.

Process outcomes

1. Verification measures are specified for verification of the hardware against the hardware requirements.
2. Verification measures are selected considering criteria, including 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.

HWE.4 with 6 Base practices

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

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

Note 1: The verification measures may cover aspects such as thermal, environmental, robustness/lifetime, and EMC.

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.

Note 2: Examples of compliance are sample reports, record of visual inspection, ICT report.

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. Traceability alone, e.g., the existence of links, does not necessarily mean that the information is consistent with each other.

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					
03-50 Verification Measure Data			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

SUP.1 Quality Assurance

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

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.

SUP.1 with 7 Base practices

SUP.1.BP1: Ensure independence of quality assurance. Ensure that quality assurance is performed independently and objectively without conflicts of interest.

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).

SUP.1.BP2: Define criteria for quality assurance. Define quality criteria for work products as well as for process tasks and their performance.

Note 2: Quality criteria may consider internal and external inputs such as customer requirements, standards, milestones, etc.

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.

Note 3: Quality assurance activities may include reviews, problem analysis and lessons learned that improve the work products for further use.

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).

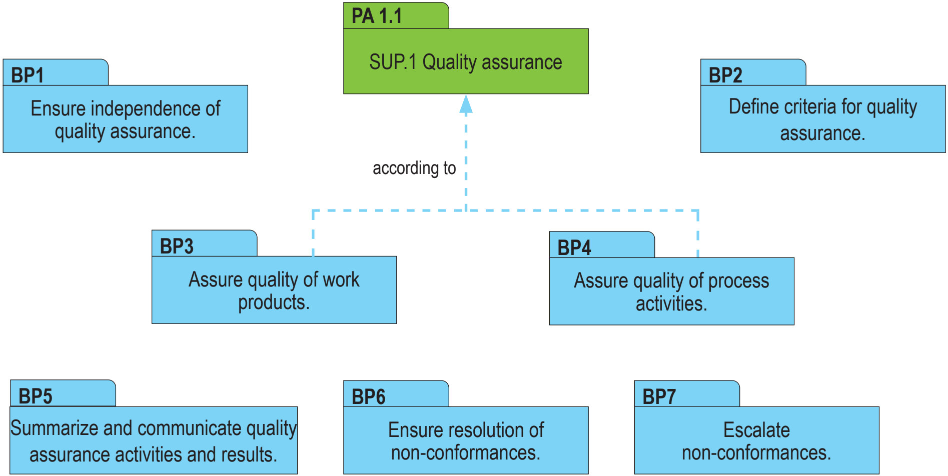
Note 6: Non-conformances detected in the process definition or implementation may be entered into a process improvement process (PIM.3).

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 Organizational structure	x				x	
18-52 Escalation path					x	x
18-07 Quality criteria		x	x	x		
13-52 Communication evidenceX			x	x	x	
13-18 Quality conformance evidence			x	x		
13-19 Review evidence			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 work products.			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

SUP.1 Rating Rules


PA 1.1
SUP.1 Quality assurance
BP1
Ensure independence of quality assurance.
BP2
Define criteria for quality assurance.
according to
BP3
Assure quality of work products.
BP4
Assure quality of process activities.
BP5
Summarize and communicate quality assurance activities and results.
BP6
Ensure resolution of non-conformances.
BP7
Escalate non-conformances.

SUP.8 Configuration Management

The purpose of the Configuration Management Process is to establish and maintain the integrity of relevant configuration items and baselines, and make them available to affected parties

Process outcomes

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.

SUP.8 with 8 Base practices

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.

Note 1: Configuration items are representing work products or group of work products which are subject to configuration management as a single entity.

Note 2: Configuration items may vary in complexity, size, and type, ranging from an entire system including all system, hardware, and software documentation down to a single element or document.

Note 3: The selection criteria may be applied to single work products or a group of work products.

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, Tested, Released, etc.), storage location, access rights, etc.

Note 6: The application of properties may be implemented by attributes of configuration items.

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

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

Note 8: This may include the application of a defined status model for configuration items.

SUP.8.BP5: Establish baselines. Define and establish baselines for internal purposes, and for external product delivery, for all relevant configuration items.

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

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

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

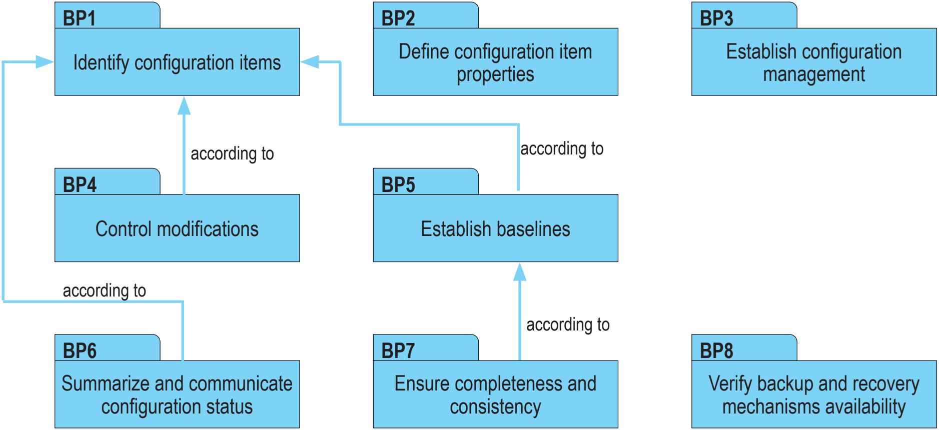
Note 10: Completeness and consistency of a baseline means that all required configuration items are included and consistent, and have the required status. This can be used to support e.g., project gate approval.

SUP.8.BP8: Verify backup and recovery mechanisms availability. Verify the availability of appropriate backup and recovery mechanisms for the configuration management including the controlled configuration items. 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	
16-03 Configuration management system			x	x	x			
13-08 Baseline					x		x	
14-01 Change history			x	x		x		
15-56 Configuration status						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	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

SUP.8 Rating Rules


BP1
Identify configuration items
BP2
Define configuration item properties
BP3
Establish configuration management
according to
according to
BP4
Control modifications
BP5
Establish baselines
according to
according to
BP6
Summarize and communicate configuration status
BP7
Ensure completeness and consistency
BP8
Verify backup and recovery mechanisms availability

SUP.9 Problem Resolution Management

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

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.

SUP.9 with 7 Base practices

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.

Note 1: Problems may relate to e.g., product, resources, or methods.

Note 2: Example values for the problem status are “new”, “solved”, “closed”, etc.

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.

Note 4: Unique identification supports traceability to changes made as needed by the change request management process (SUP.10).

SUP.9.BP2: Determine the cause and the impact of the problem. Analyze the problem, determine its cause, including common causes if existing, and impact. Involve relevant parties. Categorize the problem.

Note 5: Problem categorization (e.g., light, medium, severe) may be based on severity, criticality, urgency, etc.

SUP.9.BP3: Authorize urgent resolution action. Obtain authorization for immediate action if a problem requires an urgent resolution according to the categorization.

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.

SUP.9.BP5: Initiate problem resolution. Initiate appropriate actions according to the categorization to resolve the problem long-term, including review of those actions or initiate a change request. This includes synchronization and consistency with short-term urgent resolution actions, if applicable.

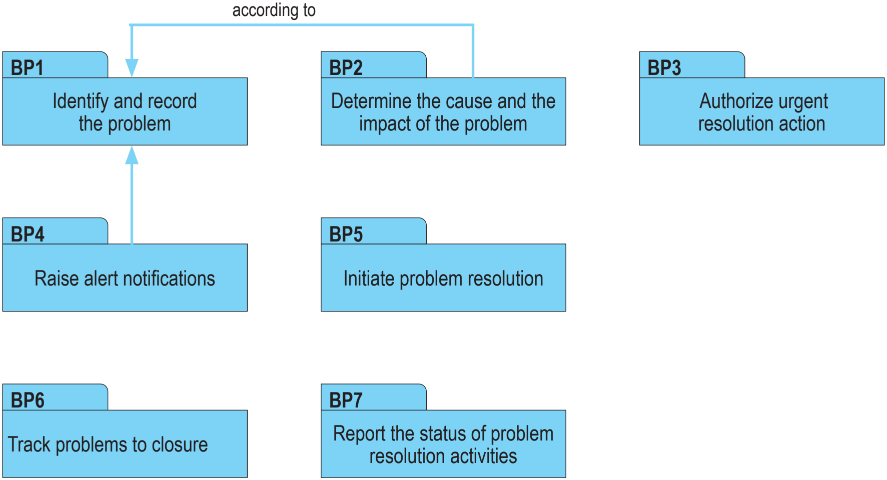
S**UP.9.BP6: Track problems to closure.** Track the status of problems to closure including all related change requests. The closure of problems is accepted by relevant stakeholders.

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 their impacts were, etc.

SUP.9 Problem Resolution Management	Outcome 1	Outcome 2	Outcome 3	Outcome 4	Outcome 5
Output Information Items					
13-07 Problem	x	x	x	x	
15-55 Problem analysis evidence		x			
15-12 Problem status					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

SUP.9 Rating Rules


according to
BP1
Identify and record the problem
BP2
Determine the cause and the impact of the problem
BP3
Authorize urgent resolution action
BP4
Raise alert notifications
BP5
Initiate problem resolution
BP6
Track problems to closure
BP7
Report the status of problem resolution activities

SUP.10 Change Request Management

The purpose of the Change Request Management Process is to ensure that change requests are recorded, analyzed, tracked, approved, and implemented.

Process outcomes

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.

SUP.10 with 6 Base practices

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.

Note 1: Change requests may be used for changes related to e.g., product, process, methods.

Note 2: Example values for the change request status are "open", "under investigation", "implemented", etc.

Note 3: The change request handling may differ across the product life cycle e.g., during prototype

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.

Note 4: Examples for analysis criteria are: resource requirements, scheduling issues, risks, benefits, etc.

SUP.10.BP3: Approve change requests before implementation. Change requests are prioritized and approved for implementation based on analysis results and availability of resources.

Note 5: A Change Control Board (CCB) is an example mechanism used to approve change requests.

Note 6: Prioritization of change requests may be done by allocation to releases.

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.

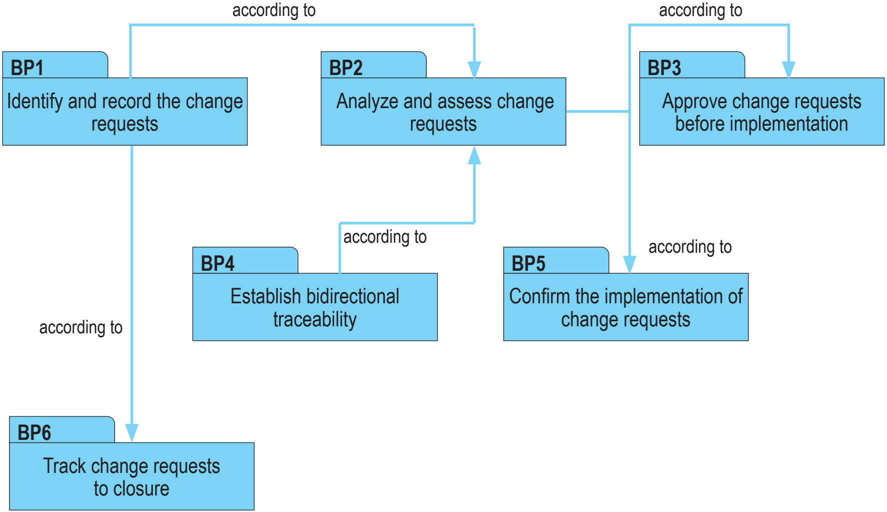
SUP.10.BP5: Confirm the implementation of change requests. The implementation of change requests is confirmed before closure by relevant stakeholders.

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 Items						
18-57 Change analysis criteria		x				
13-16 Change request	x	x	x		x	x
13-51 Consistency evidence				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

SUP.10 Rating Rules


according to
according to
BP1
Identify and record the change requests
BP2
Analyze and assess change requests
BP3
Approve change requests before implementation
according to
BP4
Establish bidirectional traceability
according to
BP5
Confirm the implementation of change requests
according to
BP6
Track change requests to closure

SUP.11 Machine Learning Data Management

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. A ML data management system including an ML data lifecycle is established.
2. A 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.

SUP.11 with 6 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, and
- 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 systemX	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

MAN.3 Project Management

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

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.

MAN.3 with 10 Base practices

MAN.3.BP1: Define the scope of work. Identify the project's goals, motivation and boundaries.

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.

Note 1: This may include the alignment of the project life cycle with the customer's development process.

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.

Note 2: The evaluation of feasibility may consider technical constraints of the project.

MAN.3.BP4: Define and monitor work packages. Define and monitor work packages and their dependencies according to defined project life cycle and estimations.

Note 3: The structure and the size of the work packages support an adequate progress monitoring.

Note 4: Work packages may be organized in a work breakdown structure.

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, product samples, or infrastructure

Note 6: Project risks (using MAN.5) 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.

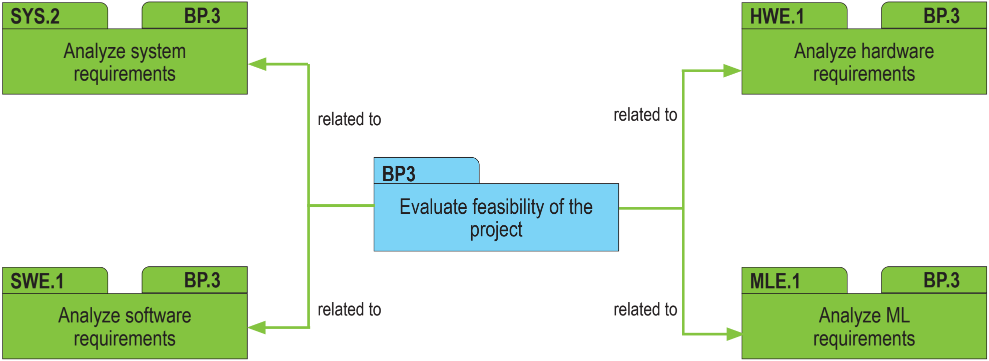
Note 10: This may include the consideration of critical dependencies, that are an input for risk management.

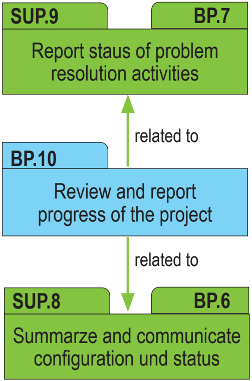
MAN.3.BP10: Review and report progress of the project. Regularly review and report the status of the project and the fulfillment of work packages against estimated effort and duration to all affected parties. Prevent recurrence of identified problems.

Note 11: Project reviews may be executed at regular intervals by the management. Project reviews 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 Items							
08-53 Scope of work	x						
08-54 Feasibility analysis		x		x			
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				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


SYS.2
BP.3
Analyze system requirements
HWE.1
BP.3
Analyze hardware requirements
related to
related to
BP3
Evaluate feasibility of the project
SWE.1
BP.3
Analyze software requirements
related to
related to
MLE.1
BP.3
Analyze ML requirements



SUP.9 BP.7

Report staus of problem resolution activities

related to

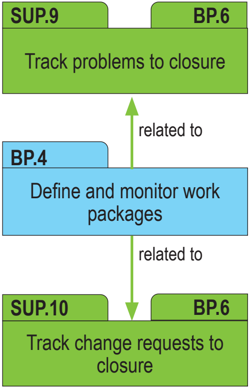
BP.10

Review and report progress of the project

related to

SUP.8 BP.6

Summarze and communicate configuration und status



SUP.9 BP.6

Track problems to closure

related to

BP.4

Define and monitor work packages

related to

SUP.10 BP.6

Track change requests to closure

MAN.5 Risk Management

The purpose is to Regularly identify, analyze, treat and monitor process related risks and product related risks.

Process outcomes

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.

MAN.5 with 7 Base practices

MAN.5.BP1: Identify sources of risks. Identify and regularly update the sources of risks with affected parties.

Note 1: Risks may include technical, economical, and schedule risks.

Note 2: Risks may include the suppliers' deliverables and services.

Note 3: The risk sources may vary across the entire project life cycle.

MAN.5.BP2: Identify potential undesirable events. Identify potential undesirable events within the scope of the risk management for the project.

MAN.5.BP3: Determine risks. Determine the probability and severity of the undesirable events to support priorities for the mitigation of the risks.

Note 4: Different methods may be used to analyze technical risks of a system, for example, functional analysis, simulation, FMEA, FTA etc.

MAN.5.BP4: Define risk treatment options. For each risk select a treatment option to accept, mitigate, avoid, or share (transfer) the risk.

MAN.5.BP5: Define and perform risk treatment activities. Define and perform risk activities for risk treatment options.

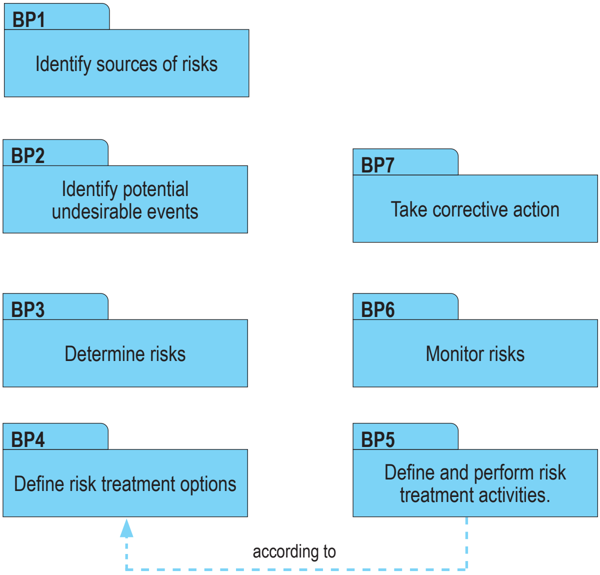
MAN.5.BP6: Monitor risks. Regularly re-evaluate the risk related to the identified potential undesirable events to determine changes in the status of a risk and to evaluate the progress of the risk treatment activities.

Note 5: 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 6: 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 Items					
15-51 Analysis results	x	x	x		x
15-09 Risk status	x		x	x	x
08-55 Risk measure				x	x
14-02 Corrective action				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



BP1

Identify sources of risks

BP2

Identify potential undesirable events

BP7

Take corrective action

BP3

Determine risks

BP6

Monitor risks

BP4

Define risk treatment options

BP5

Define and perform risk treatment activities.

MAN.6 Measurement

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

1. The measurement information needs that are necessary to evaluate the achievement of process objectives and the achievement of desired work products 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.

MAN.6 with 6 Base practices

MAN.6.BP1: Identify information needs. Identify the measurement information needs that are necessary to evaluate the achievement of process objectives and work products.

Note 1: Information needs may change over time. Therefore, the measurement process may be used in an iterative way.

MAN.6.BP2: Specify metrics. Identify and develop an appropriate set of metrics based on measurement information needs.

Note 2: Metrics may be related to processes or development results.

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.

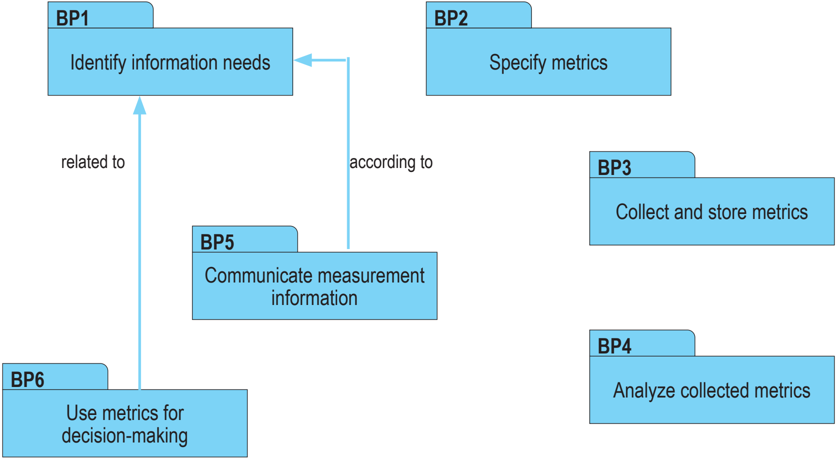
Note 3: Base metrics in the context of this process are directly gathered metrics like “number of defects found” or “number of lines of code”, where derived metrics are two or more metrics that are brought in relation to each other like “number of defects found per line of code”.

MAN.6.BP4: Analyze collected metrics. Analyze, interpret and review measured values to support decision-making.

MAN.6.BP5: Communicate analysis results. Communicate analysis results to all affected parties.

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.

MAN.6 Measurement	Outcome 1	Outcome 2	Outcome 3	Outcome 4	Outcome 5
Output Information Items					
03-03 Benchmarking data				x	x
03-04 Customer satisfaction data				x	x
03-06 Process performance information				x	x
07-51 Measurement resultX		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: Analyze collected metrics				x	x
BP5: Communicate measurement information					x
BP6: Use metrics for decision-making					x


BP1
Identify information needs
BP2
Specify metrics
related to
according to
BP3
Collect and store metrics
BP5
Communicate measurement information
BP4
Analyze collected metrics
BP6
Use metrics for decision-making

PIM.3 Process Improvement

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

PIM.3 with 8 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. Implement and apply the improvements to the processes. Update the Process documentation and train people as needed.

Note 9: Process application can be supported by establishing policies, adequate process infrastructure, process training, process coaching and tailoring processes to local needs.

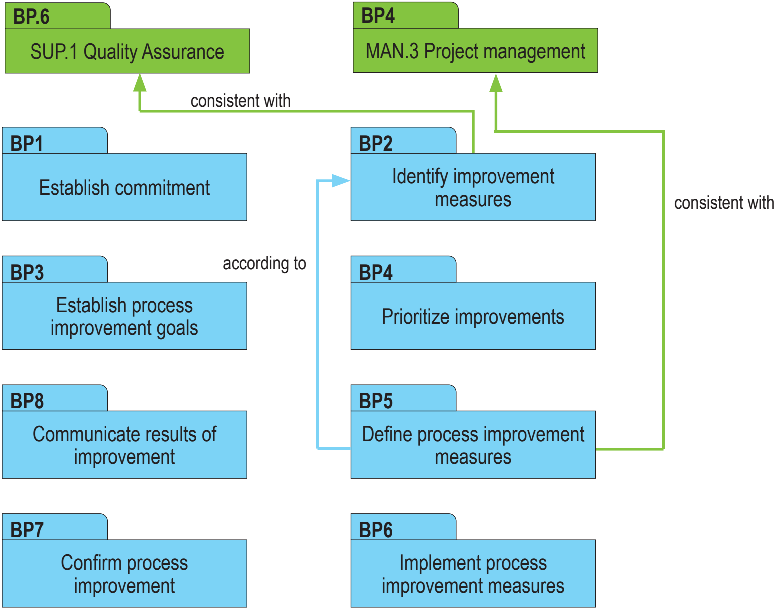
Note 10: Improvements may be piloted before roll out within the organization.

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 Items						
02-01 Commitment/agreement	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: Define process improvement measures				x		
BP6: Implement process improvement measures				x		
BP7: Confirm process improvement					x	
BP8: Communicate results of improvement						x

PIM.3 Rating Rules


BP.6
SUP.1 Quality Assurance
BP4
MAN.3 Project management
consistent with
BP1
Establish commitment
BP2
Identify improvement measures
consistent with
according to
BP3
Establish process improvement goals
BP4
Prioritize improvements
BP8
Communicate results of improvement
BP5
Define process improvement measures
BP7
Confirm process improvement
BP6
Implement process improvement measures

REU.2 Management of Products for Reuse

The purpose is to ensure that reused work products are analyzed, verified, and approved for their target context.

Process outcomes

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.

REU.2 with 6 Base practices

REU.2.BP1: Select products for reuse. Select the products to be reused using defined criteria.

Note 1: Products for reuse may be systems, hardware or software components, third party components or legacy components.

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.

Note 2: Examples for criteria can be requirements compliance, verifiability of the product to be reused in the target architecture, or portability/interoperability.

REU.2.BP3: Define limitations for reuse. Define and communicate limitations for the products to be reused.

Note 3: Limitations may address parameters of operational environment.

.

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.

Note 4: Qualification may be demonstrated by verification evidence.

Note 5: Verification may include the appropriateness of documentation.

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

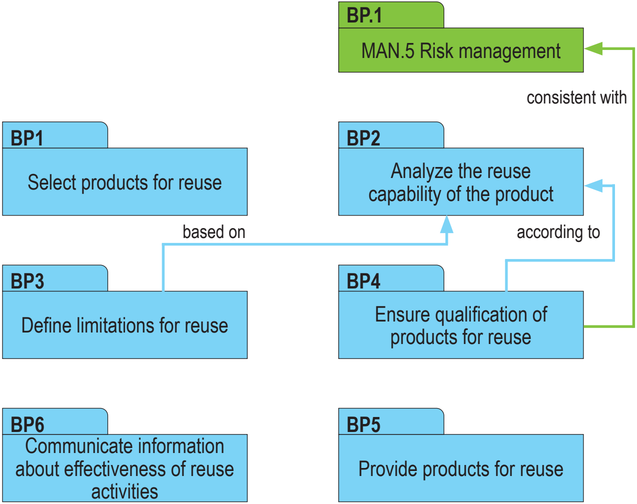
Note 6: Refer to HWE.3, SWE.5 or SYS.4 for more information on integration of hardware, software, or system components.

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 Items						
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

REU.2 Rating Rules


BP.1
MAN.5 Risk management
consistent with
BP1
Select products for reuse
BP2
Analyze the reuse capability of the product
based on
according to
BP3
Define limitations for reuse
BP4
Ensure qualification of products for reuse
BP6
Communicate information about effectiveness of reuse activities
BP5
Provide products for reuse

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.

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	**Process attribute ID** **Process attribute name** **Process attribute scope** **Process achievements**	Each process attribute is identified with a unique identifier and name. A process attribute scope statement is provided, and process achievements are defined
Process attribute acievement indicators	**Generic practices**	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. The generic practice headers are summarized at the end of a process to demonstrate their relationship to the process attribute achievements.
	Output information items	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. *Note: Refer to Annex B for the characteristics of each information item.*

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.

Process capability Level 1: Performed process

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

PA 1.1 Process performance process attribute

1. The process achieves its defined outcomes.

Generic practices GP 1.1.1

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
Output Information Items	
Process specific information items, as described in chapter 4	**x**
Generic practices	
GP 1.1.1 Achieve the process outcomes	**x**

Process capability Level 2: Managed process

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

PA 2.1 Process performance management process attribute

The performance management process attribute is a measure of the extent to which the performance of the process is managed.

Process attribute achievements

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. Persons 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.

GP 2.1.1: Identify the objectives and define a strategy for the performance of the process.

The scope of the process activities including the management of process performance and the management of work products are determined.

Corresponding results to be achieved are determined.

Process performance objectives and associated criteria are identified.

Note 1: Budget targets and delivery dates to the customer, targets for test coverage and process lead time are examples for process performance objectives.

Note 2: Performance objectives are the basis for planning and monitoring.

Assumptions and constraints are considered when identifying the performance objectives.

Approach and methodology for the process performance is determined.

Note 3: A process performance strategy may not necessarily be documented 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.

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 experience, knowledge and skill needs for the for process performance are determined based on the planning.

The needs for physical and material resources are determined based on the planning.

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.

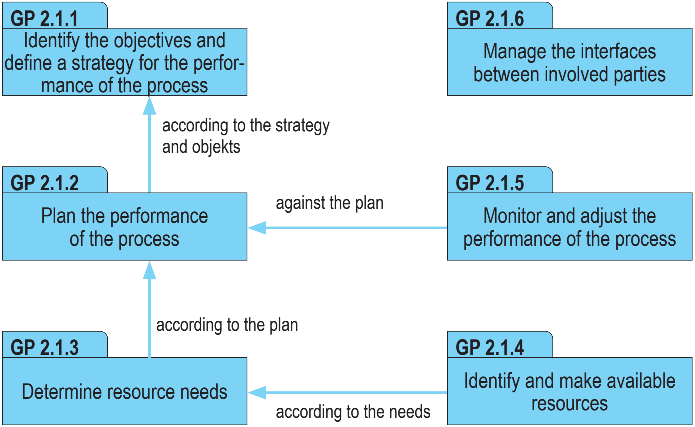
Responsibilities are assigned to the relevant individuals or parties.

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 Items								
19-01 Process performance strategy	x							
18-58 Process performance objectives	x							
14-10 Work package		x						
08-56 Schedule		x	x					
13-14 Progress status			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

PA2.1 Rating Rules


GP 2.1.1
Identify the objectives and define a strategy for the performance of the process
GP 2.1.6
Manage the interfaces between involved parties
according to the strategy and objekts
GP 2.1.2
Plan the performance of the process
against the plan
GP 2.1.5
Monitor and adjust the performance of the process
according to the plan
GP 2.1.3
Determine resource needs
GP 2.1.4
Identify and make available resources
according to the needs

PA 2.2 Work product management process attribute

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

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.s.

Generic practices GP 2.2.1 - GP 2.2.2

GP 2.2.1 Define the requirements for the work products.

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.

Note 1: Possible sources of documentation requirements may be e.g., best practices or lessons learned from other projects, standards, organization requirements, customer requirements, etc.

Note 2: There may be types of work products for which no review or approval is required, thus then there would be no need to define the corresponding criteria.

GP 2.2.2 Define the requirements for storage and control of the work products.

Requirements for the storage and control of the work products are defined, including their identification and distribution.

Note 3: Possible sources for the identification of requirements for storage and control may be e.g., legal requirements, data policies, best practices from other projects, tool related requirements, etc.

Note 4: Examples for work product storage are files in a file system, ticket in a tool, Wiki entry, paper documents etc.

Note 5: Where status of a work product is required in base practices, this should be managed via a defined status model.

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 for storage and control of the work products.

The work products including the revision status are made available through appropriate mechanisms.

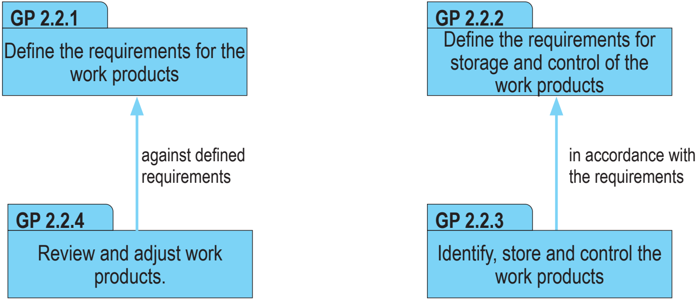
GP 2.2.4 Review and adjust work products.

The work products are reviewed against the defined requirements and 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 Items				
17-05 Requirements for work products	x	x		
18-59 Review and approval criteria for work products	x			
18-07 Quality criteria	x			
13-19 Review evidence				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

PA2.2 Rating Rules


GP 2.2.1
Define the requirements for the work products
against defined requirements
GP 2.2.4
Review and adjust work products.
GP 2.2.2
Define the requirements for storage and control of the work products
in accordance with the requirements
GP 2.2.3
Identify, store and control the work products

Process capability Level 3: Established process

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

PA 3.1 Process definition process attribute

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

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 competencies for performing the standard process are defined.
4. Tailoring guidelines for deriving the defined process from the standard process are defined.
5. Required physical and material resources and process infrastructure needs are determined as part of the standard process.
6. Suitable methods and required activities for monitoring the effectiveness, suitability and adequacy of the process are determined.

GP 3.1.1 Establish and maintain the standard process.

A suitable 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.

Process performance roles are identified and assigned to the standard process activities including their type of involvement, responsibilities, and authorities.

Note 1: An example for describing the involvement of the process roles in the activities is a RASI/RASIC representation.

Suitable guidance, procedures, and templates are provided to support the execution of the process as needed.

Note 2: Procedures may also include description of specific methods to be used.

Appropriate tailoring guidelines including predefined unambiguous criteria as well as predefined and unambiguous proceedings are defined based on identified deployment needs and context of the standard process.

The standard process is maintained according to corresponding feedback from the monitoring of the deployed processes.

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

GP 3.1.2 Determine the required competencies.

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

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

Note 4: Qualification methods are e.g., trainings, mentoring, self-study.

Note 5: 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 for performing the standard process are determined.

Note 6: 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 and adequacy of the standard process are determined.

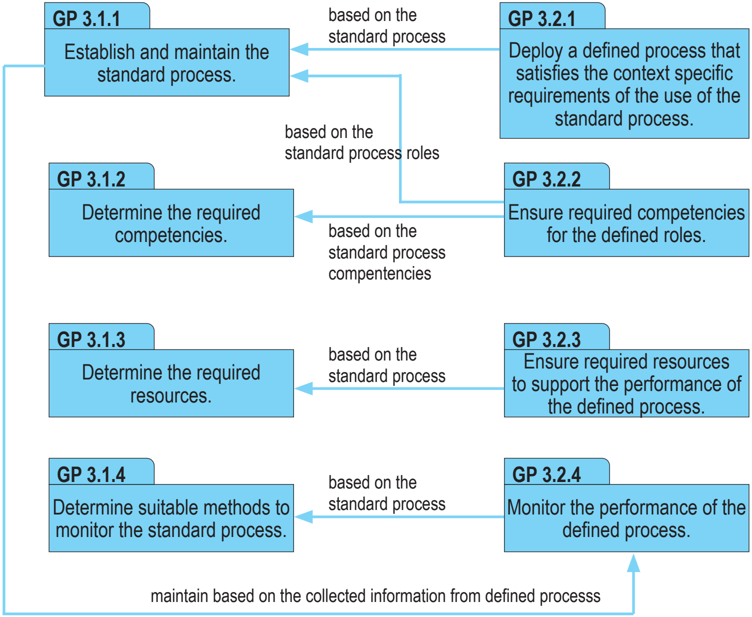
Note 7: Methods and activities to gather feedback regarding the standard process may be lessons learned, process compliance checks, internal audits, management reviews, change requests, reflection of state-of-the-art such as applicable international standards, etc.

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

Note 8: Information about process performance may be of qualitative or quantitative nature.

PA 3.1 Process definition process attribute	Achievement 1	Achievement 2	Achievement 3	Achievement 4	Achievement 5	Achievement 6
Output Information Items						
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 resources					x	
GP 3.1.4 Determine suitable methods to monitor the standard process						x

PA3.1/3.2 Rating Rules



GP 3.1.1

Establish and maintain the standard process.

GP 3.2.1

Deploy a defined process that satisfies the context specific requirements of the use of the standard process.

based on the standard process

based on the standard process roles

GP 3.1.2

Determine the required competencies.

GP 3.2.2

Ensure required competencies for the defined roles.

based on the standard process compentencies

GP 3.1.3

Determine the required resources.

GP 3.2.3

Ensure required resources to support the performance of the defined process.

based on the standard process

GP 3.1.4

Determine suitable methods to monitor the standard process.

GP 3.2.4

Monitor the performance of the defined process.

based on the standard process

maintain based on the collected information from defined processs

PA 3.2 Process deployment process attribute

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

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.
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 analyzed as a basis for understanding the behavior of the process.

GP 3.2.1 Deploy a defined process that satisfies the context specific requirements of the use of the standard process.

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.

The defined process is used as managed process to achieve the process outcomes.

Note 1: Changes in the standard process may require updates of the defined process.

GP 3.2.2 Ensure required competencies for the defined roles.

Human resources 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.

Gaps in competencies and skills are identified, and corresponding qualification measures are initiated and monitored.

Availability and usage of the project staff are measured and monitored.

GP 3.2.3 Ensure required resources to support the performance of the defined process.

Required information to perform the defined 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.

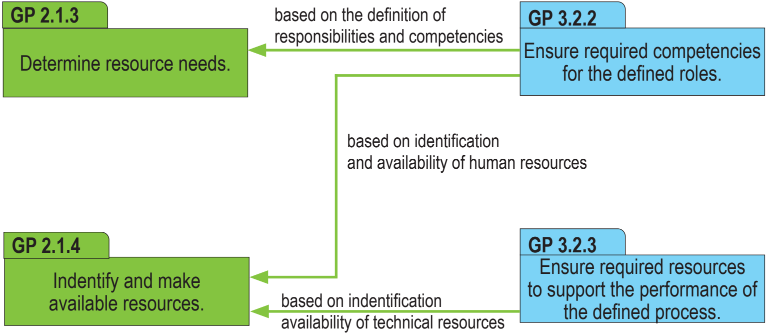
GP 3.2.4 Monitor the performance of the defined process.

Information is collected and analyzed according to the determined process monitoring methods to understand the effectiveness and adequacy of the defined process.

Results of the analysis are made available to all effected parties and 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).

PA 3.2 Process deployment process attribute	Achievement 1	Achievement 2	Achievement 3	Achievement 4	Achievement 5
Output Information Items					
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



GP 2.1.3

Determine resource needs.

based on the definition of responsibilities and competencies

GP 3.2.2

Ensure required competencies for the defined roles.

based on identification and availability of human resources

GP 2.1.4

Indentify and make available resources.

based on indentification availability of technical resources

GP 3.2.3

Ensure required resources to support the performance of the defined process.

Process capability Level 4: Predictable process

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

PA 4.1 Quantitative analysis process attribute

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

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.
   Note: Information needs typically reflect management, technical, project, process or product needs.

Generic practices GP 4.1.1 - GP 4.1.2

GP 4.1.1 Identify business goals.
Business goals are identified that are supported by the quantitatively measured process.

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.

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.

Note 1: Examples of process elements are work products, activities, tasks.

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						
18-70 Business goals	x	x				
07-61 Quantitative process metric		x	x			
07-62 Process analysis techniques				x		
07-63 Process control limits					x	
07-64 Process measurement data						x
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 defined process						x

PA 4.2 Quantitative control process attribute

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

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 GP 4.2.1 - GP 4.2.2

GP 4.2.1 Identify variations in process performance.

Deviations in the performance of process instances from the established quantitative control limits are determined based on the collected quantitative measurement data.

GP 4.2.2 Identify causes of variation.

The determined deviations in process performance are analyzed to identify potential cause(s) of variation using the defined analysis techniques.

Distributions are used to quantitatively understand the variation of process performance under the influence of potential causes of variation.

Consequences of process variation are analyzed.

GP 4.2.3 Identify and implement corrective actions to address assignable causes.

Results are provided to those responsible for taking action.

Corrective actions are determined and implemented to address each assignable cause of variation.

Corrective action results are monitored and evaluated to determine their effectiveness.

Note 1: Assignable cause may indicate a possible problem in the defined process.

PA 4.2 Quantitative control process attribute	Achievement 1	Achievement 2	Achievement 3	Achievement 4
Output Information Items				
15-57 Quantitative process analysis results	**x**	**x**	**x**	
08-66 Measures against deviations in quantitative process analysis				**x**
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**

Process capability Level 5: Innovating process

The following process attributes, together with the previously defined process attributes, demon-strate the achievement of this level.

PA 5.1 Process innovation process attribute

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

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 GP 5.1.1 - GP 5.1.3

GP 5.1.1 Define the process innovation objectives for the process that support the relevant business goals.

New business visions and goals are analyzed to give guidance for new process objectives and potential areas of process innovation.

Quantitative and qualitative process innovation objectives are defined and documented.

GP 5.1.2 Analyze quantitative data of the process.

Common causes of variation in process performance across process instances are identified and analyzed to get a quantitative understanding of their impact.

GP 5.1.3 Identify innovation opportunities.

Identify opportunities for innovation based on the quantitative understanding of the analyzed data.

Industry best practices, new technologies and process concepts are identified and evaluated. Feedback on opportunities for innovation is actively sought.

Emergent risks are considered in evaluating improvement opportunities.

PA 5.1 Process innovation process attribute	Achievement 1	Achievement 2	Achievement 3
Output Information Items			
18-80 Improvement opportunity	x		x
15-58 Common cause of variation analysis results		x	
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

PA 5.2 Process innovation implementation process attribute

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

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 is evaluated.

Generic practices GP 5.2.1 - GP 5.2.2

GP 5.2.1 Define and assess the impact of proposed changes.
Specified changes are assessed against product quality and process performance requirements and goals.
Impact of changes to other defined and standard processes is considered.
Objective priorities for process innovation are established.
Commitment to innovation is demonstrated by organizational management including other relevant stakeholders.

GP 5.2.2 Implement agreed process changes.
A mechanism is established for incorporating accepted changes into the defined and standard process(es) effectively and completely.
Process changes are implemented and effectively communicated to all affected parties.

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 1	Achievement 2	Achievement 3
Output Information Items			
18-81 Improvement evaluation results	**x**		**x**
08-66 Measures against deviations in quantitative process analysis		**x**	**x**
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**

Dependencies between processes and Process Attributes (Automotive SPICE and ME SPICE)

		PA 2.1	PA 2.2	PA 3.1	PA 3.2
MAN.3	Project Management	++			
MAN.5	Risk Management	+			+
REU.2	Reuse Program Management		+	+	
ACQ.4	Supplier Monitoring	+	+		
SUP.1	Quality Assurance		++		+
SUP.8	Configuration Management		++	+	
SUP.9	Problem Resolution Management	+	+		+
SUP.10	Change Request Management		+		+
SPL.2	Product Release		+		

A +/++ entry in a cell indicates a dependency: a weakness in the process most likely corresponds to a weakness in the related Process Attribute (PA)

MECHANICAL ENGINEERING SPICE 2.0

MEE.1 Mechanical Requirements Analysis

The purpose is to transform the mechanic related parts of the defined upper (Mechanical) System Requirements and the upper (Mechanical) System Architecture into Mechanical Requirements that will guide the design of the Mechanical System and the Mechanical Components.

Process outcomes

1. The Mechanical System Requirements and Mechanical Component Requirements are specified.
2. The Mechanical System Requirements and Mechanical Component Requirements are structured and prioritized.
3. The Mechanical System Requirements and Mechanical Component Requirements are analyzed for correctness, verifiability, and technical feasibility.
4. The impact of Mechanical System Requirements and Mechanical Component Requirements on the Operating Environment is analyzed.
5. Consistency and bidirectional traceability are established between the Mechanical System Requirements and the Upper System Requirements and/or Upper System Architecture.
6. Consistency and bidirectional traceability are established between the Mechanical Component Requirements and the Mechanical System Requirements and/or Mechanical System Architecture.
7. The Mechanical System Requirements and Mechanical Component Requirements are agreed and communicated to all affected parties.

MEE.1 with 6 Base practices

MEE.1.BP1: Specify Mechanical Requirements. Use the Upper (Mechanical) System Requirements and the Upper (Mechanical) System Architecture as well as changes to the Upper (Mechanical) System Requirements and Upper (Mechanical) Architecture to identify and document functional and non-functional Mechanical System Requirements and Mechanical Component Requirements according to defined characteristics for requirements.

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.

Note 2: Mechanical Requirements should include tolerances as necessary.

Note 3: Examples for defined characteristics of requirements shared by technical standards are verifiability (i.e., verification criteria being inherent in the requirements text), unambiguity/comprehensibility, freedom from design and implementation, and not contradicting any other requirement).

Note 4: In case of mechanical-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 mechanic.

MEE.1.BP2: Itructure Mechanical Requirements. Structure and prioritize the Mechanical System Requirements and Mechanical Component

Note 5: Examples for structuring criteria can be grouping, e.g., by functionality or expressing productvariants.

Note 6: Prioritization can be done according to project or stakeholder needs via e.g., definition of release scopes (e.g. A-/B-/C-Sample). Refer to Automotive SPICE® 4.0 (SPL.2).

MEE.1.BP3: Analyze Mechanical Requirements. Analyze the specified Mechanical System Requirements and Mechanical Component Requirements including their interdependencies to ensure correctness, technical feasibility, verifiability and to support project management regarding project estimates.

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

Note 8: Technical feasibility can be done based on given System Architectures (e.g., platform or standard product kits) or by means of prototype development.

MEE.1.BP4: Analyze the impact on the Operating Environment. Analyze the impact that the Mechanical Requirements will have on elements in the Operating Environment.

MEE.1.BP5: Ensure consistency and establish bidirectional traceability.

1. Ensure consistency and establish bidirectional traceability between the Mechanical System Requirements and the Upper System Requirements.
2. Ensure consistency and establish bidirectional traceability between the Mechanical System Requirements and the Upper System Architecture.
3. Ensure consistency and establish bidirectional traceability between the Mechanical Component Requirements and the Mechanical System Requirements.
4. Ensure consistency and establish bidirectional traceability between the Mechanical Component Requirements and the Mechanical System Architecture.

Note 9: Redundant traceability is not intended.

Note 10: Bidirectional traceability supports consistency, and facilitates impact analyses of change requests, and demonstration of verification coverage. Traceability alone, e.g., the existence of links, does not necessarily mean that the information is consistent with each other.

Note 11: In case of mechanic 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 mechanic requirements.

MEE.1.BP6: Communicate agreed Mechanical Requirements and impact on the Operating Environment. Communicate the agreed Mechanical System Requirements and Mechanical Component Requirements and results of the analysis of impact on the Operating Environment to all affected parties.

MEE.1 with 6 Base practices

MEE.1 Mechanical Requirements Analysis	Outcome 1	Outcome 2	Outcome 3	Outcome 4	Outcome 5	Outcome 6	Outcome 7
Output Information Items							
17-ME05 Mechanical System Requirement	x	x					
17-ME06 Mechanical Component 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 Mechanical Requirements	x						
BP2: Structure Mechanical Requirements		x					
BP3: Analyze Mechanical Requirements			x				
BP4: Analyze the impact on the Operating Environment				x			
BP5: Ensure consistency and establish bidirectional traceability					x	x	
BP6: Communicate agreed Mechanical Requirements and impact on the Operating Environment							x

MEE.2 Mechanical Architecture and Design

The purpose is to establish a Mechanical System Architecture and Mechanical Component Design, comprising static and dynamic aspects, consistent with the Mechanical System Requirements and Mechanical Component Requirements, and to evaluate the Mechanical System Architecture and Mechanical Component Design against defined criteria.

Process outcomes

1. The Mechanical System Architecture and Mechanical Components are designed including static and dynamic aspects.
2. The Mechanical System Architecture and Mechanical Component Design are analyzed against defined criteria and special characteristics are identified.
3. Consistency and bidirectional traceability are established between the Mechanical System Architecture and Mechanical System Requirements.
4. Consistency and bidirectional traceability are established between the Mechanical Component Design and Mechanical System Architecture and/or Mechanical Component Requirements.
5. The Mechanical System Architecture and the Mechanical Component Design are agreed and communicated to all affected parties.

MEE.2 with 6 Base practices

MEE.2.BP1: Specify static aspects of the Mechanical System and Mechanical Component.

Specify and document the

a) static structure of the elements of the Mechanical System, including their interfaces, and their relationships

b) static aspects of each Mechanical System Element

with respect to the functional and non-functional Mechanical System Requirements and Mechanical Component Requirements, including external interfaces. Document the rationale behind the Mechanical System Architecture and Mechanical Component Design decisions.

Note 1: The Mechanical System is decomposed into elements across appropriate hierarchical levels down to the Mechanical Components (the elements on the lowest level of the Mechanical System Architecture) that are described in the Mechanical Component Design.

Note 2: Examples of a design rationale can be implied by the reuse of a standard product kit, product platform, or product line, respectively, or by a make-or-buy decision, or found in an evolutionary way (e.g. set-based design).

Note 3: Model-based development (e.g. FEM, SysML) may facilitate the collaboration of the different engineering domains.

Note 4: Design for Manufacturing and Design for Assembly may be used to ensure manufacturability.

Note 5: Static aspects are determined by e.g., mechanical system structure.

Note 6: Non-functional requirements may include e.g. price per unit, maintenance, logistics, packaging, size, weight, manufacturability, environmental constraints, design guidelines, modelling guidelines, failure times.

MEE.2.BP2: Specify dynamic aspects of the Mechanical System and Mechanical Component. Specify and document the dynamic aspects of the Mechanical System and Mechanical Component with respect to the functional and non-functional Mechanical System Requirements, including the behavior of the Mechanical Elements and their interaction in the different modes.

Note 7: Dynamic aspects are determined by e.g., natural frequencies, stress, force, pressure, strain, temperature, NVH (Noise Vibration Harshness), operating modes (open, closed, in motion, misuse, emergency, etc.).

MEE.2.BP3: Analyze the Mechanical System Architecture and Mechanical Component Design. Analyze the Mechanical System Architecture and Mechanical Component Design regarding relevant technical aspects and related to the Product Lifecycle to support project estimates. Identify and document Special Characteristics. Document the rationales for the architectural and design decisions.

Note 8: Analysis criteria shall be defined. Analysis criteria may include quality characteristics (cost, weight, packaging, modularity, maintainability, expandability, scalability, reliability, safety and usability) and results of make-buy-reuse analysis.

Note 9: Analysis of the Mechanical System Architecture and Mechanical Component Design supports project feasibility analysis (MAN.3 BP3) and project estimates (MAN.3.BP5).

Note 10: The analysis may include the suitability of pre-existing Mechanical Elements for the current application.

Note 11: Examples for Product Lifecycle Phases are production, maintenance & repair, decommissioning.

Note 12: Examples for technical aspects are manufacturability, suitability of pre-existing elements to be reused, or availability of elements.

Note 13: Examples of methods suitable for analyzing technical aspects are prototypes, simulations, qualitative analyses. The simulation methods could be FEM, FMEA, CFD.

Note 14: The identification of Special Characteristics is supported by e.g., simulation, risk analyses, sizing calculations.

Note 15: Design rationales can include arguments such as proven-in-use, a make-or-buy decision, or found in an evolutionary way.

MEE.2.BP4: Consider, determine, and document Design Constraints. Determine and document Design Constraints for all Mechanical Elements and take them into account for creating the Mechanical System Architecture and Mechanical Component Design.

Note 16: Design constraints can be e.g., design guidelines, materials, manufacturing processes.

MEE.1.BP5: Ensure consistency and establish bidirectional traceability.

1. Ensure consistency and establish bidirectional traceability between the Elements of the Mechanical System Architecture and Mechanical System Requirements.
2. Ensure consistency and establish bidirectional traceability between the Mechanical Component Design and the Mechanical System Architecture.
3. Ensure consistency and establish bidirectional traceability between the Mechanical Component Design and Mechanical Component Requirements.

Note 17: Redundancy should be avoided by establishing a combination of the approaches BP4.2 and BP4.3 that covers the project and the organizational needs.

Note 18: Bidirectional traceability supports consistency and facilitates impact analyses of change requests, and demonstration of verification coverage.

Note 19: Traceability alone, e.g., the existence of links, does not necessarily mean that the information is consistent with each other.

MEE.2.BP6: Communicate agreed Mechanical System Architecture and agreed Mechanical Component Design. Communicate the agreed Mechanical System Architecture and the agreed Mechanical Component Design to all affected parties, including the Special Characteristics and updates to the Mechanical System Architecture and Mechanical Component Design.

MEE.2 Mechanical Architecture and Design	Outcome 1	Outcome 2	Outcome 3	Outcome 4
Output Information Items				
04-ME01 Mechanical System Architecture	x			
04-ME02 Mechanical Component Design	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 the Mechanical System and Mechanical Component	x			
BP2: Specify dynamic aspects of the Mechanical System and Mechanical Component	x			
BP3: Analyze the Mechanical System Architecture and Mechanical Component Design		x		
BP4: Consider, determine, and document Design Constraints	x			
BP5: Ensure consistency and establish bidirectional traceability			x	
BP6: Communicate agreed Mechanical System Architecture and Mechanical Component Design				x

MEE.3 Mechanical Component Sample Production

The purpose is to produce a Mechanical Component Sample that reflects properly the Mechanical Component Design and Mechanical Component Sample Production Specification.

Process outcomes

1. A Mechanical Component Sample Production Specification is developed, agreed on with and communicated to all affected parties.
2. Mechanical Component Samples are produced according to the Mechanical Component Sample Production Specification and Mechanical Component Design.
3. Consistency and bidirectional traceability are established between the Mechanical Component Sample Production Specification and Mechanical Component Design; and bidirectional traceability is established as chain between the Mechanical Component Sample, Production Data and Mechanical Component Sample Production Specification.
4. The Production Data are summarized to the Production Report, which is communicated to all affected parties.

MEE.3 with 5 Base practices

MEE.3.BP1: Develop Mechanical Component. Develop a Specification for Sample Production of the Mechanical Components. The Mechanical Component Sample Production Specification shall be consistent with the Mechanical Component Design
Note 1: The Mechanical Component Sample Production Specification may contain the definition of the production method(s), verification method(s) (control plan).

MEE.3.BP2: Agree on Mechanical Component Sample Production Specification. *Communicate the agreed Mechanical Component Sample Production Specification to all affected parties (e.g., engineering, sample shop, production).*
Note 2: The communication of the Mechanical Component Sample Production Specification to suppliers is handled by ACQ.4 Supplier monitoring.

MEE.3.BP3: Produce the Mechanical Component Samples. Ensure and support the Sample Production of Mechanical Components according to the Mechanical Component Design and the Mechanical Component Sample Production Specification. Record Production Data according to the Mechanical Component Sample Production Specification.

Note 3: Production here means only sample phases (e.g., prototype building, pre-series production) and does not cover the process of industrialization.

MEE.3.BP4: Ensure consistency and establish bidirectional traceability.

1. Ensure consistency and establish bidirectional traceability between the Mechanical Component Sample Production Specification and the Mechanical Component Design.
2. Establish bidirectional traceability between the recorded Production Data and the Mechanical Component Sample Production Specification.
3. Establish bidirectional traceability between the produced Mechanical Component Samples and the recorded Production Data.

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

MEE.3.BP5: Summarize and communicate the Production Data. *Summarize the Production Data of the Mechanical Component Samples to the Production Report and communicate it to all affected parties.*

Note 5: Production Data may contain:

- *Capability of chosen production method*
- *Manufacturability of the Mechanical Component Samples*
- *Improvement potentials for future releases*
- *Process Data and information*

Note 6: See SUP.9 for handling verification results that deviate from expected results.

Note 7: The communication of information mentioned above is handled by ACQ.4 Supplier monitoring in case of production at a supplier's site.

Affected parties could be:

- *Industrialization*
- *Series production*
- *Mechanical engineering*
- *Project Management*
- *Quality Assurance*

MEE.3 Mechanical Component Sample Production	Outcome 1	Outcome 2	Outcome 3	Outcome 4
Output Information Items				
19-ME01 Mechanical Component Sample Production Specification	x	x		
11-ME04 Mechanical Component		x		
13-51 Consistency Evidence			x	
15-ME01 Production Report				x
13-52 Communication Evidence				x
Base Practices				
BP1: Develop Mechanical Component Sample Production Specification	x			
BP2: Agree on Mechanical Component Sample Production Specification	x			
BP3: Produce the Mechanical Component Samples		x		
BP4: Ensure consistency and establish bidirectional traceability			x	
BP5: Summarize and communicate the Production Data				x

MEE.4 Mechanical Integration and Verification against Mechanical Architecture and Design

The purpose is:

1. to verify the Mechanical Component against the Mechanical Component Design and
2. to ensure the integration of the Mechanical Elements into an integrated Mechanical System consistent with the Mechanical System Architecture and
3. to verify the integrated Mechanical System against the Mechanical System Architecture.

Process outcomes

1. A Verification Measures are specified for Mechanical Component Design Verification based on the Mechanical Component Design.
2. Measures for Mechanical Component Design Verification are selected according to the Release Scope considering Regression Criteria.
3. The Mechanical Component Design is verified using the selected Verification Measures and the Verification Results are recorded.
4. The Mechanical Integration Sequence of the Mechanical Elements (Mechanical Component and/or Mechanical System) is specified consistent with the Mechanical System Architecture.
5. Verification Measures are specified for the Mechanical System Integration Verification based on the Mechanical System Architecture, including the interfaces and interactions between Mechanical Elements.
6. Mechanical Elements are integrated up to a complete integrated Mechanical System consistent with the Release Scope.
7. Measures for Mechanical System Integration Verification are selected according to the Release Scope considering Regression Criteria.
8. Integrated Mechanical Elements are verified using the selected Verification Measures and the Verification Results are recorded.
9. Consistency and bidirectional traceability are established between the Mechanical Component Design Verification Measures and Mechanical Component Design; and

   bidirectional traceability is established as chain between the Mechanical Component, Mechanical Component Design Verification Results and Mechanical Component Design Verification Measures.

Process outcomes

10. Consistency and bidirectional traceability are established between Integration Steps and the Mechanical System Architecture; and consistency and bidirectional traceability are established between the Mechanical System Integration Verification Measures and Mechanical System Architecture; and
    bidirectional traceability is established as chain between the Mechanical System, Integration Data and Integration Steps; and
    bidirectional traceability is established as chain between the Mechanical System, Mechanical System Integration Verification Results and Mechanical System Integration Verification Measures.
11. Bidirectional traceability is established between the Mechanical Component and Mechanical System.
12. The Integration Data are summarized to the Integration Report, which is communicated to all affected parties; the Mechanical Component Design Verification Results and the Mechanical System Integration Verification Results are summarized and communicated to all affected parties.

MEE.4 with 8 Base practices

MEE.4.BP1: Specify Verification Measures for the Mechanical Component Design. Specify Mechanical Component Design Verification Measures suitable to provide evidence for compliance of the Mechanical Component with the Mechanical Component Design. This includes:

a) techniques for the Verification Measures
b) pass/fail criteria for Verification Measures
c) a definition of Entry and Exit Criteria for the Verification Measures
d) necessary sequence of Verification Measures
e) the required Verification Infrastructure and Environment Setup

MEE.4.BP2: Select Verification measures for the Mechanical Component Design. *Document the selection of the Mechanical Component Design 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 1: Examples for Selection Criteria can be prioritization of requirements, the need for Regression Verification due to e.g. changes to the Mechanical Component Design, the intended use of the delivered product release (test bench, test track, public road etc.).

MEE.4.BP3: Perform Mechanical Component Design Verification. Perform Mechanical Component Design Verification using the selected Mechanical Component Design Verification Measures. Record the Mechanical Component Design Verification Results including pass/fail status and measured values with reference to the verified Mechanical Component.
Note 2: Capable Verification Environment as defined in the Component Design Verification Measures needs to be available for performing verification against Mechanical Component Design.
Note 3: See SUP.9 for handling verification results that deviate from expected results.

MEE.4.BP4: Define Integration Sequence Instruction and specify Mechanical System Integration Verification Measures. Identify Mechanical Elements based on the Mechanical System Architecture. Define the Integration Sequence Instruction including Integration Steps and Integration Verification Measures for the Mechanical System Integration.

The Integration Instruction shall be suitable to provide evidence for compliance of the integrated Mechanical System with the Mechanical System Architecture. This includes:

a)preconditions and techniques for the Verification Measures
b) pass/fail criteria for Verification Measures
c) a definition of Entry and Exit Criteria for Integration and the Verification Measures
d) necessary sequence of Verification Measures
e) the required Verification Infrastructure and Environment Setup

Note 4: Internal interfaces (between the mechanical elements) and external interfaces should be verified according to the Mechanical System Architecture and the specified Mechanical System Integration Verification Measures.

MEE.4.BP5: Select Mechanical System Integration Verification Measures. Document the selection of Mechanical System 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 5: Examples for Selection Criteria can be prioritization of requirements, the need for Regression Verification due to e.g., changes to the Mechanical System Architecture, or the intended use of the delivered product release.

MEE.4.BP6: Integrate Mechanical System Elements and perform Mechanical System Integration Verification.
Integrate the Mechanical Elements into an integrated Mechanical System according to the Release Scope based on the Integration Sequence Instruction.
Perform the selected Mechanical System Integration Verification Measures to verify the Mechanical Interfaces.
Record the Mechanical System Integration Verification Results including pass/fail status, integration data and the corresponding Verification Measure Data with reference to the Mechanical Elements..

Note 6: The Mechanical System Integration should be performed with verified Mechanical Elements. Otherwise, justification should be provided.

Note 7: Capable Verification Infrastructure and Environment Setup as defined in the Mechanical System Integration Verification Measures needs to be available for performing Mechanical System Integration and Mechanical System Integration Verification.

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

MEE.4.BP7: Ensure consistency and establish bidirectional traceability.

1. Ensure consistency and establish bidirectional traceability between the Mechanical Component Design Verification Measures and the Mechanical Component Design.
2. Establish bidirectional traceability between the Mechanical Component Design Verification Results and the Mechanical Component Design Verification Measures.

3. Establish bidirectional traceability between the verified Mechanical Components and the Mechanical Component Design Verification Results.
4. Ensure consistency and establish bidirectional traceability between the Integration Steps and the Mechanical System Architecture.
5. Establish bidirectional traceability between the Integration Data and the Integration Steps.
6. Establish bidirectional traceability between the integrated Mechanical System and the Integration Data.
7. Ensure consistency and establish bidirectional traceability between the Mechanical System Integration Verification Measures and the Mechanical System Architecture.
8. Establish bidirectional traceability between the Mechanical System Integration Verification Results and the Mechanical System Integration Verification Measures.
9. Establish bidirectional traceability between the verified integrated Mechanical System and the Mechanical System Integration Verification Results.
10. Establish bidirectional traceability between the Mechanical Components and integrated Mechanical System.

Note 9: Bidirectional traceability supports consistency, and facilitates impact analyses of change requests, and demonstration of verification coverage. Traceability alone, e.g. the existence of links, does not necessarily mean that the information is consistent with each other.

MEE.4.BP8: Summarize and communicate the Integration and Verification Results.

Summarize the Mechanical Component Design Verification Results.
Summarize the Integration Data into the Integration Report.
Summarize the Mechanical System Integration Verification Results.
Communicate them to all affected parties.

Note 10: Providing all necessary information from the verification measure execution in a summaryenables other parties to judge the consequences.

MEE.1 Mechanical Integration and Verification against Mechanical Architecture and Design	Outcome 1	Outcome 2	Outcome 3	Outcome 4	Outcome 5	Outcome 6	Outcome 7	Outcome 8	Outcome 9	Outcome 10	Outcome 11	Outcome 12
Output Information Items												
08-60 Verification Measure	x				x							
08-58 Verification Measure Selection Set		x					x					
15-52 Verification Results			x					x				
03-50 Verification Measure Data								x				
06-50 Integration Sequence Instruction				x								
11-ME03 Mechanical System						x						
13-51 Consistency Evidence									x	x	x	
15-ME02 Integration Report												x
13-52 Communication Evidence												x
Base Practices												
BP1: Specify Mechanical Component design verification measures	x											
BP2: Select Mechanical Component design verification measures		x										
BP3: Perform Mechanical Component design verification			x									
BP4: Define Integration Sequence Instructionand specify Mechanical System Integration Verification Measures				x	x							
BP5: Select Mechanical System Integration verification measures							x					
BP6: Integrate Mechanical System elements and perform Mechanical System integration verification						x		x				
BP7: Ensure consistency and establish bidirectional traceability									x	x		
BP8: Summarize and communicate the Integration and verification results											x	x

MEE.5 Verification against Mechanical Requirements

The purpose is to ensure that the Mechanical Components and the integrated Mechanical System are verified to provide evidence for compliance with the Mechanical System Requirements and Mechanical Component Requirements.

Process outcomes

1. Verification Measures are specified for the Mechanical System Verification and Mechanical Component Verification based on the Mechanical System Requirements and Mechanical Component Requirements.
2. Verification Measures are selected according to the Release Scope considering Regression Criteria, including criteria for Regression Verification.
3. The Mechanical System and Mechanical Components are verified using the selected Verification Measures and the Verification Results are recorded.
4. Consistency and bidirectional traceability are established between the Mechanical Component Requirements Verification Measures and Mechanical Component Requirements; and bidirectional traceability is established as chain between Mechanical Component, Mechanical Component Requirements Verification Results and Mechanical Component Requirements Verification Measures.
5. Consistency and bidirectional traceability are establised between the Mechanical System Requirements Verification Measures and Mechanical System Requirements; and bidirectional traceability is established as chain between the Mechanical System, Mechanical System Requirements Verification Results and Mechanical System Requirements Verification Measures.
6. Mechanical System Verification Results and Mechancal Component Verification Results are summarized and communicated to all affected parties.

MEE.5 with 5 Base practices

MEE.5.BP1: Specify Verification Measures for Mechanical Components and integrated Mechanical System. Specify Verification Measures for Mechanical Components and integrated Mechanical System suitable to provide evidence for compliance with the with the functional and non-functional information Mechanical System Requirements and with the functional and non-functional information *Mechanical Component Requirements, including*:

a)techniques for the Verification Measures
b) pass/fail criteria for Verification Measures
c) a definition of entry and exit criteria for the Verification Measures
d) necessary sequence of Verification Measures
e) the required Verification Infrastructure and Environment Setup

Note 1: The Verification Measures may cover aspects such as thermal, environmental, robustness/lifetime, etc.

MEE.5.BP2: Select Verification Measures for Mechanical Components and integrated Mechanical System. Select Mechanical Component Requirements Verification Measures as well as Mechanical System Requirements Verification Measures. Document the selection of the 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, the need for regression due to e.g. changes to the Mechanical Component Requirements or the Mechanical System Requirements, the intended use of the delivered product release (test bench, test track, public road etc.)

MEE.5.BP3: Verify the Mechanical Components and integrated Mechanical System. Perform the verification of the Mechanical Components and the integrated Mechanical System using the selected Verification Measures. Record the Verification Results including pass/fail status and corresponding Verification Measure Data.

Note 3: Capable verification environment as defined needs to be available for performing verification of mechanical component and integrated mechanical system.

Note 4: Mechanical Elements can be physical or virtual.

Note 5: See SUP.9 for handling verification results that deviate from expected results.

MEE.5.BP4: Ensure consistency and establish bidirectional traceability.

1. Ensure consistency and establish bidirectional traceability between the Mechanical System Requirements Verification Measures and the Mechanical System Requirements.
2. Establish bidirectional traceability between the Mechanical System Requirements Verification Results and the Mechanical System Requirements Verification Measures.
3. Establish bidirectional traceability between the integrated Mechanical System and the Mechanical System Requirements Verification Results.
4. Ensure consistency and establish bidirectional traceability between the Mechanical Component Requirements Verification Measures and the Mechanical Components Requirements.
5. Establish bidirectional traceability between the Mechnical Component Requirements Verification Results and the Mechanical Component Requirements Verification Measures.
6. Establish bidirectional traceability between the Mechnical Components and the Mechanical Component Requirements Verification Results

Note *6: Bidirectional traceability supports consistency, and facilitates impact analyses of change requests, and demonstration of verification coverage. Traceability alone, e.g. the existence of links, does not necessarily mean that the information is consistent with each other.*

MEE.5.BP5: Summarize and communicate the Verification Results.

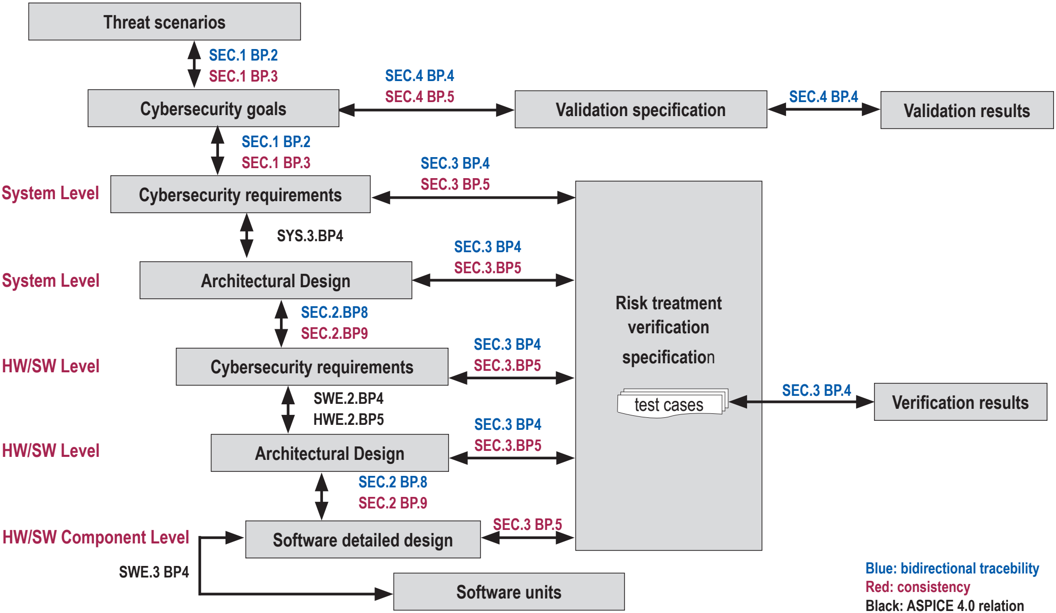
Summarize the Mechanical Component Requirements Verification Results and the Mechanical System Requirements Verification Results and communicate them to all affected parties.

Note 7: Providing all necessary information from the verification measure execution in a summary enables other parties to judge the consequences

MEE.5 Verification against Mechanical Requirements	Outcome 1	Outcome 2	Outcome 3	Outcome 4	Outcome 5	Outcome 6
Output Information Items						
08-60 Verification Measure	x					
03-50 Verification Measure Data			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 Mechanical Components and integrated Mechanical System	x					
BP2: Select Verification Measures for Mechanical Components and integrated Mechanical System		x				
BP3: Perform verification of the Mechanical Components and integrated Mechanical System			x			
BP4: Ensure consistency and establish bidirectional traceability				x	x	
BP5: Summarize and communicate the Verification Results						x

AUTOMOTIVE SPICE® CYBERSECURITY Version1.0

Bidirectional traceability and Consistency for Cybersecurity



Threat scenarios

SEC.1 BP.2

SEC.1 BP.3

Cybersecurity goals

SEC.4 BP.4

SEC.4 BP.5

Validation specification

SEC.4 BP.4

Validation results

SEC.1 BP.2

SEC.1 BP.3

System Level

Cybersecurity requirements

SEC.3 BP.4

SEC.3 BP.5

SYS.3.BP4

System Level

Architectural Design

SEC.3 BP4

SEC.3.BP5

SEC.2.BP8

SEC.2.BP9

Risk treatment verification specification

test cases

HW/SW Level

Cybersecurity requirements

SEC.3 BP4

SEC.3.BP5

SEC.3 BP.4

Verification results

SWE.2.BP4

HWE.2.BP5

HW/SW Level

Architectural Design

SEC.3 BP4

SEC.3.BP5

SEC.2 BP.8

SEC.2 BP.9

HW/SW Component Level

Software detailed design

SEC.3 BP.5

SWE.3 BP4

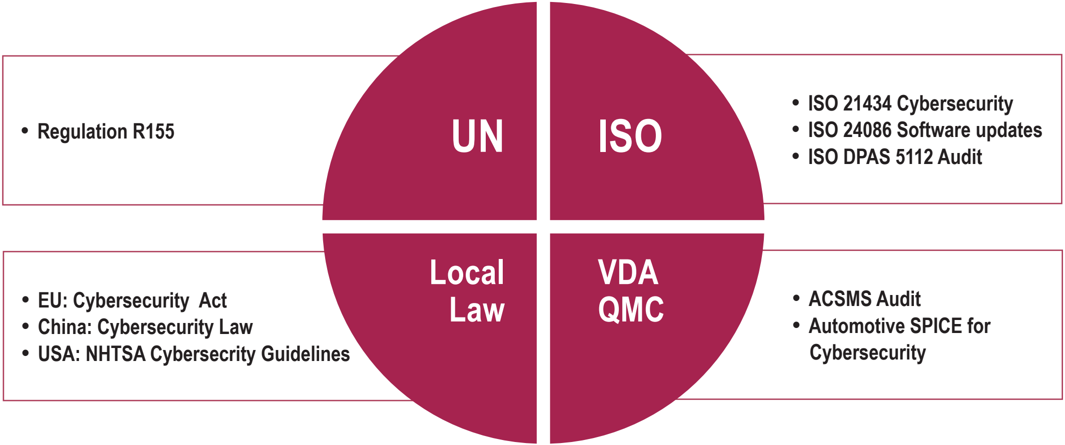
Software units

Blue: bidirectional tracebility

Red: consistency

Black: ASPICE 4.0 relation

Cybersecurity Regulation, ISO and Audit/Assesments

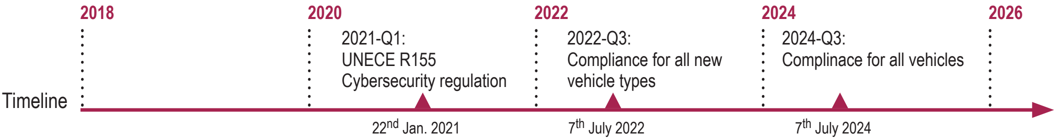


- Regulation R155

- ISO 21434 Cybersecurity
- ISO 24086 Software updates
- ISO DPAS 5112 Audit

- EU: Cybersecurity Act
- China: Cybersecurity Law
- USA: NHTSA Cybersecrity Guidelines

- ACSMS Audit
- Automotive SPICE for Cybersecurity

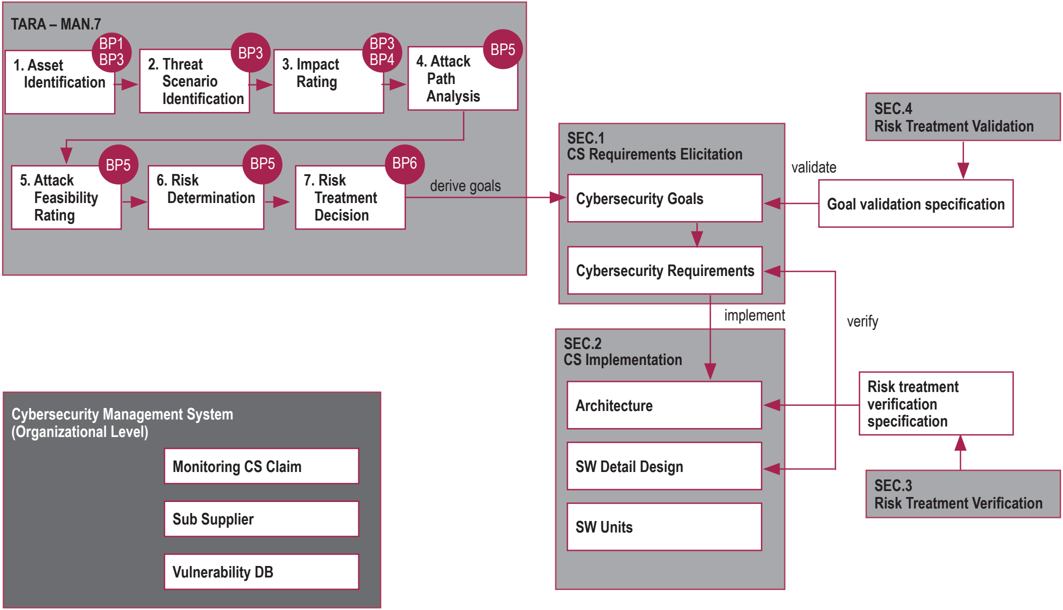


2021-Q1:
UNECE R155
Cybersecurity regulation
22nd Jan. 2021

2022-Q3:
Compliance for all new vehicle types
7th July 2022

2024-Q3:
Complinace for all vehicles
7th July 2024

Cybersecurity Processes at a Glance



TARA – MAN.7

1. Asset Identification BP1 BP3

2. Threat Scenario Identification BP3

3. Impact Rating BP3 BP4

4. Attack Path Analysis BP5

5. Attack Feasibility Rating BP5

6. Risk Determination BP5

7. Risk Treatment Decision BP6

derive goals

SEC.1 CS Requirements Elicitation

Cybersecurity Goals

Cybersecurity Requirements

validate

SEC.4 Risk Treatment Validation

Goal validation specification

implement

verify

SEC.2 CS Implementation

Architecture

SW Detail Design

SW Units

Risk treatment verification specification

SEC.3 Risk Treatment Verification

Cybersecurity Management System (Organizational Level)

Monitoring CS Claim

Sub Supplier

Vulnerability DB

TARA – Threat Analyses and Risk Assessment

Step 1 ❶ **Asset Identification**	**Step 2** ❷ **Threat Scenario Identification**			**Step 3** ❸ **Stakeholder Impact Rating**						
Asset	Security Property	Damage Scenario	Threat Scenario	Road User				OEM		
				S	F	O	P	FOEM	OOEM	POEM
Function: Synchronization between Bluetooth connected phone and internal phone number storage	*Authenticity*	*falsified phone data due to spoofed identity*	*Synchronization with a spoofed phone, unknown phone spoofs a known phone*	**0**	**1**	**0**	**1**	**1**	**0**	**0**
	Integrity	*falsified phone data due to corrupted messages*	*Tampering: Man in the middle attack allows corruption of messages*	**0**	**1**	**0**	**1**	**1**	**0**	**0**
	Non-repudiation	*falsified phone data see Authenticity*	*see Authenticity and Integrity*	**0**	**1**	**0**	**1**	**1**	**0**	**0**
	Confidentiality	*violation of privacy due to exposition of confidential data*	*Information Disclosure:* *a) unknown phone spoofs a known phone to access data* *b) man in the middle attack forwards data to attacker* *c) eavesdropping*	**0**	**2**	**0**	**3**	**2**	**0**	**0**
	Availability	*synchronization failure due to failed connection*	*a) man in the middle attack blocks communication* *b) Denial of Service attack overloads connection*	**0**	**0**	**0**	**0**	**0**	**0**	**0**
	Authortization	*loss of data due to inappropriate access rights*	*Elevation of Privileges: connected phone can access unauthorized Bluetooth profiles*	**0**	**1**	**0**	**1**	**1**	**0**	**0**

TARA – Threat Analyses and Risk Assessment

Threat Scenario Identification	Step 4 ❹ Attack Path Analysis	Step 5 ❺ Attack Feasibility Rating				
Security Property	**Attack Path Description**	**Elapsed Time**	**Specialist Expertise**	**Knowledge of the item**	**Window of opportunity**	**IT, HW/SW, or other equipment required**
Authenticity	*'1. attackers phone must emulate a paired device (CVE-2020-9770)* *2. establish connection with vehicle* *3. trigger synchronization*	≤ 1 month	Expert	Strictly confidential information	Easy	Standard
Integrity	*'1. verify Bluetooth vulnerability (CVE-2018-5383, CVE-2020-15802)* *2. establish connection with man in the middle device* *3. tamper with the data as desired*	≤ 1 month	Expert	Public information	Easy	Standard
Non-repudiation	*see Authenticity and Integrity*	≤ 1 month	Expert	Public information	Easy	Standard
Confidentiality	*'1. gather encryption key during reconnection process* *2. eavesdrop on communication, collect data* *3. copy data and use as desired*	≤ 1 month	Expert	Public information	Easy	Standard
Availability	*'1. send a flood of Bluetooth requests to occupy/overload receiver*	≤ 1 week	Expert	Public information	Easy	Standard
Authortization	*'1. attackers phone must emulate a paired device (CVE-2020-9770)* *2. establish connection with vehicle* *3. switch to an unauthorized Bluetooth profile* *4. perform unauthorized action*	≤ 1 month	Expert	Public information	Easy	Standard

TARA – Threat Analyses and Risk Assessment

Threat Scenario Identification	Step 5 ❺ Attack Feasibility Rating							
	Attack Feasibility Indication							
Security Property	Elapsed Time	Specialist Expertise	Knowledge of the item	Window of opportunity	IT, HW/SW, or other equipment required	Total Value	Attack Feasibility Indicator	Attack feasibility Value
Authenticity	4	6	11	1	0	22	Low	1
Integrity	4	6	0	1	0	11	High	3
Non-repudiation	4	6	0	1	0	11	High	3
Confidentiality	4	6	0	1	0	11	High	3
Availability	1	6	0	1	0	8	High	3
Authortization	4	6	0	1	0	11	High	3

Threat Scenario Identification	Step 6 ❻ Risk Determination							Step 7 ❼ Risk Treatment Decision	SEC.1 BP1	
Security Property	Stakeholder: Road User				Stakeholder: OEM			Option taken	Cybersecurity Goal/ Controls/ Remarks	status of controls
	S	F	O	P	FOEM	OOEM	POEM			
Authenticity	0	1	0	1	1	0	0	*reducing the risk (e.g. implementation of controls)*	*CSG1000:* *Unauthorized Users/Phones can not spoof data in communication*	*open*
Integrity	0	2	0	2	2	0	0	*reducing the risk (e.g. implementation of controls)*	*CSG1001:* *Man in the middle attacs are imposible to tamper communication*	*ongoing*
Non-repudiation	0	2	0	2	2	0	0	*reducing the risk (e.g. implementation of controls)*	*CSG1000*	*open*
Confidentiality	0	3	0	4	3	0	0	*reducing the risk (e.g. implementation of controls)*	*CSG1002:* *Information can not be disclosed in communication by any attack*	*open*
Availability	0	0	0	0	0	0	0	*accepting or retaining the risk*	*no derived CSG*	*open*
Authortization	0	2	0	2	2	0	0	*reducing the risk (e.g. implementation of controls)*	*CSG1003:* *Unauthorized Users/Phones can gain access to profile*	*open*

Criteria for road user impact rating

Impact rating	Financial	Operational	Privacy	Safety
Severe	F3: The financial damage leads to catastrophic consequences which the affected road user might not overcome.	O3: The operational damage leads to the loss or impairment of a core vehicle function.	P3: The privacy damage leads to significant or even irreversible impact to the road user.	S3: Life-threatening injuries (survival uncertain), fatal injuries
Major	F2: The financial damage leads to substantial consequences which the affected road user will be able to overcome.	O2: The operational damage leads to the loss or impairment of an important vehicle function	P2: The privacy damage leads to serious impact to the road user.	S2: Severe and life-threatening injuries (survival probable)
Moderate	F1: The financial damage leads to inconvenient consequences which the affected road user will be able to overcome with limited resources.	O1: The operational damage leads to partial degradation of a vehicle function.	P1: The privacy damage leads to inconvenient consequences to the road user.	S1: Light and moderate injuries
Negligible	F0: The financial damage leads to no effect, negligible consequences or is irrelevant to the road user.	O0: The operational damage leads to no impairment or non-perceivable impairment of a vehicle function.	P0: The privacy damage leads to no effect or, negligible consequences or is irrelevant to the road user.	S0: No injuries

Criteria for OEM impact rating

Impact rating	Financial	Operational	Privacy
Severe	F3: The financial damage leads to catastrophic consequences which the affected stakeholder might not overcome.	O3: The operational damage leads to a service not working, from non-intended operation up to the service being non-operational.	"P3: The privacy damage leads to significant or even irreversible impact to the service user OR the number of affected PII principals is very high. "
Major	F2: The financial damage leads to substantial consequences which the affected stakeholder will be able to overcome.	O2: The operational damage leads to the loss of a Service function.	"P2: The privacy damage leads to serious impact to the service user OR the number of affected PII principals is high. "
Moderate	F1: The financial damage leads to inconvenient consequences which the affected stakeholder will be able to overcome with limited resources.	O1: The operational damage leads to partial degradation of a service function or performance.	"P1: The privacy damage leads to significant inconveniences to the service user AND the number of affected PII principals is limited. "
Negligible	F0: The financial damage leads to no effect, negligible consequences or is irrelevant to the stakeholder.	O0: The operational damage leads to no effect or indiscernible degradation of a service function or performance.	P0: The privacy damage leads to no effect or can create few inconveniencies to the service user AND the number of affected PII principals is very limited.

Attack feasibility - Definitions and values

Attack Feasibility Category	Definition	Enumerate	Value
Elapsed Time	The total amount of time taken by an attacker to identify that a particular potential vulnerability may exist, to develop an attack method and to sustain the effort required in mounting the attack.	≤ 1 day	0
		≤ 1 week	1
		≤ 1 month	4
		≤ 6 months	10
		> 6 months	19
Specialist Expertise	The required level of generic knowledge of the underlying principles, product types or attack methods.	Layman	0
		Proficient	3
		Expert	6
		Multiple experts	8
Knowledge of the item	This refers to specific expertise in relation to the item under investigation. This is distinct from generic expertise, but not unrelated to it.	Public information	0
		Restricted information	3
		Confidential information	7
		Strictly confidential information	11
Window of opportunity	This has a relationship to the Elapsed time. Identification and exploitation of a vulnerability may require considerable amounts of access to a system that may increase the likelihood of detection of the attack. Some attack methods may require considerable effort off-line, and only brief access to the target to exploit. Access may also need to be continuous or over a number of sessions.	Unlimited	0
		Easy	1
		Moderate	4
		Difficult	10
Equipment	This refers to the equipment required to identify and exploit a vulnerability.	Standard	0
		Specialized	4
		Bespoke	7
		Multiple bespoke	9

SEC.1 Cybersecurity Requirements Elicitation

The purpose of the Cybersecurity Requirements Elicitation Process is to derive cybersecurity goals and requirements from the outcomes of risk management, and ensure consistency between the risk assessment, cybersecurity goals and cybersecurity requirements.

Process outcomes – As a result of successful implementation of this process

1. cybersecurity goals are defined,
2. cybersecurity requirements are derived from cybersecurity goals,
3. consistency and bidirectional traceability are established between cybersecurity requirements and goals and between the cybersecurity goals and the threat scenarios, and
4. the cybersecurity requirements are agreed and communicated to all affected parties.

Output work products

13-04 Communication record	[OUTCOME 4]	17-11 Software requirements specification	[OUTCOME 1, 2]
13-19 Review record	[OUTCOME 3]	17-12 System requirements specification	[OUTCOME 1, 2]
13-22 Traceability record	[OUTCOME 3]	17-51 Cybersecurity goals	[OUTCOME 1]
15-01 Analysis report	[OUTCOME 1, 2]		

SEC.1 with 4 Base practices

BP1 **Derive cybersecurity goals and cybersecurity requirements.** Derive cybersecurity goals for those threat scenarios, where the risk treatment decision requires risk reduction. Specify functional and non-functional cybersecurity requirements for the cybersecurity goals, including criteria for the achievement of the cybersecurity goals. [OUTCOME 1, 2]

1 *This includes the refinement of requirements during iterations of this process.*

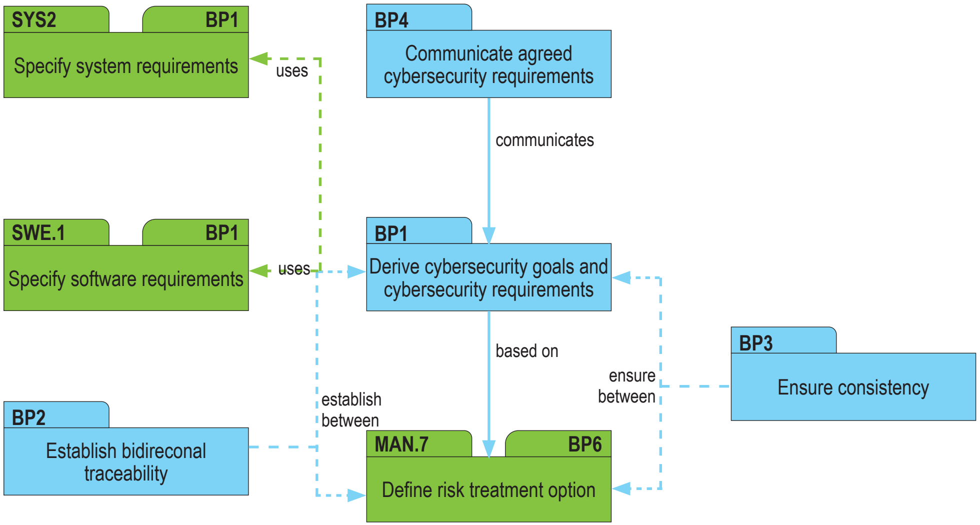
2 *This includes requirements for post-development phases which may include production, operation, maintenance and decommissioning.*

BP2 **Establish bidirectional traceability.** Establish bidirectional traceability between the cybersecurity requirements and the cybersecurity goals. Establish bidirectional traceability between the cybersecurity goals and the threat scenarios. [OUTCOME 3]

BP3 **Ensure consistency.** Ensure consistency between the cybersecurity requirements and the cybersecurity goals. Ensure consistency between the cybersecurity goals and the threat scenarios. [OUTCOME 3]

BP4 **Communicate agreed cybersecurity requirements.** Communicate agreed cybersecurity goals and cybersecurity requirements to all affected parties. [OUTCOME 4]

SEC.1 Consistency Diagram


SYS2
BP1
Specify system requirements
BP4
Communicate agreed cybersecurity requirements
uses
communicates
SWE.1
BP1
Specify software requirements
uses
BP1
Derive cybersecurity goals and cybersecurity requirements
based on
ensure between
BP3
Ensure consistency
establish between
BP2
Establish bidireconal traceability
MAN.7
BP6
Define risk treatment option

SEC.2 Cybersecurity Implementation

The purpose of the Cybersecurity Implementation Process is to allocate the cybersecurity requirements to the elements of the system and software and ensure they are implemented.

Process outcomes – As a result of successful implementation of this process

1. architectural design is refined,
2. cybersecurity requirements are allocated to elements of the architectural design,
3. appropriate cybersecurity controls are selected,
4. vulnerabilities are analyzed,
5. detailed design is refined,
6. software units are developed,
7. consistency and bidirectional traceability are established between architectural design and detailed design, and
8. the cybersecurity risk treatment implementation is agreed upon and communicated to all affected parties.

Output work products

04-04 Software architectural design	[OUTCOME 1]	13-19 Review record	[OUTCOME 7]
04-05 Software detailed design	[OUTCOME 5]	13-22 Traceability record	[OUTCOME 2, 7]
04-06 System architectural design	[OUTCOME 1]	15-50 Vulnerability analysis report	[OUTCOME 4]
11-05 Software unit	[OUTCOME 6]	17-52 Cybersecurity controls	[OUTCOME 3]
13-04 Communication record	[OUTCOME 8]		

SEC.2 with 10 Base practices

BP1 **Refine the details of the architectural design.** The architectural design is refined based on cybersecurity goals and cybersecurity requirements. [OUTCOME 1]

1 *Refinement could be on system and software level architecture.*

2 *Refinement here means to add, adapt or rework elements of the architecture.*

BP2 **Allocate cybersecurity requirements.** Allocate the cybersecurity requirements to one or more elements of the architectural design. [OUTCOME 2]

3 *Cybersecurity requirements could be on system and software level.*

BP3 **Select cybersecurity controls.** Select appropriate cybersecurity controls to achieve or support the cybersecurity requirements. [OUTCOME 3]

4 *Typically, cybersecurity controls are technical or other solutions to avoid, detect, counteract or mitigate cybersecurity risks.*

BP4 **Refine interfaces.** Refine and describe cybersecurity related interfaces between the elements of the architectural design and operating environment. [OUTCOME 1]

BP5 **Analyze architectural design.** Analyze the architectural design to identify and analyze vulnerabilities. [OUTCOME 4]

BP6 **Refine the details of the detailed design**. The detailed design is refined based on architectural design. [OUTCOME 5]

5 *Refinement here means toadd, adapt or rework components of the detailed design.*

SEC.2 with 10 Base practices

BP7 **Develop software units.** Implement the software using appropriate modeling or programming languages. [OUTCOME 6]

6 *Criteria for appropriate modeling and programming languages for cybersecurity can include the use of language subsets, enforcement of strong typing and/or the use of defensive implementation techniques.*

7 *Example to cover the defined criteria above could be the use of a coding guideline or an appropriate development environment.*

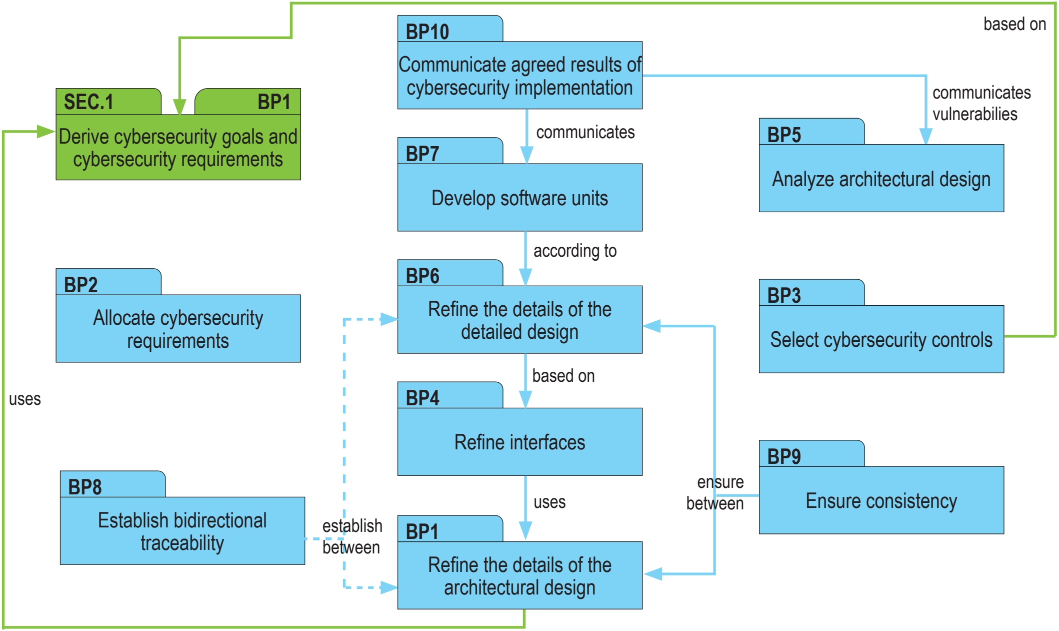
BP8 **Establish bidirectional traceability.** Establish bidirectional traceability between the refined architectural design and the detailed design. [OUTCOME 2, 7]

BP9 **Ensure consistency.** Ensure consistency between the refined architectural design and the detailed design. [OUTCOME 7]

BP10 **Communicate agreed results of cybersecurity implementation.** Communicate the agreed results of the cybersecurity implementation to all affected parties including stakeholders from post-development phases. [OUTCOME 8]

8 *The communicated contents may include both results of the cybersecurity implementation and vulnerabilities identified within the architectural design analysis.*

SEC2. Consistency Diagram


based on
SEC.1
BP1
Derive cybersecurity goals and cybersecurity requirements
BP10
Communicate agreed results of cybersecurity implementation
communicates vulnerabilies
communicates
BP7
Develop software units
BP5
Analyze architectural design
according to
BP2
Allocate cybersecurity requirements
BP6
Refine the details of the detailed design
BP3
Select cybersecurity controls
uses
based on
BP4
Refine interfaces
BP9
Ensure consistency
ensure between
uses
BP8
Establish bidirectional traceability
establish between
BP1
Refine the details of the architectural design

SEC.3 Risk Treatment Verification

The purpose of the Risk Treatment Verification Process is to confirm that the implementation of the design and integration of the components comply with the cybersecurity requirements, the refined architectural design and detailed design.

Process outcomes – As a result of successful implementation of this process

1. a risk treatment verification and integration strategy are developed, implemented, and maintained,
2. a specification for risk treatment verification is developed according to the risk treatment verification strategy suitable to provide evidence of compliance in implementing cybersecurity requirements as well as the refined architectural and detailed design,
3. identified work products are verified according to the risk treatment verification strategy for risk treatment verification. The implementation of the design and the integration of the components is tested using the defined test cases. Verification and test results are recorded,
4. bidirectional traceability between the cybersecurity requirements and risk treatment verification specification (including test cases), and bidirectional traceability between the refined architectural design (including detailed design) and the risk treatment verification specification (including test cases), and between the test cases included in the risk treatment verification specification, and verification results is established,
5. consistency between the cybersecurity requirements and risk treatment verification specification (including test cases) and consistency between the refined architectural design (including detailed design) and the risk treatment verification specification (including test cases) is established, and
6. results of the verification are summarized and communicated to all affected parties.

Output work products

08-50 Test specification	[OUTCOME 2]	13-22 Traceability record	[OUTCOME 4]
08-52 Test plan	[OUTCOME 1]	13-25 Verification results	[OUTCOME 3, 6]
13-04 Communication record	[OUTCOME 6]	13-50 Test result	[OUTCOME 3, 6]
13-19 Review record	[OUTCOME 3, 5]	19-10 Verification strategy	[OUTCOME 1]

BP1 **Develop a risk treatment verification and integration strategy.** Develop and implement a risk treatment verification and integration strategy, including a regression strategy. This contains:

- activities with associated methods, techniques and tools,
- work products or processes under verification,
- degree of independence for verification for performing these activities, and
- verification criteria. [OUTCOME 1]

1 *The risk treatment verification may provide objective evidence that the outputs of a particular phase of the system and software development lifecycle (e.g., requirements, design, implementation, testing) meet the specified requirements for that phase.*

2 *The risk treatment verification strategy may include*

- *requirements-based testing and interface testing on system and software level,*
- *check for any unspecified functionalities,*
- *resource consumption evaluation,*
- *control flow and data flow verification, and*
- *static analysis; for software: static code analysis e.g. industry recognized security-focused coding standards.*

3 *The risk treatment verification methods and techniques may include*

- *network tests simulating attacks (non-authorized commands, signals with wrong hash key, flooding the connection with messages, etc.), and*
- *simulating brute force attacks.*

4 *The risk treatment verification methods and techniques may also include audits, inspections, peer reviews, walkthroughs, code reviews, and other techniques.*

BP2 **Develop specification for risk treatment verification.** Develop the specification for risk treatment verification (including test cases) according to the risk treatment verification strategy. It shall be suitable to provide evidence of compliance of the implementation with the cybersecurity requirements and the refined architectural design and detailed design. [OUTCOME 2]

5 *Methods of deriving test cases may include*

- *analysis of requirements,*
- *generation and analysis of equivalence classes,*
- *boundary values analysis, and/or*
- *error guessing based on knowledge or experience.*

BP3 **Perform verification activities.** Verify identified work products according to the specified strategy in order to confirm that the work products meet their specified requirements.

Test the implementation of the design and component integration according to the risk treatment verification specification.

Record the risk treatment verification results and logs. [OUTCOME 3]

BP4 **Establish bidirectional traceability.** Establish bidirectional traceability between the cybersecurity requirements and risk treatment verification specification, including test cases comprised in the risk treatment verification specification.

Establish bidirectional traceability between the refined architectural design, detailed design, software units and the risk treatment verification specification.

Establish bidirectional traceability between the test cases included in the risk treatment verification specification, and verification results. [OUTCOME 4]

6 *Bidirectional traceability supports coverage, consistency, and impact analysis.*

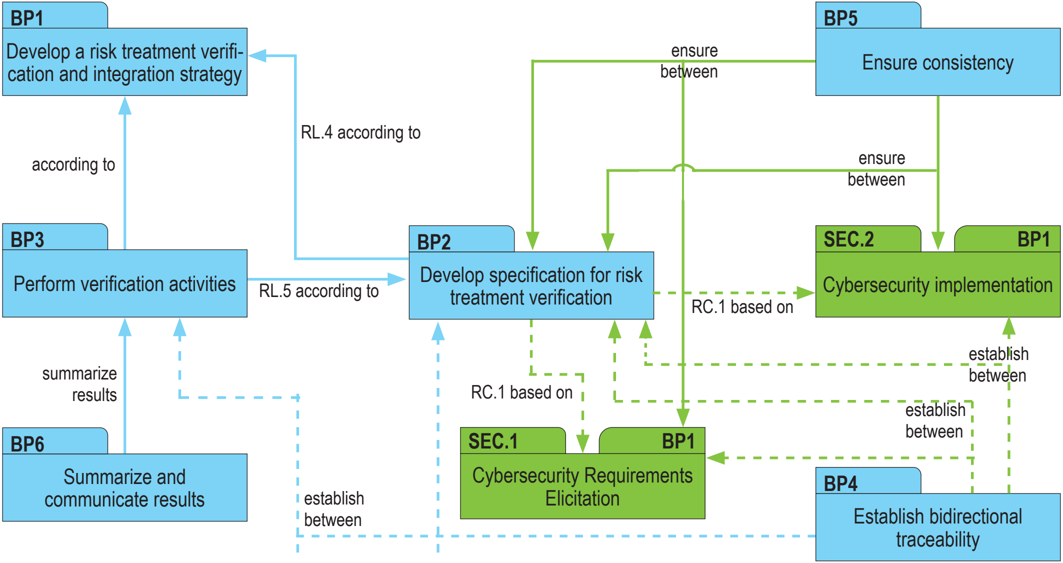
BP5 **Ensure consistency.** Ensure consistency between the cybersecurity requirements and the risk treatment verification specification, including test cases comprised in the risk treatment verification specification.
Ensure consistency between the refined architectural and detailed design and the risk treatment verification specification. [OUTCOME 5]

7 *Consistency is supported by bidirectional traceability and can be demonstrated by review records.*

BP6 **Summarize and communicate results.** Summarize the risk treatment verification results and communicate them to all affected parties. [OUTCOME 6]

8 *Providing all necessary information from the risk treatment verification execution in a summary enables other parties to judge the consequences.*

SEC3. Consistency Diagram


BP1
Develop a risk treatment verification and integration strategy
BP5
Ensure consistency
ensure between
RL.4 according to
according to
ensure between
BP3
Perform verification activities
BP2
Develop specification for risk treatment verification
SEC.2
BP1
Cybersecurity implementation
RL.5 according to
RC.1 based on
establish between
summarize results
RC.1 based on
establish between
BP6
Summarize and communicate results
SEC.1
BP1
Cybersecurity Requirements Elicitation
BP4
Establish bidirectional traceability
establish between

SEC.4 Risk Treatment Validation

The purpose of the Risk Treatment Validation Process is to confirm that the integrated system achieves the associated cybersecurity goals.

Process outcomes – As a result of successful implementation of this process

1. a risk treatment validation strategy is developed, implemented and agreed upon with relevant stakeholders and maintained suitably to provide evidence that the implementation achieves the associated cybersecurity goals,
2. the implemented design and integrated components are validated according to the defined risk treatment validation strategy,
3. validation activities are documented and the results are recorded,
4. bidirectional traceability between the cybersecurity goals, risk treatment validation specification and validation results is established,
5. consistency between the cybersecurity goals and the risk treatment validation specification is established, and
6. results of the validation are summarized and communicated to all affected parties.

Output work products

08-50 Test specification	[OUTCOME 2]	13-22 Traceability record	[OUTCOME 4]
13-04 Communication record	[OUTCOME 6]	13-24 Validation results	[OUTCOME 3]
13-19 Review record	[OUTCOME 2, 5]	19-11 Validation strategy	[OUTCOME 1]

SEC.4 with 6 Base practices

BP1 **Develop a risk treatment validation strategy.** Develop and implement a validation strategy. [OUTCOME 1]

1 *Risk treatment validation methods and techniques typically include cybersecurity-relevant methods to detect unidentified vulnerabilities (e.g., penetration testing).*

2 *Risk treatment validation examines whether the associated cybersecurity goals are achieved.*

SEC.4 with 6 Base practices

BP2 **Develop specification for risk treatment validation.** Develop the specification for risk treatment validation (including test cases) according to the risk treatment validation strategy. It shall be suitable to provide evidence of achievement of the associated cybersecurity goals. [OUTCOME 2]

3 *Methods of deriving test cases may include*

- *analysis of requirements,*
- *generation and analysis of equivalence classes,*
- *boundary values analysis, and/or*
- *error guessing based on knowledge or experience.*

BP3 **Perform and document risk treatment validation activities.** Validate the implemented design and the integrated components according to the defined risk treatment validation strategy.

The risk treatment validation activities are documented, and the results are recorded. [OUTCOME 2, 3]

4 *See SUP.9 for handling of non-conformances and vulnerabilities.*

BP4 **Establish bidirectional traceability.** Establish bidirectional traceability between the cybersecurity goals and the risk treatment validation specification. Establish bidirectional traceability between the risk treatment validation specification and the validation results. [OUTCOME 4]

5 *Bidirectional traceability supports coverage, consistency and impact analysis.*

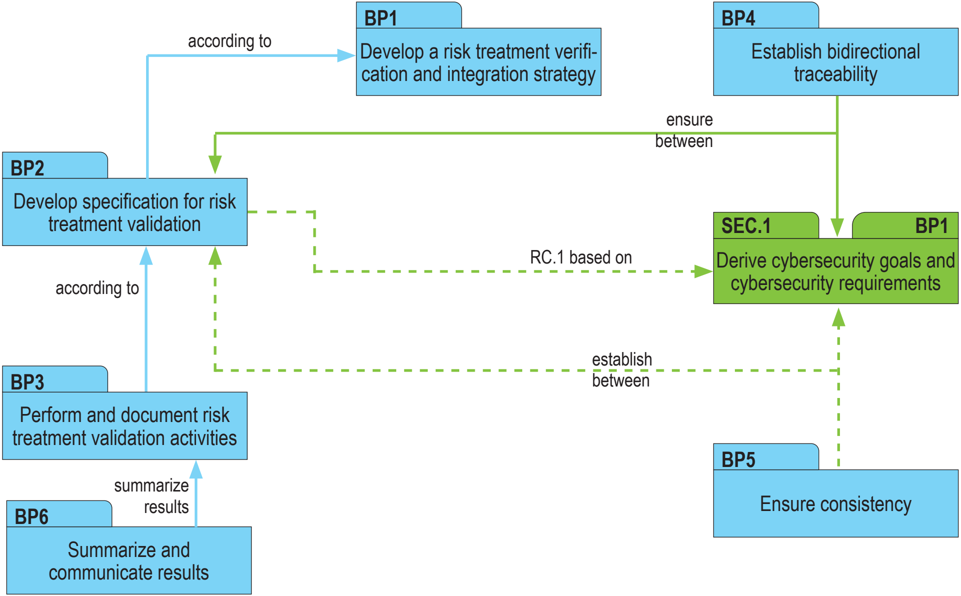
BP5 **Ensure consistency.** Ensure consistency between the cybersecurity goals and the risk treatment validation specification. [OUTCOME 5]

6 *Consistency is supported by bidirectional traceability and can be demonstrated by review records.*

BP6 **Summarize and communicate results.** Summarize the risk treatment validation results and communicate them to all affected parties. [OUTCOME 3, 6]

7 *This includes typically information from the risk treatment validation activities and important findings concerning additional vulnerabilities that enables other parties to judge the consequences.*

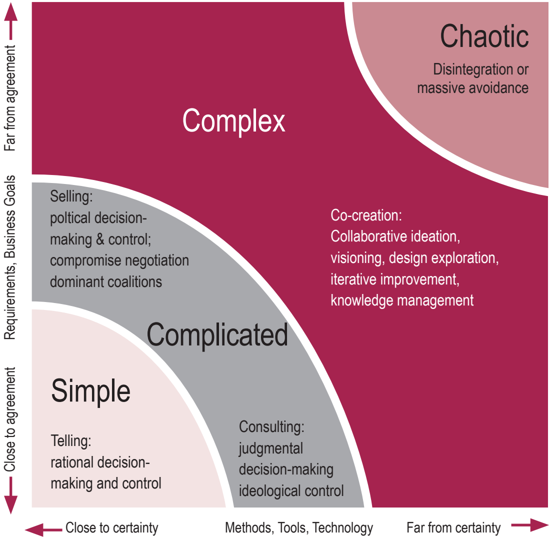
SEC4. Consistency Diagram


BP1
Develop a risk treatment verification and integration strategy
BP4
Establish bidirectional traceability
according to
ensure between
BP2
Develop specification for risk treatment validation
SEC.1
BP1
Derive cybersecurity goals and cybersecurity requirements
RC.1 based on
according to
establish between
BP3
Perform and document risk treatment validation activities
summarize results
BP6
Summarize and communicate results
BP5
Ensure consistency

AGILE SPICE® Version 1.3

Agile SPICE® is a bridge to Automotive SPICE®

- Growing complexity and faster changes during development drive increased use of agile approaches.
- OEMs ensure the overall quality and safety/security of those products, leading to increased pressure to show SPICE capabilities.
- SPICE as is can be applied and interpreted for any kind of development.
- However, both agile organizations and those in an agile transition struggle in implementing and interpreting SPICE.
- There is a need to reduce misunderstandings and help to interpret the terminology for both sides.
- This bridge shall …
  - reduce the variation in interpretation of assessors,
  - increase the acceptance of SPICE in the agile community.



Why Agile SPICE®?

Viewpoint from an agile organization

- Driving acceptance of process capability requirements in agile environments especially in automotive industry
- Practices describing „what" is expected not the „how"
  - Avoiding discussions about specific agile approaches
  - Ensuring expected process capability by automotive industry
- Helping in both implementing agile good practices and achieving Automotive SPICE expectations at the same time
- Resolving past misunderstandings of how to implement and assess relevant Base Practices in agile environments
- Providing a "SPICE bridge" to agile approaches

Viewpoint a conventional organization

- How to keep existing capability while moving toward agile approaches?
- Learning about agile terminology
- Avoiding typical pitfalls in an agile transformation (e.g., how to ensure a minimum of governance)

Viewpoint of an assessor

- Very broad interpretations of existing SPICE processes in agile organizations
- Lack of common terminology for assessing agile
- Need a mapping of ratings between classical and agile approaches
- Requires comparability of ratings of agile and classical approaches to development work

Wishful thinking in the long run: Use of Agile SPICE® as a bridge to other PAMs.

Agile SPICE® Assumptions

- Providing agile practices for the what and not the how
- Condensing existing best agile practices without favoring a specific one (e.g., Scrum, Kanban, SAFe, LeSS, Nexus, Scrum of Scrums….)
- Practice is within the guiding principles of the agile manifesto and our definition of agility
- Bridging existing PAM processes and outcomes (do their intentions match?)
- Retaining existing process attributes / generic practices (capability dimension) on CL2-5

> **Agility is the timely adaption of an organization (or team) to an ever-changing environment while continuously delivering value to their customers at sustainable pace.**
>
> Definition by Kugler Maag Cie and Knüvener Mackert GmbH.

Agile Principles for Automotive*

1. Our highest priority is to satisfy our customers through early and continuous delivery of valuable and usable system functions.

2. Requirement changes are mastered, prioritized and systematically integrated into our continuous development work. Agile processes make use of changes to the competitive advantage of the customer.

3. We deliver regularly usable and enhanced system features, preferring shorter time periods within a few weeks or months.

4. Experts from all domains should collaborate intensively during product development.

5. We organize the product development around motivated individuals. We design an environment and support to achieve maximum value. In doing so, we trust that the individuals do their jobs independently and in the best possible way.

6. The most efficient and effective way to communicate information to and within a development team is face-to-face.

7. Usable and extended system functions are the most important measure of progress. Agile processes promote sustainable development.

8. Clients, developers and users should be able to maintain a steady pace for an unlimited period of time.

9. Continuous attention to technical excellence and good design promotes agility.

10. Simplicity - the art of maximizing the amount of work not done - is essential.

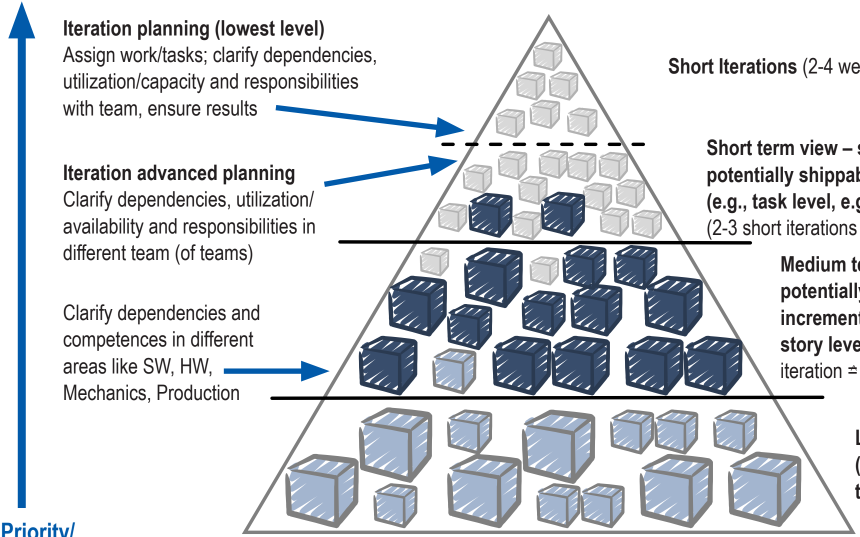
11. The best architectures, requirements and designs are created by self-organized teams.

12. At regular intervals, the team reflects on how it can become more effective and adjusts its behavior accordingly

* Adapted by Kugler Maag Cie

Agile Work Management – Assumptions

Granularity of planning



Iteration planning (lowest level)
Assign work/tasks; clarify dependencies, utilization/capacity and responsibilities with team, ensure results

Iteration advanced planning
Clarify dependencies, utilization/availability and responsibilities in different team (of teams)

Clarify dependencies and competences in different areas like SW, HW, Mechanics, Production

Short Iterations (2-4 weeks)

Short term view – small potentially shippable work increment (e.g., task level, e.g. sprints) (2-3 short iterations ≙ 4-10 weeks)

Medium term view – larger potentially shippable work increment(e.g., epic/feature/story level, e.g. releases) (1 large iteration ≙ 4-6 smaller iterations)

Long term view (e.g., vision/strategic themes level)

Picture by Kugler Maag Cie GmbH 2019

AGL.1 Agile Work Management

The purpose of Agile Work Management is to collaboratively manage the work of a team (of teams) to develop iteratively within work boundaries and product vision and goals to generate business and customer value.

Purpose outcomes

As a result of successful implementation of this process:

1) the product vision and goals are jointly defined and kept up to date.
2) the right set of competencies and adequate resources are planned and adapted as needed.
3) a work approach is defined and continuously improved.
4) dependencies, interfaces, stakeholders, and their commitment are planned for and monitored.
5) the needed infrastructure and work environment is planned and operationalized.
6) the feasibility is evaluated for critical elements.
7) the backlog is estimated and prioritized as basis for both for short term and long-term planning.
8) the content of iterations and (potentially) shippable increment is planned and realized.
9) progress and status of work completion is made transparent and impacts on product vision and goals are managed to ensure consistency.
10) impediments are identified and resolved when planned work or product vision and goals are significantly affected; recurrence of selected issues is prevented.

AGL.1 with 11 base practices

AGL.1.BP1: Identify demand and work boundaries. Identify the customer demand and work boundaries collaborating closely with stakeholders. Derive the product vision and goals to achieve business and customer value. Keep demand, boundaries as well as product vision and goals up to date. [OUTCOME 1]

Note 1: Collaboration can be internal and external as well as across organizations, e.g., within a program, product line or organization, and together with (multiple) customers and suppliers.

Note 2: [Def] A customer demand is a recorded customer need or statement on the problem to be solved.

Note 3: [Def] A product vision and goals define the product capabilities potentially leading to business value. They typically run across iterations and address a mid and long-term perspective.

Note 4: Product vision and goals are typically input to requirements engineering activities.

Note 5: A customer can be a stakeholder within (e.g., product management) or external to the organization (an individual customer).

Note 6: Boundaries can include system context, solution space, link to feasibility, organizational constraints and business objectives, platform, and product line constraints as well as regulatory requirements.

Note 7: Ensure agreement with stakeholders on change mechanism for changes to the demand and boundaries.

AGL.1.BP2: Build team. Form, empower and enable a team (of teams) fitting to the vision and work boundaries. Ensure the right skill and experience set within each team (of teams). [OUTCOME 2]

Note 8: Self-organization is a guiding principle for founding an agile team (of teams) and their efficient performance.

Note 9: Teams of teams are used for larger scopes using a fitting agile scaling approach for teams.

Note 10: In scaling approaches all teams have to agree on ground rules to ensure a common purpose.

Note 11: Best performing teams are typically stable in composition and are working at a sustainable pace.

Note 12: Team building includes defined ground rules for collaboration within a team (of teams).

Note 13: [Def] Empowerment is a conscious delegation of decision authority to a team (of teams).

Note 14: Enablement ensures the right skills and experience set within a team (of teams) including needed training.

Note 15: Responsibility for technical releases and product liability must be defined.

AGL.1.BP3: Define work approach. Establish, record, and keep the work approach of the team (of teams) up to date. Ensure that the work approach reflects the given level of complexity, fulfils the work boundaries, and defines team policies, iteration cycles, agile events, artefacts, and roles. [OUTCOME 3]

Note 16: Typically, agility is based on the pull principle as well as on limiting the amount of work in progress and on reducing multi-tasking.

Note 17: [Def] The Definition of Done (DoD) is a build-in quality measure containing a minimum, agreed to, and recorded set of criteria upon which a task or increment is considered to be done.

Note 18: [Def] The Definition of Ready (DoR) is a build-in quality measure containing a minimum, agreed to, and recorded set of criteria upon which a content is considered as ready to be pulled into an iteration.

Note 19: Typical team (of teams) policies include Definitions of Ready and Done based on quality criteria.

Note 20: Stakeholder requirements for DoD and DoR include quality requirements, and possibly regulatory requirements imposed by safety or security considerations.

Note 21: An agile approach often encompasses a whole product lifecycle and a clear alignment with the customer work approach. (see also Note 28)

Note 22: Typically, there is an agreement in place with the customer on the chosen agile approach and how delivered work is accepted for each shippable increment.

Note 23: If an iteration-based approach has been chosen, the types of iterations and their respective length must be in line with the chosen agile approach and is influenced by the customer and technical feasibility as well as safety and security aspects.

AGL.1.BP4: Manage stakeholder and interfaces. Stakeholders, interfaces, and dependencies within and outside the team (of teams) are identified, planned for, recorded, and involved. [OUTCOME 4]

Note 24: Dependencies and interfaces include both technical and organizational ones:

- *technical: e.g., among major architectural elements or safety/security related activities/roles;*
- *organizational: e.g., to other teams/stakeholders and between different iteration cycles;*
- *supplied products or services or tools: e.g., from suppliers or from customers.*

Note 25: Critical dependencies should be identified from the start, their sequence identified, and their status tracked across multiple iteration cycles. The elements of the backlog typically contain a path to address them.

Note 26: Stakeholder involvement includes tracking the commitment of involved and affected parties as well as ensuring the active involvement of stakeholders.

Note 27: Maintain communication records for all interfaces. Continuously verify the necessity of all interfaces and remove unnecessary interfaces.

Note 28: In case of mixed agile and traditional approaches, e.g., on system and other discipline level, clearly define how different approaches integrate and synchronize.

Note 29: Agile practitioners are typically organized in networks or communities (e.g., Community of Practice, tribe, circle).

AGL.1.BP5: Plan and provide infrastructure. Identify, plan for, provide, and keep the needed work and development infrastructure up to date. [OUTCOME 5]

Note 30: Infrastructure typically includes but is not limited to an engineering tool chain, ticketing and backlog database, verification and integration environment (i.e., for continuous integration and deployment), communication and collaboration tools, physical and online workspaces, work environment, licensing, etc.

Note 31: Align infrastructure needed for the interfaces to all stakeholders within and outside the organization.

Note 32: Agile approaches typically focus on as much automation of processes as possible, e.g., for ticketing, development, continuous integration and deployment, and transparency of status.

AGL.1.BP6: Evaluate feasibility. Evaluate and act on the feasibility of the product vision and critical elements. [OUTCOME 6]

Note 33: Criticality is related to fundamental and conscious decisions during the complete product lifecycle.

Note 34: Critical elements address contents related to e.g., product risks, architectural challenges, key features, key technology decisions, key supplied features, relevant safety, and security contents.

Note 35: The identification, prioritization and monitoring of critical elements is a continuous activity.

Note 36: Critical elements should be addressed by potentially shippable increments.

Note 37: Many agile approaches are built for adapting to uncertainty in early stages of development (see MAN.5 for risk management).

AGL.1.BP7: Understand backlog and estimate work. Perform a high-level estimate on backlog items. Estimate and prioritize work for the upcoming iteration cycles to ensure a common understanding of the work content within the team (of teams). Refine and adapt estimates to continuously improve team collaboration and the quality of the product backlog. [OUTCOME 7]

Note 38: A high-level estimate helps to gauge the overall feasibility of product vision and goals.

Note 39: Work estimation requires sufficiently small backlog items fitting within the upcoming iteration cycle, i.e., satisfying a DoR and considering (potential) impediments.

Note 40: The selection of work items for estimation should be driven by priority and business value.

Note 41: Typically, agile estimation is based on experts discussing estimates to achieve a shared understanding and consensus.

Note 42: Appropriate estimation method and recorded estimation data from previous estimates should be used, i.e., based on estimating complexity, relative size via a reference story, historical data, or analogy.

Note 43: Usually, the estimation method is evaluated and refined on a regular basis.

AGL.1.BP8: Plan Work. Plan and realize the content of the upcoming iteration cycles based on estimates, team capacity, and definitions of ready and done. [OUTCOME 8]

Note 44: Typically, there are different levels of planning related to iteration cycles depending on the complexity of the product and team of teams (e.g., sprints and releases).

Note 45: In general, work planning should consider actual team member capacity, availability, and velocity as well as the recorded work approach, dependencies, and feasibility.

Note 46: Planning is usually based on selecting (pulling) content from the prioritized product backlog meeting DoR criteria, estimates, overall product vision and goals.

Note 47: Typically, planned work for each level is the result of a planning workshop for the upcoming iteration cycles by focusing on the near and known future.

Note 48: Typically, the team decides on the work distribution mechanisms within the team (of teams), e.g., principles like push, pull, round-robin.

Note 49: Typically, work is planned to avoid multitasking and by limiting the amount of work in progress, i.e., interruption and segmentation of activities is reduced to a minimum.

Note 50: Typically, one or multiple iterations form a potentially shippable increment. Being potentially shippable does not mean the results have to be delivered to customers. Shipping is a recorded business decision and should provide customer value and feedback.

Note 51: The term “sample” is often used in automotive for a larger shippable increment containing results of multiple iteration cycles and disciplines.

AGL.1.BP9: Inspect and visualize. Inspect, measure, and visualize the status and progress of work. Identify impediments towards the consistency of planning and communicate impacts on product vision and goals. [OUTCOME 9]

Note 52: Intervals and events to inspect the work status depend on the complexity of the product and team of teams (e.g., small iterations on team level and larger increments on team of teams or program level).

Note 53: Typically, progress and status of work of a team is visible and reviewed daily.

Note 54: Measurements are typically related to team (of teams) capacity, velocity, and rate of completion based on agreed to Definition of Ready and Definition of Done.

Note 55: The status is usually visible on demand at any time by physical or online team boards or charts.

Note 56: Transparency supports achieving consistency among current and overall planning, product backlog, team skills and team capacity to ensure a sustainable pace.

AGL.1. BP10: Manage impediments. Collect, monitor, and resolve impediments related to work management within and across iteration cycles by teams (of teams). [OUTCOME 10]

Note 57: [Def] An impediment is any issue keeping the team (of teams) from achieving either the planned work within an iteration *cycle or the agreed to product vision and goals. Risks are regarded as potential impediments.*

Note 58: Managing impediments supports consistency among current planning and vision, product backlog, and team capacity to ensure a sustainable pace.

Note 59: Impediments related to work management are typically identified by the team (of teams) while reviewing the progress of work. Addressing them helps in adapting work planning as well as product vision and goals.

Note 60: Work resulting from impediments is typically managed as backlog item.

Note 61: In general, impediments are a type of problems and handled according to a problem management strategy; see SUP.9. Impediments related to quality objectives are identified in SUP.1. Potential impediments are handled in MAN.5 as risks.

AGL.1. BP11: Improve work approach. Inspect and improve the work approach based on short learning cycles within the team (of teams). [OUTCOME 3]

Note 62: The way of working in the team (of teams) is regularly discussed to identify improvements to the recorded work approach.

Note 63: Typically, current processes, interfaces, and work products are inspected as part of the work approach and adapted at least per iteration cycle and event driven.

Note 64: Improvement work from identified improvements is typically managed as part of the backlog.

Output work products

05-AGL32	Product visions and goals → [OUTCOME 1, 9, 10]
05-AGL28	Demand statement → [OUTCOME 1]
14-AGL02	Product Backlog → [OUTCOME 1, 4, 5, 6, 7, 8, 9, 10]
14-AGL30	Team setup → [OUTCOME 2]
08-AGL10	Work approach → [OUTCOME 2, 3, 4, 11]
08-AGL31	Training plan → [OUTCOME 2]
13-AGL36	Training records → [OUTCOME 2]
05-AGL07	Definition of Done → [OUTCOME 3, 11]
05-AGL08	Definition of Ready → [OUTCOME 3, 11]
08-AGL33	Work environment → [OUTCOME 5]
11-AGL06	Potentially shippable increment → [OUTCOME 6]
13-AGL11	Impediment record→ [OUTCOME 6]
14-AGL12	Impediment backlog → [OUTCOME 6, 10]
13-AGL13	Task → [OUTCOME 7, 8, 9, 10, 11]
13-AGL37	Progress record → [OUTCOME 9]

AGL.2 Partner Collaboration Management

The purpose of the Partner Collaboration Management Process is to achieve common project goals in collaboration with a partner.

Purpose outcomes

As a result of successful implementation of this process:

1) information is shared as agreed between partners.
2) the collaboration model is documented in the collaboration agreement and up to date.
3) transparency is established by continuous alignment and synchronization between partners.
4) risks and impediments are managed collaboratively.
5) technical approach and content are in line with shared product vision.
6) progress of all partners is continuously and jointly inspected.

AGL.2 with 6 base practices

AGL.2.BP1: Establish a collaboration model. Establish and maintain an agreement on how partners collaborate, how and how often they align and synchronize, and which information is shared among the partners. [OUTCOME 1, 2, 3]

Note 1: Ensure that the collaboration model is based on a common understanding regarding roles, meetings, and artifacts. The collaboration model could be included in the work approach.

Note 2: While cooperation is a loosely coupled group work, agile collaboration is characterized by a teamwork with a common goal, continuous alignment, and synchronization.

Note 3: A collaboration agreement typically includes e.g. communication mechanisms, meeting structure, technical and organizational interfaces, infrastructure, roles including skills and responsibility split, and ownership of artifacts.

Note 4: Collaboration in an agile environment requires a common Definition of Ready and Definition of Done. It is up to the team (of teams) to complement the agreed minimum set by own criteria.

AGL.2.BP2: Share information. Establish communication to share all agreed information between partners. [OUTCOME 1, 2, 3]

Note 5: In agile context, a suitable meeting structure may encompass queue replenishment, product backlog refinement, iteration planning, stand-up, iteration review, retrospective. These collaborative activities highly depend on self-organization of the team (of teams).

Note 6: Information to be shared may include technical content, acceptance criteria, status, risks, impediments, and open points.

AGL.2.BP3: Establish a technical approach. A technical approach is jointly developed, agreed upon and kept up to date on a regular basis in line with the collaboration agreement. [OUTCOME 1, 3, 4, 5]

Note 7: Constraints, risks, assumptions, and open items are considered.

Note 8: The technical approach is aligned with the agile work approach (see AGL.1).

AGL.2.BP4: Review technical progress. Perform reviews on jointly developed technical content regarding acceptance criteria and Definition of Done on regular basis as agreed in the collaboration agreement. [OUTCOME 1, 2, 3, 5]

Note 9: If rework is required, prioritize, plan and track work packages to closure. Ensure that the joint technical content is in line with shared product vision.

Note 10: Acceptance criteria are applicable to individual work packages.

AGL.2.BP5: Inspect work progress. Measure, visualize, and inspect joint progress continuously regarding quality, effort, cost, and schedule against plan. Identify technical and business risks, impediments, and deviations from desired status collaboratively. [OUTCOME 1, 3, 4, 5, 6]

Note 12: Quality, effort, cost, and schedule can be monitored by metrics and KPIs agreed by the partners.

Note 13: Inspect is the ability of a team (of teams) to detect deviations from desired status.

AGL.2.BP6: Act on risks, impediments, and deviations. Mitigate technical and business risks, impediments, and deviations from desired status collaboratively and adapt accordingly. [OUTCOME 1, 4]

Note 14: Prioritize, plan, and track open points to closure. In case of deviations from the desired status, jointly identify root causes, introduce countermeasures, and prevent reoccurrence.

Note 15: Adaption is the ability of a team (of teams) to timely adjust to an everchanging environment.

Output work products

02-AGL14 Collaboration agreement → [OUTCOME 1, 2, 6]
05-AGL15 Product vision and goals → [OUTCOME 1, 2, 3, 4, 6]
05-AGL08 Definition of Ready → [OUTCOME 2, 3]
05-AGL07 Definition of Done → [OUTCOME 1, 2, 3, 4, 6]
14-AGL02 Product Backlog → [OUTCOME 2, 3, 4, 6]

14-AGL18 Iteration Backlog → [OUTCOME 3, 4]
15-09 Risk list → [OUTCOME 2, 3, 5, 6]
14-AGL12 Impediment Backlog → [OUTCOME 2, 5, 6]
13-19 Review record → [OUTCOME 3, 4]
13-AGL37 Progress record → [OUTCOME 3, 4]

AGL.3 Agile Quality Assurance

The purpose of Agile Quality Assurance is to objectively and independently identify, track to closure, escalate and further prevent impediments affecting the achievement of quality objectives.

Purpose outcomes

As a result of successful implementation of this process:

1) quality objectives in line with work boundaries, vision, strategic themes, work approach and governance criteria are collaboratively identified.
2) a quality assurance strategy to identify and independently communicate impediments in achieving quality objectives is jointly defined and kept up to date.
3) conformance of actual work products and processes to agreed work approach and applicable governance criteria is ensured.
4) impediments affecting quality objectives are recorded and communicated to relevant stakeholders.
5) impediments affecting quality objectives are tracked to closure and their re-occurrence is prevented.
6) mechanisms and authority to escalate non-resolution of impediments within and beyond the agile team (of teams) are established and effective.

AGL.3 with 8 base practices

AGL.3.BP1: Identify quality objectives. Collaboratively identify and agree on quality objectives in line with work boundaries, product vision and goals, work approach and governance criteria. [OUTCOME 1]

Note 1: *Quality objectives*

- *encompass the work approach selected by the team (of teams) as well as the relevant governance processes of the organization,*
- *are agreed collaboratively among identified stakeholders,*
- *are achieved by the team (of teams) across the iterations.*

Note 2: *Quality objectives support the establishment of suitable work processes and quality characteristics for the work products.*

Note 3: Among other, criteria for contents definition, structure, review, approval of work products as well as any other criterion derived from standards are part of the quality objectives.

AGL.3.BP2: Define quality assurance strategy. Jointly define and record a quality assurance strategy to identify and independently communicate impediments in achieving quality objectives. [OUTCOME 2]

Note 4: The team (of teams) is jointly responsible for the quality of their processes and work products.

Note 5: The strategy encompasses the quality of work processes and developed products.

Note 6: The strategy includes the quality assurance of supplier deliverables.

Note 7: The strategy addresses methods to identify, track to closure, escalate and further prevent impediments affecting the achievement of quality objectives.

Note 8: The strategy includes the communication of impediments independently from the development team.

Note 9: Agile quality assurance may make use of the results of other processes such as e.g., iteration reviews and retrospectives, problem analysis, reports.

Note 10: The strategy is agreed to, recorded, and communicated.

AGL.3.BP3: Ensure conformance. Ensure the conformance of actual work products and processes as implemented by the team (of teams) to agreed work approach and applicable governance criteria as well as against quality objectives. [OUTCOME 3]

Note 11: Typically, criteria for the quality of processes and work product are defined, e.g., in a DoR and DoD.

Note 12: Quality assurance activities are planned, e.g., as part of the backlog and within iteration planning.

Note 13: Quality assurance checks objectively for each release that work products and processes are aligned with the work approach and the applicable governance criteria.

Note 14: The agile methodology followed by the team (of teams) is also verified.

AGL.3.BP4: Record impediments. Record impediments affecting the achievement of quality objectives and communicate them to relevant stakeholders. [OUTCOME 4]

Note 15: The term impediment is used here instead of non-conformance to emphasize the need to act upon non-adherence to agreed work approach and applicable governance criteria.

Note 16: Communicate regularly with the team (of teams) regarding impediments and desired performance. This can happen at various levels, during daily stand-ups, during iteration reviews and retrospectives, etc.

Note 17: The team (of teams) plans in the backlog the corrective actions necessary to resolve the recorded impediments.

Note 18: Communication can include metrics, risks, qualitative feedback, etc.

Note 19: Metrics relevant to process and product quality can be present. This can include (but not limited to) velocity forecasts, defect trends, increments released per gate, etc.

Note 20: Communication should be based on the perspective relevant for the recipient – team (of teams), management, customer.

AGL.3.BP5: Resolve impediments. Track impediments affecting the achievement of quality objectives to closure and prevent their re-occurrence. [OUTCOME 5]

Note 21: Ensure that impediments are correctly understood by relevant stakeholders.

Note 22: Ensure that resolving and closing of impediments are planned by the team (of teams).

Note 23: Track and verify the closure of impediments.

Note 24: Analyse and communicate root causes of re-occurring impediments.

Note 25: Support the team (of teams) to avoid impediment re-occurrence.

AGL.3.BP6: Establish escalation mechanisms. Establish mechanisms and authority to escalate non-resolution of impediments and related root causes within and beyond the team (of teams). [OUTCOME 6]

Note 26: The escalation path is known by each team member and those that contribute to the team.

Note 27: The escalation path begins with the team as the starting point. Typically, the path ends with the highest authority in the organisation.

Note 28: The escalation path is typically recorded e.g.by an organigram.

Note 29: The escalation path is accessible to anyone in the organisation.

Note 30: The impediment backlog is visible at least to anyone in the escalation path and the team (of teams).

AGL.3.BP7 Manage escalations. Ensure the resolution of escalated impediments and related root causes by adequate governance and decision-making bodies. [OUTCOME 6]

Note 31: Primarily, the team (of teams) is jointly responsible for resolving impediments and collaborates closely with the person who tracks their impediments to closure.

Note 32: Each member in the escalation path maintains a seamless communication to its lower and its upper partners in the path.

Note 33: The agile quality assurance responsible acts as 'escalation guard' and is empowered to involve higher authorities of the escalation path, if necessary.

Note 34: Any escalation rationale, decision, action, and effect are recorded.

AGL.3.BP8 Inspect and improve. Inspect and improve the quality assurance strategy based on short learning cycles within the team (of teams). [OUTCOME 1, 2]

Note 35: The quality strategy typically considers different organizational levels of development.

Note 36: Inspect and improve events take place regularly in a collaborative way with the team (of teams) and relevant stakeholders.

Note 37: The learning aspects consider multiple views, including management, teams, and customer.

Note 38: The improvements of quality strategy and/or objectives are consistent with the organisation strategy and reflect customer quality needs (e.g., recorded in DoD).

Note 39: The improvements and learnings are reflected and made visible, e.g., in retrospectives by maintaining DoD and DoR criteria.

Output work products

08-13 Quality plan → [OUTCOME 1, 2]
13-19 Review record → [OUTCOME 3]
14-AGL12 Impediment Backlog → [OUTCOME 4, 5, 6]
14-AGL02 Product Backlog → [OUTCOME 4, 5]
14-AGL24 Communication record → [OUTCOME 1, 4, 5, 6]
13-AGL26 Escalation record → [OUTCOME 6]

ORGANIZATION SPICE 3.0

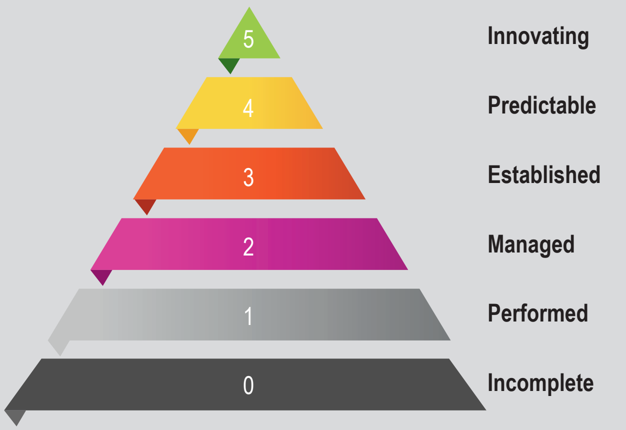
Organization SPICE

In addition to capability evaluations of single processes, the capability level of an entire organization may be evaluated. One refers to Organizational Maturity Levels (OML) in this case.

Project assessments dominate currently, but the interest in Organizational Maturity Assessments is growing because of the desire to reduce the effort needed for assessments.

Organizational assessments examine the entire company, including a majority of its projects. Ultimately, it is the organization that makes it possible for the employees in the projects to apply processes effectively.

These organizational assessments evaluate the capability and maturity of the company, to deliver quality systematically. The basis for this assessment is ISO / IEC 15504-7, which defines the concept of "Organizational Maturity Model". In assessments multiple process instances are investigated.

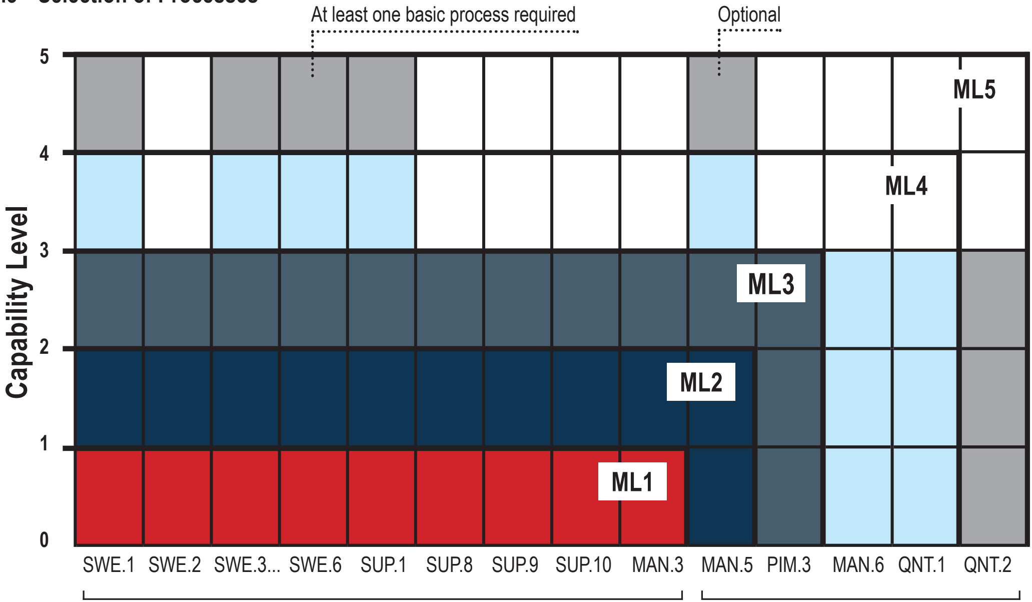


The ISO/IEC 15504-7 Organizational Maturity Levels

3.1 Maturity Level Definition

Maturity Level	Maturity Level Name	Maturity Level Description
Level 0	Incomplete	The organization does not demonstrate effective implementation of its processes to support the organization's business
Level 1	Performed	The organization demonstrates achievement of the purpose of the processes that are fundamental to support the organization's business.
Level 2	Managed	The organization demonstrates management of the processes that are fundamental to support the organization's business.
Level 3	Established	The organization demonstrates effective definition and deployment of processes and process standards that are fundamental to support the organization's business.
Level 4	Predictable	The organization demonstrates a quantitative understanding of relevant processes that are fundamental to achieving the organization's business goals, in order to establish consistent and predictable performance.
Level 5	Innovating	The organization demonstrates the ability to change and adapt the performance of the processes that are fundamental to achieving the organization's business goals in a systematically planned and predictable manner.

3.3 Selection of Processes



The intacs Maturity Level Concept using an example with Automotive SPICE PAM 3.1

3.5 Achievement of a Maturity Level

Maturity	Achievement
Level 1	• All analyzed process instances have achieved capability level 1.
Level 2	• All analyzed process instances have achieved capability level 2.
Level 3	• All analyzed process instances have achieved capability level 3.
Level 4	• All analyzed process instances have achieved capability level 3 and • all analyzed process instances with target capability level 4 have achieved capability level 4. • In the organization, processes with target capability level 4 contribute a significant amount of work results in the majority of projects.
Level 5	• All analyzed process instances have achieved capability level 3 and • all analyzed process instances with target capability level 4 have achieved capability level 4 and • all analyzed process instances with capability target level 5 have achieved capability level 5. • In the organization, processes with target capability level 5 contribute a significant amount of work results in the majority of projects.

PIM.1 Process Establishment

The purpose of the Process Establishment process is to establish the processes of the organization's management system [ISO9001, Clause 4.4.1] required to consistently provide products that meet customer and applicable statutory and regulatory requirements. [ISO9001, Clause 1.b]

Process outcomes – As a result of successful implementation of this process

1. A defined and maintained standard set of processes are established in line with the quality policy and quality objectives, along with an indication of each process's applicability;
2. The standard processes are defined, including their inputs, activities, out-comes, and further necessary information;
3. A strategy for tailoring the standard process for the product or service is developed in accordance with the needs of the project; and
4. Information and data related to the use of the standard process for specific projects exist and is maintained.

Output work products

09-00 Policy	[OUTCOME 1, 3]	03-06 Process performance data	[OUTCOME 4]
09-02 Quality policy	[OUTCOME 1, 3]	13-18 Quality record	[OUTCOME 4]
14-10 (Process improvement) work package	[OUTCOME 2]	13-19 Review record	[OUTCOME 4]
10-00 Process description	[OUTCOME 3, 4]	15-01 Analysis report	[OUTCOME 4]
06-51 Tailoring Guideline	[OUTCOME 3]	16-06 Process repository	[OUTCOME 4]

PIM.1.BP1: Define process architecture. Define a standard set of processes meeting the define quality objectives and quality policy. Determine the sequence and interaction between processes so that they work as an integrated system of processes. Determine required inputs and expected outputs of each process. [OUTCOME 1]

Note 1: A process architecture typically contains a clear distinction of process, project, product and/or service documents.

PIM.1.BP2: Define standard processes. Define and maintain a description of each standard process according to the process needs of the organization. [OUTCOME 2]

Note 2: This includes activities, input and output work products, roles and responsibilities, phases and/or milestones/gates, resources, templates, checklists, instructions, training material, and the trigger for activities.

PIM.1.BP3: Establish process tailoring guidelines. Establish organizational guidelines for tailoring the organization's standard process to meet the specific needs of projects. [OUTCOME 1, 3]

PIM.1.BP4: Communicate process. Communicate and ensure awareness for the standard processes, the quality policy, and the quality objectives. [OUTCOME 1, 4]

Note 3: This includes all required training.

PIM.1.BP5: Monitor and adjust process performance. Monitor process performances. Capture and maintain information and data related to the use of standard processes to facilitate the maintenance of the pro-cess architecture and the standard processes. Adjust as needed to achieve the quality objectives. [OUTCOME 4]

PIM.1.BP4: Report process performance. Report process performance data, trend and target values. [OUTCOME 1, 4]

RIN.2 Skill Development

The purpose of the Skill Development process is to provide the organization and project with individuals who possess the needed skills and knowledge to perform their roles effectively.

Process outcomes – As a result of successful implementation of this process

1. Training is provided to address the training needs in the organization; and
2. Training is conducted to ensure that all individuals have the skills required to perform their assignments

Output work products

06-04 Training material	[OUTCOME 2]	15-22 Training evaluation report	[OUTCOME 2]
08-02 Acquisition plan	[OUTCOME 1]	15-ORG01 Skill analysis report	[OUTCOME 2]
08-24 Training plan	[OUTCOME 2]	19-08 Training strategy	[OUTCOME 1]
13-23 Training record	[OUTCOME 2]		

RIN.2.BP1: Develop a strategy for skill development. Develop a strategy for skill development and training including how the competence needs will be identified, how the needed development and training will be developed or acquired and provided. [OUTCOME 1]

Note 1: The strategy typically includes how the skill needs are identified and required trainings are provided in time. The strategy typically includes methods to ensure that no individual in the organization perform any activities without sufficient skills.

Note 2: The skill needs of each individual are typically identified based on potentials roles and responsibility the person might be assigned to.

Note 3: Typically, the skill needs of each individual covers the specific skills for the assigned activities, roles and responsibilities, the soft skills, the process knowledge, and the skills required to ensure a specific culture for quality, learning, and aspects of safety and security.

Note 4: Usually, established standards and recognized certificates are required as proof of the required skills.

Note 5: Typical training methods are classroom training, online training, exer-cises, workshops, on-the-job training, coaching concepts, or a combination of them. At a minimum, the training objectives and criteria for achievement need to be clearly defined.

RIN.2.BP2: Identify new skills and competencies. Identify and evaluate skills and competencies to be provided or improved through development and training. [OUTCOME 1]

Note 6: Sources or reasons for new skills and competencies might be new individuals, new tasks, new tools or techniques, new laws and guidelines, and renewal requirements.

RIN.2.BP3: Develop or acquire training. Develop or acquire training that addresses the common skill development needs. Maintain training. [OUTCOME 1]

RIN.2.BP4: Train individuals. Train the individuals of the organization so that they attain the knowledge and skills needed to perform their roles. Maintain adequate skill records. [OUTCOME 2]

RIN.2.BP5: Evaluate training effectiveness. Identify and evaluate added value provided by each training session, including the satisfaction in training results. [OUTCOME 2]

QNT.1 Quantitative Performance Management

The purpose of the Quantitative Performance Management process is to establish and maintain a quantitative understanding of the performance of selected organization’s processes through measurement and the use of statistical and other quantitative techniques to predict the performance of the organization’s implemented processes and to support the achievement of the organization’s relevant business goals.

Process outcomes – As a result of successful implementation of this process

1. Processes or process elements are selected for quantitative measurement based on of their relevance and significance to the achievement of business objectives;
2. Measures and analytical techniques to be used in statistically managing the processes or process elements are established and maintained;
3. Process performance data is collected and analyzed using appropriate statistical or other quantitative techniques to establish an understanding of the variation of the selected processes or process elements;
4. Results of data analysis are used to identify special causes of variation (assignable causes) in process performance;
5. Corrective and preventive actions are implemented to address the special and other causes of variation;
6. Performance of the selected processes or process elements is monitored and improved to establish stable, capable and predictable processes within established control limits; and
7. Projects are managed using statistical and quantitative techniques to optimize the probability of achieving critical business objectives

Output work products

07-10 Process Performance Models	[OUTCOME 7]	10-00 Process Description	[OUTCOME 7]
08-ORG01 Description of measurement and analytical techniques	[OUTCOME 1, 2]	10-06 Process control limit (also known as Process Performance Baselines)	[OUTCOME 3, 4, 6]
14-02 Corrective action register	[OUTCOME 5]	16-06 Process repository	[OUTCOME 1, 6]
14-12 Preventive action register	[OUTCOME 5]	16-07 Measurement repository	[OUTCOME 3]
15-01 Analysis report	[OUTCOME 4]	19-13 Decision-making strategy	[OUTCOME 1, 7]
15-08 Risk analysis report	[OUTCOME 1, 4]	19-14 Selection criteria	[OUTCOME 1, 7]
15-18 Process performance report	[OUTCOME 6]		

QNT.1 with 8 Base practices

QNT.1.BP1: Identify critical business and/or project objectives. Select the relevant business and/or project objectives to be addressed through quantitative management. [OUTCOME 1]

QNT.1.BP2: Establish quantitative process performance objectives. Select the processes or process elements that impact the identified business/project objectives. Then establish and maintain quantitative process performance objectives, both for the selected process or process elements. [OUTCOME 1]

Note 1: Process Performance objectives can cover various characteristics of a process, such as quality, elapsed effort, duration, etc.

QNT.1.BP3: Establish an appropriate set of measures and analytic techniques for selected processes/process elements. Define measures and analytical techniques to be used in quantitative management of the processes and process element that impact achievement of the process performance objectives. Define the handling and usage of data sources and storage. Ensure consistency to the performance, project and business objectives. Perform measurements and use the analytical techniques as defined. [OUTCOME 1, 2]

Note 2: Analytical techniques might be statistical or other quantitative techniques.

QNT.1.BP4: Establish distributions that characterize Process Performance. Collect and analyze the process performance data of selected processes and process elements and establish distributions that characterize process performance. [OUTCOME 3, 4, 6]

Note 3: The distribution that characterizes the process performance is also known as a process performance baseline.

QNT.1.BP5: Analyze and correct special causes of variation. Perform root cause analysis when process performance is out of its control limits or in cases where process performance objectives are not achieved. Identify and implement corrective and if needed preventive actions. [OUTCOME 4, 5]

Note 4: Special causes of variation are also known as assignable causes.

QNT.1.BP6: Establish Process Performance Models. Develop process performance models based on the relationship of measurable characteristics of one or more processes or process elements in order to statistically predict the probability of achieving the desired process performance objectives. [OUTCOME 1, 3, 6, 7]

QNT.1.BP7: Compose and deploy a defined process. Use statistical and other quantitative techniques to compose and deploy a defined process that enables the project to achieve its process performance objectives. [OUTCOME 1, 7]

Note 5: Project planning uses process performance baselines and process perfor-mance models.

QNT.1.BP8: Use statistical and other quantitative techniques to monitor and manage the project. Use statistical and quantitative techniques to monitor the performance of selected process elements and manage the project so that its objectives will be met. [OUTCOME 1, 7]

QNT.2 Process Innovation

The purpose of the Process Innovation Process is to improve the performance of selected processes that are fundamental to achieve an organization's busi-ness objectives in a systematically planned and predictable manner, based on statistical and quantitative analysis of the impact of the proposed changes, that might be small improvement or bigger innovations.
NOTE 1: Any process improvements according to this purpose falls within the term process innovation.

Process outcomes – As a result of successful implementation of this process

1. Quantitative business objectives are maintained based on business strategies and current results;
2. Process performance objectives are established by analysis using statistical and quantitative techniques to understand the organization's ability to meet the identified business objectives;
3. Process performance shortfalls and common causes of variation are understood and areas that could contribute to achieving business objectives are identified for innovation;
4. Identified innovation areas are analyzed for their relevance, significance and possible impact on the organization's process performance objectives;
5. Process innovations are piloted in order to validate the expected impact on process performance objectives; and
6. Validated improvements are selected for deployment based on an evaluation of cost, benefits and other factors.

Output work products

03-06 Process performance data	[OUTCOME 2, 3]	14-12 Preventive action register	[OUTCOME 6]
05-02 Business goals	[OUTCOME 1]	15-01 Analysis report	[OUTCOME 1,2,3,4,5]
05-07 Process performance objectives	[OUTCOME 2]	15-05 Evaluation report	[OUTCOME 4, 5]
07-09 Quantitative analysis technique	[OUTCOME 2, 3]	15-16 Improvement opportunity	[OUTCOME 3, 4]
07-10 Process performance model	[OUTCOME 3, 4]	16-07 Measurement repository	[OUTCOME 1, 2, 3]
08-29 Improvement plan	[OUTCOME 4, 5]	19-02 Process strategy	[OUTCOME 1, 2]
09-02 Quality policy	[OUTCOME 21]	19-13 Decision-making strategy	[OUTCOME 3, 4, 5]
10-05 New process concept	[OUTCOME 4]	19-14 Selection criteria	[OUTCOME 3, 4]
14-02 Corrective action register	[OUTCOME 5, 6]		

QNT.2.BP1: Maintain business objectives. New business visions and objectives are monitored and analyzed on a regular basis. Process innovation objectives are adjusted if needed. [OUTCOME 1, 2]

QNT.2.BP2: Identify common causes of variation and shortfalls in achieving process performance objectives. Use statistical and quantitative techniques to analyze performance of processes and process elements that directly impact business objectives in order to understand common causes of variation and shortfalls in achieving process performance objectives. [OUTCOME 3]

QNT.2.BP3: Identify innovation opportunities. Identify potential innovation opportunities for processes, arising from new technologies and process concepts. Analyze feedback on opportunities for innovation and select the improvement opportunities based on their relevance and significance to the achievement of business objectives. [OUTCOME 4]

QNT.2.BP4: Establish process innovation objectives for innovation opportunities. Analyze the costs, benefits, and risks of selected innovation opportunities and the impact on the organization's process performance objectives and determine an implementation strategy. [OUTCOME 4]

QNT.2.BP5: Pilot potential beneficial innovation opportunities. Select and plan the pilot innovations including criteria to be used for evaluating results in order to gain early feedback on the potential benefits the impact on process performance objectives. [OUTCOME 4, 5]

QNT.2.BP6: Review the results of pilot innovations. Review the results of pilots to determine whether to deploy the innovations to other parts of the organization. [OUTCOME 5, 6]

QNT.2.BP7: Select innovations to be deployed.

Prioritize and select candidate innovations for implementation based on

- Economic factors (productivity, profit, growth, efficiency, quality, competition, resources, and capacity);
- Human factors (job satisfaction, motivation, morale, conflict/cohesion, goal consensus, participation, training, span of control);
- Management factors (skills, commitment, leadership, knowledge, ability, organizational culture and risks);
- Technology factors (sophistication of system, technical expertise, development methodology, need of new technologies).

[OUTCOME 6]

QNT.2.BP8: Establish process innovation objectives for innovation opportunities. Plan and monitor the implementation of the innovations based on the implementation strategy. [OUTCOME 6]

QNT.2.BP9: Measure progress towards achieving process innovation objectives. Use quantitative and statistical techniques to evaluate effectiveness of process change against process objectives and historical data, with respect to common causes of variations. [OUTCOME 1]

QNT.2.BP10: Take corrective actions if process innovation objectives are not achieved. Take corrective actions when process innovations fail to meet the defined process innovation objectives. [OUTCOME 1]

Annex A2: Process set for Automotive SPICE® PAM 4.0 maturity level assessments

ML	Process ID & Name		Scope	Scource
ML-1	SYS.2	System Requirements Analysis	Plug-In	Automotive SPICE®
	SYS.3	System Architectural Design	Plug-In	Automotive SPICE®
	SYS.4	System Integration & Integration Verification	Plug-In	Automotive SPICE®
	SYS.5	System Verification	Plug-In	Automotive SPICE®
	SWE.1	Software Requirements Analysis	Plug-In	Automotive SPICE®
	SWE.2	Software Architecture Design	Plug-In	Automotive SPICE®
	SWE.3	SW Detailed Design and Unit Construction	Plug-In	Automotive SPICE®
	SWE.4	Software Unit Verification	Plug-In	Automotive SPICE®
	SWE.5	Software Component Verification and Integration Verification	Plug-In	Automotive SPICE®
	SWE.6	Software Verification	Plug-In	Automotive SPICE®
	HWE.1	Hardware Requirement Analysis	Plug-In	Automotive SPICE®
	HWE.2	Hardware Design	Plug-In	Automotive SPICE®
	HWE.3	Verification against Hardware Design	Plug-In	Automotive SPICE®
	HWE.4	Verification against Hardware Requirements	Plug-In	Automotive SPICE®
	MLE.1	Machine Learning Requirements Analysis	Plug-In	Automotive SPICE®

ML-1	MLE.2	Machine Learning Architecture	Plug-In	Automotive SPICE®
	MLE.3	Machine Learning Training	Plug-In	Automotive SPICE®
	MLE.4	Machine Learning Model Testing	Plug-In	Automotive SPICE®
	SUP.1	Quality Assurance	Basic	Automotive SPICE®
	SUP.8	Configuration Management	Basic	Automotive SPICE®
	SUP.9	Problem Resolution Management	Basic	Automotive SPICE®
	SUP.10	Change Request Management	Basic	Automotive SPICE®
	SUP.11	Machine Learning Data Management	Plug-In	Automotive SPICE®
	MAN.3	Project Management	Basic	Automotive SPICE®
	ACQ.4	Supplier Monitoring	Extened*	Automotive SPICE®
ML-2	SYS.1	Requirements Elicitation	Extened*	Automotive SPICE®
	MAN.5	Risk Management	Extened	Automotive SPICE®
	SPL.2	Product Release	Extened*	Automotive SPICE®
ML-3	PIM.1	Process Establishment	Extened*	This document
	RIN.2	Skill development	Extened*	This document
	PIM.3	Process Improvement	Extened	Automotive SPICE®

	REU.2	Reuse Program Management	Extened*	Automotive SPICE®
ML-4	QNT.1	Quantitative Performance Management	Extened	This document
	MAN.6	Measurement	Extened	Automotive SPICE®
ML-5	QNT.2	Quantitative Process Innovation	Extened	This document

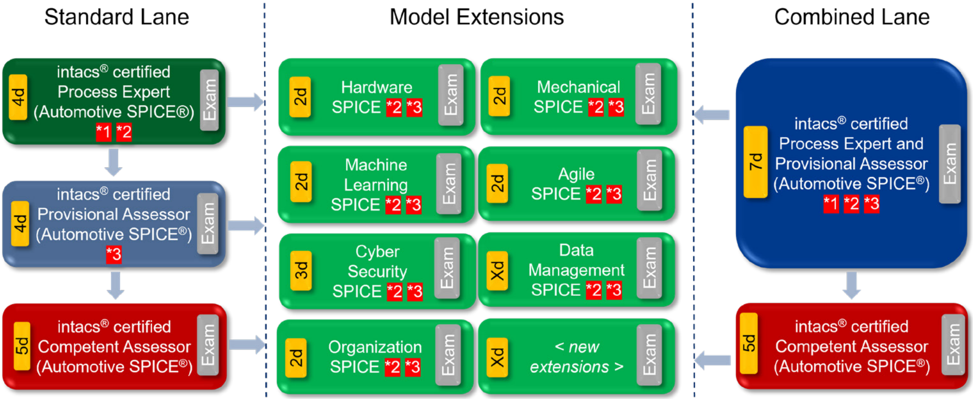
ASSESSMENT GUIDE

intacs™ Certification Levels for Assessors and Instructors

Certification Level	**Additional Requirements & Capability** (compared to the lower assessor grade)
"intacs™ certified **Instructor Competent Level**" (qualified for all passed PE/PA model and extension trainings/exams; domain specific qualifications may apply*)	▪ Proven teaching skills ▪ Approval by an accredited instructor (observation process) ▪ No training course or exam
"intacs™ certified **Instructor Provisional Level**" (qualified for all passed PE/PA model and extension trainings/exams; domain specific qualifications may apply*)	
"intacs™ certified **Principal Assessor**" (qualified for all passed PE/PA model and extension trainings/exams; domain specific qualifications may apply*)	▪ Continuously and actively contributes to the international SPICE community's knowledge & best practices ▪ No training course or exam
"intacs™ certified **Competent Assessor**" (qualified for all passed PE/PA model and extension trainings/exams; domain specific qualifications may apply*)	▪ Active assessment experience ▪ Passed CA training course & exam ▪ Approval by an accredited assessor (observation process) ▪ Accredited of leading assessments
"intacs™ certified **Provisional Assessor**" (qualified for all passed PE/PA model and extension trainings/exams; domain specific qualifications may apply*)	▪ Little or no assessment experience ▪ Passed PA training course & exam ▪ Capable of performing internal evaluations & acting as a co-assessor in official assessments
"intacs™ certified **Process Expert**" (Automotive SPICE®, Test SPICE, Medical SPICE, ...)	▪ Little or no assessment experience ▪ Passed PE training course & exam ▪ Capable of supporting internal improvement activities

** e.g. VDA-specific requirements for qualification*

intacs® Training Architecture Automotive SPICE, October 1st, 2024


Standard Lane

Model Extensions

Combined Lane

4d intacs® certified Process Expert (Automotive SPICE®) *1 *2 Exam

4d intacs® certified Provisional Assessor (Automotive SPICE®) *3 Exam

5d intacs® certified Competent Assessor (Automotive SPICE®) Exam

2d Hardware SPICE *2 *3 Exam

2d Mechanical SPICE *2 *3 Exam

2d Machine Learning SPICE *2 *3 Exam

2d Agile SPICE *2 *3 Exam

3d Cyber Security SPICE *2 *3 Exam

Xd Data Management SPICE *2 *3 Exam

2d Organization SPICE *2 *3 Exam

Xd *< new extensions >* Exam

7d intacs® certified Process Expert and Provisional Assessor (Automotive SPICE®) *1 *2 *3 Exam

5d intacs® certified Competent Assessor (Automotive SPICE®) Exam

*1 includes Base / System / Software processes without HWE and MLE, emphasizes Process Improvement

*2 includes all Guideline aspects regarding content

*3 includes all Guideline aspects regarding rating rules

Certification Requirements for Assessors

	Provisional Assessor	Competent Assessor	Principal Assessor
INITIAL CERTIFICATION	• Process Expert training completed, and exam passed • Provisional Assessor training completed, and exam passed	• Certificate Provisional level • Competent Assessor training completed, exam and observation passed • Upgrade training ASPICE 4.0 attended • Lead Assessor observation passed • within 60 months: AT AT AT AT AT	• Certificate Competent level • within recent 36 month: AL AL AL AC AC any
LICENSE RENEWAL	(no requirements)	• within 36 months: AT,AL AL AL EP,AC any any	• within 36 months: AT,AL AL AL AC AC any

TYPES OF EXPERIENCE EVIDENCE

Assessment participation

AT Assessment team member ≥ 50 h (Preparation, Execution, Reporting)

AL Assessment team lead ≥ 50 h (Preparation, Execution, Reporting)

Passive event participation

intacs®-accepted, ≥ 6 net hours

IP Internal event

EP External event

Active contribution

intacs®-accepted, ≥ 6 net hours

AC Examples:

- Presentation at an intacs®-accepted event
- Member of an iNTACS Working Group
- Publication of a SPICE-related paper

Certification Requirements for Assessors

	INSTRUCTOR PROVISIONAL	INSTRUCTOR COMPETENT
INITIAL CERTIFICATION	• Certificate Principal level • Didactic skills • Training observation passed (Provisional course) • within 60 months: AT,AL AL AL AC AC,CT any	• Certificate Instructor (Provisional) • Training observation passed (Competent course)
LICENSE RENEWAL	• within 36 months: AT,AL AL AL AC AC,CT any CT	

ADDITIONAL TYPE OF EXPERIENCE EVIDENCE

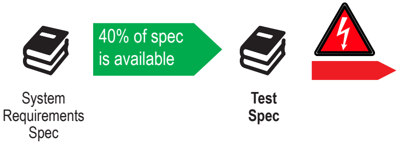
Assessor training course held

CT

- intacs® Provisional Assessor training
- intacs® Competent Assessor training

The process context definition impacts the rating.

1. Example:
The entire product is defined as process context.



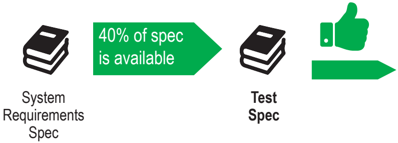
The test process can only be executed for the given 40% of the system requirements.
⟶ **BPs and PA1.1 rating < L; Level 0**

Idea of completeness

Good process performance with insufficient input from other processes leads to a lower rating. The argument is, that with insufficient input, the process cannot be executed without risks. The evidence is missing, that the process is capable to fulfill the purpose.

Typically the customer is interested to identify the process related product risks.

2. Example:
The 40% of the specification is defined as process context.



For the given 40% of the system requirements the test process is executed well ⟶ **rating F**

Idea of capability

Good process performance with insufficient input from other processes leads to a good rating (see example above).
The argument is, that the process is executed in a good way based on the available input.

Assessment Input Part 1 of 3: Assessment Plan

Roles and Assignments			
Sponsor, Position			
Local Coordinator			
Lead Assessor, Certification ID, valid until			
Co Assessor(s), Certification ID(s), valid until			
Standards and Classifications			
Process Assessment Model version	Automotive SPICE 4.0	Assessment class	1, 2 or 3
Assessment Standard	ISO/IEC 33002	Independence category	A, B, C or D
Assessment Process	KM Assessment Process 4.1	Rating scale	NPLF+-
		VDA Guideline version	2
Application of VDA Assessment Guideline chapter 2.5: Assessing specific application environments			
Model based development	Yes/No	Development external to the assessed project DEX	Yes/No
Agile environments	Yes/No	Application parameter	Yes/No

Assessment Input Part 2 of 3: Assessed Project and Assessment Scope

Project & Product			
Company, Organizational Unit(s)	<Name(s) of the assessed …>		
Project(s)	<Name(s) of the assessed project(s)>		
Product(s), ASIL Level(s)	e.g., Braking Control Unit, ASIL D		
Application Domain(s)	Body Electronics, Integrated Services, Infotainment, Powertrain, Chassis Control		
Organization Unit Classification	Automotive Tier 1	Small/Medium-sized Enterprise	Yes/No
Assessment Scope			
Assessment purpose	“Evaluation of process improvement” or “Identify process related product risk”		
Process instances and locations	e.g., VDA scope incl. MAN.5, REU.2 in location A; SWE.1-5 for Basic-SW in location B; SWE.1-5 for Appl.-SW in location C;		
Target capability level	e.g., CL 3 for all assessed processes		
Process context	Specify e.g., "Entire product" OR, "A subset of stakeholder requirements valid for a specific release" OR "All changes between two defined project milestones" OR "All software requirements implemented by improved processes"		
Constraints			

Assessment Input Part 3 of 3: Assessment Agenda

Day 1			
Start	End	Topic	Participants
9:00	9:15	Kick off	Sponsor, Assessment Team
9.15	10:50	MAN.3 - Project Management - Project Level	
10:50	11:00	Break	
11:00	12:00	MAN.3 - Project Mngt. - Sub-Project/Teams	
12:00	12:30	Break	
12:30	13:30	Consolidation	
13:30	15:00	SUP.1 - Quality Assurance	
15:00	15:30	Consolidation	
15:30	17:00	SUP.8 - Configuration Management	
17:00	17:30	Consolidation	
Day 2			
Start	End	Topic	Participants
9:00	10:30	SYS.2 - System Requirements	
10:30	11.00	Consolidation	
11:00	12:15	SYS.3 - System Architecture	
12.15	13:00	Break	
13:00	13.30	Consolidation	
13.30	15:00	SWE.1 - SW Requirements	
15:00	15:30	Consolidation	
15:30	17:00	SWE.2 - SW Architecture	
17:00	17:30	Consolidation	

Assessment Input Part 3 of 3: Assessment Agenda

Day 3			
Start	End	Topic	Participants
9:00	10:15	SWE.3 - SW Construction	
10:15	10:45	Consolidation	
10:45	12:15	SWE.4 - SW Module Test	
12:15	13:00	Break	
13:00	13:30	Consolidation	
13:30	15:00	SWE.5 - SW Integration and Integr. Test	
15:00	15:30	Consolidation	
15:30	17:00	SWE.6 - SW Qualification	
17:00	17:30	Consolidation	
Day 4			
Start	End	Topic	Participants
9:00	10:15	SYS.4 - System Integration and Integr. Test	
10:15	10.45	Consolidation	
10:45	12:15	SYS.5 - System Qualification	
12.15	13:00	Break	
13:00	13.30	Consolidation	
13.30	14:45	SUP.9 - Problem Resolution Management	
14:45	15:15	Consolidation	
15:15	17:00	SUP.10 - Change Request Management	
17:00	17:30	Consolidation	

Assessment Input Part 3 of 3: Assessment Agenda

Day 5			
Start	End	Topic	Participants
9:00	11:00	Consolidation	
11.00	12:00	Open Session	
12:00	13:00	Break	
13:00	15:00	Consolidation	
15:00	16:00	Presentation of Preliminary Results	

Assessment class / Assessment type

Assessment class:

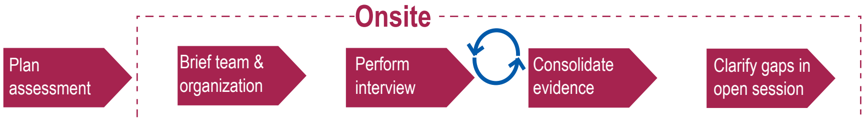
1. 4 or more process instances per process
   If there are fewer than the required number of process instances available in the organization, all process instances shall be selected.
2. 2 or more process instances per process
   If there are fewer than … (ditto)
3. else

Example for a specific application environment:
2.2.5 Management of platform and legacy SW

No (i.e. the Management of platform and legacy software is not applied): *The assessment covers the current project and former projects in which platform and legacy software were developed. The platform and legacy software development is assessed and rated in separate instances.*

	Type A	Type B	Type C	Type D
Body performing the assessment	The body performing the assessment is independent of the organization being assessed		The body performing the assessment is part of the organization being assessed	The body performing the assessment may or may NOT be independent being assessed
Competent assessor	Independent of the organization being assessed	Independent of the organization being assessed	Adequate separation of responsibility from personnel in other functions	Need NOT be independent of the organization being assessed
Assessors (other than competent asessor)		May be from the organization being assessed provided clear separation of the reponsibilities of the assessors from personnel in other functions		

KM Assessment Process 4.1 (Overview) Part 1



	Plan assessment	Brief team & organization	Perform interview	Consolidate evidence	Clarify gaps in open session
SP: Sponsor	S	S			
LA: Lead Assessor	R	R	R	R	R
CA: Co-Assessor	S	S	S	S	S
LC: Local Contact	S	S	S		S
IN: Interviewee	S	S	S		S
Description	The LA clarifies planning input & plans the assessment. All participants agree to the agenda. The assessors sign the NDA.	The LA briefs the CA & LC. The SP explains the assessment purpose. The LA briefs the organization.	The LA assign an interviewer to lead the interview. The IN answers. The assessors take notes. The LC collects evidences.	The assessors consolidate the information & identify gaps. The assessors draft a 1st rating	The assessors ask what is needed to close the information gaps. The IN answers. The assessors take notes. The LC collects evidence.
Output	Assessm. Scope Assessm. Plan Assessm. Agenda NDA	Kick-off presentation	Notes & evidence	Strengths, potentials & information gaps	Notes & evidence

KM Assessment Process 4.1 is compliant to the documented assessment process in the VDA Automotive Spice Guideline, 2.0

KM Assessment Process 4.1 (Overview) Part 2

Onsite

	Rate PAs & capability levels	Feedback to organization	Feedback to sponsor	Report assessment	Plan improvement
SP: Sponsor		S	S	S	R
LA: Lead Assessor	R	R	R	R	S
CA: Co-Assessor	S	S		S	S
LC: Local Contact		S		S	S
IN: Interviewee		S			S
Description	The assessors consolidate information. The assessors perform the ratings and calculate levels. The assessors prepare the feedback.	The LA presents the feedback.	(optional step) The LA explains the results and answers the questions from the SP	The LC provides the list of evidences. The assessors draft & review the report. The LA sends the report. The SP, LA & CA sign the log.	The SP sets priorities and initiates related improvement workshops.
Output	Process profiles	Feedback presentation	-	Assessment report Assessment log	Improvement plan

Guideline For Interviewees

Prepare for the interview

Understanding

- Ensure correct understanding of the SPICE process, its purpose and practices
- Ensure correct understanding of the strategy, process description, and needed artifacts
- Reflect on the last "Process Audit" and weaknesses that were detected

Presentation

- Prepare introduction slides
- Choose at least 5 examples to show that all actions are performed throughout the whole process
- Exercise presentation several times

Artifacts

- Update, release and check-in artifacts as planned
- Be prepared to show coverage
- Ensure you are able to guide the assessor through each artifact

Convince in the interview

Strategy

- Report your responsibility – very compactly; use the role description or ONE introduction slide for your work
- Show the defined process as the basis for your explanation

Consistency

- Explain what you do using the defined process step by step In parallel open the artifacts and show your work
- Use the prepared examples and show consistency
- Be prepared to show different examples

Completeness

- Report progress and status
- Show coverage
- Show trends and derived actions

Guideline For Interviewer

Start of interview

- Be friendly. Reduce stress. Ensure, that the interviewee does not feel he is being grilled. Start with a very easy question.

How to raise questions

- Start with an ‘open’ question to get a lot of information. Use ‘closed’ questions to get precise or detailed answers.
- Be an active listener. Repeat what you have understood if you need additional confirmation.

Do not …

- Never assume any activities or work products
- Never phrase a question in a way that indicates a certain answer or expectation
- Never blame a person. Never provide feedback (i.e. rating or indicators for it) during interviews. Never be ironic.

Find the gaps

- Follow the feature / change request / bug report from the source to the realization, to the qualification and to closure
- Repeat at least 3 times. Check consistency and completeness. Do not get distracted by sidetracks of ‘storytellers’

End of interview

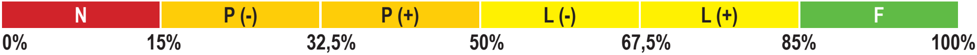
- Be friendly. Reduce stress. Ensure, that the interviewee does not feel he is being grilled.
- Invite participants to the feedback presentation.

NPLF Rating

N	Not achieved 0% to 15%	There is little or no evidence of achievement of the defined attribute in the assessed process. **Outcome:** Outcome/achievement not existent, or content judged inacceptable.
P	Partially achieved >15% to 50%	There is some evidence of an approach to, and some achievement of, the defined attribute in the assessed process. Some aspects of achievement of the attribute may be unpredictable. **Outcome:** Some outcomes/achievements implemented, but projects/OUs still incapable of reaching quality, time, or budget goals and targets
L	Largely achieved > 50% to 85%	There is evidence of a systematic approach to, and significant achievement of, the defined attribute in the assessed process. Some weakness related to this attribute may exist in the assessed process. **Outcome:** Outcome/achievement implies a significant likelihood, however no certainty, of reaching quality, time, and budget goals and targets.
F	Fully achieved > 85% to 100%	There is evidence of a complete and systematic approach to, and full achievement of, the defined attribute in the assessed process. No significant weaknesses related to this attribute exist in the assessed process. **Outcome:** No process risk with respect to quality, time, budget. Goals and targets identified, even in presence of imperfections.

How to Rate Practices and Process Attributes

1. **Interview** the team members who perform the process and gather sufficient evidences like affirmations and work products for each practice (of both Base Practices and Generic Practices) within the assessment scope.
2. **Clarify** how each practice is expected to be applied within the specific project, scope and schedule.
3. **Use NPLF for rating** the achievement of each base and generic practice
   (Not achieved / Partly achieved / Largely achieved / Fully achieved)



   a. **Rate N**, if there is little or no evidence despite sufficient questions.

   b. Else **rate F**, if the activities expected by the schedule are performed completely and systematically AND there are no significant risks.

   c. Else **rate L,** if the activities according to the schedule are performed largely and systematically.

   d. Else **rate P.**

   e. **Indicate with +/-** the distance to 15% - 50% - 85%.

   f. **Consider** the rating rules in the Automotive SPICE® Guidelines

4. **Rate** the degree to which the related Process Attributes (PAs) are achieved in the same way.

How to Aggregate Ratings and Calculate Levels

5. If required, aggregate the PA ratings of several process instances (but never of different processes)

a. Assign values for each NPLF rating: N → 0; P– → 1; P+ → 2; L– → 3; L+ → 4; F → 5

b. Assign each value a pre-defined weighting, round the arithmetic mean and convert back

6. Calculate the capability level of the related process: a ‘F’ rating of the PAs is expected – start with capability level 1 and climb up level for level. Only at the highest level is a PA rating of ‘L’ accepted.

Process xyz	PA1.1	PA2.1	PA2.2	PA3.1	PA3.2	PA4.1	PA4.2	PA5.1	PA5.2
Capability Level 1	L/F								
Capability Level 2	F	L/F	L/F						
Capability Level 3	F	F	F	L/F	L/F				

Assessment Report

The lead assessor reports the assessment team result to the sponsor.

The report includes as a minimum:

- The assessment input including constraints, if any
- The strengths and potentials per process
- The resulting capability and process attribute profiles
- The practice ratings and objective evidences
- The reasons for rule deviations, if applicable
- Unique title, version number, change history, distribution list

The report should also include:

- A Management Summary
- Basic recommendation for next improvement steps

Assessment Log

The lead assessor asks the sponsor to sign the assessment log.

INFORMATION ITEM CHARACTERISTICS

Only in PDF included

Automotive SPICE 4.0 Annex B Information Item Characteristics

Characteristics of information items are defined using the schema in table B.1. See Section 3.3.2 on the definition and explanation on how to interpret information items and their characteristics.

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 use these in evaluating the samples provided by the organizational unit. It is not intended to use the listed characteristics as a checklist. Some characteristics may be contained in other work products, as it would be found appropriate in the assessed organization.

Table B.1 — Structure of information item characteristics (IIC) table

Automotive SPICE 4.0 Annex B Information Item Characteristics

Characteristics of information items are defined using the schema in table B.1. See Section 3.3.2 on the definition and explanation on how to interpret information items and their characteristics.

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 use these in evaluating the samples provided by the organizational unit. It is not intended to use the listed characteristics as a checklist. Some characteristics may be contained in other work products, as it would be found appropriate in the assessed organization.

Table B.1 — Structure of information item characteristics (IIC) table

ID	Name	Characteristics
01-03	Software component	• 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	• Software executable (e.g, simulator with stubbing, debug-able, object code) including: - application parameter files (being a technical implementation solution for configurability-oriented requirements) - all configured software elements
01-52	Configuration item list	• Items under configuration control • The name of work products and an associated reference (to file, to tool artifact) • Configuration item attributes and properties
01-53	Trained ML model	• 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	• Hyperparameters are used to control the ML model which has to be trained, e.g.: - Learn rate of training - Scaling of network (number of layers or neurons per layer) - Loss function • Minimum characteristics: - Description - Initial value - Final value upon communicating the results of the ML training
02-01	Commitment / agreement	• 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: - time - people - budget - equipment - facilities

03-06	Process performance information	• Measurements about defined quantitative or qualitative measurable indicators, that match defined information needs. • Measurement metrics for the calculation of the quantitatively or qualitatively measurable indicators • Data comparing process performance against expected levels • Examples for project performance information: - resource utilization against established target - time schedule against established target - activity or task completion criteria met

		- defined input and output work products available - process quality against quality expectations and/or criteria - product quality against quality expectations and/or criteria - highlight product performance issues, trends • Examples for service level performance information: - references any goals established - real time metrics related to aspects such as: - capacity - throughput - operational performance - operational service - service outage time - up time - job run time
03-50	Verification Measure data	• Verification measure data are data recorded during the execution of a verification measure, e.g.: - for test cases: raw data, logs, traces, tool generated outputs - measurements: values - calculations: values - simulations: protocol - reviews such as optical inspections à findings record - analyses: values

03-51	ML data set	• Selection of ML Data for e.g., ML model training (ML Training and Validation Data Set) or test of the trained and deployed ML model (ML Test Data Set).
03-53	ML data	• Datum to be used for Machine Learning. The datum has to be attributed by metadata, e.g., unique ID and data characteristics. Examples: – 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	• Consists of bill of materials • Consists of layout e.g, GERBER data • Specifies requirements for EOL test e.g.: – Test type (AOI, ICT, boundary scan) – Test coverage – Electrical loads – Acceptance criteria • In case of semiconductor development: mask data (GDS2)

04-04	Software architecture	• A justifying rationale for the chosen architecture. • Individual functional and non-functional behavior of the software component • Settings for application parameters (being a technical implementation solution for configurability-oriented requirements) • Technical characteristics of interfaces for relationships between software components such as: - 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 components and software states such as: - Logical software 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 components • Explanatory annotations, e.g, with natural language, for single elements or entire diagrams/models.

04-05	Software detailed design	• Elements of a software detailed design: - Control flow definition - Format of input/output data - Algorithms - Defined data structures - Justified global variables - Explanatory annotations, e.g, with natural language, for single elements or entire diagrams/models • Examples for expression languages, depending on the complexity or criticality of a software unit: - natural language or informal languages - semi-formal languages (e.g, UML, SysML) - formal languages (e.g, model-based approach)
04-06	System architecture	• A justifying rationale for the chosen architecture. • Individual behavior of system elements • Interrelationships between system elements Settings for system parameters (such as application parameters) Manual/human control actions, e.g., according to STPA • Interface Definitions: - Technical characteristics of interfaces for relationships between two system elements

		• Interfaces between system elements e.g.: - 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.: - 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.: - 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.:
  - connector pin configurations and floating IOs for µCs/MOSFETs
  - signal scaling & resolution to be reflected by the application software

- Mechanical-hardware interfaces e.g.
  - 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:
  - 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

04-51	ML architecture	• An ML architecture is basically a special part of a software architecture (see 04-04). Additionally - 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	• Describes the initial floorplan and 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 behavior • Identifies the relationship and dependency between hardware components • Describes all hardware variants to be developed • Describes power supply, thermal and grounding concepts

04-53	Hardware detailed design	• 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	• 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	• 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	• is defined by output, input, type, and electrical characteristics including signal tolerances. • Examples of interfaces are - high level interfaces like SPI, I2C, CAN, LIN, Ethernet - electrical interconnections - thermal interfaces between hardware elements (heat dissipation)
06-04	Training material	• 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	• Identification of required physical elements (e.g., hardware, mechanical, wiring elements), and software executables and application parameters (being a technical implementation solution for configurability-oriented requirements) • necessary sequence or ordering of integration • preconditions for starting system integration
06-51	Tailoring guideline	• Criteria for tailoring, • Proceeding of tailoring describing how to derive and document the defined process from the standard process including responsibility for tailoring and corresponding approval • 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	• Description / confirmation of existing backup and recovery mechanisms • References to corresponding procedures or regulations
07-04	Process metric	• Measurements about the process' performance: – 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	• Monitors key processes and critical tasks, provides status information to the project on: – 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: – tasks are performed on schedule – product's development is within the resource commitments allocated • References any goals established

07-06	Quality metric	• Measures quality attributes of the work products defined: – functionality – reliability – usability – efficiency – maintainability – portability • Measures quality attributes of the “end customer” quality perception *Note: Refer ISO/IEC 25010 for detailed information on measurement of product quality.*
07-08	Service level metric	• Real time metrics taken while a system is operational, it measures the system’s performance or expected service level • Identifies aspects such as: – capacity – throughput – operational performance – operational service – service outage time – up time – job run time

07-51	Measurement result	Result of gathering qualitative or quantitative data, e.g., Process metric • Measurements about the process' performance: - 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 Project metric • Monitors key processes and critical tasks, provides status information to the project on: - 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: • tasks are performed on schedule • product's development is within the resource commitments allocated • References any goals established Quality metric

		• Measures quality attributes of the work products defined: - functionality - reliability - usability - efficiency - maintainability - portability • Measures quality attributes of the “end customer” quality perception Service level metric • Benchmarking data • Customer satisfaction survey
07-61	Quantitative process metric	• Quantitatively measurable indicators that match information needs derived from business goals • Relation of the quantitatively measurable indicators to process elements in process descriptions or repositories and tools • Process measurement metrics for the calculation of the quantitatively measurable indicators, based on data from related process elements, repositories, or tools
07-62	Process analysis technique	• Methods for statistical analysis of process data • Frequency of data collection.
07-63	Process control limits	• Quantitative control limits for the quantitative process metrics

07-64	Process measurement data	• Data collected across process instances • Attributes of data, e.g., timestamps • Relation to process measurement metrics • Storage and retrieval • Effective controls over access
15-57	Quantitative process analysis results	• Deviations, and distributions, of the quantitative performance of individual process instances performance from the established quantitative control limits (special causes of variations)
08-66	Measures against deviations in quantitative process analysis	• Definition of counter measures actions to address each assignable cause of special causes of variation, or common causes of variation • Effective implementation of these counter measures
15-58	Common cause of variation analysis results	• Identification of common causes - deviations of the quantitative performance of all process instances from the established quantitative control limits - distributions of the quantitative performance of all process instances within established quantitative control limits

08-53	Scope of work	• 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 • Reuse options • Integration of third party deliveries
08-54	Feasibility analysis	• Statement about the ability of the project to achieve the project objectives with available resources
08-55	Risk measure	• Identifies - the risk to be mitigated, avoided, or shared (transferred) - the activities to mitigate, avoid, or share (transfer) the risk - the originator of the measure - criteria for successful implementation - criteria for cancellation of activities - frequency of monitoring

		• Risk treatment alternatives: - treatment option selected- avoid/reduce/transfer - alternative descriptions - recommended alternative(s) - justifications
08-56	Schedule	• Identifies the activities to be performed • Identifies the expected, and actual, start and completion date for required activities against progress/completion of activities • Identifies dependencies between activities and critical path • Has a mapping to scheduled resources and input data • Identifies resource allocation, resource workload, and critical resources *NOTE: A schedule is consistent with the defined work packages, see 14- 10*
08-57	Validation Measure Selection Set	• Include criteria for re-validation in the case of changes (regression). • Identification of validation measures, also for regression
08-58	Verification Measure Selection Set	• Include criteria for re-verification in the case of changes (regression). • Identification of verification measures, also for regression testing

08-59	Validation Measure	• 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 - 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	• 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 - 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 modelbased development, ICT) • Necessary verification environment & infrastructure • Necessary sequence or ordering

08-61	Resource allocation	• Detailed / named resources are allocated to process tasks • Overall resource workload is considered (e.g., allocation of resources to multiple projects) *NOTE: Work breakdown structure may be used to refine the detailed resource allocation* *NOTE: A resource allocation may be integrated in a/ be a part of the schedule, see 08-56* *NOTE: Resources to be allocated are e.g., personnel/human resources for project roles and physical and material resources such as (special/limited) equipment, tool, licenses, test hardware, test vehicle, climate chambers etc.*
08-62	Communication matrix	• 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	• 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	• The ML test approach describes - ML test scenarios with distribution of data characteristics (e.g., gender, weather conditions, street conditions within the ODD) defined by ML requirements - quantity of each ML 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	• The ML Training and Validation approach describes at least: - 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 robustification methods like random dropout

10-00	Process description	• Process description of a standard or defined process (e.g., after tailoring), including: - scope and the intended use of the process - process activities including description and dependencies - entry and exit criteria such as input information needed and expected outputs for activities - Roles assigned to process activities (e.g., as RASIC) or work products - guidelines - templates - specific methods/work instructions
10-50	Role description	• Name/identifier (unique within the organization) • Assigned activities (e.g., as RASIC) • Responsibilities and authorities • Required competencies, skills, and experience
10-51	Qualification method description	• Training courses • Training materials • Mentoring/coaching concepts • Self-learning material

10-52	Process resource and infrastructure description	• Required facilities • Required tools and corresponding licenses • Required networks • Required services • Required samples
11-03	Release note	• 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: - 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): - hardware / software / product elements, libraries, etc. - associated documentation list • New/changed parameter information (e.g., for application parameters or global variables) and/or commands. Note that application parameters are a technical implementation solution for configurability-oriented requirements) • 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	• Includes the hardware/software/product • Includes and associated release elements such as: - system hardware/software/product elements - associated customer documentation - application parameter definitions defined - command language defined - installation instructions - release letter

11-05	Software Unit	Can be • a representation of a software element at the lowest level in a conceptual model, which is decided not to be further subdivided and that is a part of a software component, or • a representation of a software unit under verification such as commented source code, auto-code, an object file, a library, an executable, or an executable model as input to verification
11-06	Integrated System	• Integrated product • Application parameter files (being a technical implementation solution for configurability-oriented requirements) • All configured elements for the product release are included
11-50	Deployed ML model	• 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.

12-03	Reuse candidate	• 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
13-06	Delivery evidence	• Evidence of items shipped/delivered electronically to customer • Identification of: - to whom it was sent - address, where delivered - delivery date - receipt of delivered product

13-07	Problem	• 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 - 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	• 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 *NOTE: This should be established before a release to identify consistent and complete delivery*
13-09	Meeting support evidence	• Agenda and minutes that are records that define: - purpose of meeting - attendees - date, place held - reference to previous minutes - what was accomplished - identifies issues raised

		- any open issues - next meeting if any
13-13	Product release approval	• Content information of what is to be shipped or delivered • Identification of: - for whom it is intended - the address where to deliver - the date released - Evidence of supplier approval
13-14	Progress status	• Status of a plan(s) (actual against planned) such as: - 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

13-16	Change request	• 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 - 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 conformance evidence	• 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
13-19	Review evidence	• Provides the context information about the review: - what was reviewed - lists reviewers who attended and their area of responsibility - status of the review

		• Provides information about the scope of the review: - checklists - review criteria - requirements - compliance to standards • Effort information about: - preparation time spent for the review - time spent in the review • Review findings: - non-conformances - improvement suggestions
13-24	Validation results	• Validation data, logs, feedback, or documentation • Validation measure passed • Validation measure not passed • Validation measure not executed, and a rationale • Information about the validation execution (date, participants etc.) • Abstraction or summary of validation results

13-25	Verification results	• Verification data and logs • Verification measure passed • Verification measure not passed • Verification measure not executed, and a rationale • Information about the verification execution (date, “object-under-verification”, etc.) • Abstraction or summary of verification results
13-50	ML test results	• Test data and logs • Test data with correct results • Test data with incorrect results • Test data not executed, and a rationale • Information about the test execution (date, participants, model version etc.) • Abstraction or summary of ML test results
13-51	Consistency Evidence	• Demonstrates bidirectional traceability between artifacts or information in artifacts, throughout all phases of the life cycle, by e.g., - 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 - 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 *Note: This evidence can be accompanied by e.g., Definition of Done (DoD) approaches.*
13-52	Communication Evidence	• All forms of interpersonal communication such as - e-mails, also automatically generated ones - tool-supported workflows - meeting, verbally or via meeting minutes (e.g., daily standups) - podcast - blog - videos - forum - live chat - wikis - photo protocol

13-55	Process resource and infrastructure documentation	• Information on availability, allocation, and usage of - Facilities - Tools and corresponding licenses - Networks - Services - Samples • for non-standard and critical resources and infrastructure.
14-01	Change history	• Historical records of all changes made to an object (document, file, software component, etc.): - description of change - version information about changed object - date of change - change requester information - change control record information
14-02	Corrective action	• 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

14-10	Work package	• 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 • Estimates of effort, duration *NOTE: The work package descriptions may be integrated into the/be a part of a schedule, see 08-56*
14-50	Stakeholder groups list	• Identifies: • involved parties • weight/importance of each stakeholder group • representative(s) for each stakeholder group • information needs of each stakeholder group
14-53	Role Assignment	• Assignment of person(s) to roles - 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	• Uniquely identifies type, supplier, and amount of the complete set of all hardware parts of the hardware
15-06	Project status	• Status of in regards to progress and consistency of schedule, work item content, tasks, resources (human resources, infrastructure, hardware/materials, budget), skills and competence of human resources • planned progress and expenditure against dates/deadlines and actual expenditure
		• reasons for variance from planned progress • threats to continued progress • issues which may affect the ability of the project to achieve its goals • contingency actions
15-07	Reuse analysis evidence	• Identification of reuse opportunities • Identification of constraints for reuse • Identification of regression test cases • Identification of reuse infrastructure • Identification of known defects

15-09	Risk status	• Identifies the status, or the change, of an identified risk: - risk statement - risk source - risk impact and risk probability - categories and risk thresholds, e.g., for prioritization or setting a status - risk treatment activities in progress
15-12	Problem status	• Indicates progress of problem resolution • Status of problem e.g., - by problem categories/classification - by problem resolution stage

15-13	Assessment/audit report	• States the purpose of assessment • Method used for assessment • Requirements used for the assessment • Assumptions and limitations • Identifies the context and scope information required: – date of assessment – organizational unit assessed – sponsor information – assessment team – attendees – scope/coverage – assesses and information – assessment tool used
		• Records the result: – Data – identifies the gaps, potentials, weaknesses or non-conformances that require corrective actions
15-16	Improvement opportunity	• 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

15-51	Analysis Results	• Identification of the object under analysis. • The analysis criteria used, e.g.: - selection criteria or prioritization scheme used - decision criteria - quality criteria • The analysis results, e.g.: - what was decided/selected - reason for the selection - assumptions made - potential negative impact • Aspects of the analysis may include - correctness - understandability - verifiability - feasibility - validity
15-52	Verification Results	• Verification data and logs • 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

15-54	Tailoring documentation	• Applied criteria for tailoring, • Evidence that the defined process is tailored from the standard process according to the defined criteria
15-55	Problem analysis evidence	• Author and involved parties • Date of the analysis • Context and root cause of the problem • Analysis result may include - Impact - Potential negative impact - Affected parties - Potential solution (if known)
15-56	Configuration status	• Summary of configuration management records including relevant status • Analysis of the configuration management overall state • Identification of baselines made
16-03	Configuration management system	• 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	• Contains process descriptions • Supports multiple presentations of process assets
16-50	Organizational structure	• Disciplinary reporting line • Organizational units and sub-units, if applicable
16-52	ML data management system	• 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

17-00	Requirement	• An expectation of functions and capabilities (e.g., non-functional 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 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 - heat dissipation

		- dimensions - weight - materials
		• Examples of aspects related to requirements about system interfaces are - connectors cables - housing • Examples for requirements at the hardware level are - 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

17-05	Requirements for work products	• Requirements for content and structure, storage and control - Identifies documentation specific meta data, such as id, date, author information, ownership, access rights, review and approval status with, where applicable, status model and workflow, 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
17-54	Requirement Attribute	• Meta-attributes that support structuring and definition of release scopes of requirements. • Can be realized by means of tools. *NOTE: usage of requirements attributes may further support analysis of requirements.*

17-55	Resource needs	• Identification of required resources for process performance • Staff including competencies, skills and authorities needs • Material, equipment, and infrastructure • Time and budget *NOTE: Needs are derived from Work Breakdown structure and schedule*
17-57	Special Characteristics	• 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 *NOTE: A typical method for identifying and rate special characteristics is an FMEA.*
18-00	Standard	• 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	• Defines expectations for product release: – 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	• Defines the expectations for work products and process performance • Including thresholds/tolerance levels, required measurements, required checkpoints • Defines what is an adequate work product (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
18-52	Escalation path	• 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	• Identify types of work products to be subject to configuration control

18-57	Change analysis criteria	• Defines analysis criteria, such as - resource requirements - scheduling issues - risks - benefits
18-58	Process performance objectives	• Objectives for the process of creating the process outcomes and capability level 2 achievements, and corresponding evaluation criteria • Assumptions and constraints, if applicable • Used as the basis for deriving a detailed planning • Examples: - Effort, costs, or budget targets (e.g., min/max limits) - Process-specific deadlines in line with milestones, or frequency of activities (of e.g., dates for deliveries to the customer, quality gates) - Metrics (e.g., max. number of open change requests per release, max. ratio of configuration items in status “in work” at certain milestones before next delivery / release date)
18-59	Review and approval criteria for work products	• Specifies for each type of work products review and approval needs - If and when a review is required - Who shall review it - Who shall approve it - Review method(s) to be used - Criteria for approval

18-70	Business goals	• Explanation of the business goals • Requirements for the business needs • Associations to other goals • Reasons for the existence of the goals and needs, level of degree of the need and effect on the business not having that need • Conditions, constraints, assumptions • Timeframe for achievement • Authorization at the highest level
18-80	Improvement opportunity	• Cause of the improvement need, e.g., - from qualitative or quantitative process performance analysis, evaluations, and monitoring - industry best practice review, state-of-the-art observations, market studies etc. • Improvement objectives derived from organizational business goals and improvement needs • Organizational scope • Process scope • Activities to be performed to keep all those affected by the improvement informed • Priorities

18-81	Improvement evaluation results	• Operational impacts of identified changes on the product(s) and processes • Expected benefit • Conditions, constraints, assumptions
19-01	Process performance strategy	• The operational approach to achieve the process outcomes, consistent with the Process Performance Objectives (18-58), e.g.: - proceedings, including the monitoring of the performance of the process - methodology • scope(s) of the strategy within the process, e.g.: - development sites - application domain-specific differences (e.g., software drivers versus. powertrain software) - disciplines (e.g., different configuration management approaches for software and hardware, or combined approaches) - options due to socio-cultural differences
19-50	ML data quality approach	• 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 requirementsv • Describes analysis activities of the data • Describes activities to ensure the quality of the data to avoid issues e.g., data bias, bad labeling

IT ISN'T ONLY AUTOMOTIVE SPICE WHICH WE SIMPLIFY

Acceptance of processes is the key

Accepted processes are lived and make it possible to learn from each other.

Accepted processes can fulfil norms – and provide benefit.

The focus lies on the most important process results.
The responsibility for content and release are clear.
Participants agree on the most direct path to the goal.

Accepted processes are based on simple rules.

TRAININGWORKSHOPS@KNÜVENERMACKERT

Easy and effective learning in training workshops

We offer trainings for

- SPICE and Process Management
- Engineering
- Safety and Security
- Management
- Leadership

Please find more information on www.knuevenermackert.com.

SERVICES – WE DELIVER!

We deliver more than just process definition

We deliver by driving the change to your next level

- Trainings
- Assessments Automotive SPICE, SPICE, CMMI, Safety
- Process management
- Process standardization
- Process roll out
- Coaching

WWW.KNUEVENERMACKERT.COM

Knüvener Mackert GmbH
Rote-Äcker-Str. 13
72766 Reutlingen / Germany

www.knuevenermackert.com
info@knuevenermackert.com

ISBN 978-3-00-061926-7
9 783000 619267