VDA Verband der Automobilindustrie

Joint Quality Management in the Supply Chain

Automotive SPICE®

- Guidelines

Process assessment using Automotive SPICE in the development of software-based systems

1st. edition, September 2017

Quality Management in the Automotive Industry

Automotive SPICE®

Guidelines

Process assessment using Automotive SPICE in the development of software-based systems.

1st. edition, September 2017

Verband der Automobilindustrie e. V. (VDA)
German Association of the Automotive Industry (VDA)

ISSN 0943-9412
Printed: December 2017

Verband der Automobilindustrie e. V. (VDA)
Qualitätsmanagement-Center (QMC)
10117 Berlin, Behrenstraße 35

Germany

Printing house:

Henrich Druck + Medien GmbH
Schwanheimer Straße 110, 60528 Frankfurt am Main, Germany

Printed on chlorine-free bleached paper

Non-binding VDA standard recommendation

The Association of the German Automotive Industry (VDA) recommends its members to apply the following standard for the implementation and maintenance of quality management systems.

Exclusion of liability

VDA volumes are recommendations available for general use. Anyone applying them is responsible for ensuring that they are used correctly in each case.

This VDA volume takes into account state of the art technology, current at the time of issue. Implementation of VDA recommendations relieves no one of responsibility for their own actions. In this respect everyone acts at their own risk. The VDA and those involved in VDA recommendations shall bear no liability.

If during the use of VDA recommendations, errors or the possibility of misinterpretation are found, it is requested that these be notified to the VDA immediately so that any possible faults can be corrected.

Copyright

Preface

The predominant and continually increasing number of innovative vehicle functions related to user-friendliness, safety, security and the protection of the environment by economic efficiency can only be achieved by the introduction of complex and highly-networked software systems.

Market demands require permanent innovations with increasing complexity within shorter time frames. Together with rising demand for reliability, the shorter development periods make it essential to improve the software development processes and methods for product creation so as to ensure the development and manufacturing of products in time and in the quality which satisfies customer's expectations.

In 2005 the Automotive Special Interest Group published the automotive domain specific "Automotive SPICE Process Assessment Model" and "Automotive SPICE Process Reference Model" derived from the ISO/IEC 15504 International Standard for Process Assessments.

Between 2012 and 2015 the "Automotive SPICE Process Assessment Model" Version 2.5 and "Automotive SPICE Process Reference Model" Version 4.5 was significantly reworked by the VDA QMC working group 13. This was based on a mandate of the VDA Quality management board to take appropriate steps to improve the quality and comparability of assessment results.

In July 2015 the Automotive SPICE process reference and assessment model version 3.0 was released in a combined document that is improved regarding the structure of the processes with added clarifications, additional concepts and by removing inconsistencies. A version 3.1 with minor updates will be available with the publication of this document.

The "Automotive SPICE Process Assessment Model" is increasingly used within the global automotive industry for the objective evaluation of processes and the subsequent improvement of processes at project and organization level. The objective in drawing up this document was to support the interpretation and application of the model for the automotive industry and to provide guidance and recommendations to increase the comparability of assessments results.

Acknowledgements

Our thanks go to the following companies, in particular to the employees involved, for their participation in the preparation of this document:

- BMW AG
- Robert Bosch GmbH
- Brose Fahrzeugteile GmbH & Co. KG
- Continental AG
- Daimler AG
- Ford Motor Company
- Knorr Bremse GmbH
- Kugler Maag CIE GmbH
- Schaeffler AG
- Volkswagen AG
- ZF Friedrichshafen AG

Thanks are also due to all who have provided suggestions for improvement as well as those organizations represented in the editorial circle.

Content

Terms and glossary

In the following, definitions of terms used in the present volume are provided. When applicable, a citation of the definition provided in the ISO/IEC330xx process assessment series of standards is given in italic letters.

Please refer to ISO/IEC 33001:2015 for a full glossary of the terms used the ISO/IEC 330xx series *[ISO33001]*.

Term	Definition
Assessing organization	The organization which performs the assessment. Usually the lead assessor and other assessment team members are part of the assessing organization.
Assessment log	The formal documentation of the execution of an assessment drawn up by the assessor. The assessment log is the evidence of the assessor's assessment activities and is provided to the certification body.
Assessment scope	*Definition of the boundaries of the assessment, provided as part of the assessment input, encompassing the boundaries of the organizational unit for the assessment, the processes to be included, the quality level for each process to be assessed, and the context within which the processes operate.* *→ [ISO/IEC 33001:2015, 3.2.8]*
Assessment sponsor	*Individual or entity, internal or external to the organizational unit being assessed, who requires the assessment to be performed and provides financial or other resources to carry it out.* *→ [ISO/IEC 33001:2015, 3.2.9]*
Assessment team	*One or more individuals who jointly perform a process assessment.* *→ [ISO/IEC 33001:2015, 3.2.10]*
Assessor	*Individual who participates in the rating of process attributes.* *→ [ISO/IEC 33001:2015, 3.2.11]*
Audit	*A systematic, independent and documented process for obtaining audit evidence [records, statements of fact or other information which are relevant and verifiable] and evaluating it objectively to determine the extent to which the audit criteria [set of policies, procedures or requirements] are fulfilled.* *→ [ISO 19011]*

Term	Definition
Automotive SPICE	A process assessment and reference model conformant to the requirements of ISO/IEC 33002:2015. It is primarily addressing the development of embedded software-based systems within the automotive domain. It can be downloaded free of charge on *www.automotivespice.com*.
AUTOSAR	**AUT**omotive **O**pen **S**ystem **AR**chitecture: an initiative by the automotive industry for standardization of software in electronic control units (www.autosar.org).
AUTOSAR domains	Categories used to classify electronic control units by their area of application, e.g. chassis, powertrain, telematics, body.
Process capability	A characterization of the ability of a process to meet current or projected business goals.
Capability level	Point on a scale of achievement of the process capability derived from the process attribute ratings for an assessed process.
Certification body	A central body which administrates the certification information of the trained assessors and classifies the trained assessors by their qualifications and practical experience according to a certification scheme.
Certification scheme	A set of rules and procedures used by a certification body to certify assessors.
Evidence	Artifact or information reflecting practice performance. Evidence are used during assessment to understand process performance and can be documents, oral information, data or information from tools or other sources.
Evidence repository	Repository for storing evidences which have been obtained.
Feedback presentation	A process step at the end of the assessment, when the assessment team provides early feedback on the results of the assessment. It usually covers the main strengths and potential improvements. The set of provisional process capability profiles is also presented if appropriate.
Findings	The evaluations documented by assessors regarding strengths and potential improvements of the organizational unit which was evaluated, based on verbal affirmations from interviews and work products presented (→*Evidence*).

Term	Definition
HIS	"**H**ersteller**I**nitiative **S**oftware" (Manufacturer Initiative Software):Different working groups of German automotive manufacturers (Audi AG, BMW Group, Daimler AG, Porsche AG, VOLKSWAGEN AG), formerly working together in fields not involving competition on the development of software for ECUs, including the subject of process assessments. The activities of the HIS have been terminated in spring 2016.
HIS process scope	A selected set of processes from Automotive SPICE which are assessed (where applicable) in every assessment carried out by the automotive manufacturers represented in the HIS. Due to the termination of the HIS work the HIS scope has been replaced by the → *VDA process scope*.
Indicator	*Sources of objective evidence used to support the assessor's judgment in rating process attributes.* → *[ISO/IEC 33001:2015, 3.3.1]*
ISO/IEC330xxseries	The set of International Standards ISO/IEC 33001:2015, ISO/IEC 33099 defines the requirements and resources needed for process assessment.
Lead assessor	*Assessor who has demonstrated the competencies to conduct an assessment and to monitor and verify the conformance of a process assessment.* → *[ISO/IEC 33001:2015, 3.2.12]*
Process measurement framework	*Schema for use in characterizing a process quality characteristic of an implemented process* → *[ISO/IEC 33001:2015, 3.4.6]*
NDA	**N**on-**D**isclosure **A**greement
OEM	"**O**riginal **E**quipment **M**anufacturer". In the automotive industry this term is used to describe the vehicle manufacturers. (See also →*Tier 1…n*).
Organization assessed	The organizational unit which is assessed. This usually refers to projects in one or more departments in the assessed organization.
Practice level	Lowest level of granularity within the Automotive SPICE process assessment model, determined by the "base practices" and "generic practices" of the processes. Strengths and potential improvements should be traceable to this level and are derived from expectations regarding a state-of-the-art implementation of the practices. Although these expectations constitute good practices in engineering, their achievement might not be satisfied in all cases because "state-of-the art" is highly depending on the context and on individual interpretation.

Term	Definition
Process assess-ment model (PAM)	*Model suitable for the purpose of assessing a specified process quality characteristic, based on one or more process reference models* → *[ISO/IEC 33001:2015, 3.3.9]*
Process reference model (PRM)	*Model comprising definitions of processes in a domain of application described in terms of process purpose and outcomes, together with an architecture describing the relationships between the processes.* → *[ISO/IEC 33001:2015, 3.3.16]*
Process context	*Set of factors, documented in the assessment input, that influence the judgment, comprehension, and* *comparability of process attribute ratings* → *[ISO/IEC 33001:2015, 3.2.16]*
Process (capability) profile	*Set of process attribute ratings for an assessed process* → *[ISO/IEC 33001:2015, 3.2.18]*
Process quality characteristic	*Measurable aspect of process quality; category of process attributes that are significant to process quality.* → *[ISO/IEC 33001:2015, 3.2.10]*
Set of process (ca-pability) profiles	The collective representation of the capability profiles of each process in the scope of the assessment.
SPICE	**S**oftware **P**rocess **I**mprovement and **C**apability d**E**termination Name of the starting project, elaborating the draft of ISO/IEC TR 15504. These days the term “SPICE” is used colloquially to refer to ISO/IEC 330xx.
Tier 1…n	The term “Tier 1…n” is used to refer to suppliers at various levels in the supply chain. Direct suppliers to the OEM are referred to as “Tier 1”, a supplier to a Tier 1 supplier is referred to as a “Tier 2”, etc.
VDA	“**V**erband **d**er **A**utomobilindustrie”, the German association of Auto-motive Industry
VDA process scope	Standard set of processes to be considered in the Automotive domain.

Introduction

The starting point for the development of this publication was the mandate given by the Quality Management Board (QMB) of the VDA QMC to revise Automotive SPICE 2.5 and the existing Blue-Gold Book in order to improve the quality and reproducibility of assessment results. This task was assigned by the QMB to the working group 13 in the VDA QMC.

The objective of working group 13 is the definition of the Automotive SPICE process reference and assessment model. In addition to that, it is the objective of the working group to give necessary clarifications and recommendations for the application of Automotive SPICE in terms of performing assessments and monitoring of resulting process improvements in the development of software-based systems.

To fulfill this mandate, the following activities were performed:

Improving the Automotive SPICE Process Assessment and Reference Model regarding structure, inconsistencies, clarifications and additional concepts. This was done with the publication of the 3.0 version of Automotive SPICE in July 2015 *[AS30]*.

Giving guidelines on the interpretation of Automotive SPICE and on Assessment performance. This is provided by the current publication.

Setting requirements for the qualification of assessors and update existing procedures, training materials and examinations. This will be done in collaboration with the international assessor certification scheme (intacs) to accompany the release and roll-out of this publication *[intacs]*.

The current publication will replace the existing Blue-Gold Print and will be valid with its official publication in the VDA QMC online shop.

The present publication addresses the mandate by proving two parts:

Part 1: Interpretation and rating guidelines

This part provides rules and recommendations for the rating performed in an assessment.

Part 2: Guidelines for performing the assessment

By defining the requirements for the assessment process, it is intended to standardize the procedure, so that the companies involved in an assessment are able to follow a defined assessments approach. This present volume specifies the requirements related to the assessment process, as well as the qualification of assessors carrying out assessments based on Automotive SPICE.

All rules and recommendations for carrying out assessments reflect best practices from assessors having extensive experience in assessments based on Automotive SPICE in various applications.

Besides the knowledge of the participating members and third-party members involved, the present publication leverages other sources giving valuable input, which has been proven in many years of assessment practice and assessor trainings, in particular:

- Book: "*Automotive SPICE - Capability Level 2 und 3 in der Praxis"* by Dr. Pierre Metz *[Metz2016]*

  This book gave significant and valuable input to chapter 2.1.4, "*Strategy and plan"*, chapter 4, *"Rating guidelines on process capability level 2"* and chapter 5, *"Rating guidelines on process capability level 3"*.

- Intacs white paper "*Clarifying Myths with Process Maturity Models vs. Agile" [IntAgile]*

  This publication was considered to provide input to chapter 2.2.2 *"Agile environments"*.

- Intacs training materials for provisional and competent course "Automotive SPICE".

  The existing training material was reviewed and evaluated to consider the current body of knowledge in the domain of assessor education *[intacs]*.

Document scope

The scope of the current document is to support assessments using Automotive SPICE. It addresses the process of performing the assessment and in detail the rating performed in an assessment. It is based on the 3.1 version of the Automotive SPICE Process Assessment and Reference Model.

The intention of this publication is NOT to replace or extend the Automotive SPICE PAM or PRM. Automotive SPICE 3.1 is a full process assessment model (incl. reference model) complying to the requirements of ISO/IEC 33002. It can be used on its own to perform assessments.

The aim of this document is to set guidelines for the application of Automotive SPICE to assist the assessors while planning, executing and reporting the assessment. Beside this it specifically addresses the improvement process which should resolve issues found in an assessment.

The target audience is predominantly assessors which are active in the automotive domain, but can also be seen as an additional input for assessments in other domains. It also addresses other parties or roles involved or affected by an Automotive SPICE assessment like the assessing organization, the assessed organization or the assessment sponsor.

Furthermore, the document is intended to support the understanding of the assessment process and should be taken in case of dissent about the result of an assessment as a basis for clarification.

Relation to ISO/IEC 330xx series

The ISO/IEC 330xx series of international standards define the requirements and resources needed for process assessment. Several standards in the ISO/IEC 330xx family were intended to replace and extend parts of the former ISO/IEC 15504 series.

ISO/IEC 330xx process assessments are conducted based on three core elements:

- process models that define processes, the entities that are the subject to assessment;
- process measurement frameworks that provide scales for evaluating specified attributes; and
- a specification of the process to be followed in conducting assessments.

The intention of the Working Group 13 of the VDA QMC was to provide a domain specific set of documents covering these three elements for performing assessments conformant to ISO/IEC 33002. This has been achieved

- by providing the Automotive SPICE process reference and assessment model *[AS30];*
- by referencing ISO/IEC 33020 *[ISO33020]* as the mandatory process measurement framework for assessment of process capability in the Automotive SPICE PAM and;
- by providing a documented assessment process conformant to ISO/IEC 33002 in chapter 6 of this volume.

Relation to Automotive SPICE

At the beginning of the development of Automotive SPICE 31 different processes have been compiled by the Automotive Special Interest Group (AutoSIG) from ISO/IEC 12207:2008 to provide a process set suitable for assessments in the automotive domain. To allow assessments with reasonable effort the "Herstellerinitiative Software" (HIS) selected a prioritized subset of 15 processes as a standard process scope for assessments. In the past 10 years the majority of assessments have been performed using this so-called HIS scope. With the termination of the work of the HIS at the beginning of 2016 the VDA QMC working group took over the task to adapt this scope to the new versions of Automotive SPICE.

Since this scope has been well-proven over years of assessments the working group 13 decided to maintain the set of processes in principle. This was done in order not to increase the effort for performing an assessment and provide comparability to former assessment results.

It is a principle of process assessments according to the ISO/IEC 330xx series that the process scope (the selection of processes to be investigated in an assessment) might be adapted in accordance with the sponsor and with respect to the purpose of the assessment.

The following VDA process scope provides a standard set, which is recommended to give a sound overview of an assessed project. According to the purpose of the assessment, the process scope for an assessment might be tailored to fewer processes or extended by other processes from Automotive SPICE or other process reference models like ISO/IEC 12207 *[ISO12207]*.

The VDA Scope is based on the release 3.1 of the Automotive SPICE process reference and assessment model *[Automotive SPICE]*.

The following 16 processes are clustered in the VDA process scope:

VDA Scope	
ACQ.4	Supplier Monitoring
SYS.2	System Requirements Analysis
SYS.3	System Architectural Design
SYS.4	System Integration and Integration Test
SYS.5	System Qualification Test
SWE.1	Software Requirements Analysis
SWE.2	Software Architectural Design
SWE.3	Software Detailed Design and Unit Construction
SWE.4	Software Unit Verification
SWE.5	Software Integration and Integration Test
SWE.6	Software Qualification Test
SUP.1	Quality Assurance
SUP.8	Configuration Management
SUP.9	Problem Resolution Management
SUP.10	Change Request Management
MAN.3	Project Management

This process set correlates with the former HIS scope. The additional process was necessary to reflect the structural changes in the engineering processes.

Part 1: Interpretation and rating guidelines

1 Application of interpretation and rating guidelines

1.1 Overview

The purpose of part one of the current publication is to support the assessors in interpreting the Automotive SPICE process reference and assessment models and rating the process attributes for the given target capability level. Since most of the assessments in the automotive domain do not address capability levels higher than 3, no guidelines are provided for level 4 or 5.

Chapter 1, "*Application of interpretation and rating guidelines*" provides an overall guideline on rating in an assessment. It introduces a clearer definition of how to set-up and consider the assessment scope and how to rate based on this assessment scope.

An integral part of the interpretation and rating guidelines is rules and recommendations addressing specific key concepts, application environments and the different capability levels.

In chapter 2, "*Key concepts and overall guidelines*" rules and recommendations related to key concepts introduced or modified with the 3.1 version of Automotive SPICE are given. Further, rules and recommendations for rating in specific application environments are provided.

Chapter 3, "*Rating guidelines on process performance (level 1)*" is related to the process specific outcomes, base practices and work products associated with the capability level1. In this chapter, specific rating rules and recommendations are given for each process of the VDA Scope.

In chapter 4, "*Rating guidelines on process capability level 2*" and chapter 5, "*Rating guidelines on process capability level 3*" specific rating rules and recommendations for each process attribute of level 2 and 3 are given.

1.2 Assessment scope

1.2.1 Motivation

Every process needs a certain input delivered by other processes to produce its desired outputs. A root cause for diverging assessment results is a different consideration of these inputs when rating the process performance attribute on level 1. In case the input is not complete, the output of the receiving process will also be incomplete in terms of a given assessment scope. As an example, the completeness of the system test cases in the System Qualification Test Process SYS.5 will directly depend on the completeness of the system requirements.

Recent years have shown that there are different approaches regarding whether and to what extent an incomplete input affects the rating of the receiving process. This is often due to different assessment purposes of the assessment. In fact, these assessment purposes may influence the decision of an assessor when rating the process performance attribute.

Typically, the following scenarios are considered:

- **Assessment purpose "process improvement"**
  The purpose of the assessment is to provide a base for process improvement.

- **Assessment purpose "process-related product risk"**
  The purpose of the assessment shall give evidence for process risk impacting on the quality of a specific product release.

"Process improvement":

If improvement potentials to one specific process are to be identified, a weakness in other processes should not affect the rating. If all base practices in the process are fully performed and the corresponding outcomes are achieved with respect to the locally available input, there is no reason to downrate the process attribute PA 1.1. For this assessment purpose, it is not relevant in terms of improvement of the process, whether its input is complete with respect to a defined product release.

In this case, a set of exemplary input for the assessed process might be sufficient to provide a sound capability rating and identify any necessary improvement potential. Any input with a sufficient quantity and quality will allow a prediction, how this process will perform in a comparable development environment.

Example: *Consider the situation where all base practices of the SYS.5 process are fully performed and the corresponding outcomes are achieved, but only 20% of the stakeholder requirements of the project have been processed in SYS.2.*

With respect to process improvement the conclusion can be drawn that the process would be fully capable (F) in principle if applied as is in another development. There will be no need to derive any improvement actions based on the assessment result for the SYS.5 process.

“Process-related product risk”:

On the other hand, a successful implementation of the process performance might be interpreted as signifying that the system has been fully tested with respect to the full functionality to be delivered. This means that at the end the process output needs to cover all stakeholder requirements. If this is not the case, the process attribute PA 1.1 might be downrated.

For process-related product risk, the most important criterion is, whether a given set of top level requirements has been processed correctly and completely in the chain of all assessed processes, thus resulting in a product which is “ready for delivery”.

Example: *Besides the achievement of the outcomes within the SYS.5 process, the simple question arises: “Was the system completely tested for delivery according to the given top-level stakeholder requirements?” In the scenario described above, the answer would clearly be “no”, because only 20% of the stakeholder requirements have been tested (see example above) resulting in a significant risk that specified system functionalities will not operate as defined.*

Within this scenario, an assessor might consider the completeness of the SYS.5 output with respect to the stakeholder requirements. This may lead to the decision that the process attribute PA 1.1 is only largely (L) achieved, although all outcomes of SYS.5 are fully achieved with respect to the locally available input.

Both approaches lead to diverging assessment results. Engineering organizations have been confused about the capability achieved, depending on the approach the assessor has had in mind. As a result, necessary information needed for process improvement might have been lost, if the completeness of the process output with respect to the stakeholder requirements was in focus of the assessment and vice versa.

There is no clear indication given in the PAM, how such assessment purposes shall be considered. It therefore is the aim of the authors of this publication that the process assessment model can be applied consistently in order to produce comparable and reproducible assessment results, which provide the necessary information needed for both approaches. Restricting the application of Automotive SPICE to either one or the other is not seen as a reasonable and constructive approach.

In general, there is no conflict between the both use-cases; the differences and confusion due to different, and often imprecisely defined, assessment scopes. To solve this issue, a clearer definition of the assessment scope and a definition how to consider the process context is provided in chapter 1.2.2 and 1.2.3 of this publication. The latter has been elaborated under the following constraints and assumptions:

- Comparability of assessment results cannot be reached unless the project scope comprising the process context is comparable.
- A rating of the base practices shall always be made with respect to the input locally available to the process under investigation.
- The coverage of the scope in terms of completeness shall always be made using distinctive rules for rating the process performance attribute.
- The rules for downrating the process performance attribute shall inhibit a “Fully” rating of the process attribute PA 1.1.
- Completeness with respect to the scope of the assessment may be justified by the assessed organization by means of reviews or other techniques.

To enable a consistent application of the Automotive SPICE PAM, chapter 1.2.3 provides such distinctive rating rules and gives a guideline to define the process context in the assessment scope. Templates and examples are given in the annex.

1.2.2 Defining the assessment scope

As defined in ISO/IEC 33001, 3.2.8 the assessment scope shall provide

"The definition of the boundaries of the assessment, provided as part of the assessment input, encompassing

- *the boundaries of the organizational unit for the assessment, the processes to be included, the quality level for each process to be assessed and*
- *the context within which the processes operate (process context)".*

1.2.2.1 Defining the boundaries of the assessed organizational unit

As defined in ISO/IEC 33002, the boundaries of the assessed organizational unit according to the definition in ISO/IEC 33001 shall be given in the assessment scope. The definition of the organizational boundaries shall be given in terms of

- the localization of the involved organizational unit(s) and
- the responsibilities of the involved organizational unit(s).

These boundaries shall always be defined with respect to the defined processes (see chapter 1.2.2.2) and the defined process context (see chapter 1.2.3). In summary, the boundaries shall identify which part of the organization is responsible for the performance of the given processes in the scope and provide information about the location of the development sites. This is a necessary input for the planning of the assessment.

1.2.2.2 Defining the processes to be included

The VDA Scope defined at the beginning of this document provides a standard selection of processes that are recommended for assessment to give a comprehensive overview of an assessed project. Depending on the purpose of the assessment, this may be tailored or extended.

The processes to be assessed shall be identified. In case of tailoring of the VDA scope a rationale shall be documented for choosing this specific set of processes with respect to the purpose of the assessment.

Each process in the assessment scope shall be assessed and the result shall be documented in the assessment report. To ensure a sufficient base for rating, each process defined in the scope shall be at least once performed.

Exceptionally there might be the need to exclude or add processes after agreement of the assessment scope, e.g. during the execution of the assessment. Any exclusion of processes in the rating shall be documented by a modified assessment scope and shall be approved by the sponsor of the assessment. An exclusion of a process must not be done based on a “not applicable” classification of the process.

1.2.2.3 Defining the target capability level

Since the measurement framework used in Automotive SPICE is applicable for rating capability, the term “*capability level”* as a refinement for the *“quality level”* is used. There are five capability levels specified in ISO/IEC 33020 for the assessment. As mentioned before, the rules and recommendations given in this publication are only considering capability levels 1 to 3, due to the fact that this covers most of the Automotive SPICE assessments in the automotive domain.

Since the planning of the assessment is significantly influenced by the choice of the target capability level, the intended maximum capability level to be assessed shall be defined as part of the assessment scope.

1.2.3 Defining the process context in the assessment scope

In ISO/IEC 33001, 3.2.16 the process context is defined as

> *“the set of factors, documented in the assessment input, that influence the judgment, comprehension, and comparability of process attribute ratings”.*

When defining the process context, the boundaries within which the processes operate in terms of

- a set of stakeholder requirements and
- a set of change requests

shall be identified.

As a default, we suggest the following categories of process contexts are offered when defining the assessment scope:

Process context category A (Parts of a product / delivery)

Exemplarily, for this category the process context may be defined as:

- All stakeholder requirements and changes assigned to a defined organizational unit.
- All stakeholder requirements and changes related to a defined architectural element.
- All stakeholder requirements and changes to be implemented between two defined project milestones.
- All changes and affected stakeholder requirements in a (delta) project developing additional functionalities based on an existing system or software.
- All changes between two defined project milestones.
- All software requirements implemented by changed processes.
- A set of stakeholder requirements and changes from different projects to enable a capability rating of all processes in the scope.

A process context of category A can be chosen for various assessment purposes such as

- internal process improvement
- supplier capability determination or benchmarking, or

Process context category B (Entire product / delivery)

Exemplarily, for this category the process context may be defined as:

- All stakeholder requirements and changes valid for a specific product release
- All software requirements and changes valid for a specific software release operating in a defined system environment.
- Complete system delivered by a Tier 1 supplier
- A complete software platform delivered by an internal or external organization.

A process context of category B can be chosen for various assessment purposes such as

- the process-related product risk of the delivery in terms of evaluating the current delivery status, and identifying corrective actions
- internal process improvement

- supplier capability determination or benchmarking

Each process attribute rating for the project instances (see chapter 1.2.4) shall strictly remain within the boundaries of the process context in the assessment scope. The following general rules shall be applied to address the completeness of the process output with respect to the defined process context.

[CPL.RL1] If only some, non-critical gaps related to the coverage of the process output regarding the defined assessment scope have been identified, GP 1.1.1 must not be downrated.

[CPL.RL2] If many, and/or critical gaps related to the coverage of the process output regarding the scope have been identified, GP 1.1.1 must not be rated higher than L.

Note: A further downrating to P or N may be justified by significant criticality of the identified gaps or an only partially coverage of the assessment scope.

1.2.4 Defining instances when setting up the assessment scope

Depending on different constraints, the same process might be applied in different process instances within the same project e.g. for parts that are developed using model-based approaches in comparison to parts that are manual coded. Therefore, different process attribute ratings might be derived for different instances of the rated process. The corresponding rating methods are provided in the measurement framework of ISO/IEC 33020, 5.4.1.

There are different use cases, where a separation of a process into instances may be reasonable. Building instances may reflect the need of a higher granularity of the assessment findings due to the execution of the process with different approaches or in separate organizations or locations.

Setting up instances doesn’t change the given scope and process context of an assessment. In terms of the rating rules described in chapter 1.2.3, the assessment scope still applies. If instances are defined, they all shall be rated according to the given scope and the rules shall be applied on each process performance attribute rating of each single instance.

To provide a more detailed understanding of the term “process instance”, the following exemplary use cases are given:

- A project has used standard process version 2 until March 2016, and standard process version 3 since then. If the assessor can clearly see that the usage of these two standard process versions actually do not overlap, a reasonable instantiation may be:
    - A rating of process instance “SWE.3 until March 2016”
    - A separate rating of Process instance “SWE.3 after March 2016”
- Parties responsible for different hierarchical levels in the architecture of a mechatronic product development project use different requirements engineering approaches, e.g.:
    - A rating of process instance “SYS.2 / Mechatronic level”
    - A separate rating of process instance “SYS.2 / ECU level”
    - A rating of process instance “SWE.2 / Application SW level”
    - A separate rating of process instance “SWE.2/ Basis SW level”
- Different reuse strategies used for different parts of the overall SW, e.g.
    - A rating of process instance “SWE.x / Platform code”
    - A rating of process instance “SWE.x / Project specific developed code”
- Different SW development paradigms are used for different parts of the overall SW, e.g.:
    - A rating of process instance “SWE.3 / Model-based development”
    - A rating of process instance “SWE.3 / Manual coding”
- Different sub-projects use different project management approaches, e.g.:
    - A rating of process instance “MAN.3 / SW level”
    - A separate rating of process instance “MAN.3 / Overall project”
- Different organizational units develop different parts of the software. These organizational units might even be located in different geographical locations and regions, with probably different social-cultural backgrounds, e.g.:
    - A rating of the process instances “SWE.x / Standard SW components in the reusable platform – Asia”
    - A rating of the process instances “SWE.x / Standard SW components in the reusable platform – Europe”

- A rating of the process instances “SWE.x / All customer-/application-specific SW components – Germany”

Reasons for assessing different process instances separately can be meaningful e.g.

- in order to have company-internal benchmarking
- for a more accurate understanding of the various characteristics in the organization in order to better launch precise process improvement initiatives

The ratings of the process attributes for each process instance shall be documented in the assessment report.

In case instances are defined, a process is rated independently for each instance thus resulting in separate ratings of the process. This requires an aggregation of the results to a single process attribute rating considering the impact of the instance on the overall rating. The recommendations how to perform the aggregation can be found in chapter 1.3.3.

1.3 General rating practice

1.3.1 Rating outcomes and indicators

According to the ISO/IEC 33002, which defines the requirements for performing process assessments, it is always mandatory to rate the process attributes *[ISO/IEC 33002:2015]*.

Nevertheless, in terms of achieving a structured approach to determine the rating of a process attribute, ISO/IEC 33020 provides the following possibility:

> *Process outcomes and process attribute outcomes may be characterized as an intermediate step to providing a process attribute rating. [ISO/IEC 33002:2015, 5.4]*

As mentioned in the overview chapter, part 1 of this publication provides rating rules and recommendations. These rules and recommendations either affect directly the process attribute rating or address this so-called “characterization of process (attribute) outcomes”.

In the recent years of application of Automotive SPICE, the terminology “rating a base practice” or “rating a generic practice” has been established as a synonym for performing this step of characterization. To avoid confusion in the community, the present publication continues using this terminology when defining rules and recommendations which are not directly affecting process attribute ratings.

Formally, since base and generic practices are indicators and thereby only sources of objective evidence used to support the assessor's judgment in rating process attributes, a rating of indicators is not a defined term in the ISO/IEC 330xx series.

In this context – and since Automotive SPICE has a defined relationship between process outcomes and base practices – the terminology "rating a … practice" means:

> **"Characterizing the outcome based on the indicator to compile a consistent process attribute rating"**

1.3.2 Sampling of work products for rating

The selection of the work products has to be carried out carefully to ensure that work product samples are representative, comprehensive, and provide evidence of the implemented process.

1.3.2.1 Selection of work product samples

The following aspects apply for the selection of work products:

- Coverage of the most important functions, which are relevant for the assessment scope
- Coverage of new functionality, adapted functionality, reused software and platform software according to the assessment scope
- Coverage of the whole spectrum of ASIL levels applied within the assessment scope
- Coverage of manual coding (all programming languages used) and model based development (all modeling tools used), where applicable

Metrics (e.g. number of requirements, cyclomatic complexity, lines of code, number of change requests) can support the selection of work product samples. It can be useful to select units with different complexity to sample the corresponding detailed designs.

For the engineering processes (SYS.x, SWE.x) the following approach is recommended: The assessor chooses stakeholder requirements based on above-mentioned aspects. The work products selected for evaluating the indicators of the processes should mark a clear path through the engineering life cycle. The same approach should be applied when evaluating supporting processes such as change management or problem management.

Although the assessed organization may propose certain work products, it remains the assessor's decision to which extent these work products are considered for the process attribute.

1.3.2.2 Plausibility checks of work product samples

All documents used as candidate for objective evidence have to be checked for consistency, in terms of plausibility of the last change time stamp and appropriateness of the change history. The latter can be easily checked by inspecting the history of the work product in the respective tool which is used for configuration or document management. If a document has been initially generated shortly before the assessment it should not be considered for the rating of the process attribute in question unless there is a plausible reason for the late documentation.

The history of the work product should show an appropriate life cycle and a number of versions which correlates with the update cycle of the respective work product.

For instance, it could be expected that if a schedule should be updated on a weekly basis there is at least one version per week (or some evidence that an update was not necessary). Technical documents tend to have more versions than plans. However, if the architecture is based on a platform, there may not be that many versions. It is up to the assessors to check whether the number of versions reflects appropriately the life cycle and status of the project and fulfills the purpose of the process attribute which is assessed.

1.3.2.3 Content-related examination

The content-related examination of the work products should always cover the whole scope of the assessment.

This means based on the criteria for the selection of the work products samples the whole scope shall be represented.

In the limited time it is not possible to cover all aspects of the project. Nevertheless, the samples shall also be checked regarding the right content. For the content of work products, the work product characteristics can be used as guidelines.

The system requirements for example are not only to be checked to determine whether there are linked stakeholder requirements but also if the system requirements reflect the intention of the stakeholder requirements. Another example would be to check the unit tests against the detailed design. The engineer should explain the detailed design. The unit tests are then checked against the detailed design. Inconsistencies found between the test cases and the explanation of the detailed design shall be considered when rating the process attributes.

Automotive SPICE shall not be mistaken for a checklist. The assessor has the duty to check appropriate instantiation of documentation to cover the different process attributes. Appropriateness is based on e.g. the scope, the size and complexity of the project team (e.g. distributed development), the size and complexity of the product, the timeline, and other influencing factors as defined in the process context.

1.3.3 Aggregation of process attribute ratings

It is recommended to use the rating method R2 from ISO/IEC 33020 for the rating of each process attribute.

This means,

1) firstly, to rate each process attribute for each process within the scope of the assessment for each process instance;
2) and secondly, aggregating the process attribute ratings of the process instances.

An aggregation of the process attribute ratings of all process instances is mandatory. This means, in the assessment report there will be one additional set of process attribute ratings for the aggregation.

The aggregation is done according to the following schema ("one dimensional aggregation using arithmetic mean" according to ISO/IEC 33020):

1) Firstly, in accordance with ISO/IEC 33020 NPLF rating values can be expressed as interval values as follows:

   N → 0; P → 1; L → 2; F → 3

   with rounding the result to the nearest integer (by rounding up or down), and converting the result back to the corresponding ordinal rating. Rounding rules are: rounding down to the nearest integer when the average value is less than the midpoint between consecutive integers; rounding up if the average value is at or above the midpoint between consecutive integers.

2) Secondly, the aggregation can be done

   a. by calculating an arithmetic mean, or
   b. by assigning these internal values a percentage weighting first, and then converted back to the ordinal NPLF rating scale. Weightings and their rationale must be explained in the assessment report, and may depend on e.g.
      - size of personnel of organizational unit/ sub-project
      - strategic significance of the product, e.g. commodity vs. new innovative products
      - contribution to the revenue in %
      - criticality of product parts, e.g. a risk class according to ISO 26262

	Process instance **A**	Process instance **B**	Process instance **C**	Aggregated rating
2a. Arithmetic mean without any weighting of process instances	**L (2)**	**L (2)**	**F (3)**	(2+2+3) / 3 → **L (2.33)**
	P (1)	**L (2)**	**F (3)**	(1+2+3) / 3 → **L (2)**
	N (0)	**P (1)**	**F (3)**	(0+1+3) / 3 → **P (1.33)**
2b. Arithmetic mean with weighting	**L (2)** 70%	**L (2)** 15%	**F (3)** 15%	(2*0.7+2*0.15+3*0.15) → **L (2.15)**
	P (1) 70%	**L (2)** 20%	**F (3)** 10%	(1*0.7+2*0.2+3*0.1) → **P (1.4)**
	N (0) 30%	**P (1)** 20%	**F (3)** 50%	(0*0.3+1*0.2+3*0.5) → **L (1.7)**

Each row represents a process as defined in the assessment scope.

1.4 Application of rating rules and recommendations

1.4.1 Objective

In chapter 1 to 5 of this document, rules and recommendations are provided. They are intended to reduce the spread of rating decisions for the same specific process attributes or indicators in terms of interpretation, dependencies and consistency. This is seen as one of the key factors by the authors of this publication to improve the quality and reproducibility of assessment results.

Due to the different impact an identified weakness may have on the capability of an assessed process, a differentiation between rules and recommendations with different levels of rigor has been established:

1.4.1.1 Rules

Looking at the formal definition of the term “Rule” taken from Oxford dictionaries *[Oxford]*, the following explanation is provided:

> *“One of a set of explicit or understood regulations or principles governing conduct or procedure within a particular area of activity.”*

A lot of assessments need a specific analysis of the current situation and context, in which the project operates. Each rating requires the assessors’ knowledge to consider the specific circumstances when rating process attributes. That means, exceptions from rules might be necessary to provide an objective and adequate rating. With respect to the formal definition above, “*governing conduct or procedure”* shall not be interpreted in terms of a strict and rigorous regulation which shall be followed under all circumstances. The aim of a rule is to provide rating principles, which are valid in the majority of assessment situations.

Therefore, in terms of rating in an assessment a rule sometimes might be infringed. In this case a justification shall be documented, which describes why the assessor did not follow the specific rule when rating a specific process attribute or indicator. The documented list of infringed rules and corresponding justifications shall be communicated to the assessment sponsor.

The activity on documenting deviations from rules is provided in chapter 6.4.2.3 *“Consolidation”*.

1.4.1.2 Recommendations

The formal definition of the term “Recommendation” from Oxford dictionaries *[Oxford]* is as following:

> *“A suggestion or proposal as to the best course of action, especially one put forward by an authoritative body.”*

The aim of giving rating recommendations is to provide proposals for best course of actions as stated in this formal definition. In an assessment the assessor may consider a recommendation or may not, depending on his objective judgment whether the recommendation is applicable in the context of the rating decision. Nevertheless, also recommendations should provide the best approach in the majority of assessment situations. So an assessor should normally follow this recommendation, but if he does not, there is no need to document it.

The differentiation of recommendation from rules has been made by the authors of this publication based on the expected impact of the respective indicator and based on the analysis of typical incorrect ratings and root causes for differing assessment results which have occurred in the past.

1.4.1.3 Accumulation of rating rules and recommendations

There might be cases in which for one process attribute or indicator rating different rules and/or recommendations apply in parallel. If for instance two different rules are requiring a downrating of at least one step, there should be no automatism to downrate at least by two steps if applying both rules. It is the task of the assessor to decide, whether the specific weaknesses found in the assessment requires an accumulation of the rules or not.

1.4.2 Terminology

For the formulation of rules and recommendations a defined terminology is used in this document. An overview of used terminology and an additional explanation is given in the following table, if applicable:

1.4.2.1 Rules [RL]

	Wording	Explanation
1	If …, PAx.y must not be rated F. If …, the indicator … must not be rated F.	Any rating other than F might be chosen, depending on the impact of the detected weakness.
2	If ..., the indicator … must not be rated higher than N / P / L.	-
3	If …, the indicator … must not be down-rated.	The found issue shall not lead to a down-rating.
4	If ..., the indicator … shall be downrated. If ..., the corresponding indicators … shall be downrated.	The indicator(s) shall be downrated for at least one step of the rating scale. It is the decision of the assessor, if a further down-rating is necessary to reflect the identified weakness.
5	If ... the indicator A is downrated / rated N / P / L, the indicator B must not be rated higher.	See Rule 2. This rule is used to ensure consistency within the rating.
6	If ... the indicator A is downrated / rated N / P / L, the indicator B shall be downrated.	See Rule 4. This rule is used to ensure consistency within the rating.
7	If ... the indicator A is downrated / rated N / P / L due to …, the indicator B shall be downrated.	See Rule 6, in case a specific aspect of in-dicator A was the root cause for its down-rating.
8	If ..., this must not be used to downrate the … indicator ….	See Rule 3, in case a specific aspect shall be excluded as a root cause for downrat-ing.

In general, the term “downrate” means that the initial rating of the indica-tor(s) without applying the rule shall be reduced. The degree of downrating depends on the significance and number of identified weaknesses.

1.4.2.2 Recommendations [RC]

	Wording	Explanation
1	If ..., the indicator … should not be rated F.	A rating other than F should be chosen, depending on the impact of the detected weakness.
2	If ..., the indicator … should not be rated higher than N / P / L.	-
3	If ..., the indicator … should not be down-rated.	The found issue should not lead to a down-rating.
4	If ..., the indicator … should be downrated.	The indicator should be downrated for at least one step of the rating scale.
5	If ..., this should not be used to downrate the … indicator ….	See Recommendation 3, in case a specific aspect should be excluded as a root cause for downrating.
6	If ... the indicator A is downrated / rated N / P / L, the indicator B should not be rated higher.	See Recommendation 2. This recommendation is used to support consistency within the rating.
7	If ... the indicator A is downrated / rated N / P / L, the indicator B should be downrated.	See Recommendation 4. This recommendation is used to support consistency within the rating.
8	If ... the indicator A is downrated / rated N / P / L due to …, the indicator B should be downrated.	See Recommendation 7, in case a specific aspect of indicator A was the root cause for its downrating.
9	If ... the indicator A is downrated / rated N / P / L, it should have no influence on …	This rule is used to support consistency within the rating.
10	If ... the indicator A is downrated / rated N / P / L due to ..., this should be in line with the rating of the indicator …	"To be inline" does not mean that the ratings should be the same. It should be checked, whether both ratings have been performed based on the same insight. This is especially related to the findings obtained during the assessment. Ratings which differ by more than one step of the rating scale might be an indicator of inconsistency.

1.4.3 Consistency of process attribute rating

In each part of the rating guidelines in chapter 3 to 5, specific chapters for assuring the consistency of the process attribute ratings are provided. The aim of these chapters is to describe and visualize the relations which are explicitly and implicitly contained in the Automotive SPICE process assessment model and give appropriate rules and recommendations to address the dependencies. It is not the aim of these guidelines to cover all dependencies that exists in the complete set of the Automotive SPICE PAM processes for each capability level of the measurement framework. The given rules and recommendations are restricted to

- the dependencies having significant impact on the process attribute rating and those which are not obvious;
- the processes of the VDA scope; and
- the capability levels 1 to 3.

The processes described in the Automotive SPICE PAM are intended to cover self-contained topics like project management or system requirements analysis, which can be rated individually on the defined capability scale.

On the one hand base practices are applicable to a specific process. As the input to a base practice for a specific process may be produced by another base practice of the same process or a base practice of another process there are relations and dependencies within and between processes.

On the other hand, there are generic practices, which are applicable to all assessed processes. There are dependencies of generic practices within the same capability level or between different levels and between generic practices on a specific capability level to level 1 base practices for connected processes.

The following types of dependencies have been identified:

Explicit dependencies:

A Dependencies between base practices within one specific process indicated in the PAM by a specific wording, e.g.:

SWE.6.BP4: Test integrated software ↔ SWE.6.BP3: Select test cases: "Test the integrated software ***using*** *the selected test cases".*

B Dependencies between base practices of one process to base practices / process attribute of another process indicated in the PAM by a specific wording, e.g.:

SWE.6.BP5: Establish bidirectional traceability ↔ SWE.1.BP1: Specify software requirements:
*"**Establish** bidirectional traceability **between** software requirements and test cases included in the software qualification test specification"*

C Dependencies between generic practices of a defined capability level indicated in the PAM by a specific wording, e.g.:

GP 2.1.3: Monitor the performance of the process against the plans ↔ GP 2.1.2: Plan the performance of the process:
*"The process is performed **according to** the plan(s). Process performance is monitored …"*

Implicit dependencies:

D Dependencies between generic practices of a defined capability level and specific connected processes, e.g.:

The general practices for the PA2.1 "Performance Management" are strongly linked to the MAN.3 process "Project Management".

E Dependencies related to the logical process workflow within one process and related to the quality of the input to a specific base practice, e.g.:

Following a strategy requires the strategy to be available, meaningful and in a sufficient quality.

F Dependencies related to the logical workflow between processes and related to the quality of the input to a specific base practice, e.g.:

Elaborating test cases in a software test specification requires the linked requirements to be documented with a sufficient level of quality.

G Dependencies related to the logical workflow between processes and related to the quantity of the input necessary to achieve a specific process purpose e.g.:

Testing the software requires the software requirements to be available in a sufficient volume.

As dependencies of type A and C are explicitly contained in the PAM, they have to be considered when rating the processes, the process outcomes or the corresponding indicators with respect to a specific capability level.

Dependencies of type B and D are caused by relations in the PAM crossing the boundary of a specific process or capability level.

In terms of rating consistency of each process attribute rating, the identified dependencies between indicators are addressed by a corresponding rule or recommendation depending on its impact on the process attribute rating.

In case of type E and F dependencies, the rating of an indicator requires a certain quality of an input from a specific base practice. This means, if the delivering process is not capable of delivering its output or parts of it (e.g. the output is not complete or has low quality), then it seems likely that the receiving process is not capable of producing those parts of the intended output which are based on the output of the delivering process.

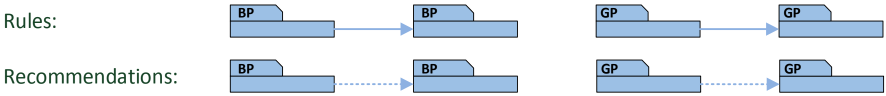
In fact, the receiving process might have compensated the weakness of the delivering process. As a consequence, for these dependencies only rating recommendations have been provided in the following generic wording:

“[XXX.RCy] If the indicator BPn of process X is downrated, this should be in line with the rating of the indicator BPm of process Y.”

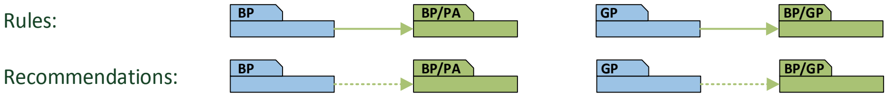
In case of dependencies of type G, the rating of an indicator requires a certain **quantity** of an input from a specific base practice. The links for the corresponding relation can be obtained from the given diagrams. The dependencies themselves are only addressed by the rules considering the scope of the assessment as described in chapter 1.2, *“Assessment Scope”*.

To illustrate relevant dependencies of type A to F and their consideration in the rules and recommendations specific diagrams are provided in the rating consistency chapters of chapter 3 to 5.

In these diagrams in-process (A,E) or in-level (C) dependencies are indicated by solid blue arrows for those resulting in a rule and dashed blue arrows for those resulting in a recommendation:



Out-of-process (B,F) or out-of-level (D) dependencies are indicated by solid (rule) and dashed (recommendation) green arrows:



In case a blue target box is shown, the dependency is located within the same process or capability level. A green box represents a target indicator which is located outside the rated process or capability level.

2 Key concepts and overall guidelines

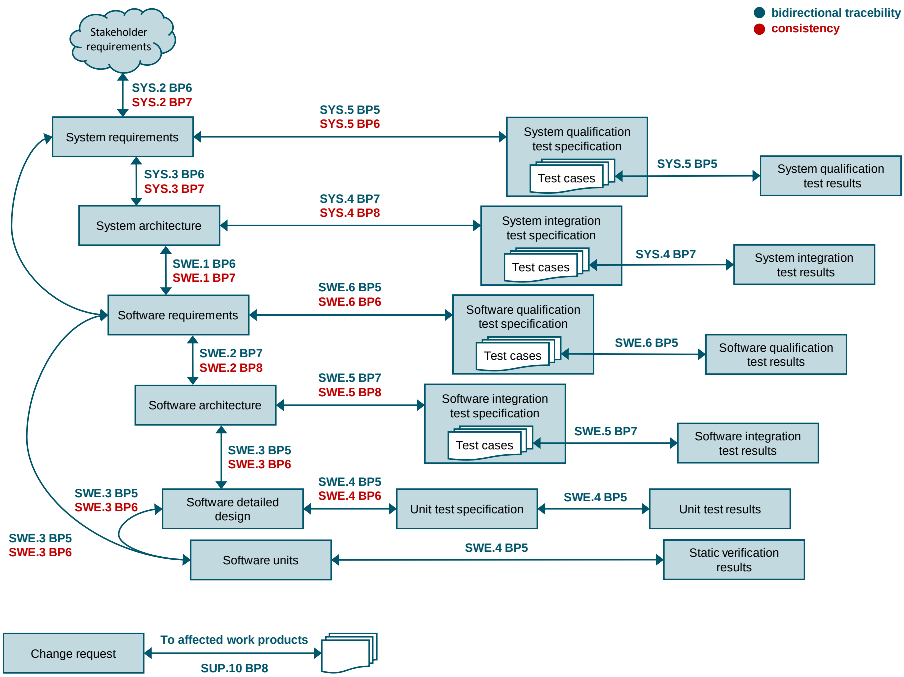
2.1 Specific terms used in base practices

2.1.1 Traceability and consistency

In the Automotive SPICE PAM VDA scope traceability and consistency are addressed by base practices in the engineering processes and in the Change Request Management process. Furthermore, consistency is addressed in the Project Management process.

In the engineering processes in the PAM the previously combined assessment of traceability and consistency is split into separate base practices to enable a more precise rating of each e.g. in case that traceability is given but no thorough consistency.

The following figure shows the relationships respectively for traceability and consistency:



In the engineering processes traceability base practices are applied between

- affected work products of the processes on the left side of the “V” model
- affected work products of the processes on the left side and the corresponding processes on the right side of the “V” model
- test cases and test results on the right side of the “V” model

In the engineering processes consistency base practices are applied between

- affected work products of the processes on the left side of the “V” model
- affected work products of the processes on the left side and the corresponding processes on the right side of the “V” model

In the Change Request Management process in PAM 3.1 a new base practice is applied for

- traceability between change requests and the corresponding problem reports and
- based on the traceability of the engineering processes the traceability between change requests and affected work products to support consistency, completeness and impact analysis.

In the Project Management process, a base practice is applied for

- consistency of estimates, activities, schedules, plans, interfaces, and commitments for the project across affected parties.

On level 2 the way to achieve traceability between work products may be defined within the work product requirements (GP 2.2.2 Define the requirements for documentation and control of the work products).

2.1.1.1 Rating recommendations

Purpose of traceability

According to ISO/IEC/IEEE 24765 traceability is “the degree to which a relationship can be established between two or more products of the development process, especially products having a predecessor-successor or master-subordinate relationship to one another” *[ISO24765]*.

Bidirectional Traceability enables or supports

- analysis of dependencies in both directions,
- analysis of requirements coverage,

- status tracking for implementation of requirements, establishing of test cases, applying of tests etc.,
- debugging,
- impact analysis and risk assessment for changes,
- impact analysis and risk assessment for changing technology,
- impact analysis on cost, schedule and technical impact,
- impact analysis to the operating environment,
- maintenance of all affected work products in case of applying changes,
- maintenance of work products across revisions and
- consistency.

Granularity of traceability

The granularity is required to be respectively at least on the lowest granularity mentioned in the PAM:

- single stakeholder requirement
- single system requirement
- single system architecture element
- single software requirement
- single software architecture component
- single software detailed design element
- single software unit
- single verification criterion
- single test case
- single test result
- single change request
- single problem record

Recommendations and rules:

[TAC.RC.1] If the granularity is not at least on the lowest granularity mentioned above, the traceability indicator should be downrated.

Related to:

- SYS.2.BP6 “Establish bidirectional traceability”
- SYS.3.BP6 “Establish bidirectional traceability”
- SYS.4.BP7 “Establish bidirectional traceability”
- SYS.5.BP5 “Establish bidirectional traceability”

- SWE.1.BP6 “Establish bidirectional traceability”
- SWE.2.BP7 “Establish bidirectional traceability”
- SWE.3.BP5 “Establish bidirectional traceability”
- SWE.4.BP5 “Establish bidirectional traceability”
- SWE.5.BP7 “Establish bidirectional traceability”
- SWE.6.BP5 “Establish bidirectional traceability”
- SUP.9.BP7 “Initiate problem resolution”
- SUP.10.BP8 “Establish bidirectional traceability”
- Output WP 13-22 “Traceability record”
- Output WP 13-21 “Change control record”

Evidence for traceability

As evidence for traceability is defined in the PAM:

- For the engineering processes the traceability record (WP 13-22)
- For the change request management process the change control record (WP 13-21)

Recommendations and rules:

[TAC.RC.2] If there is no documented evidence for the traceability between related work products on the required granularity (see above) the traceability indicator should be downrated.

Related to:

- SYS.2.BP6 “Establish bidirectional traceability”
- SYS.3.BP6 “Establish bidirectional traceability”
- SYS.4.BP7 “Establish bidirectional traceability”
- SYS.5.BP5 “Establish bidirectional traceability”
- SWE.1.BP6 “Establish bidirectional traceability”
- SWE.2.BP7 “Establish bidirectional traceability”
- SWE.3.BP5 “Establish bidirectional traceability”
- SWE.4.BP5 “Establish bidirectional traceability”
- SWE.5.BP7 “Establish bidirectional traceability”
- SWE.6.BP5 “Establish bidirectional traceability”
- SUP.10.BP8 “Establish bidirectional traceability”
- Output WP 13-22 “Traceability record”
- Output WP 13-21 “Change control record”

Methodology/approach (tool support) for traceability

The PAM in general does not prescribe any methodology/approach or tools. The same applies for the application of traceability.

The selected methodology/approach for traceability shall be appropriate to handle the complexity of the product. For complex systems/projects a tool support is recommended to handle the complexity.

Recommendations and rules:

> **[TAC.RC.3]** If the project is not using an automatized tool based approach but a sample based check confirmed that the project complexity is covered sufficiently by maintaining the traceability manually, this should not be used to downrate the traceability indicator.

Related to:

- SYS.2.BP6 “Establish bidirectional traceability”
- SYS.3.BP6 “Establish bidirectional traceability”
- SYS.4.BP7 “Establish bidirectional traceability”
- SYS.5.BP5 “Establish bidirectional traceability”
- SWE.1.BP6 “Establish bidirectional traceability”
- SWE.2.BP7 “Establish bidirectional traceability”
- SWE.3.BP5 “Establish bidirectional traceability”
- SWE.4.BP5 “Establish bidirectional traceability”
- SWE.5.BP7 “Establish bidirectional traceability”
- SWE.6.BP5 “Establish bidirectional traceability”
- SUP.10.BP8 “Establish bidirectional traceability”
- Output WP 13-22 “Traceability record”
- Output WP 13-21 “Change control record”

Purpose of consistency

Consistency

- addresses content and semantics by ensuring that all project related work products are in line with each other across affected parties and not in contradiction to each other and
- reduces the risk of misinterpretation and faults.

Evidence for consistency

As evidence for consistency the Review Record (WPC 13-19) is defined.

Recommendations and rules:

[TAC.RL.1] If there is no documented evidence for the consistency between related work products on the required granularity (see above) the consistency indicator shall be downrated.

Related to:

- SYS.2.BP7 “Ensure consistency”
- SYS.3.BP7 “Ensure consistency”
- SYS.4.BP8 “Ensure consistency”
- SYS.5.BP6 “Ensure consistency”
- SWE.1.BP7 “Ensure consistency”
- SWE.2.BP8 “Ensure consistency”
- SWE.3.BP6 “Ensure consistency”
- SWE.4.BP6 “Ensure consistency”
- SWE.5.BP8 “Ensure consistency”
- SWE.6.BP6 “Ensure consistency”
- MAN.3.BP9 “Ensure consistency”
- Output WP 13-19 “Review record”

2.1.1.2 Rating consistency within the engineering processes

For the engineering processes consistency is supported by bidirectional traceability.

Recommendations and rules:

> **[TAC.RC.4]** If for engineering processes the traceability indicator is downrated, the consistency indicator should not be rated higher.

Related to:

- SYS.2.BP6 “Establish bidirectional traceability”
- SYS.2.BP7 “Ensure consistency”
- SYS.3.BP6 “Establish bidirectional traceability”
- SYS.3.BP7 “Ensure consistency”
- SYS.4.BP7 “Establish bidirectional traceability”
- SYS.4.BP8 “Ensure consistency”
- SYS.5.BP5 “Establish bidirectional traceability”
- SYS.5.BP6 “Ensure consistency”
- SWE.1.BP6 “Establish bidirectional traceability”
- SWE.1.BP7 “Ensure consistency”
- SWE.2.BP7 “Establish bidirectional traceability”
- SWE.2.BP8 “Ensure consistency”
- SWE.3.BP5 “Establish bidirectional traceability”
- SWE.3.BP6 “Ensure consistency”
- SWE.4.BP5 “Establish bidirectional traceability”
- SWE.4.BP6 “Ensure consistency”
- SWE.5.BP7 “Establish bidirectional traceability”
- SWE.5.BP8 “Ensure consistency”
- SWE.6.BP5 “Establish bidirectional traceability”
- SWE.6.BP6 “Ensure consistency”
- Output WP 13-22 “Traceability record”
- Output WP 13-19 “Review record”

2.1.1.3 Redundancy

In the engineering processes for SWE.1 und SWE.3 there are parallel paths for traceability and consistency established e.g. for SWE.1:

- First path from system requirements to system architecture (in SYS.3) to software requirements (in SWE.1) and

- Second path from system requirements (in SYS.2) directly to software requirements (in SWE.1).

As long as applicable and no added value would be provided by such traceability (e.g. by supporting differing views) it is not necessarily required to maintain both paths with full granularity, e.g. in case of a one microcontroller system the link from system architecture to software requirements is relatively trivial.

The same applies in case of software development only where consistency and bidirectional traceability has to be ensured between stakeholder requirements and software requirements directly. Redundancy via system requirements should be avoided if it would not provide any added value.

Recommendations and rules:

[TAC.RL.2] If traceability and consistency is only established for one path and not for the other redundant path, the traceability indicator must not be downrated.

[TAC.RL.3] If only one path is explicitly established and the other path can't be derived from the established path, the traceability indicator shall be downrated.

Related to:

- SYS.2.BP6 “Establish bidirectional traceability”
- SYS.2.BP7 “Ensure consistency”
- SYS.3.BP6 “Establish bidirectional traceability”
- SYS.3.BP7 “Ensure consistency”
- SWE.1.BP6 “Establish bidirectional traceability”
- SWE.1.BP7 “Ensure consistency”
- SWE.2.BP7 “Establish bidirectional traceability”
- SWE.2.BP8 “Ensure consistency”
- SWE.3.BP5 “Establish bidirectional traceability”
- SWE.3.BP6 “Ensure consistency”
- Output WP 13-22 “Traceability record”
- Output WP 13-19 “Review record”

2.1.2 Summarize and communicate

2.1.2.1 Introduction

Communication is a key element for an effective and efficient project performance. The purpose of communication is to provide relevant work products to stakeholders. The work products that need to be communicated at a minimum are mentioned in the respective BPs of the PAM. Stakeholders are at a minimum those who need certain work products to begin and perform their work properly and those who are involved in the management of activities and work products. Depending on the work product to be provided the stakeholders of a project may encompass:

- Project participants
- Project managers
- Customers
- Platform developers
- External service/product providers
- Foreign locations
- Senior managers
- System and software architects
- Testers
- Quality assurance staff
- Configuration managers
- Change control board members
- Purchasing staff

The concept of communication has been revised in the Automotive SPICE PAM 3.1.

In the requirements analysis processes (SYS.1, SYS.2, SWE.1) and the architecture and design processes (SYS.3, SWE.2, SWE.3) the emphasis is on agreed work products that will be communicated to affected parties.

In the testing processes (SWE.4, SWE.5, SWE.6, SYS.4, SYS.5) the communication is focused on the information about the results of testing and verification.

MAN.3.BP7 requires the set-up of formal communication (e.g. regular meetings with project participants). GP 2.1.5 requires implicitly the definition of providers and addressees of process related information by defining responsibility and authority.

On capability level 1 evidence for communication may consist of any tangible artifact (e.g. emails, meeting minutes, voice recordings, etc.). The term “affected parties” is used here for the group of stakeholders who are directly processing the work products of a certain process in their work. Communication at level 1 does not follow necessarily a plan or procedure.

On level 2 a mechanism for the communication on project level is required to ensure an effective and efficient exchange of information to all relevant stakeholders. Communication channels and artifacts to be used are defined using MAN.3.BP7 in general and in GP 2.1.7 for a particular process.

Communication on level 2 is formalized and planned. Regular meetings and defined communication media are established; efficient communication is also based on decisions and actions recording as well as regular actions follow-up monitoring

On level 3 the communication is defined for the organization in a standard process. The standard may be tailored to the project needs.

The exchange of information can be effective even when the sender and receiver of the information do not directly communicate to each other.

> **[SAC.RC.1]** If the BP for communication of work products (e.g. SYS.2.BP8 for agreed system requirements) is downrated or the BP for summarize and communicate of test results is downrated, this should be in line with the rating of the indicator GP 2.1.7.
>
> **[SAC.RC.2]** If there is evidence that necessary information is not provided to all relevant stakeholders (see examples in the list above), the indicator for “communicate agreed…” and/or the indicator for “summarize and communicate …” should be downrated.

Related to:

- SUP.1.BP4 “Summarize and communicate quality assurance activities and results”
- MAN.3.BP10 “Review and report progress of the project”
- ACQ.4.BP2 “Exchange all agreed information”
- SYS.4.BP9 “Summarize and communicate results”
- SYS.5.BP7 “Summarize and communicate results”

- SWE.4.BP7 “Summarize and communicate results”
- SWE.5.BP9 “Summarize and communicate results”
- SWE.6.BP7 “Summarize and communicate results”
- Output WP 13-04 “Communication record”

2.1.2.2 Rating Recommendations

Agree

The term “agree” that is used in SYS.2, SYS.3, SWE.1, SWE.2 and SWE.3 means that the work products of these processes are analyzed and discussed with the relevant stakeholders and a common understanding has been reached upon these work products. This agreement has to be documented to identify those work products that are ready for further use in the engineering process. Examples for evidences for “agree” are:

- Involvement of experts in the analysis is documented in meeting minutes
- Written feedback from customer
- Status tag “agreed” in the requirements specification
- Affected parties state independently in an assessment that they agree

  [SAC.RL.1] If there are evidences that work products are communicated but not agreed, the respective indicator for communicate must not be rated higher than P.

Related to:

- SYS.2.BP8 “Communicate agreed system requirements”
- SYS.3.BP8 “Communicate agreed system architectural design”
- SWE.1.BP8 “Communicate agreed software requirements”
- SWE.2.BP9 “Communicate agreed software architectural design”
- SWE.3.BP7 “Communicate agreed software detailed design”
- Output WP 13-04 “Communication record”

Summarize

The term “summarize” implies that test results in SYS.4, SYS.5, SWE.4, SWE.5 and SWE.6 are collected, structured, condensed and documented to meet the information needs of the relevant stakeholders. Examples are:

a) Number of performed tests
b) Percentage of failed tests
c) Number of test cases that were skipped
d) Reasons for cancellation
e) Exceptions
f) Abnormal test results

> **[SAC.RL.2]** If test results are not summarized appropriately to cover the aspects a) and b), the respective indicator for “summarize and communicate …” must not be rated higher than P.

> **[SAC.RC.3]** If test results are not summarized appropriately to cover all aspects above, the respective indicator for “summarize and communicate …” should be downrated.

Related to:

- SYS.4.BP9 “Summarize and communicate results”
- SYS.5.BP7 “Summarize and communicate results”
- SWE.4.BP7 “Summarize and communicate results”
- SWE.5.BP9 “Summarize and communicate results”
- SWE.6.BP7 “Summarize and communicate results”
- Output WP 13-04 “Communication record”

Tool based communication

Some development tool suites provide the service of automated emailing when a certain status of work products has been changed (e.g. “Test finished”). Even though it might be beneficial that all project participants receive a message about all status changes, the flood of emails can lead to ignoring of relevant information as it is simply too much information. On capability level 1 this is still acceptable. On capability level 2 the project shall have a communication mechanism ensuring that information needs for all project participants are satisfied efficiently.

> **[SAC.RC.4]** If automated emailing is used to inform all project participants about status changes of work products, the respective indicator for communicate should not be downrated
>
> **[SAC.RC.5]** If automated emailing is used to inform all project participants about status changes of work products and evidence is gathered that emails are systematically not read, the respective indicator for communicate should be downrated
>
> **[SAC.RC.6]** If automated emailing is used to inform all project participants about status changes of work products without customizing to meet the specific information needs communicated by the project participants, the indicator GP 2.1.7 for the respective process should be downrated.

Related to:

- SYS.2.BP8 “Communicate agreed system requirements”
- SYS.3.BP8 “Communicate agreed system architectural design”
- SWE.1.BP8 “Communicate agreed software requirements”
- SWE.2.BP9 “Communicate agreed software architectural design”
- SWE.3.BP7 “Communicate agreed software detailed design”
- SWE.4.BP7 “Summarize and communicate results”
- SWE.5.BP9 “Summarize and communicate results”
- SWE.6.BP7 “Summarize and communicate results”
- SYS.4.BP9 “Summarize and communicate results”
- SYS.5.BP7 “Summarize and communicate results”
- Output WP 13-04 “Communication record”

2.1.3 Verification criteria

Verification criteria define the qualitative and quantitative measures for the verification of a requirement. Verification criteria demonstrate that a requirement can be verified within agreed constraints (e.g. “should work with dirty diesel in winter”).

Verification criteria are not the same as test cases, but are input to them. Verification criteria are necessary especially for non-functional requirements (what shall be checked?) or to understand the preconditions for the test of a single requirement or a set of requirements. The requirements engineer should have analyzed all requirements and should understand the dependencies between the requirements. He shares this knowledge with the tester through the verification criteria. It is a good practice to create the verification criteria when the requirement is created to ensure the verifiability of the requirement.

With Automotive SPICE 3.1 the use of verification criteria changed and are now associated to system/software requirements only (SYS.2.BP5 and SWE.1.BP5 Develop verification criteria). For the architecture, the term evaluation is introduced to ensure that the architecture is suitable to implement the requirements (see SYS.3/SWE.2).

The term “criteria for verification” is used in SWE.4 and should not be confused with verification criteria. SWE.4 “criteria for verification” encompass test cases as well as criteria for other verification methods such as unit test cases, unit test data, static verification, coverage goals and coding standards such as the MISRA rules.

Examples:

- Requirement #1: “The noise of the motor at 2500 rpm shall be less than or equal to 67dBA.”
- Verification criteria for requirement #1: “The measurement of the noise level is performed at 25° Celsius ambient temperatures, with an ambient pressure of 1013 mbar at 1 m distance and the motor in horizontal position.”
- Requirement #2: “The software shall be implemented in the programming language Java.”
- Verification criterion for requirement #2: “Check if only byte code files (.class) are released.”

- Requirement #3: “The system shall react to every input within 25 ms.”
- Verification criterion for requirement #3: “The measurement of the reaction time is done with maximum CPU load between the input interfaces i1 and i2 and the debug output o1.”

2.1.3.1 Rating recommendations

Create verification criteria

The verification criteria shall cover the following aspects:

a) Identification of the requirement to be verified
b) Verification method (e.g. tests, inspections, peer reviews, audits, walkthroughs or analysis)
c) Verification environment
d) Preconditions and special conditions (e.g. with winter diesel)
e) Constraints
f) Success criteria

Identification of a verification method or verification step (e.g. software test, system test) is necessary, but not sufficient for the verification criteria. If special test methods, environments, additional information or constraints are needed to be conducted or to be considered by the verification then this special information has to be documented.

Recommendations and rules:

[VEC.RL.1] If one of the aspects a), b) or f) is missing in the verification criteria, the indicator SYS.2.BP5 / SWE.1.BP5 must not be rated higher than P.

[VEC.RL.2] If the corresponding requirements or corresponding work products (e.g. test plan) contain all aspects above and there are no additional verification criteria defined, the indicators SYS2.BP5 / SWE1.BP5 must not be downrated.

Related to:

- SYS.2.BP5: “Develop verification criteria”
- SWE.1.BP5: “Develop verification criteria”
- Output WP 17-50 “Verification criteria”

2.1.4 Strategy and plan

2.1.4.1 Rating recommendations

Understanding of “strategy”

Having a strategy means that all parties involved in achieving the process outcomes have agreed on the methodological approach, and on how to deal with constraints, in order to achieve these process outcomes.

At a first glance, having a strategy at capability level 1 seemingly implies an overlapping with GP 2.1.1, GP 2.1.5, GP 2.1.6, and GP 2.1.7. This impression appears to be further supported by the fact that e.g. SUP.9 and SUP.10 require for their strategies the definition of responsibilities and defined interfaces, and that SUP.8 further demands resource definition. However, the above does not contradict the distinction between capability level 1 and capability level 2 as we can see in the following:

- SUP processes need a higher degree of formalism at capability level 1 because they cut across all processes. This also applies to ACQ.4, even though this process does not have a distinct strategy BP
- The achievement of capability level 1 of the SUP processes only contributes to the achievement of capability level 2 of all other processes, i.e. the SUP processes do not completely represent all aspects of a capability level 2 capability for a particular process
- For achieving capability level 2 of the SUP and testing processes themselves there still is much more to achieve beyond a strategy at CL1. This includes systematic planning, tracking, and adjustment of schedule, effort, resource consumption, and work product management by a structured team (PA 2.2). In contrast, a worthy capability level 1 strategy may still be the opinion of a single person, so that the overall capability level 1 performance may be achieved e.g. by means of “heroes”, or “firefighters”.

A strategy does not need to be a specific text processing document titled “strategy”. A strategy can be evident in, or indicated by, any physical indicators or other accessible information. Examples:

1) A standard departmental slide representation for an organizational unit describing their purpose, objectives, and an abstract but sufficient explanation of their proceedings, e.g.
   - A centralized, small basis software department with members sitting close to each other, and having already worked together for a considerable time
2) Existence of tools that enforce a certain workflow including GUIs with mandatory edit fields, e.g. document management systems, configuration or change request management
3) Automated, or partially automated, workflows implemented by tools and scripts, e.g.
   - automatically generated test result report frame with traceability links to the test case specification
   - build tools including a static software verification step
   - continuous integration approaches
   - continuous delivery approaches
4) An attribute column in a requirements management tool showing for each requirement which test method is going to be used for verifying it
5) An appropriate flipchart drawing, of which a photograph was taken and stored in a particular location accessible to everyone affected

In addition, the following remains essential:

- The objective of a strategy is that it must be adhered to, and must be effective; just documenting a strategy does not necessarily ensure that it is followed and effective.
- Therefore,
   - the necessary comprehensiveness, and detail of information indicated by example 1) and 2) is always context-dependent;
   - further, people interviewed must independently confirm the strategy, e.g. two testers responsible for different SW components.

The task of the assessor is to check whether a strategy exists in terms of being effective with regard to fulfilling the process outcomes in the concrete context.

[SAP.RL.1] If a strategy is not documented as a specific text processing document titled “strategy” but there is evidence of a strategy known by all relevant parties (see examples above) the strategy-related indicators must not be downrated.

[SAP.RL.2] If a strategy is not effective in terms of achieving the process outcomes, or not adhered to by all relevant parties, then the strategy-related indicators shall be downrated.

Related to:

- SYS.4.BP1 “Develop system integration strategy”
- SYS.4.BP2 “Develop system integration test strategy including regression test strategy”
- SYS.5.BP1 “Develop system qualification test strategy including regression test strategy”
- SWE.4.BP1 “Develop software unit verification strategy including regression strategy”
- SWE.5.BP1 “Develop software integration strategy”
- SWE.5.BP2 “Develop software integration test strategy including regression test strategy”
- SWE.6.BP1 “Develop software qualification test strategy including regression test strategy”
- SUP.1.BP1 “Develop a project quality assurance strategy”
- SUP.8.BP1 “Develop a configuration management strategy”
- SUP.9.BP1 “Develop a problem resolution management strategy”
- SUP.10.BP1 “Develop a change request management strategy”
- Output WP 08-52 “Test plan”
- Output WP 08-13 “Quality plan”
- Output WP 08-04 “Configuration management plan”
- Output WP 08-27 “Problem management plan”
- Output WP 08-28 “Change management plan”

Understanding of “plan”

In Automotive SPICE there are two distinct perspectives on the term “plan”:

1) At capability level 1 the “plan” is the work product indicator for the BP “strategy”. In this respect, a plan always has process-specific content. Accordingly, Annex D.7 of Automotive SPICE explains that:

   - For capability level 1 only the work product indicators of specific WPC “08-xy <specific> Plan” shall be considered.
   - For CL 2 both the work product indicators of the specific WPC e.g. “08-12 Project Plan” and the generic WPC “08-00 Plan” shall be considered.

2) At capability level 2 a “plan” requires more (see Automotive SPICE Annex B), such as

   - tasks to be accomplished including
     a) schedules, milestones and target dates;
     b) effort estimations and allocation; and
     c) critical dependencies.
   - Includes contingency plan for non-completed tasks.

   Accordingly, Annex D.7 of Automotive SPICE explains that for capability level 2 the work product characteristics of both generic WPC “08-00 Plan” and specific WPC “08-xy <specific Plan>” are to be considered.

[SAP.RL.3] If the work product characteristics as explained by the generic work product characteristics ID “08-00 Plan” are missing, this must not be used to downrate the Strategy-BP indicator of the assessed process.

Related to:

- SYS.4.BP1 “Develop system integration strategy”
- SYS.4.BP2 “Develop system integration test strategy including regression test strategy”
- SYS.5.BP1 “Develop system qualification test strategy including regression test strategy“
- SWE.4.BP1 “Develop software unit verification strategy including regression strategy”
- SWE.5.BP1 “Develop software integration strategy”
- SWE.5.BP2 “Develop software integration test strategy including regression test strategy”
- SWE.6.BP1 “Develop software qualification test strategy including regression test strategy”
- SUP.1.BP1 “Develop a project quality assurance strategy”
- SUP.8.BP1 “Develop a configuration management strategy”
- SUP.9.BP1 “Develop a problem resolution management strategy”
- SUP.10.BP1 “Develop a change request management strategy”
- Output WP 08-52 “Test plan”
- Output WP 08-13 “Quality plan”
- Output WP 08-04 “Configuration management plan”
- Output WP 08-27 “Problem management plan”
- Output WP 08-28 “Change management plan”

Work product organization of strategy and plans

The strategy or plan of a process does not necessarily have to be separate artifacts. Strategies of several processes may well be merged into one document. For example, this often happens for

- the system testing-oriented processes
- the change request, and configuration management, strategies, as change requests are to be placed against concrete versions of artifacts, or entire product baselines.

Further,

- strategies and parts of a plan may well be merged into one document, e.g. methods and proceedings are described together with roles and human resources allocated to certain work packages (Example: configuration management tool usage for the tool administrator and baseline-responsible),

- while concrete date/time-oriented information is kept in schedules.

[SAP.RL.4] If the strategies of different processes are combined in the same document this must not be used to downrate the corresponding strategy-BP indicators.

Related to:

- SYS.4.BP1 “Develop system integration strategy”
- SYS.4.BP2 “Develop system integration test strategy including regression test strategy”
- SYS.5.BP1 “Develop system qualification test strategy including regression test strategy”
- SWE.4.BP1 “Develop software unit verification strategy including regression strategy”
- SWE.5.BP1 “Develop software integration strategy”
- SWE.5.BP2 “Develop software integration test strategy including regression test strategy”
- SWE.6.BP1 “Develop software qualification test strategy including regression test strategy”
- SUP.1.BP1 “Develop a project quality assurance strategy”
- SUP.8.BP1 “Develop a configuration management strategy”
- SUP.9.BP1 “Develop a problem resolution management strategy”
- SUP.10.BP1 “Develop a change request management strategy”
- Output WP 08-52 “Test plan”
- Output WP 08-13 “Quality plan”
- Output WP 08-04 “Configuration management plan”
- Output WP 08-27 “Problem management plan”
- Output WP 08-28 “Change management plan”

2.2 Application in specific environments

2.2.1 Model based development

The approach of model-based development can be used for different purposes within the system and software development e.g. models can support the requirements elicitation process or support the development of complex algorithms.

2.2.1.1 Rating recommendations

Models need additional description

Models can be used in different use cases within the development process (e.g. for requirements elicitation, architectural design, detailed design, code generation, verification). It has to be defined and documented what the use case of the model is, e.g. “the system architecture is documented using SysML”.

Modeling notations may be graphical, textual or a mixture of both and may differ depending on the use-case of the model. The syntax and semantics of the notations shall be defined in a more or less stringent way (formal, semi-formal or informal).

Aspects (e.g. design decisions) that the modeling notations cannot express require additional description in natural language that could be provided e.g. in text boxes in the model. The corresponding work product characteristics (Annex B of Automotive SPICE PAM) give guidance for the aspects of the additional description.

The corresponding indicator within the rules depends on the use case of the model in the development process e.g. if the model is used for software requirement elicitation, the corresponding indicator is SWE.1.BP1 or if the model is used for software detailed design, the corresponding indicators are SWE.3.BP1, SWE.3.BP2, SWE.3.BP3.

Recommendations and rules:

> **[MBD.RL.1]** If the use cases for the modeling are not explicitly defined and this aspect is significant in the context of the corresponding indicator, the corresponding indicator shall be downrated.
>
> **[MBD.RL.2]** If the syntax and semantics of the model notation is not defined or not appropriate for the use case and this aspect is significant in the context of the corresponding indicator, the corresponding indicator shall be downrated.
>
> **[MBD.RL.3]** If the additional description is missing or insufficient and this aspect is significant in the context of the corresponding indicator, the corresponding indicator shall be downrated.

[MBD.RL.4] If the additional description is documented in association with the model, the corresponding indicator must not be downrated.

Consistency of additional description

Aspects that cannot be expressed by the modeling notation might be missing, if not documented in some other appropriate form.

If the model itself is part of a development artifact, e.g. for the use case of requirement elicitation the model is part of the requirement specification, it has to be ensured that this additional description in natural language of the model is considered in the following development process.

Recommendations and rules:

[MBD.RL.5] If the additional description in natural language of the model is not considered in the following development process and this aspect is significant in the context of the corresponding indicator, the corresponding indicator must not be rated F.

Refer to chapter 2.1.1 for the generic concept of consistency and traceability.

Models for code generation

If automated code generation is used (a.k.a. graphical programming), then the basis for the code generation is

- already a part of the design or
- derived from the design (traceability between model and design has to be established).

In a software design, there is information which is not usable for code generation but is important to guide the understanding of the software. Examples are textual annotations to graphical elements or additional description (e.g. design decisions).

The unit verification done at the model level shall provide evidence for compliance of the software units with the software detailed design and with the non-functional software requirements.

Traceability and consistency support the compliance of a model and code part. The compliance of additional description (e.g. design decisions) with the model and/or the code is normally shown by reviews.

Recommendations and rules:

[MBD.RL.6] If there is no or insufficient evidence for compliance of the parts of the model used for code generation with

- the detailed design (or the parts of the model used for the detailed design) or
- the non-functional software requirements

and one of these aspects is significant in the context of the SWE.3.BP6, the indicator SWE.3.BP6 must not be rated higher than P.

[MBD.RL.7] If the parts of the model for code generation are not verified using static verification and not tested to provide evidence for compliance of the software units with the software detailed design and with the non-functional software requirements and this aspect is significant in the context of SWE.4.BP3, the indicator SWE.4.BP3 shall be downrated.

[MBD.RL.8] If software units that are generated from the verified model by using a qualified tool chain (and without any further modification after generation) are not statically verified, the indicator SWE.4.BP3 must not be downrated.

Qualified tool chain for the code generation means that there is evidence that the generated code is correct and consistent with the model.

[MBD.RL.9] If software units that are generated from the verified model by using a qualified tool chain (and without any further modification after generation) are not unit tested, the indicator SWE.4.BP4 must not be downrated.

[MBD.RL.10] If software units generated from the verified model are modified and not explicitly statically verified, the indicator SWE.4.BP3 shall be downrated.

[MBD.RL.11] If software units generated from the verified model are modified and not explicitly unit tested, the indicator SWE.4.BP4 shall be downrated.

Not generated code (manual code)

If the model is used for code generation and it contains also parts that are not automatically generated, the rules and recommendations from chapter 3 apply.

2.2.2 Agile environments

Agile software development is based on principles of the Agile Manifesto with the objective to create lightweight development methods. Popular frameworks for agile software development are SCRUM, KANBAN and eXtreme Programming.

Automotive SPICE describes meaningful process principles but does not predefine any concrete lifecycle model, method, tool, templates, metrics, proceedings etc (the WHAT level). This means the Automotive SPICE content resides at a higher level of abstraction than any process implementation (the HOW level) in order to allow for maximum freedom, and, also, for benchmarking. In contrast, agile methods rather reside at the HOW level. Therefore, Automotive SPICE and agile approaches cannot, by definition, contradict each other. The only valid question would be to ask whether concrete process implementations, following or including agile methods or not, actually satisfy the Automotive SPICE principles. Automotive SPICE does not predefine any type of lifecycle model like V- or Waterfall-model.

Agile Methods may support Automotive SPICE requirements and should be compliant to required rules and standards. For example, non-functional requirements, review and documentation criteria or coding guidelines are valid in an agile and non-agile life cycle.

2.2.2.1 Rating recommendations

The rating recommendations in this chapter are based on practical experience and have no pretention of completeness.

The documented practical experience within this chapter are partly not specific to agile development (e.g. missing software architecture) but have been detected often in Automotive SPICE Assessments of projects with agile development methods.

Planning in agile environment

Customer planning requirements are equal in agile and non-agile development. Projects have to ensure that features are delivered and bugs are fixed as agreed and scheduled. The planning methods may differ.

Therefore, the agile project has to ensure that the project planning is in line with the customer release planning.

For example, an agile SCRUM project will ensure that the sequence of sprint cycles will deliver the needed functionality corresponding to the customer requirements, i.e. the planning has to ensure that the agreed features are developed and tested within the sprints before the planned release, and the planning has to be consistent across affected parties and agreed plans.

> **[AGE.RC.1]** If evidences from project planning (e.g. backlog, burn down chart and/or sprint planning) show gaps regarding the release planning and this aspect is significant in the context of MAN.3.BP4, MAN.3.BP9 and SPL.2.BP1, the indicators MAN.3.BP4, MAN.3.BP9 and SPL.2.BP1 should be downrated.

Project life cycle

The chosen project life cycle should fit to the project scope, requirements, deliveries, complexity, etc. Therefore, it may be necessary to create a life cycle according to a standard process with tailoring to meet the project needs.

For example, the customer might continuously deliver requirements to the project and expect continuous integration by the project in order to monitor the progress of the product. An agile development process (e.g. SCRUM or Kanban) may support the customer requirements regarding progress monitoring and incremental requirements delivery.

A negative example would be the following scenario:

a) The customer requires that the supplier has to transfer all engineering work products to the customer including project requirements, software architecture and design, source code, black and white box test cases.

b) But the supplier's development process for the project uses an agile project life cycle (e.g. SCRUM based) which does not produce a software architecture.

[AGE.RC.2] If the defined project life cycle does not fit to project scope, requirements, deliveries, etc., the base practices MAN.3.BP2 should be downrated.

Management of project requirements

In practice, some projects manage the project requirements in a change management or tracking tool in which the requirements are managed within tasks or change requests only. These solutions may have the benefit to trace requirements to tasks and code easily but have the disadvantage that no overview of all project requirements is established. Without an overview of project requirements, the maintenance of requirements is very difficult in regard to impact analysis of changes and getting evidence that all requirements are implemented completely.

For example, a feature has different functions. In development, a first task is issued for development the feature. During the development period, different change requests/tasks to the feature are assigned and implemented to add, change or delete functions of the feature. At project end the requirements of the feature can only be determined by assessing all tasks of the feature.

[AGE.RC.3] If the project development is based on change management without a complete and consistent overview of all project requirements and this aspect is significant in the context of SWE.1.BP3 (for software) and SYS.2.BP3 (for system), the base practices SWE.1.BP3 (for software) and SYS.2.BP3 (for system) should be downrated.

Risk management

Customers, company or project requirements often require integrating risk management for the development projects, and this risk management needs to be integrated into the agile project.

For example, if the customer requires managing of project and technical risks then the project has to identify, mitigate and manage project risks at project management level and technical risks on requirements and architecture level.

[AGE.RC.4] If risk management is required for the project but not integrated in the agile project, the base practices MAN.3.BP5 and MAN.5.BP1 should be downrated.

Software architecture

A software architectural design has to be defined that identifies the elements of the software and software requirements are to be allocated to the elements of the software.

Agile projects have to ensure that a software architecture is developed and maintained and that traceability between requirements and architecture, between architecture and design and between architecture and integration tests is established.

Example of a proceeding for creating a software architecture within an agile environment can be that a basic architecture and architecture rules are defined at project start and the architecture is incrementally completed within Sprints (for SCRUM based projects). For all architectural modifications, an impact analysis is performed.

[AGE.RC.5] If no software architecture is developed and maintained, the base practice SWE.2.BP1 should be downrated.

[AGE.RC.6] If the software architecture is modified incrementally including impact analysis, this should not be used to downrate the indicator SWE.2.BP1.

Software testing

Software Unit Verification, Software Integration Test and Software Qualification Tests need to be established in software development projects which require all these 3 levels of testing.

Agile methods may combine these test levels within other methods or levels. For example, testing can be integrated into Sprints in SCRUM based projects. Then the agile project has to ensure that the process purposes of all 3 software testing processes (SWE.4, SWE.5 and SWE.6) are fulfilled by the defined activities in project Sprints.

[AGE.RC.7] If the test level Software Unit Verification is not consistently integrated in the agile life cycle, the base practices SWE.4.BP1 should be downrated.

[AGE.RC.8] If the test level Software Integration Test is not consistently integrated in the agile life cycle, the base practices SWE.5.BP1 should be downrated.

[AGE.RC.9] If the test level Software Qualification Tests is not consistently integrated in the agile life cycle, the base practices SWE.6.BP1 should be downrated.

Independent quality assurance

Agile development methodologies may define generic role descriptions which need to be derived for the roles and responsibilities in the development project. By defining the responsibilities, the project has to ensure that work product and process quality assurance are performed at project level independently and performed objectively without conflicts of interest.

For example, the agile project ensures the independency by an organization structure in which a quality assurance role is defined to ensure that work products and process quality assurance are checked independently and without conflicts of interest.

[AGE.RC.10] If the project does not ensure that work product and process quality assurance is performed at project level independently and objectively without conflicts of interest, the base practice SUP.1.BP1 should be downrated.

Pair programming

Agile methods may use pair programming in which two software developers work together at one computer. One writes code while the other reviews each line of code as the other developer types it in. The developers frequently switch roles.

[AGE.RC.11] If the used pair programming method is not in conflict with code review requirements (e.g. inspection is required due to safety context), the base practices SUP.1.BP2 and SWE.4.BP3 should not be downrated.

2.2.2.2 Rating consistency / Dependencies to processes

Agile development has relationships to all processes of the complete project scope. Important relations are addressed by the recommendations above.

2.2.3 Distributed development

Engineering of automotive software based systems within an organization is not always performed at one location. In the context of a project for the development of a particular product the necessary engineering resources, supporting resources and management resources may be distributed across separate departments, locations, buildings, third-party service providers etc...

In the planning phase of an assessment the sponsor and the assessor have to determine whether all associated entities will be covered with one assessment or with separated assessments for each entity.

If all associated entities are performing their work based on a standard process, it may be optimal to include them all into the assessment scope. If one location is solely responsible for software testing the interviews for this process shall be only at that location.

When associated entities have different processes, separate assessments could be performed, or a single assessment may be organized for these entities that are providing the necessary instances for the processes performed with the same purpose & outcomes (e.g.: project management, QA, CM).

2.2.3.1 Rating recommendations

Responsible roles within the project have to maintain an effective collaboration and communication including the definition of a consistent set of responsibilities to achieve the project goals.

In an assessment the following aspects have to be considered:

- Scope of work for all associated entities
- Definition of responsibilities
- Interrelationships between overall plans, sub-project plans and plans for support organizations
- Effectiveness of monitoring of sub-project activities

- Assignment of activities to sub-projects
- Availability of resources for sub-projects
- Effectiveness of communication between all entities
- Compatibility of status models for work products
- Providing of necessary information and work products to sub-projects
- Readiness criteria to integrate work products from sub-projects
- Escalation mechanisms when work product requirements are not met
- Verification criteria for the integration of system or software items that were developed at different locations

[DID.RL.1] If the scope of work is not defined for all sub-projects, the indicator MAN.3.BP1 must not be rated higher than L.

[DID.RL.2] If the plans of the overall project and sub-projects show inconsistencies and this aspect is significant in the context of MAN.3.BP9, the indicator MAN.3.BP9 shall be downrated.

[DID.RL.3] If the monitoring of the overall project does not recognize deviations in sub-projects and this aspect is significant in the context of MAN.3.BP4 and/or MAN.3.BP5, the indicator MAN.3.BP4 and/or MAN.3.BP5 shall be downrated.

[DID.RL.4] If the assignment of activities to sub-projects does not include a consistent and suitable set of responsibilities and/or commitments for a distributed environment, the indicator GP 2.1.5 for the respective process shall be downrated.

[DID.RL.5] If sub-projects shall work with the same work environment as the overall project but the provided work environment appears to be insufficient (e.g. visible by floating license limitations, insufficient response time or tool performance), the indicator GP 2.1.6 for the respective process shall be downrated.

[DID.RL.6] If sub-projects shall work with the same status models for work products but status information appears to be incompatible, the indicator GP 2.2.2 for the respective process shall be downrated.

[DID.RL.7] If sub-projects do not have the necessary information and work products to perform the process, the indicator GP 2.1.7 for the respective process shall be downrated.

[DID.RL.8] If readiness criteria for work products from sub-projects to be integrated at overall project level are missing, the indicator GP 2.2.1 for the respective process shall be downrated.

[DID.RL.9] If escalation mechanisms across the sub-projects that are not defined and this aspect is significant in the context of SUP.1.BP6, the indicator SUP.1.BP6 shall be downrated.

[DID.RL.10] If the system and/or software integration strategy does not cover the verification of items that were developed at different locations and this aspect is significant in the context of SWE.5.BP1 and/or SYS.4.BP1, the indicators SWE.5.BP1 and/or SYS.4.BP1 shall be downrated.

2.2.4 Management of third-party software

Software projects often include software which has not been developed by the projects themselves but has been delivered from another party ("third-party") and integrated by the project.

The following types of third-party software are mainly used:

- Purchased software (e.g. commercial of the shelf software)
- Software which is developed by a supplier on basis of customer requirements (e.g. engineering service)
- Free- and open-source software
- Software supplied by customer
- Software supplied by another company department ("internal supplier")
- In case the software is delivered from an internal supplier, the project has to determine whether the interface to the internal supplier will be managed by project management directly or by supplier management within the project. Criteria for managing internal suppliers according the needs of Supplier Monitoring (ACQ.4) may be that a contract (e.g. statement of work, license agreement) between the project and the internal supplier is necessary, otherwise the corporation and communication between the project and the internal supplier is not ensured.
- Examples for internal suppliers are:
  - A separate internal group delivering an AUTOSAR device driver to the project

- Application software on basis of license agreement of another company department, which is also sold to customers.

ACQ.4 applies to third-party software which is purchased from suppliers and delivered by customer, on basis of mutually agreed contracts and with defined interfaces (e.g. exchanging, monitoring and tracking all relevant information between both parties).

Software without any support (e.g. open-source software) needs to be managed according to the project needs (see rating recommendations below) but the usage of ACQ.4 is not necessary.

In case the assessed project uses a significant number of third-party software or third-party software products with a high impact (e.g. large size), the scope of the assessment may need to be adapted. For example, software requirements analysis SWE.1 includes the analysis of the software requirements of the third-party software.

2.2.4.1 Rating recommendations

Licenses for third-party software

All kinds of third-party software are valid under specific software license agreements. The project has to ensure that license agreements must be valid for the project purpose. For example, if the developed software including the third-party software is part of an ECU which is intended to go into mass production, then all third-party software licenses must be valid for mass production.

> **[TPS.RC.1]** If it turns out in the assessment that a valid license agreement is absent for the project (e.g. mass production license is needed but not in place) and this aspect is significant in the context of MAN.3.BP5, the base practice MAN.3.BP5 should be downrated.

Functional and non-functional software requirements

The specification or the contractual basis of third-party software has to cover functional and non-functional software requirements.

The functional software requirements of the third-party software have to be in line with software requirements of the project. In case of

- “software which is developed by a supplier on basis of project requirements” the project has to transfer these requirements to the supplier and should use the associated tests as acceptance tests.
- “commercial of the shelf software” the project has to ensure that the commercial of the shelf software complies with the requirements specified for the purchased software. The specified requirements should build the basis for acceptance tests of the third-party software.

The non-functional requirements include for example quality requirements (e.g. specific coding guidelines, metrics target), which are often used as criteria for acceptance tests.

In case the third-party software is software without any support (e.g. Free and open-source software) the project has to ensure that non-functional requirements are met or whether the third-party software (e.g. non-automotive commercial of the shelf software) is treated according legacy software rules (see chapter 2.2.5).

> **[TPS.RC.2]** If the software requirements of the third-party software are not in line with the functional requirements for the project and this aspect is significant in the context of ACQ.4.BP2 and SWE.1.BP8, the indicators ACQ.4.BP2 and SWE.1.BP8 should be downrated.

> **[TPS.RC.3]** If relevant non-functional software requirements for the project (e.g. quality requirements which are valid for the complete project) are not agreed with the provider of the third-party software and this aspect is significant in the context of ACQ.4.BP2 and SWE.1.BP8, the indicators ACQ.4.BP2 and SWE.1.BP8 should be downrated. Excluded is third-party software without any support and third-party software which is treated as legacy software.

Software architecture

The third-party software and its interfaces (e.g. external API) have to be part of the software architecture.

For example, a purchased operating system has to be defined in the software architecture together with its interfaces and how the operating system is connected to the relevant software architecture elements.

[TPS.RC.4] If third-party software is not part of the software architecture and this aspect is significant in the context of SWE.2.BP1, the base practice SWE.2.BP1 should be downrated.

[TPS.RC.5] If the external interfaces of the third-party software are not defined in the software architecture and this aspect is significant in the context of SWE.2.BP3, the base practice SWE.2.BP3 should be downrated.

Managing of free and open-source software

Free Software is source code that allows users to use and modify the software for any purpose. It is not covered by copyright law or other restrictions. Free Software normally has no support, the project has to define and check rules whether the free software elements fit to the project (non-functional) requirements.

Note: Open-source software is source code under an open-source software license agreement (e.g. GNU General Public License (GPL)).

Because open-source software normally has no support, the project has to define and check rules whether the open-source software elements and the license fit to the project (non-functional) requirements. Especially the open-source license agreement has to be fulfilled by the project. Otherwise the project does not have the right to integrate and use the open-source software (e.g. open-source licenses shall be transferred to customer; open-source licenses require to disclose the complete source code of the developed system).

Note: The rules for managing open-source software within a company are often called open-source Policy.

[TPS.RC.6] If free and open-source software is not managed according to rules, which ensure that the open-source software fits to software requirement and this aspect is significant in the context of SWE.2.BP1, the base practices SWE.2.BP1 should be downrated.

[TPS.RC.7] If open-source software is not managed according to rules, which ensure that the open-source software license agreement is fulfilled and this aspect is significant in the context of MAN.3.BP5, the base practices MAN.3.BP5 should be downrated.

Managing of supplied software from customer

Based on customer strategies, customer delivers source or object code to the supplier's software project. I.e. the customer or a customer department acts as a supplier within the project.

The delivered customer software is valid under a software license agreement and support rules needs to be agreed by both customer and supplier. Both parties must keep the conditions of these agreements.

> **[TPS.RC.8]** If the supplier project does not comply with the agreements and the agreed rules for the customer-supplied software and this aspect is significant in the context of ACQ.4.BP1, the base practice ACQ.4.BP1 should be downrated.
>
> Excluded is customer-supplied software for which the customer takes over all responsibility (e.g. customer delivers a software library which the supplier needn't test and for which the supplier has no responsibility in case of identified non-conformances).
>
> **[TPS.RC.9]** If the customer does not comply with the agreements and the agreed rules for the supplied customer software, the base practice ACQ.4.BP1 should not be downrated but the noncompliance of the customer should be documented in the assessment report.

Interface definition to third-party provider

The interface between the third-party software provider (supplier or customer for supplied customer software) and the project needs to be defined and agreed for managing for example deliveries, acceptance, problem and change management and release management. Only in the case that the software is intended to be used without any support from a third-party provider (e.g. open-source software), is not required to be specify the interface.

> **[TPS.RC.10]** If the interface between third-party software provider and the project is not defined and agreed and this aspect is significant in the context of ACQ.4.BP1, the base practice ACQ.4.BP1 should be downrated. An exception is software without any support.

Acceptance of third-party software

Evidence is needed that third-party software has been verified according to acceptance criteria which are defined in an agreement between the third-party software provider and the project. These acceptance criteria may contain for example the review of the release documentation, fulfillment of coding guidelines and/or code coverage of manual and automated tests in compliance with the agreed requirements.

For software without any support from a third-party provider (e.g. open-source software) the project has to define acceptance criteria based on their integration and test strategy.

> **[TPS.RC.11]** If no acceptance criteria and tests are defined to check the compliance of acceptance criteria for third-party software and this aspect is significant in the context of SWE.5.BP3 and ACQ.4.BP1, the base practices SWE.5.BP3 and ACQ.4.BP1 should be downrated.

> **[TPS.RC.12]** If no acceptance tests are performed to check the compliance of third-party software according the defined acceptance criteria and this aspect is significant in the context of SWE.5.BP6 and ACQ.4.BP4, the base practices SWE.5.BP6 and ACQ.4.BP4 should be downrated.

Responsibility for third-party software

The responsibility for third-party software should be defined and agreed within an agreement (e.g. software license agreement, statement of work) between the third-party software provider and the project. This responsibility defines for example who warrants for non-conformances and which tests are done by third-party software provider and which by the user of the third-party software.

> **[TPS.RC.13]** If the responsibility for the third-party software is not defined between third-party software provider and the project and this aspect is significant in the context of ACQ.4.BP1 and MAN.3.BP7, the base practices ACQ.4.BP1 and MAN.3.BP7 should be downrated.

2.2.4.2 Rating consistency / Dependencies to processes

Third-party software has relationships to all processes of the complete project scope. Important relations are addressed by the recommendations above.

2.2.5 Management of platform and legacy software

In the context of this guideline *platform software* is a set of software elements including all the related work products that share a common, managed set of features satisfying the specific needs of a mission. The intent of the mission supports the reuse of the software platform in different projects.

In the context of this guideline *legacy software* was / has been developed in a previous finished project (previous with regard to the project in the assessment scope) and has been in production at least once. In the assessment the development process used when developing the legacy software is unknown or differs from the process used in the assessed project.

From the perspective of Automotive SPICE, software platform development underlies the same rules and regulations as the development of project specific software. Therefore, software platform development itself can be the focus of a process assessment or may be part of the process assessment for a specific project if the development of the platform software runs parallel to this specific project. In this case, this chapter is not relevant.

If for any reason the consideration of the development process for platform software or legacy software in the assessment is not possible or not required by the assessments sponsor, the rules und recommendations in this chapter support the following scenario:

Based on the stakeholder requirements the platform or legacy software is part of the assessment scope (process context) but the platform or legacy software development process itself is not part of the assessment scope. So the assessed project has to show that the interfaces to the software platform (development) or to the legacy software are managed and therefore the following rules and recommendations address this interface management.

Note: The application of the following rating rules shall consider the proportion of the platform/legacy elements used in relation to the scope of the project specific software (reflected in the recommendations by the phrase “and this aspect is significant in the context of”).

2.2.5.1 Rating recommendations

Identification of platform and/or legacy software

> **[PLS.RC.1]** If the boundary to platform and/or legacy software intended for use within the project is not consistently reflected in the scope of the assessed project and this aspect is significant in the context of MAN.3.BP1, the indicator MAN.3.BP1 (scope of work) should be downrated.

> **[PLS.RC.2]** If the platform and/or legacy software used in the assessed project is not consistently reflected in the software architectural design and this aspect is significant in the context of SWE.2.BP1, the indicator SWE.2.BP1 (software architectural design) should be downrated.

Responsibility for platform and/or legacy software

The responsibility for platform software and/or legacy software should be defined as an important project interface and communicated within the assessed project. The responsible person for platform software is the interface to the organization which develops the software platform and is the contact person for the project in case of problems or changes. The responsible for legacy software is the contact person for the project in case of legacy software support.

> **[PLS.RC.3]** If the responsibility for platform software and/or legacy software is not defined and active or problems concerning responsibility for platform software and/or legacy software were not identified and escalated to the organization and this aspect is significant in the context of MAN.3.BP7, the indicator MAN.3.BP7 (project interfaces) should be downrated.

Transparency of requirements fulfilled by platform and/or legacy software

For the assessed project, it is important to know which are the functional and non-functional requirements covered by the used platform software. This is necessary for example for analyzing software requirements and their impact on the operating environment or ensuring consistency with system requirements.

[PLS.RC.4] If the functional and non-functional requirements covered by the used platform software are not known in the assessed project and this aspect is significant in the context of SWE.1.BP1, the indicator SWE.1.BP1 (specification of software requirements) should be downrated.

For legacy software, a detailed list of covered functional and non-functional requirements as requested above typically is not available for the assessed project. In this case the assessed project has to ensure that the used legacy software fits the project requirements by other measures. For example, if the source code of the legacy software is available the requirements may be re-engineered from the source code. In addition, or if the source code is not available for minimizing risks the assessed project may investigate how often the used legacy software was/is used in other projects, which risks and problems these projects were/are faced with and if the environment of the assessed project is similar to the environments of other projects which used/uses the legacy software. Based on the gained insight by these measures the alignment of the used legacy software to the project requirements should be demonstrated.

[PLS.RC.5] If there is no evidence for measures proving that legacy software fits the project requirements of the assessed project and this aspect is significant in the context of SWE.1.BP1, the indicator SWE.1.BP1 (specification of software requirements) should be downrated.

Requirements changes

Requirements changes may have an impact on whether the platform software and/or legacy software used by the assessed project still fits. As a consequence, change requests which may have a relation to the platform software and/or legacy software should be analyzed and assessed accordingly.

[PLS.RC.6] If change requests are not analyzed with respect to an impact on the used platform software and/or legacy software and this aspect is significant in the context of SUP.10.BP4, the indicator SUP.10.BP4 (analyze change requests) should be downrated.

Testing of used platform and/or legacy software

Software items developed in the assessed project have to be considered in the integration and test strategies of the processes SWE.5 and SWE.6 for identifying problems or risks. The same is necessary for used platform and/or legacy software.

For legacy software, the type of measures mentioned in chapter "*Transparency of requirements fulfilled by platform and/or legacy software"* for showing alignment to the project requirements is a starting point. The gained insight from these measures should be used for deriving verification criteria.

For platform software, the verification criteria that shall be used by the assessed project have to be synchronized with the department responsible for the software platform development to avoid redundancy and gaps.

> **[PLS.RC.7]** If the used platform software and/or legacy software is not reflected in the test strategies of the processes SWE.5 and/or SWE.6 and this aspect is significant in the context of the corresponding base practices for developing test strategies, the indicators SWE.5.BP1 (software integration strategy) and/or SWE.5.BP2 (software integration test strategy) and/or SWE.6.BP1 (software qualification test strategy) should be downrated, respectively.

> **[PLS.RC.8]** If there is no evidence for derivation of verification criteria for platform and/or legacy software considering the aspects mentioned above and this aspect is significant in the context of SWE.1.BP5, the indicator SWE.1.BP5 (software verification criteria) should be downrated.

2.2.5.2 Rating consistency / Dependencies to processes

Platform software and legacy software development have relationships to all processes of the complete project scope. Important relations are addressed by the recommendations above.

2.2.6 Application parameters

Interpretation of terms

In the following, the terms "calibration parameters" and "application parameters" are used synonymously.

Automotive SPICE 3.1 defines “application parameters” as follows:

> *“An application parameter is a parameter containing data applied to the system or software functions, behavior or properties. The notion of application parameter is expressed in two ways: firstly, the logical specification (including name, description, unit, value domain or threshold values or characteristic curves, respectively), and, secondly, the actual quantitative data value it receives by means of data application.”*

Application parameters can therefore generally be used for two scenarios:

1) Influencing the implemented system behavior

   The software makes the system behave according to the stored application parameter data not containing any executable or interpretable code, e.g.

   - The range of the window glass in a door system within which antitrap protection shall be active
   - Values for low idle speed, motor characteristic diagrams etc.
   - Product vehicle impacting system behavior, e.g. such as country codes, left-hand/right-hand steering etc.

2) Code selection

   Code variants can be determined at compile-time by e.g. preprocessor commands or preprocessor variable settings of e.g. the programming language C; as a result, the built program only contains code that is to be executed. In contrast, the expected executed code can also be determined later, i.e. at runtime, depending on application parameter values evaluated if-clauses.

In both scenarios, the actual data set can be flashed into the system by e.g. diagnosis jobs or end-of-line.

Typical problems in practice

Typical problems are:

- Parameter information and data sets are not subject to configuration management.
- No strategy for changes to application parameters defined in the context of change request management, in particular if different parties are responsible for different parameters.

- Testing strategies at the various testing levels do not reflect permutations or interdependencies of parameters, and parameter values, in particular if different parties are responsible for different parameters.
- Quality assurance activities tend to forget interdependencies of parameters and parameter values.

2.2.6.1 Rating recommendations

Distinguishing between requirements and design activity decisions

Application parameters influencing the implemented system's behavior

The requirements perspective, being a black-box perspective, must require configurability of a behavior, e.g.

- "The movement range in centimeters of a window shall be configurable"

In contrast, deciding on how many application parameters are to be implemented in the software, and on specific logical information (i.e. the parameters' variable names, technical data types, default values etc.) is an architectural or detailed design decision because it is not a black-box perspective.

The requirements specification must inform about configurability expectations as otherwise the testing processes will miss to set up the test strategy, and test cases, accordingly.

Note that this does not mean that a separate application parameter specification document is not allowed. What is important is to be aware that the perspectives of a black-box requirement statement and technical implementation decisions are different disciplines.

> **[APA.RL.1]** If the requirements are not consistent with the implemented application parameters and their values, and if this aspect is significant in the context of BP1 of SYS.2 and/or SWE.1, respectively, then the indicator BP1 of SYS.2 and/or SWE.1 shall be downrated.

> **[APA.RL.2]** If the definition of the application parameters is not consistent with their implementation and values, and if this aspect is significant in the context of SWE.3.BP1, then the indicator BP1 of SWE.3 shall be downrated.

Application parameters for code selection at runtime

Application parameters for code selection at runtime represent code variants, e.g.

- a navigational system for customer A additionally offers Points-Of-Interest while the variant for customer B does not;
- a fault diagnosis for a stuck relay is not required for the technology of a pulse-width based activation of the power stage.

Therefore, application parameters for code selection are not to be described in requirements specification as explained above; rather, each given requirement is marked as to in which variant it must be implemented.

Deciding on how many application parameters are to be implemented in the software in order to express this, and on specific logical information (i.e. the parameters' variable names, technical data types, default values etc.) is an architectural or detailed design decision.

> **[APA.RL.3]** If those implemented application parameters which represent product variants and their values are not consistent with the requirements related to that variant, and if this aspect is significant in the context of SYS.2 and SWE.1, then the indicator BP1 of SYS.2 and SWE.1, respectively, shall be downrated.

Responsibility for application parameters

Application parameters influencing the implemented system's behavior

Often the division of responsibility for application parameters does not follow the exact customer-supplier boundary.

Examples:

- A controller device supplier defines, and implements, all application parameters but the customer retains the right to alter some of them after the supplier's delivery

- Owners of different reusable standard SW components maintain their own local parameters

Some of the parameters shall not even be accessible to the customer. In such a situation, for e.g. product liability purposes, the responsibility for each of the application parameters should be explicitly defined. This may be done by e.g. an addendum to a development agreement interface.

> **[APA.RC.1]** If application parameter values can be, or are, altered by a party other than the developers of the product, but responsibilities are not clearly defined, and if this aspect is significant in the context of MAN.3.BP7, then the indicator BP7 of MAN.3 should be downrated.

Application parameters for code selection at runtime

The responsibility of such parameters is upon the supplier. Therefore, they must not be altered by the customer, so no application parameter information is exposed.

Treating application parameter information as configuration items

For any application parameter, the following aspects

a) the data value sets
b) variable names
c) technical data types
d) default values
e) the corresponding memory maps

are configuration items. These configuration items are part of baselines.

[APA.RL.4] If all aspects above are not treated as configuration items, and if this is significant in the context of SUP.8.BP2, then the indicator BP2 of SUP.8 shall be downrated.

Change management related to application parameters

Furthermore, in the context of change request management (SUP.10) the impact of a change on application parameter information must explicitly be analyzed. For

- application parameters for code selection at runtime this means activating or deactivating features, and, thus, changing product variants;
- application parameters influencing the implemented system's behavior this means changing the product application

[APA.RL.5] If for change requests their impact on application parameters is not evaluated, and if this aspect is significant in the context of SUP.10.BP4, then the indicator BP4 of SUP.10 shall be downrated.

[APA.RL.6] If the change request management strategy does not define how changes to application parameters are to be proceeded, and if this aspect is significant in the context of SUP.10.BP1, then the indicator BP1 of SUP.10 shall be downrated.

Quality assurance on parameter information

Application parameters influencing the implemented system's behavior

Application parameter information, both in requirements and design specifications, has to undergo quality assurance (see SUP.1). Thus, quality assurance methods must not only evaluate whether data ranges, default values, and final values are correct, but must also check for consistency of this information across all parameters. This is particularly important if different parties are responsible for different application parameters (see chapter "Responsibility for application parameters").

[APA.RL.7] If application parameters do not receive quality assurance at least with respect to technical correctness and cross-parameter-consistency, and if this aspect is significant in the context of SUP.1.BP2 then the indicator BP2 of SUP.1 shall be downrated.

Application parameters for code selection at runtime

Application parameters for code selection at runtime represent product variants. Thus, quality assurance methods must evaluate whether the chosen data values represent the desired product variants.

Example 1:

- The customer wants Feature F_1 only. Therefore, it was decided to choose product variant V_2. However, erroneously both parameters X and Y were activated which results in the product actually realizing F_1 *and* F_2, i.e. Varian V_1. This error should have been detected by e.g. design or code reviews against the table.

	Variant V_1	Variant V_2	Variant V_3
Feature F_1, activated by parameter X	x	x	-
Feature F_2, activated by parameter Y	x	-	-

Example 2:

- The customer wants features F_1 and F_2 only. Therefore, it was decided to choose variant V_1. Correspondingly, parameters X and Y were set. However, during requirements reviews, design reviews, and code reviews it remained unnoticed that parameter Y also activates feature F_3 which was never wanted.

	Variant V_1	Variant V_2	Variant V_3
Feature F1, activated by parameter X	x	x	-
Feature F2, activated by parameter Y	x	-	-
Feature F3, also activated by parameter Y	-	x	-

See also **[APA.RL.8]** here.

Application parameters and testing

Application parameters influencing the implemented system's behavior

It is important to consider

- which application parameters are relevant for which test cases.
  Examples:
  - interdependencies between application parameters may require that particular test cases are to be performed as a coherent sequence
  - given, or reused, test cases may need to be redesigned
  - a particular parameter A may be exclusive to parameter B and C, i.e. test cases for B and C are no longer required
- and which values exactly e.g., determined via the methods "boundary values" and/or "equivalence classes".

The same considerations apply to regression test definition.

> **[APA.RL.8]** If test methods, and test case definition, do not reflect the respective application parameters, and if this aspect is significant in the context of testing processes, the corresponding indicator BP1 of SWE.4, SWE.5, SWE.6, SYS.4, or SYS.5, respectively, shall be downrated.

Application parameters for code selection at runtime

Application parameters for code selection at runtime represent product variants. The code variants are expressed in requirements specifications, see the corresponding chapter above. Therefore, the testing parties need to receive a product sample that realizes the requirements of the desired variants as otherwise, test cases will fail anyway according to the corresponding base practices of the testing processes. In such a case, this situation will already be covered by the rating of the base practices or process outcomes, respectively.

However, the allocation of requirements to variants may be complex (e.g. a method such as "feature trees" may be needed in preference to simple attribute columns in a requirements management tool). Therefore, for test case selection understanding the interdependencies between application parameters for code selection at runtime is important.

3 Rating guidelines on process performance (level 1)

3.1 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 customer has to introduce a supplier monitoring process for the following relationships with suppliers:

- Supplier develops a component on basis of the customer requirements
- Supplier delivers and maintains a component which is provided off the shelf to the customer (e.g. operating system, device drivers, system with hard- and software)
- Supplier delivers a component with off the shelf sub-components and development on basis of customer requirements
- Excluded are suppliers which deliver products without any support (e.g. open-source software)

Interfaces between supplier and customer have to be established for exchanging, monitoring and tracking all relevant information between both parties. Even for a small number of deliveries (e.g. commercial off the shelf component) interfaces have to be set up and maintained for at least component deliveries and managing changes and problem reports.

3.1.1 Rating recommendations

3.1.1.1 Monitoring all suppliers

All project relevant suppliers have to be tracked and their performance against the agreed requirements has to be assessed. This includes suppliers for engineering service, commercial of the shelf products, firmware, etc. Excluded are suppliers which deliver products without any support (e.g. open-source software).

> **[ACQ.4.RL.1]** If not all suppliers, excluding suppliers without any support, involved in the project are monitored according ACQ.4, PA 1.1 must not be rated F.

3.1.1.2 Incomplete agreements with supplier

Agreements between supplier and customer have to be established and maintained, which cover:

- supplier's project content and scope
- exchanged information between customer and supplier
- joint activities
- joint processes and interfaces
- responsibilities and stakeholders
- joint project management
- test specification and testing activities
- joint problem and change management
- joint reporting and reviews
- escalation mechanism

Examples for such agreed documents are distributed interface agreements, statements of work, license agreements, etc.

> **[ACQ.4.RL.2]** If agreements between supplier and customer are incomplete with respect to all aspects above, the indicator BP1 shall be downrated.

Related to:

- BP1 "Agree on and maintain joint processes"
- Output WP 13-04 "Communication record"
- Output WP 13-09 "Meeting support record"

3.1.1.3 Consistency to main customer agreements

Agreements of the customer's customer (e.g. OEM) have to be taken into consideration for establishing the agreements between supplier (e.g. TIER 2) and customer (e.g. TIER 1). E.g. quality requirements in the agreements between supplier and customer have to be in line with OEM quality agreements.

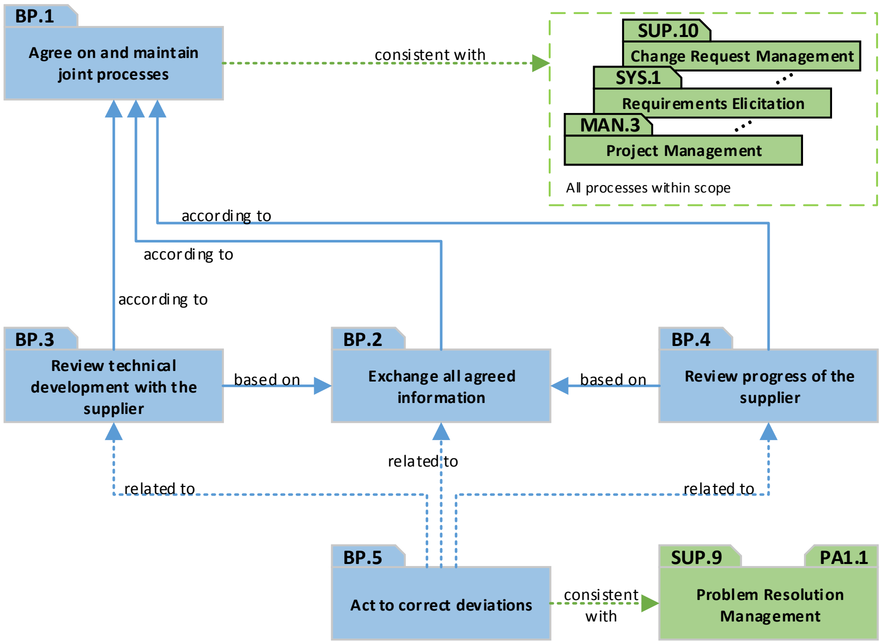
> **[ACQ.4.RC.1]** If relevant agreed requirements of the customer's customer (e.g. OEM), are not part of agreements between supplier and customer, the indicator BP1 should be downrated.

Related to:

- BP1 "Agree on and maintain joint processes"
- Output WP 13-04 "Communication record"
- Output WP 13-09 "Meeting support record"

3.1.2 Rating consistency

The following figure shows the relationships between ACQ.4 base practices as well as their relationships to other processes:



These relationships are used as the basis for the rating rules and recommendations defined in the following subchapters.

Generic aspects regarding communication (2.1.2) shall also be considered for rating.

3.1.2.1 Rating consistency within ACQ.4

Within ACQ.4, the following base practices have relationships to each other.

> **[ACQ.4.RL.3]** If the indicator BP1 is downrated due to incomplete agreements between supplier and customer (see ACQ.4.RL.2), the corresponding indicators (BP2, BP3, BP4) shall be downrated.

BP3 Review technical development with the supplier

> **[ACQ.4.RL.4]** If the indicator BP2 is downrated due to incomplete exchange of all agreed information necessary for reviewing technical development, the indicator BP3 shall be downrated.

BP4 Review progress of the supplier

> **[ACQ.4.RL.5]** If the indicator BP2 is downrated due to incomplete exchange of all agreed information necessary for reviewing the progress of the supplier, the indicator BP4 shall be downrated.

BP5 Act to correct deviations

> **[ACQ.4.RC.2]** If the indicators BP2, BP3 or BP4 are downrated due to identified non-conformances which are not managed as corrective actions, the indicator BP5 should be downrated.

3.1.2.2 Rating consistency to other processes at level 1

The following base practices of ACQ.4 have relationships to other processes.

BP1 Agree on and maintain joint processes

Supplier management can have relationships to all processes of project scope.

Example 1: The supplier is to develop a software component on the basis of customer requirements (SWE.1). Relevant customer requirements have to be transferred completely to the supplier.

Example 2: The purchased operating system is part of the software architecture (SWE.2) with detailed information about constraints and interfaces. All relevant architecture information has to be part of the supplier contract.

> **[ACQ.4.RC.3]** If BP1 is downrated due to incomplete agreements between supplier and customer (see ACQ.4.RL.2), this should be in line with the rating of the related BP indicators of relevant processes of the project scope.

BP5 Act to correct deviations

> **[ACQ.4.RC.4]** If BP5 is downrated due to gaps in analyzing, tracking and control of deviations from the agreed project plans, this should be in line with the rating of PA 1.1 of SUP.9 Problem Resolution Management.

3.2 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 System Requirements Analysis process uses the stakeholder requirements that were processed in the Requirements Elicitation process as an input. Such stakeholder requirements can be either functional or non-functional. The level of detail can be very generic (e.g., “the vehicle shall have a powerful acceleration”) or specific (e.g., the angular speed of the wiper arm shall be in a range of x…y) down to design restrictions that have to be considered in the design phase.

Stakeholder requirements can be in contradiction to each other e.g., legal regulations with specific customer needs. System Requirements will be in such a case derived as a trade-off between such stakeholder requirements in dialog with the customer.

In the majority of automotive embedded software project stakeholder requirements are mainly customer requirements that address certain functionality to be implemented into the system.

The term Stakeholders is not limited to the customer. Examples for stakeholders are legal entities, end users and internal organizational units like purchasing, platform development (Reuse), manufacturing etc.

Normally the functional content grows over the releases. Therefore, the complete set of requirements is not necessarily available at the project start. This has to be considered by rating the completeness of the work products.

3.2.1 Rating recommendations

3.2.1.1 System requirements

System Requirements are particular desired characteristics of a system. Their implementation will be verified in the integrated system.

System Requirements may address among others:

- Functions that are implemented in mechanics, hardware or software or cover a combination of these elements
- Parameters influencing the system behavior
- Processing of signals from other systems
- Non-functional Requirements

System requirements have to be granular and understandable. Unclear or generic requirements have to be clarified with the individual stakeholders.

The existence of a set of system requirements shall be demonstrated as a populated list or data base that allows the structuring of the system requirements.

Recommendations and rules:

[SYS.2.RL.1] If unclear or inconsistent requirements are not clarified with the individual stakeholders, the indicator BP1 shall be downrated.

[SYS.2.RL.2] If the system requirements specification is not reflecting latest changes, the indicator BP1 must not be rated higher than L.

[SYS.2.RL.3] If system requirements are not derived from customer requirements but from platform requirements according to a reuse strategy the indicator BP1 must not be downrated.

Related to:

- BP1 “Specify system requirements”
- Output WP 17-12 “System requirements specification”

3.2.1.2 Structure system requirements

System requirements are structured by grouping, sorting and categorizing to support a prioritization and to map the required functionality to future releases. The structure and categorization of the system specification enables the project to manage the requirements in terms of e.g., organizational, technical, legal and internal topics.

Recommendations and rules:

[SYS.2.RL.4] If the categorization is not appropriate as mentioned above, the indicator BP2 must not be rated higher than L.

[SYS.2.RL.5] If the mapping of functionality to the releases does not reflect the customer and other stakeholder needs, the indicator BP2 shall be downrated.

[SYS.2.RC.1] If there is no evidence for prioritization but a release plan exists that demonstrates the assignment of functionality to future releases the indicator BP2 should not be downrated.

Related to:

- BP2 “Structure system requirements”
- Output WP 15-01 “Analysis report”

3.2.1.3 Analyze

The analysis of system requirements is the basis for a correct implementation. Even though requirements sometimes are very detailed or their implementation seems to be very simple, a well-founded analysis has to be conducted for those requirements, too. The scope and appropriateness of the analysis and its documentation depend on the context of product (e.g. platform) and organization. The result of analysis can vary from a simple attribute in a list to a complex simulation or the building of a demonstrator to evaluate the feasibility of system requirements. Doubts in feasibility of functionality have to be reflected in MAN.5.

Recommendations and rules:

[SYS.2.RL.6] If the system requirements and their interdependencies are not evaluated in terms of correctness, technical feasibility and verifiability, the indicator BP3 must not be rated F.

[SYS.2.RC.2] If the analysis of impact on cost and schedule is covered by the estimation of work packages in the project planning, this must not be used to downrate the indicator BP3.

Related to:

- BP3 “Analyze system requirements”
- Output WP 15-01 “Analysis report”
- Output WP 17-50 “Verification criteria”

3.2.1.4 Impact on the operating environment

The analysis of the impact on the operating environment covers the impact on the system in scope as well as the impact on other systems and the entire vehicle considering the following possible aspects:

- Interfaces
  - Mounting
  - Energy flow (mechanic, hydraulic, pneumatic, electric, temperature etc.)
  - Material flow (fuel, oil, water etc.)
  - Signals and signal quality
  - Noise, vibration, harshness
- Environment
  - Temperature
  - Humidity
  - Exhaust
  - EMC
  - Radiation
- Performance
  - Interface response times (mechanic, hydraulic, pneumatic, electric see 4.10.1.4)
  - Subsystem response times (e.g. micro controller processing time)
- Resources
  - Energy flow
  - Material flow
  - Memory usage (RAM, ROM, EEPROM/DataFlash)

Recommendations and rules:

[SYS.2.RC.3] If the analysis of the impact on the operating environment is not considering aspects from the list above or other aspects that are relevant for the project the indicator BP4 should be downrated.

[SYS.2.RC.4] If there are insufficient reserves of memory, processor time, and/or peripheral resources the indicator BP4 should be downrated.

Rationale: Insufficient reserves of memory, processor time and/or peripheral resources are signs for inappropriate analysis of technical feasibility or inappropriate analysis of impact on the operating environment.

Related to:

- BP4 “Analyze the impact on the operating environment”
- Output WP 15-01 “Analysis report”
- Output WP 17-08 “Interface requirements specification”

3.2.1.5 Verification criteria

Refer to chapter 2.1.3 for the generic concept of the term “Verification Criteria”.

Recommendations and rules:

> **[SYS.2.RL.7]** If verification criteria are not documented as a separate work product but demonstrably contained in the requirement or test specification the indicator BP5 must not be downrated.

Related to:

- BP5 “Develop verification criteria”
- Output WP 17-50 “Verification Criteria”

3.2.1.6 Customer will not update his requirements

System requirements are derived from stakeholder requirements. During the process of analysis of system requirements inconsistencies between stakeholder requirements and system requirements may occur as the customer will not update his requirements.

Recommendations and rules:

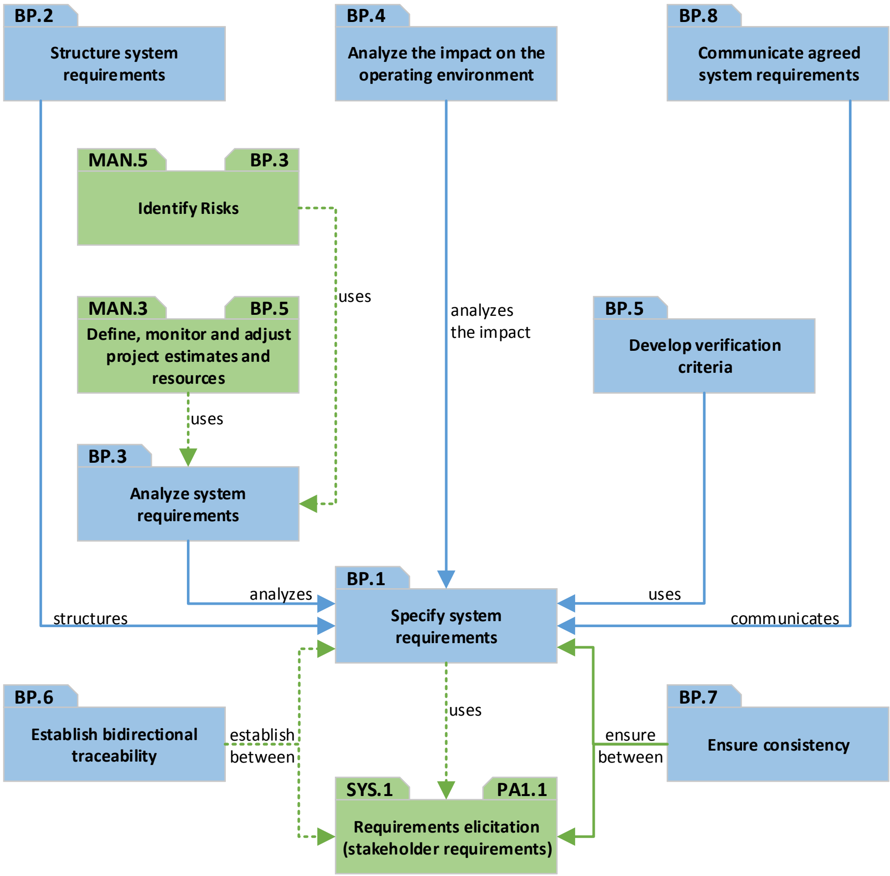
> **[SYS.2.RL.8]** Customer requirements are not necessarily updated as a result of system requirement analysis. If in this case the result of analysis is documented and comprehensibility and traceability from system requirements to the corresponding sources (customer confirmation e. g. via emails, meeting records, presentations) is given the indicator BP7 must not be downrated.

Related to:

- BP7 “Ensure consistency”
- Output WP 13-22 “Traceability record”

3.2.2 Rating consistency

The following figure shows the relationships between SYS.2 base practices as well as their relationships to other processes:



These relationships are used as the basis for the rating rules and recommendations defined in the following subchapters.

Generic aspects regarding traceability and consistency (2.1.1), communication (2.1.2), and verification criteria (2.1.3) shall also be considered for rating.

3.2.2.1 Rating consistency within SYS.2

The following rating rule is related to the specification of the system requirements and thus influences several base practices of the process:

> **[SYS.2.RL.9]** If the indicator for the specification of system requirements (BP1) is downrated, PA 1.1 shall be downrated as all indicators (BP2, BP3, BP4, BP5, BP6, BP7 and BP8) are affected.

3.2.2.2 Rating consistency to other processes at level 1

The following base practices of SYS.2 have relationships to other processes.

BP1 Specify system requirements

> **[SYS.2.RC.5]** If PA 1.1 for SYS.1 is downrated, this should be in line with the rating of the indicator BP1.

BP3 Analyze system requirements

> **[SYS.2.RC.6]** If the indicator BP.3 is downrated, this should be in line with the rating of the indicator about project estimates and resources (MAN.3.BP5).

> **[SYS.2.RC.7]** If the indicator BP.3 is downrated, this should be in line with the rating of the indicator about risk identification (MAN.5.BP3).

BP6 Establish bidirectional traceability

> **[SYS.2.RC.8]** If PA 1.1 for SYS.1 is downrated, this should be in line with the rating of the indicator BP6 (see 2.3.1).

BP7 Ensure consistency

> **[SYS.2.RC.9]** If PA 1.1 for SYS.1 is downrated, this should be in line with the rating of the indicator BP7.

3.3 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.

For technical projects in most cases the solution space for an architecture is manifold and not biunique. In addition, the solution for the architecture is influenced by several other not necessarily technical drivers (non-functional technical requirements).

Possible system requirements for the definition of an architecture are e.g.

- Non-functional technical requirements
  - Performance (response time, cycle time, deadline, flow)
  - Safety (non-functional safety aspects e.g. two microcontroller system)
  - Security
  - COTS (Commercial Of The Shelf) elements with defined interfaces
  - etc.

- Maintainability requirements
  - Usability
  - Simplicity
  - Maximum cohesion and minimum coupling
  - Testability
  - Analyzability
  - Modifiability
  - etc.

- Business requirements
  - Costs
  - Portability (reuse, platform, legacy interfaces)
  - Scalability
  - etc.

Some of these aspects are in contradiction to each other so that in most cases the finally selected architecture is a compromise between these criteria.

3.3.1 Rating recommendations

3.3.1.1 Develop system architectural design

The system architectural design is the highest level design description of the system with different (high level) abstraction views reflecting concerns of different stakeholders. The term “stakeholder” is not limited to the customer and could include as well e.g. strategic planning, project management, development, testing, quality assurance, safety etc. of the supplier and other entities such as legal bodies.

These views are architecture visualizations that are required for communication, discussion, reviews, analysis, evaluation, planning, change request analysis, impact analysis, maintenance etc. of the system.

There is no common definition of which views are required and no criteria for the completeness of the sum of views. There are some approaches in the industry that specify the kind of information that is required for the view (“viewpoints” which are collections of patterns, templates, and conventions for constructing one type of view) and the integration of the views in a thoroughly architectural design description.

In most cases the system architectural design is a graphical representation of the system supplemented by textual explanations. The graphical representation consists at least of a static view providing an overview of the structure and a dynamic view describing the designated behavior of the system.

Static system architecture views allow the decomposition of the system into manageable elements with high cohesion and low coupling. This decomposition supports the assignment of requirements to these architecture elements and will help the organization to distribute the work. Architecture elements of the system that are developed external to the assessment scope (e. g. platform parts, third-party parts, etc.) will also be included as dedicated elements in the system architectural design and have to be considered as well for interface analysis, dynamic behavior etc.

As appropriate the architecture elements are detailed further in the architectural design. The number of hierarchical levels needed to define manageable elements of the system may be different for each element of the highest level. The appropriate level of detail is e.g. driven by:

- The need for encapsulation and modularization of elements
- The need for integration of reused elements
- The complexity of the element
- The need for maintainability
- The allocation of requirements to elements
- The distribution of work
- The need to enable parallel work on elements

On each layer of the static view of the system architectural design the interfaces between the elements are required to be identified.

The detailed aspects of the interface descriptions of system architectural design and the detailed dynamic aspects of system architectural design are described below.

Recommendations and rules:

[SYS.3.RL.1] If the system architecture does not reflect dynamic views the indicator BP1 shall be downrated.

[SYS.3.RL.2] If the system architecture does not reflect applicable non-functional requirements the indicator BP1 shall be downrated.

Related to:

- BP1 “Develop system architectural design”
- Output WP 04-06 “System architectural design”
- Output WP 17-08 “Interface requirement specification”

3.3.1.2 Allocate system requirements

For system requirements that are structured as per SYS.2.BP2 it is required to allocate the requirements to the elements of the system architectural design derived by BP1. At the stage of system architectural design, the allocation typically is done on the level of suitable requirement clusters (e.g. a chapter in requirements specification) and not on the level of single requirements.

The allocation shall be traceable e.g. by a matrix or based on additional corresponding attribute/information in the used requirements management tool.

Each requirement or requirement cluster is required to be mapped to at least one element of the system architectural design ("no requirement is forgotten"). This mapping could be one direction of the bidirectional traceability addressed by BP6.

Recommendations and rules:

> **[SYS.3.RL.3]** If the allocation of system requirements to elements of the system architectural design is done based on clusters but not on single requirements, the indicator BP2 must not be downrated.

Related to:

- BP2 "Allocate system requirements"

3.3.1.3 Define interfaces of system elements

System interfaces represent the interaction between the elements of the system architecture and the interaction between the system and the system environment. The system interfaces are derived by any linkage (intended or not intended) as e.g.

- Mounting
- Energy flow (mechanic, hydraulic, pneumatic, electric, temperature etc.)
- Material flow (fuel, oil, water etc.)
- Signals and signal quality
- Noise, vibration, harshness

Recommendations and rules:

> **[SYS.3.RL.4]** If the system interface definition is absent or not all links are considered the indicator BP3 shall be downrated.

Related to:

- BP3 “Define interfaces of system elements”
- Output WP 17-08 “Interface requirement specification”

3.3.1.4 Describe dynamic behavior

For describing the dynamic behavior of a system at runtime behavioral descriptions are required e.g.

- State transition diagrams
- Sequence diagrams
- Message sequence charts
- Use-case diagrams

Which ones are required or suitable depends on the application. In addition, the response times have to be considered.

Recommendations and rules:

> **[SYS.3.RL.5]** If evidence of describing dynamic behavior regarding the topics mentioned above is missing the indicator BP4 shall be downrated.

Related to:

- BP4 “Describe dynamic behavior”
- Output WP 04-06 “System architectural design”
- Output WP 17-08 “Interface requirement specification”

3.3.1.5 Evaluate alternative system architectures

One of the following three approaches for architecture development should be used in practice and should be identifiable for an assessor:

1) Development of alternative solutions (e.g. for development of a completely new system):

   Several potential solutions for the system architecture are described at least up to an abstraction level that allows the identification of the main differences between the architectures and that allows the evaluation of the most important architecture criteria for each of the potential solutions. Based on this first evaluation at least one of the proposed solutions is elaborated further by performing the base practices (BP1 to BP4). It has to be ensured that the proposed solutions that are chosen for further elaboration are able to cover the required needs of the project. At the end, these proposed and refined solutions are evaluated based on the defined evaluation criteria (BP5) and a decision is made:

   - Selection/Confirmation of one/the proposed solution as the used architecture for further development or
   - Rejection of the previous proposed solution(s) and step back to architecture development (BP1)

2) Iterative architecture development:

   During the development of an architecture by performing the base practices (BP1 to BP5) several variants for the used architecture arises. A variant can be a completely different architecture or a variant can differ from an already identified proposed solution only in some aspects or viewpoints. As a consequence, the evaluation of the chosen criteria (BP5) can take place several times during the elaboration of the architecture that is chosen at the end and that is used as the basis for the further development.

3) Carry over and adaption of an existing architecture (e.g. for platform development):

Although only one solution approach is used for the architecture development it has to be ensured that the chosen approach is suitable for the project and valid according to the chosen evaluation criteria. So BP5 is reduced to the evaluation of one solution only. Identified weaknesses during the evaluation should be eliminated or the consequences of the weaknesses in the chosen architecture have to be made transparent.

In any case it has to be ensured that all relevant parties and all necessary competencies are involved in the agreement on the selection of the final system architecture.

Recommendations and rules:

[SYS.3.RL.6] If none of the three described approaches for architecture development is observable in the assessed project, PA 1.1 shall be downrated.

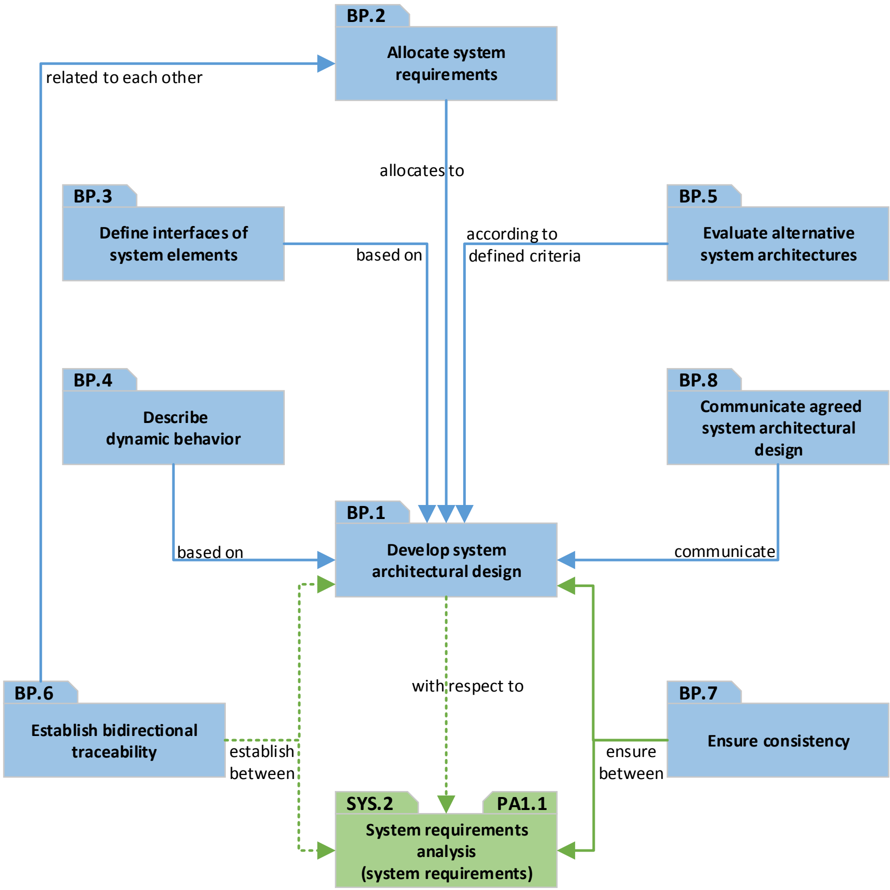
[SYS.3.RC.1] If the used procedure for architecture selection does not involve the required parties or competencies, the indicator BP5 should be downrated.

Related to:

- BP5 “Evaluate alternative system architectures”
- Output WP 04-06 “System architectural design”

3.3.2 Rating consistency

The following figure shows the relationships between SYS.3 base practices as well as their relationships to other processes:



These relationships are used as the basis for the rating rules and recommendations defined in the following subchapters.

Generic aspects regarding traceability and consistency (2.1.1), and communication (2.1.2) shall also be considered for rating.

3.3.2.1 Rating consistency within SYS.3

The following rating rule is related to the development of the system architectural design and thus influences several base practices of the process:

> **[SYS.3.RL.7]** If the development of the system architectural design (BP1) is downrated, PA 1.1 shall be downrated as all indicators (BP2, BP3, BP4, BP5, BP6, BP7 and BP8) are affected.

BP6 Establish bidirectional traceability

> **[SYS.3.RL.8]** If the allocation of the system requirements to elements of the system architectural design (BP2) is downrated, the indicator BP6 shall be downrated.

3.3.2.2 Rating consistency to other processes at level 1

The following base practices of SYS.3 have relationships to other processes.

BP1 Develop system architectural design

> **[SYS.3.RC.2]** If PA 1.1 for SYS.2 is downrated, this should be in line with the rating of the indicator BP1.

BP6 Establish bidirectional traceability

> **[SYS.3.RC.3]** If PA 1.1 for SYS.2 is downrated, this should be in line with the rating of the indicator BP6.

BP7 Ensure consistency

> **[SYS.3.RC.4]** If PA 1.1 for SYS.2 is downrated, this should be in line with the rating of the indicator BP7.

3.4 SYS.4 System Integration and Integration Test

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.

3.4.1 Rating recommendations

3.4.1.1 Integration strategy

In general, the planning and extent of all integration activities is achieved by following a documented integration strategy.

The expectations for an integration strategy cover these aspects:

- A definition of the intended approach for the integration of system elements (bottom-up, top-down, according to availability, according to criticality level, items on a critical path etc.) which leads to:
  - A definition of the integration steps and their sequence in relation to project plan and release plan.
  - A definition of which items, defined in the system architecture, need to be ready for the defined integration steps.
- A definition of how the level of complexity regarding the product and the organization is handled (e.g. Multi-site development, technical complexity of the system).
- A definition of the preconditions for system items to be ready for integration (e.g. predefined test steps or quality criteria).

Recommendations and rules:

[SYS.4.RL.1] If the integration strategy does not cover all aspects above, the indicator BP1 must not be rated F.

Related to:

- BP1 “Develop system integration strategy”
- Output WP 08-52 “Test plan”

3.4.1.2 Test strategy

In general, all testing activities follow a test strategy documented in a test plan.

The expectations for a test strategy cover these aspects:

a) A definition of the test scope
b) A definition of how specific requirements regarding testing (e.g. test-specific stakeholder requirements, ISO 26262) are covered.
c) A definition of the methods for test case and test data development (e.g. development of positive / negative tests, test of static and dynamic behavior, equivalence partitioning).
d) A definition of the criteria to select test cases including

   - the coverage of new or changed requirements
   - the coverage of changes in the architecture or interface specifications
   - the coverage of change requests
   - the coverage of item changes
   - the consideration of dependencies, based on the analysis of changes (e.g. causal chain analysis) and
   - the selection of appropriate test cases for regression testing including a set of test cases selected as a basis set to be executed.

e) A definition of the test environment regarding each test method
f) The assignment of test methods to project phases (e.g. stress test, smoke test and fault injection test).
g) A definition of the test coverage in relation to the project plan and release plan.
h) A definition of entry and exit criteria for the test
i) A documentation of sufficient test coverage of each test level, if the test levels (e.g. software qualification test, software integration test and unit test) are combined
j) An approach for the handling of failed tests

   Note: This aspect of the test strategy should refer to the Problem Resolution Management strategy (SUP.9.BP1).

Recommendations and rules:

> **[SYS.4.RL.2]** If the test strategy does not cover all aspects above, the indicator BP2 must not be rated F.
>
> **[SYS.4.RL.3]** If the test strategy does not cover aspect b), c) or d), the indicator BP2 must not be rated higher than P.

Related to:

- BP2 “Develop system integration test strategy including regression test strategy”
- Output WP 08-52 “Test plan”

3.4.1.3 Develop specification for integration test

The test specification has to be developed according to the strategy. For details see chapter 3.4.1.2, *“Test strategy”*, aspect b), c) and e).

Recommendations and rules:

> **[SYS.4.RL.4]** If the test specifications are not based on the architecture and interface specifications, the indicator BP3 must not be rated higher than P.

Related to:

- BP3 “Develop specification for system integration test”
- Output WP 08-50 “Test specification”

3.4.1.4 Select test cases

Test cases are selected from the integration test specification.

The expectations for a successful implementation of the test case selection cover the following aspects:

a) The selection of the test cases has to be performed according to the defined strategy
b) The selection of test cases has to consider the intended use of the deliverable item (test bench, test track, use on public road, …)
c) The used selection criteria (defined in the strategy) have to be documented
d) The selection of the test cases has to be documented

Recommendations and rules:

> **[SYS.4.RL.5]** If the test case selection does not cover the aspect a) and b), the indicator BP5 must not be rated F.

> **[SYS.4.RC.1]** If the test case selection does not cover the aspect c) and d), the indicator BP5 should not be rated F.

Related to:

- BP5 “Select test cases”
- Output WP 08-50 “Test specification”

3.4.1.5 Test implementation using automation

On the different levels of testing the execution of the test cases shall follow a test plan including the test strategy. According to the test cases specified this can be done by manual testing or by an automated approach using test scripts processed by a test automation tool or specific programmed test routines.

The expectations for a successful implementation cover these aspects:

- Completeness of test scripts and programs with respect to the test cases assigned to an automated test in the test specification
- Consistency of test scripts and programs with respect to each test case.

Recommendations and rules:

> **[SYS.4.RL.6]** If the test implementation is not complete in terms of all aspects above, the indicator BP6 must not be rated F.

Related to:

- BP6 “Perform system integration test”
- Output WP 13-50 “Test result”

3.4.1.6 Test logs as evidence for test results

By testing the system and the software on different levels a large amount of logged data may be generated, which have to be documented in test logs. This is especially true for automated tests. Also in tests performed manually the results may be provided in different levels of detail.

The expectations for a successful implementation cover this aspect:

- Test logs supplying a meaningful summary of the logged data as an adequate evidence for each test result.

Recommendations and rules:

[SYS.4.RL.7] If the test logs do not cover the aspect above, the indicator BP6 must not be rated F.

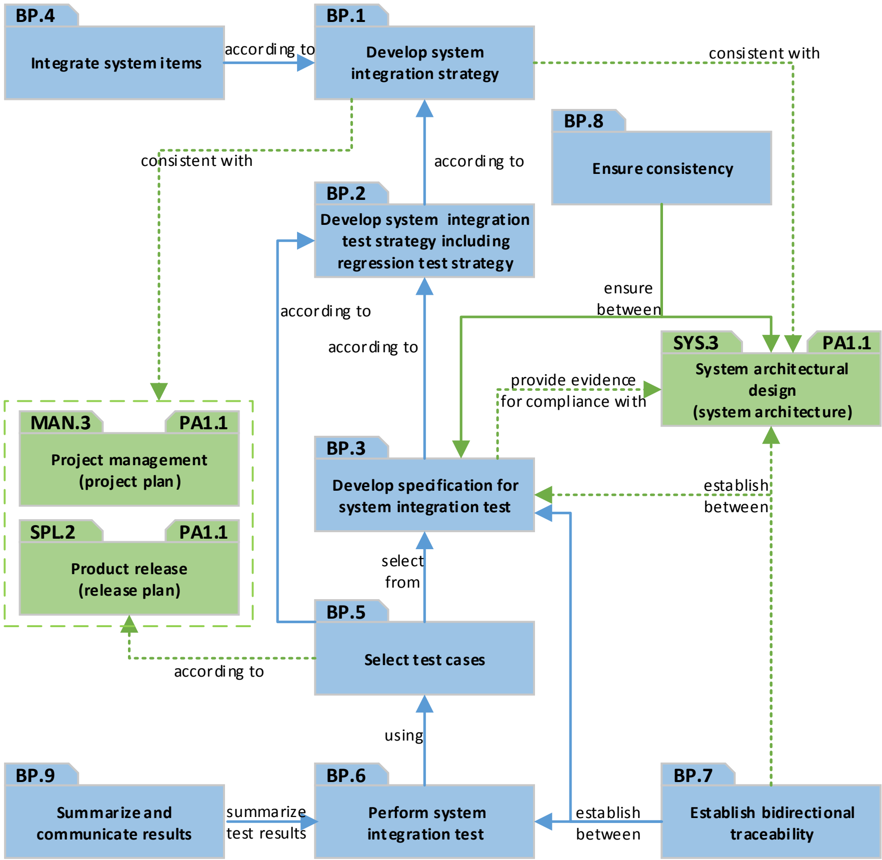
[SYS.4.RL.8] If the test results contain only a pure passed/failed information without a supporting test log, the indicator BP6 must not be rated higher than P

Related to:

- BP6 “Perform system integration test”
- Output WP 13-50 “Test result”

3.4.2 Rating consistency

The following figure shows the relationships between SYS.4 base practices as well as their relationships to other processes:



These relationships are used as the basis for the rating rules and recommendations defined in the following subchapters.

Generic aspects regarding traceability and consistency (2.1.1), summarize and communication (2.1.2), and strategy and plan (2.1.4) shall also be considered for rating.

3.4.2.1 Rating consistency within SYS.4

The following rating rule is related to the system integration test strategy and thus covers several base practices of the process:

> **[SYS.4.RL.9]** If the strategy-related activities are not performed according to the defined strategy (BP2), the indicators BP3 and BP5, respectively, shall be downrated.

Within SYS.4, the following base practices have relationships to each other:

BP2 Develop system integration test strategy including regression test strategy.

> **[SYS.4.RL.10]** If the test strategy is not developed according to the defined integration strategy (BP1), the indicator BP2 shall be downrated.

BP3 Develop specification for system integration test

> **[SYS.4.RL.11]** If the indicator for developing the test strategy (BP2) is downrated due to missing or inadequate definitions of methods for test case and test data development, the indicator BP3 shall be downrated.

BP4 Integrate system items

> **[SYS.4.RL.12]** If the strategy-related activities are not performed according to the defined strategy (BP1), the indicator BP4 shall be downrated.

BP5 Select test cases

> **[SYS.4.RL.13]** If the indicator for developing the test specification (BP3) is downrated, the indicator BP5 must not be rated higher.

> **[SYS.4.RL.14]** If the indicator for developing the test strategy (BP2) is downrated due to a missing or inadequate definition of the test case selection criteria, the indicatorBP5 shall be downrated.

BP6 Test integrated system

> **[SYS.4.RL.15]** If the indicator for selecting test cases (BP5) is rated P or N, the indicator BP6 shall be downrated.

3.4.2.2 Rating consistency to other processes at level 1

The following base practices of SYS.4 have relationships to other processes.

BP1 Develop system integration strategy

[SYS.4.RC.2] If project plan or release plan are not adequate, this should not be used to downrate the indicator BP1.

[SYS.4.RC.3] If the PA 1.1 for SYS.3 is downrated, this should be in line with the rating of the indicator BP1.

BP3 Develop specification for system integration test

[SYS.4.RC.4] If the PA 1.1 for SYS.3 is downrated, this should be in line with the rating of the indicator BP3.

BP5 Select test cases

[SYS.4.RC.5] If only the release plan is not adequate, but the test cases are selected according to the strategy, this should not be used to downrate the indicator BP5.

BP7 Establish bidirectional traceability

[SYS.4.RC.6] If PA 1.1 for SYS.3 is downrated, this should be in line with the rating of the indicator BP7.

BP8 Ensure consistency

[SYS.4.RC.7] If PA 1.1 for SYS.3 is downrated, this should be in line with the rating of the indicator BP8.

3.5 SYS.5 System Qualification Test

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.

3.5.1 Rating recommendations

3.5.1.1 Test strategy

In general, all testing activities follow a test strategy documented in a test plan.

The expectations for a test strategy cover these aspects:

a) A definition of the test scope
b) A definition of how specific requirements regarding testing (e.g. test-specific stakeholder requirements, ISO 26262) are covered.
c) A definition of the methods for test case and test data development (e.g. development of positive/negative tests, equivalence partitioning).
d) A definition of the criteria to select test cases including
    - the coverage of new or changed requirements
    - the coverage of change requests
    - the coverage of item changes
    - the consideration of dependencies, based on the analysis of changes (e.g. causal chain analysis) and
    - the selection of appropriate test cases for regression testing including a set of test cases selected as a basis set to be executed.
e) A definition of the test environment regarding each test method
f) The assignment of test methods to project phases (e.g. stress test, smoke test and fault injection test).
g) A definition of the test coverage in relation to project plan and release plan.
h) A definition of entry and exit criteria for the test
i) A documentation of sufficient test coverage of each test level, if the test levels (e.g. software qualification test, software integration test and unit test) are combined
j) An approach for the handling of failed tests

    Note: This aspect of the test strategy should refer to the Problem Resolution Management strategy (SUP.9.BP1).

Recommendations and rules:

> **[SYS.5.RL.1]** If the test strategy does not cover all aspects above, the indicator BP1 must not be rated F.

> **[SYS.5.RL.2]** If the test strategy does not cover aspect b), c) or d), the indicator BP1 must not be rated higher than P.

Related to:

- BP1 "Develop system qualification test strategy including regression test strategy"
- Output WP 08-52 "Test plan"

3.5.1.2 Develop specification for qualification test

The test specification has to be developed according to the strategy. For details see chapter 3.5.1.1, *"Test strategy"*, aspect b), c) and e).

Recommendations and rules:

> **[SYS.5.RL.3]** If the test specifications are not based on the requirement specifications and the verification criteria, the indicator BP2 must not be rated higher than P.

Related to:

- BP2 "Develop specification for system qualification test"
- Output WP 08-50 "Test specification"

3.5.1.3 Select test cases

Test cases are selected from the qualification test specification.

The expectations for a successful implementation of the test case selection cover the following aspects:

a) The selection of the test cases has to be performed according to the defined strategy
b) The selection of test cases has to consider the intended use of the deliverable item (test bench, test track, use on public road, …)
c) The used selection criteria (defined in the strategy) have to be documented
d) The selection of the test cases has to be documented

Recommendations and rules:

> **[SYS.5.RL.4]** If the test case selection does not cover the aspect a) and b), the indicator BP3 must not be rated F.
>
> **[SYS.5.RC.1]** If the test case selection does not cover the aspect c) and d), the indicator BP3 should not be rated F.

Related to:

- BP3 “Select test cases”
- Output WP 08-50 “Test specification”

3.5.1.4 Test implementation using automation

On the different levels of testing the execution of the test cases shall follow a test plan including the test strategy. According to the test cases specified this can be done by manual testing or by an automated approach using test scripts processed by a test automation tool or specific programmed test routines.

The expectations for a successful implementation cover these aspects:

- Completeness of test scripts and programs with respect to the test cases assigned to an automated test in the test specification
- Consistency of test scripts and programs with respect to each test case.

Recommendations and rules:

> **[SYS.5.RL.5]** If the test implementation is not complete in terms of all aspects above, the indicator BP4 must not be rated F.

Related to:

- BP4 “Test integrated system”
- Output WP 13-50 “Test result”

3.5.1.5 Test logs as evidence for test results

By testing the system and the software on different levels a large amount of logged data may be generated, which have to be documented in test logs. This is especially true for automated tests. Also in tests performed manually the results may be provided in different levels of detail.

The expectations for a successful implementation cover this aspect:

- Test logs supplying a meaningful summary of the logged data as an adequate evidence for each test result.

Recommendations and rules:

[SYS.5.RL.6] If the test logs do not cover the aspect above, the indicator BP4 must not be rated F.

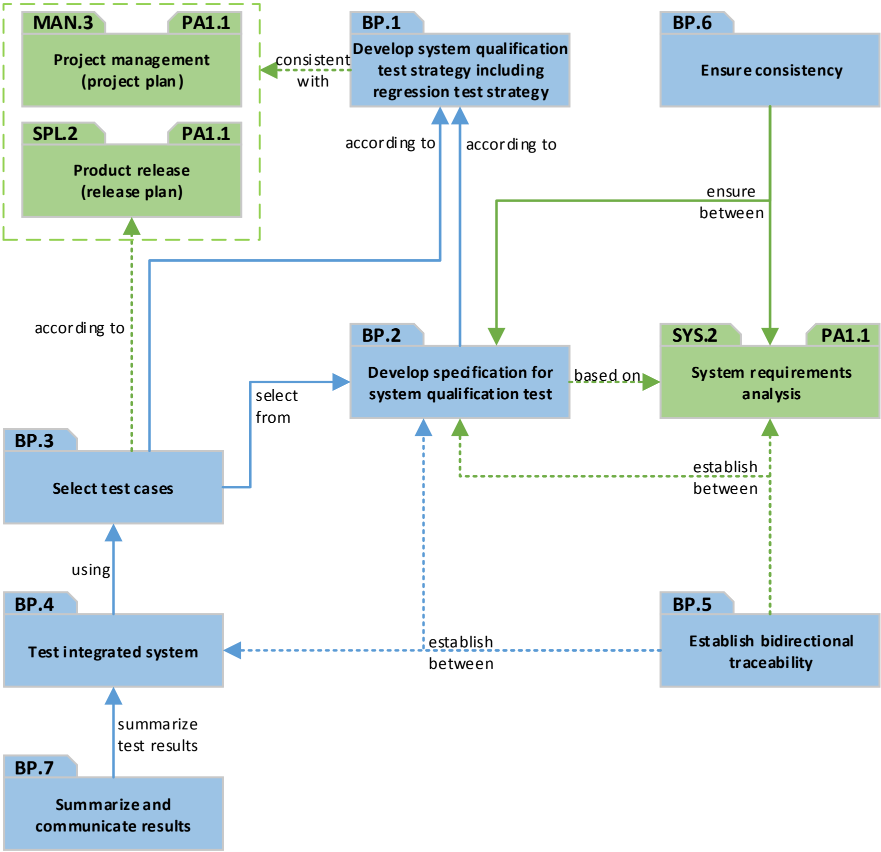
[SYS.5.RL.7] If the test results contain only a pure passed/failed information without a supporting test log, the indicator BP4 must not be rated higher than P

Related to:

- BP4 “Test integrated system”
- Output WP 13-50 “Test result”

3.5.2 Rating consistency

The following figure shows the relationships between SYS.5 base practices as well as their relationships to other processes:



These relationships are used as the basis for the rating rules and recommendations defined in the following subchapters.

Generic aspects regarding traceability and consistency (2.1.1), summarize and communication (2.1.2), and strategy and plan (2.1.4) shall also be considered for rating.

3.5.2.1 Rating consistency within SYS.5

The following rating rule is related to the system qualification test strategy and thus covers several base practices of the process:

> **[SYS.5.RL.8]** If the strategy-related activities are not performed according to the defined strategy (BP1), the indicators BP2 and BP3, respectively, shall be downrated.

Within SYS.5, the following base practices have relationships to each other:

BP2 Develop specification for system qualification test

> **[SYS.5.RL.9]** If the indicator for developing the test strategy (BP1) is downrated due to missing or inadequate definitions of methods for test case and test data development, the indicator BP2 shall be downrated.

BP3 Select test cases

> **[SYS.5.RL.10]** If the indicator for developing the test specification (BP2) is downrated, the indicator BP3 must not be rated higher.

> **[SYS.5.RL.11]** If the indicator for developing the test strategy (BP1) is downrated due to a missing or inadequate definition of the test case selection criteria, the indicator BP3 shall be downrated.

BP4 Test integrated system

> **[SYS.5.RL.12]** If the indicator for selecting test cases (BP3) is rated P or N, the indicator BP4 shall be downrated.

3.5.2.2 Rating consistency to other processes at level 1

The following base practices of SYS.5 have relationships to other processes.

BP1 Develop system qualification test strategy including regression test strategy

> **[SYS.5.RC.2]** If project plan or release plan are not adequate, this should not be used to downrate the indicator BP1.

BP2 Develop specification for system qualification test

> **[SYS.5.RC.3]** If the PA 1.1 for SYS.2 is downrated, this should be in line with the rating of the indicator BP2.

BP3 Select test cases

[SYS.5.RC.4] If only the release plan is not adequate, but the test cases are selected according to the strategy, this should not be used to downrate the indicator BP3.

BP5 Establish bidirectional traceability

[SYS.5.RC.5] If PA 1.1 for SYS.2 is downrated, this should be in line with the rating of the indicator BP5.

BP6 Ensure consistency

[SYS.5.RC.6] If PA 1.1 for SYS.2 is downrated, this should be in line with the rating of the indicator BP6.

3.6 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 Software Requirements Analysis process uses the system requirements that were processed in the System Requirements Analysis process and the system architecture as an input. Such system requirements can be either functional or non-functional.

Normally the functionality and software content grows over the releases. Therefore, the complete set of requirements is not necessarily available at project start. This has to be considered by rating the completeness of the work products.

3.6.1 Rating recommendations

3.6.1.1 Software requirements

Functions are implemented in mechanics, hardware or software or as a combination of these elements. Software requirements are derived from the system requirements that are categorized to be implemented in software.

In case of software development only, the software requirements may refer directly to the stakeholder requirements.

Software Requirements may address:

- Functional parts to be implemented in software including safety and/or security related functions
- Hardware related software functions
- Receiving signals from electronic sensors
- Processing of signals
- Control of electronic hardware (actuators)
- Structure and storage of data
- Parameters defining the software behavior
- Non-functional requirements (e.g. safety, security, quality requirements)

Software requirements have to be granular and understandable. Unclear or generic requirements have to be clarified with the system requirement owner.

The existence of a set of software requirements applicable for the project shall be demonstrated as a populated list or data base that allows the structuring of the software requirements.

The implementation of software requirements will be verified in the integrated software (SWE.6).

Recommendations and rules:

> **[SWE.1.RL.1]** If there is no evidence that unclear or inconsistent requirements are not clarified with the respective system requirement owner, the indicator BP1 shall be downrated.
>
> **[SWE.1.RL.2]** If the software requirements specification is not reflecting latest changes, the indicator BP1 must not be rated higher than L.
>
> **[SWE.1.RL.3]** If software requirements are not derived from system requirements but from platform requirements according to a reuse strategy the indicator BP1 must not be downrated.

Related to:

- BP1 “Specify software requirements”
- Output WP 17-11 “Software requirements specification”
- Output WP 17-08 “Interface requirements specification”
- Output WP 13-21 “Change control record”

3.6.1.2 Structure software requirements

Software requirements are structured by grouping, sorting and categorizing to support a prioritization and to map the required functionality to future releases. The structure and categorization of the software specification enables the project to manage the requirements in terms of e.g., organizational, technical, legal and internal topics.

Recommendations and rules:

> **[SWE.1.RL.4]** If the categorization is not appropriate as mentioned above, the indicator BP2 must not be rated higher than L.
>
> **[SWE.1.RL.5]** If the mapping of functionality to the releases does not reflect the customer and other stakeholder needs, the indicator BP2 shall be downrated.

[SWE.1.RC.1] If there is no evidence for prioritization but a release plan maps the functionality to future releases the indicator BP2 should not be downrated.

Related to:

- BP2 “Structure software requirements”
- Output WP 15-01 “Analysis report”

3.6.1.3 Analyze

The analysis of software requirements is the basis for a correct implementation. Even though requirements sometimes are very detailed or their implementation seems to be very simple, a well-founded analysis has to be conducted for those requirements, too. The scope and appropriateness of the analysis and its documentation depend on the context of product (e.g., platform) and organization. The result of analysis can vary from a simple attribute in a list to a complex simulation or the building of a demonstrator to evaluate the feasibility of software requirements. Doubts in feasibility of functionality have to be reflected in MAN.5.

Recommendations and rules:

[SWE.1.RL.6] If the software requirements and their interdependencies are not evaluated in terms of correctness, technical feasibility and verifiability the indicator BP3 must not be rated F.

[SWE.1.RC.2] If the analysis of impact on cost and schedule is covered by the estimation of work packages in the project planning this should not be used to downrate the indicator BP3.

Related to:

- BP3 “Analyze software requirements”
- Output WP 15-01 “Analysis report”
- Output WP 17-50 “Verification criteria”

3.6.1.4 Impact on the operating environment

The analysis of the impact on the operating environment covers the impact on the software in scope as well as the impact on other software parts, other systems or the entire vehicle considering the following possible aspects:

- Interfaces
  - Signals and signal quality
  - Voltage and current
- Environment
  - Temperature
  - EMC
- Performance:
  - Interface response times (signal response, sample time, cycle time, bus load, signal delay, jitter)
  - Micro controller response time (processing time)
- Resources
  - RAM / ROM memory usage
  - EEPROM / DataFlash memory usage

Recommendations and rules:

[SWE.1.RC.3] If the analysis of the impact on the operating environment is not considering aspects from the list above or other aspects that are relevant for the project the indicator BP4 should be downrated.

[SWE.1.RC.4] If there are insufficient reserves of memory, processor time, and/or peripheral resources the indicator BP4 should be downrated.

Rationale: Insufficient reserves of memory, processor time and/or peripheral resources are signs for inappropriate analysis of technical feasibility or inappropriate analysis of impact on the operating environment.

Related to:

- BP4 “Analyze the impact on the operating environment”
- Output WP 15-01 “Analysis report”
- Output WP 17-08 “Interface requirements specification”

3.6.1.5 Verification criteria

Refer to chapter 2.1.3 for the generic concept of the term Verification Criteria.

Recommendations and rules:

> **[SWE.1.RL.7]** If verification criteria are not documented as a separate work product but demonstrably contained in the requirement or test specification the indicator BP5 must not be downrated.

Related to:

- BP5 “Develop verification criteria”
- Output WP 17-50 “Verification Criteria”

3.6.1.6 Software development only

In case of software development only, the software requirements refer directly to the stakeholder requirements. In consequence consistency and bidirectional traceability have to be ensured between stakeholder requirements and software requirements.

Recommendations and rules:

> **[SWE.1.RL.8]** In the case of software development only, if the traceability from software requirements to stakeholder requirements is established, the indicator BP6 must not be downrated.

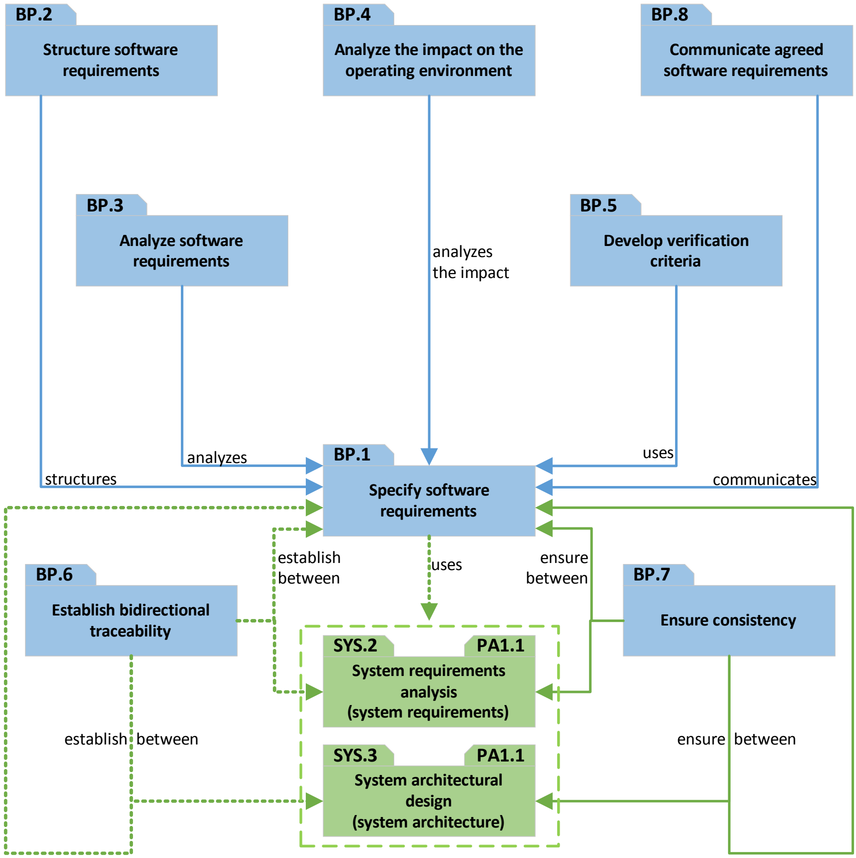
> **[SWE.1.RL.9]** In the case of software development only, if the consistency from software requirements to stakeholder requirements is established, the indicator BP7 must not be downrated.

Related to:

- BP6 “Establish bidirectional traceability”
- BP7 “Ensure consistency”
- Output WP 13-22 “Traceability record”

3.6.2 Rating consistency

The following figure shows the relationships between SWE.1 base practices as well as their relationships to other processes.



These relationships are used as the basis for the rating rules and recommendations defined in the following subchapters.

Generic aspects regarding traceability and consistency (2.1.1), communication (2.1.2), and verification criteria (2.1.3) shall also be considered for rating.

3.6.2.1 Rating consistency within SWE.1

The following rating rule is related to the specification of the software requirements and thus influences several base practices of the process:

[SWE.1.RL.10] If the specification of software requirements (BP1) is downrated, PA 1.1 shall be downrated as all indicators (BP2, BP3, BP4, BP5, BP6, BP7 and BP8) are affected.

3.6.2.2 Rating consistency to other processes at level 1

The following base practices of SWE.1 have relationships to other processes:

BP1 Specify software requirements

[SWE.1.RC.5] If PA 1.1 for SYS.2 is downrated, this should be in line with the rating of the indicator BP1.

[SWE.1.RC.6] If PA 1.1 for SYS.3 is downrated, this should be in line with the rating of the indicator BP1.

BP3 Analyze system requirements

[SYS.2.RC.7] If the indicator BP.3 is downrated, this should be in line with the rating of the indicator determine, monitor and adjust project estimates and resources (MAN.3.BP5).

[SYS.2.RC.8] If the indicator BP.3 is downrated, this should be in line with the rating of the indicator evaluate feasibility of the project (MAN.5.BP3) with regard to risks.

BP6 Establish bidirectional traceability

[SWE.1.RC.9] If PA 1.1 for SYS.2 is downrated, this should be in line with the rating of the indicator BP6 (see 2.1.1).

[SWE.1.RC.10] If PA 1.1 for SYS.3 is downrated, this should be in line with the rating of the indicator BP6 (see 2.1.1).

BP7 Ensure consistency

[SWE.1.RC.11] If PA 1.1 for SYS.2 is downrated, this should be in line with the rating of the indicator BP7 (see 2.1.1).

[SWE.1.RC.12] If PA 1.1 for SYS.3 is downrated, this should be in line with the rating of the indicator BP7 (see 2.1.1).

3.7 SWE.2 Software Architectural Design

The purpose of the Software Architectural Design Process is to establish an architectural design and to identify which software requirements are to be allocated to which elements of the software, and to evaluate the software architectural design against defined criteria.

For technical projects in most cases the solution space for an architecture is manifold and not biunique. In addition, the solution for the architecture is influenced by several other not necessarily technical drivers (non-functional technical requirements).

Possible software requirements for the definition of an architecture are e.g.

- Non-functional technical requirements
  - Performance (response time, sample time, cycle time, deadline, flow)
  - Safety (non-functional safety aspects e.g. fault tolerant software architecture)
  - Security
  - COTS (Commercial Of The Shelf) elements with defined interfaces
  - etc.
- Maintainability requirements
  - Usability
  - Simplicity
  - Maximum cohesion and minimum coupling
  - Testability
  - Analyzability
  - Modifiability
  - Application interface, coder
  - etc.
- Business requirements
  - Costs
  - Portability (reuse, platform, legacy interfaces)
  - Scalability
  - etc.

Some of these aspects are in contradiction to each other so that in most cases the finally selected architecture is a compromise between these criteria.

3.7.1 Rating recommendations

3.7.1.1 Develop software architectural design

The software architectural design is the highest level design description of the software with different (high level) abstraction views reflecting concerns of different stakeholders. The term “stakeholder” is not limited to the customer and could include as well e.g. strategic planning, project management, development, testing, quality assurance, safety etc. of the supplier and other entities e.g. legal bodies.

These views are architecture visualizations that are required for communication, discussion, reviews, analysis, evaluation, planning, change request analysis, impact analysis, maintenance etc. of the software.

There is no common definition which views are required and no criteria for the completeness of the sum of views. There are some approaches in the industry that specify the kind of information that is required for the view (“viewpoints” which are collections of patterns, templates, and conventions for constructing one type of view) and the integration of the views in a thoroughly architectural design description.

In most cases the software architectural design is a graphical representation of the software supplemented by textual explanations. The graphical representation consists at least of a static view providing an overview of the structure and a dynamic view describing the designated behavior of the software.

Static software architecture views allow the decomposition of the software into manageable elements with high cohesion and low coupling. Is decomposition supports the assignment of requirements to these architecture elements and will help the organization to distribute the work to the developers. Architecture elements of the software that are developed external to the assessment scope (e. g. open-source software, platform software, third-party software, etc.) will also be included as dedicated elements in the software architectural design and have to be considered as well for interface analysis, dynamic behavior, resource consumption objectives etc.

As appropriate the architecture elements are detailed further in the architectural design down to the components as the lowest level elements. The components consist of one or more units and are subject of the software detailed design process (SWE.3) (See “Annex C Terminology” of the PAM for definition of the terms element and component).

The number of hierarchical levels needed to define manageable elements of the software may be different for each element of the highest level. The appropriate level of detail is e.g. driven by:

- The need for encapsulation and modularization of components
- The need for integration of reused components
- The complexity of the element
- The need for maintainability
- The allocation of requirements to components
- The distribution of work
- The need to enable parallel work on components

On each layer of the static view of the software architectural design the interfaces between the elements are required to be identified.

The detailed aspects of the interface descriptions of software architectural design and the detailed dynamic aspects of software architectural design are described below.

Recommendations and rules:

> **[SWE.2.RL.1]** If the software architecture does not reflect dynamic views the indicator BP1 shall be downrated.
>
> **[SWE.2.RL.2]** If the software architecture does not reflect applicable non-functional requirements the indicator BP1 shall be downrated.

Related to:

- BP1 “Develop software architectural design”
- Output WP 04-04 “Software architectural design”
- Output WP 17-08 “Interface requirement specification”

3.7.1.2 Allocate software requirements

For software requirements that are structured as per SWE.1.BP2 it is required to allocate the requirements to the elements of the software architectural design derived by BP1. At the stage of software architectural design, the allocation typically is done on the level of suitable requirement clusters (e.g. a chapter in requirements specification) and not on the level of single requirements.

The allocation shall be traceable e.g. by a matrix or based on additional corresponding attribute/information in the used requirements management tool.

Each requirement or requirement cluster is required to be mapped to at least one element of the software architectural design ("no requirement is forgotten"). This mapping could be one direction of the bidirectional traceability addressed by BP7.

Recommendations and rules:

> **[SWE.2.RL.3]** If the allocation of software requirements to elements of the software architectural design is done based on clusters but not on single requirements, the indicator BP2 must not be downrated.

Related to:

- BP2 "Allocate software requirements"

3.7.1.3 Define interfaces of software elements.

Software interfaces represent the input and the output of an element of the software architecture. A software interface is defined by sender, receiver, format, size, resolution, quality information, frequency etc. of the data being transferred.

Recommendations and rules:

> **[SWE.2.RL.4]** If the software interface definition is absent or incomplete regarding the definition above the indicator BP3 shall be downrated.

Related to:

- BP3 "Define interfaces of software elements"
- Output WP 17-08 "Interface requirement specification"

3.7.1.4 Describe dynamic behavior

For describing the dynamic behavior of a software system at runtime behavioral descriptions are required e.g.

- State transition diagrams
- Sequence diagrams
- Message sequence charts
- Use-case diagrams

Which ones are required or suitable depends on the application.

In addition, the response times have to be considered for defining e.g.:

- Tasks
- Threading concept
- Time slices
- Interrupts
- Interfaces

Recommendations and rules:

[SWE.2.RL.5] If evidence of describing dynamic behavior regarding the topics mentioned above is missing the indicator BP4 shall be downrated.

Related to:

- BP4 “Describe dynamic behavior”
- Output WP 04-04 “Software architectural design”
- Output WP 17-08 “Interface requirement specification”

3.7.1.5 Define resource consumption objectives

For the software architectural design, the required response times and resources for memory (ROM, RAM, external / internal EEPROM or Data Flash) are required to be derived from stakeholder requirements for all resource-critical elements of the software architectural design and may depend on project phases. The term “stakeholder” is not limited to the customer and could include as well the system and platform development.

For CPU load the execution environment has to be considered as well e.g. usage profile and external triggers (e.g. interrupts).

Recommendations and rules:

> **[SWE.2.RL.6]** If evidence of describing resource consumption objectives regarding the definition mentioned above is missing the indicator BP5 shall be downrated.

Related to:

- BP5 “Define resource consumption objectives”
- Output WP 04-04 “Software architectural design”

3.7.1.6 Evaluate alternative software architectures

One of the following three approaches for architecture development should be used in practice and should be identifiable for an assessor:

1) <u>Development of alternative solutions (e.g. for development of a completely new system):</u>

   Several potential solutions for the software architecture are described at least up to an abstraction level that allows the identification of the main differences between the architectures and that allows the evaluation of the most important architecture criteria for each of the potential solutions. Based on this first evaluation at least one of the proposed solutions is elaborated further by performing the base practices (BP1 to BP5). It has to be ensured that the proposed solutions that are chosen for further elaboration are able to cover the required needs of the project. At the end, these proposed and refined solutions are evaluated based on the defined evaluation criteria (BP6) and a decision is made:

   - Selection/Confirmation of one/the proposed solution as the used architecture for further development or
   - Rejection of the previous proposed solution(s) and step back to architecture development (BP1)

2) <u>Iterative architecture development:</u>

   During the development of an architecture by performing the base practices (BP1 to BP5) several variants for the used architecture arises. A variant can be a completely different architecture or a variant can differ from an already identified proposed solution only in some aspects or viewpoints. As a consequence, the evaluation of the chosen criteria (BP6) can take place several times during the elaboration of the architecture that is chosen at the end and that is used as the basis for the further development.

3) Carry over and adaption of an existing architecture (e.g. for platform development):

   Although only one solution approach is used for the architecture development it has to be ensured that the chosen approach is suitable for the project and valid according to the chosen evaluation criteria. So BP6 is reduced to the evaluation of one solution only. Identified weaknesses during the evaluation should be eliminated or the consequences of the weaknesses in the chosen architecture have to be made transparent.

In any case it has to be ensured that all relevant parties and all necessary competencies are involved in the agreement on the selection of the final software architecture.

Recommendations and rules:

[SWE.2.RL.7] If none of the three described approaches for architecture development is observable in the assessed project, PA 1.1 shall be downrated.

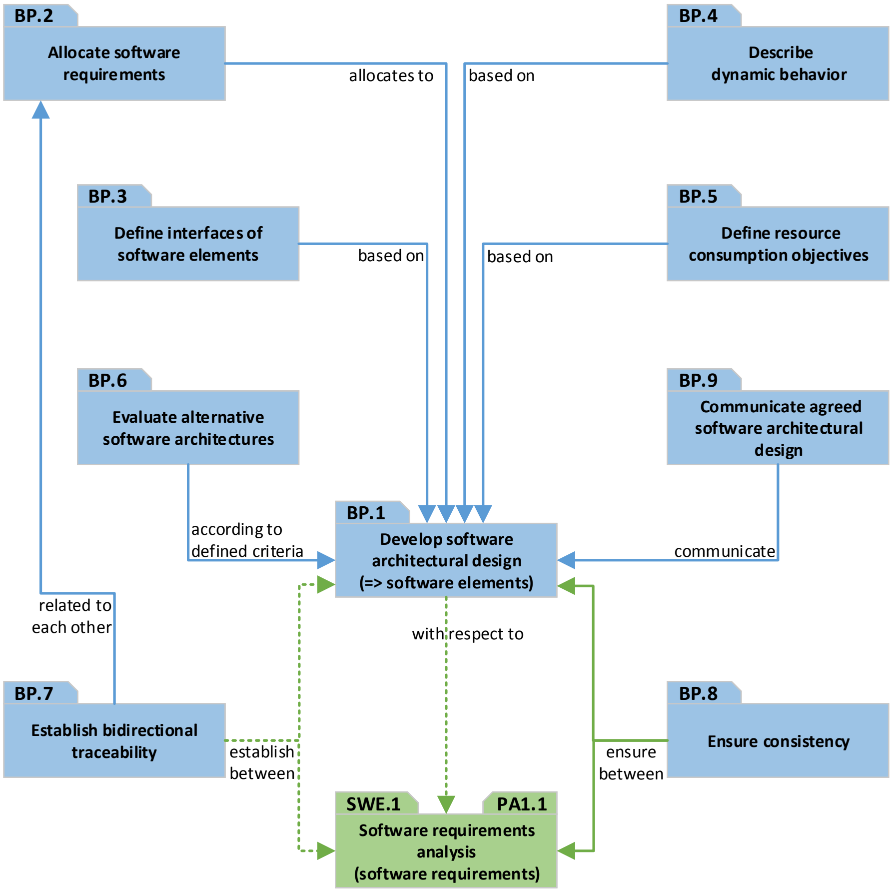
[SWE.2.RC.1] If the used procedure for architecture selection does not involve the required parties or competencies, the indicator BP6 should be downrated.

Related to:

- BP6 “Evaluate alternative software architectures”
- Output WP 04-04 “Software architectural design”

3.7.2 Rating consistency

The following figure shows the relationships between SWE.2 base practices as well as their relationships to other processes:



These relationships are used as the basis for the rating rules and recommendations defined in the following subchapters.

Generic aspects regarding traceability and consistency (2.1.1), and communication (2.1.2) shall also be considered for rating.

3.7.2.1 Rating consistency within SWE.2

The following rating rule is related to the development of the software architectural design and thus influences several base practices of the process:

[SWE.2.RL.8] If the development of the software architectural design (BP1) is downrated, PA 1.1 shall be downrated as all indicators (BP2, BP3, BP4, BP5, BP6, BP7, BP8 and BP9) are affected.

[SWE.2.RL.9] If the allocation of the software requirements to elements of the software architectural design BP2 is downrated, the indicator BP7 shall be downrated.

3.7.2.2 Rating consistency to other processes at level 1

The following base practices of SWE.2 have relationships to other processes:

BP1 Develop software architectural design

[SWE.2.RC.3] If PA 1.1 for SWE.1 is downrated, this should be in line with the rating of the indicator BP1.

BP7 Establish bidirectional traceability

[SWE.2.RC.4] If PA 1.1 for SWE.1 is downrated, this should be in line with the rating of the indicator BP7.

BP8 Ensure consistency

[SWE.2.RC.5] If PA 1.1 for SWE.1 is downrated, this should be in line with the rating of the indicator BP8.

3.8 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 software detailed design refines the components specified in the Software Architecture Design process into software units and their interfaces. These software units that are not further refined on the design level and their interfaces are the basis for generating or developing the source code for the derived software units.

The detailed design for a component shall describe the approach to satisfy the mapped software requirements by describing plans of how code will be organized both statically and dynamically. It shall also describe how different modules will interact.

For technical projects in most cases the solution space for a detailed design is not biunique. In addition, the solution for the detailed design is influenced by several other not necessarily technical drivers (non-functional technical requirements.

Possible software requirements for the definition of a detailed design are e.g.:

- Non-functional technical requirements
  - Performance (response time, sample time, cycle time, deadline, flow)
  - Safety (non-functional safety aspects e.g. program flow monitoring)
  - Security
  - COTS (Commercial of the Shelf) elements with defined interfaces
  - etc.
- Maintainability requirements
  - Usability
  - Simplicity
  - Maximum cohesion and minimum coupling
  - Testability
  - Analyzability
  - Modifiability

- Application interface, coder
- etc.

- Business requirements
  - Costs
  - Portability (reuse, platform, legacy interfaces)
  - Scalability
  - etc.

Some of these aspects are in contradiction to each other so that in most cases the finally selected detailed design is a compromise between these criteria.

3.8.1 Rating recommendations

3.8.1.1 Develop software detailed design

The description of the software detailed design consists of different abstraction views reflecting concerns of different stakeholders: These are mainly a structural view of the relations and interfaces between software units, a dynamic view of the dynamic behavior of the interaction of software units, and a dynamic view of the internal behavior of the software units. The term “stakeholder” is not limited to the customer and could include as well e.g. strategic planning, project management, development, testing, quality assurance, safety etc. of the supplier and other entities e.g. legal bodies.

These views are detailed design visualizations or representations that are required for communication, discussion, reviews, analysis, evaluation, planning, change request analysis, impact analysis, maintenance etc. of the software.

There is no common definition which views are required and no criteria for the completeness of the sum of views. There are some approaches in the industry that specify the kind of information that is required for the view (“viewpoints” which are collections of patterns, templates, and conventions for constructing one type of view) and the integration of the views in a thoroughly detailed design description.

In most cases the software detailed design is a mix of graphical representation and/or textual explanations. The graphical representation consists at least of a static view providing an overview of the software units of a component and its interfaces and a dynamic view describing the designated behavior of the interaction of the software units. For the description of the internal behavior of the software units graphical and/or textual (pseudo code) representations are used.

Static software detailed design views allow the decomposition of the software into manageable software units with high cohesion and low coupling. These software units support the allocation of requirements to these detailed design elements and will help the organization to distribute the work to the developers. Software units that are developed external to the assessment scope (e. g. open-source software, platform software, third-party software, etc.) will also be included as dedicated elements in the software detailed design and have to be considered as well for interface analysis, dynamic behavior etc.

The decomposition of the components into software units is e.g. driven by:

- The need for encapsulation and modularization of software units
- The need for integration of reused software units
- The need for testability
- The complexity of the software units
- The need for maintainability
- The distribution of work
- The need to enable parallel work on software units

In the static view of the software detailed design the interfaces between the software units are required to be identified and defined.

The detailed aspects of the interface descriptions of software detailed design and the detailed dynamic aspects of software detailed design are described below.

Recommendations and rules:

[SWE.3.RL.1] If the software detailed design does not reflect dynamic views the indicator BP1 shall be downrated.

[SWE.3.RL.2] If the software detailed design does not reflect applicable non-functional requirements the indicator BP1 shall be downrated.

Related to:

- BP1 “Develop software detailed design”
- Output WP 04-05 “Software detailed design”

3.8.1.2 Define interfaces of software units

Software interfaces represent the input and the output of a software unit. A software interface is defined by sender, receiver, format, size, resolution, quality information, frequency etc. of the data being transferred.

Recommendations and rules:

> **[SWE.3.RL.3]** If the software interface definition is absent or incomplete regarding the definition above the indicator BP2 shall be downrated.

Related to:

- BP2 “Define interfaces of software units”
- Output WP 04-05 “Software detailed design”

3.8.1.3 Describe dynamic behavior

For describing the dynamic behavior of a software component at runtime behavioral descriptions are required e.g.

- State transition diagrams
- Sequence diagrams
- Message sequence charts
- Use-case diagrams

Which ones are required or suitable depends on the intended application of the component. The same applies to the derived software units: e.g. except for execution time it may be not necessary to describe dynamic behavior for software units which are not complex.

In addition, the response times have to be considered for defining e.g.:

- Tasks
- Threading concept
- Time slices
- Interrupts
- Interfaces

Recommendations and rules:

[SWE.3.RL.4] If evidence of describing dynamic behavior regarding the topics mentioned above is missing the indicator BP3 shall be downrated.

Related to:

- BP3 “Describe dynamic behavior”
- Output WP 04-05 “Software detailed design”

3.8.1.4 Evaluate software detailed design

One of the following three approaches for detailed design development should be used in practice and should be identifiable for an assessor:

1) <u>Development of alternative solutions (e.g. for development of a completely new component):</u>

   Several potential solutions for the software detailed design are described at least up to an abstraction level that allows the identification of the main differences between the detailed designs and that allows the evaluation in terms of interoperability, interaction, criticality, technical complexity, risks and testability for each of the potential solutions. Based on this first evaluation at least one of the proposed solutions is elaborated further by performing the base practices (BP1 to BP3). It has to be ensured that the proposed solutions which are chosen for further elaboration are able to cover the required needs of the project. Finally, these proposed and refined solutions are evaluated based on the mentioned evaluation criteria (BP4) and a decision is made:

   - Selection/Confirmation of one/the proposed solution as the used detailed design for further development or
   - Rejection of the previous proposed solution(s) and step back to detailed design development (BP1)

2) <u>Iterative detailed design development:</u>

   During the development of a detailed design by performing the base practices (BP1 to BP3) several variants for the used detailed design arise. A variant can be a completely different detailed design or a variant can differ from an already identified proposed solution only in some aspects or viewpoints. As a consequence the evaluation of the chosen criteria (BP4) can take place several times during the elaboration of the detailed design that is chosen at the end and that is used as the basis for the further development.

3) Carry over and adaption of an existing detailed design (e.g. for platform development):

   Although only one solution approach is used for the detailed design development it has to be ensured that the chosen approach is suitable for the project and valid according to the mentioned evaluation criteria. So BP4 is reduced to the evaluation of one solution only. Identified weaknesses during the evaluation should be eliminated or the consequences of the weaknesses in the chosen detailed design have to be made transparent.

In any case, it has to be ensured that all relevant parties and all necessary competencies are involved in the agreement on the selection of the final software detailed design.

Recommendations and rules:

> **[SWE.3.RL.5]** If none of the three described approaches for detailed design development is observable in the assessed project, PA 1.1 shall be downrated.
>
> **[SWE.3.RC.1]** If the used procedure for detailed design selection does not involve the required parties or competencies, the indicator BP4 should be downrated.

Related to:

- BP4 “Evaluate alternative software detailed designs”
- Output WP 04-05 “Software detailed design”

3.8.1.5 Develop software units

In case of model-based development the source code for software units is typically produced by adequate code generation tools.

Refer to chapter 2.2.1 for further guidance when using model-based development.

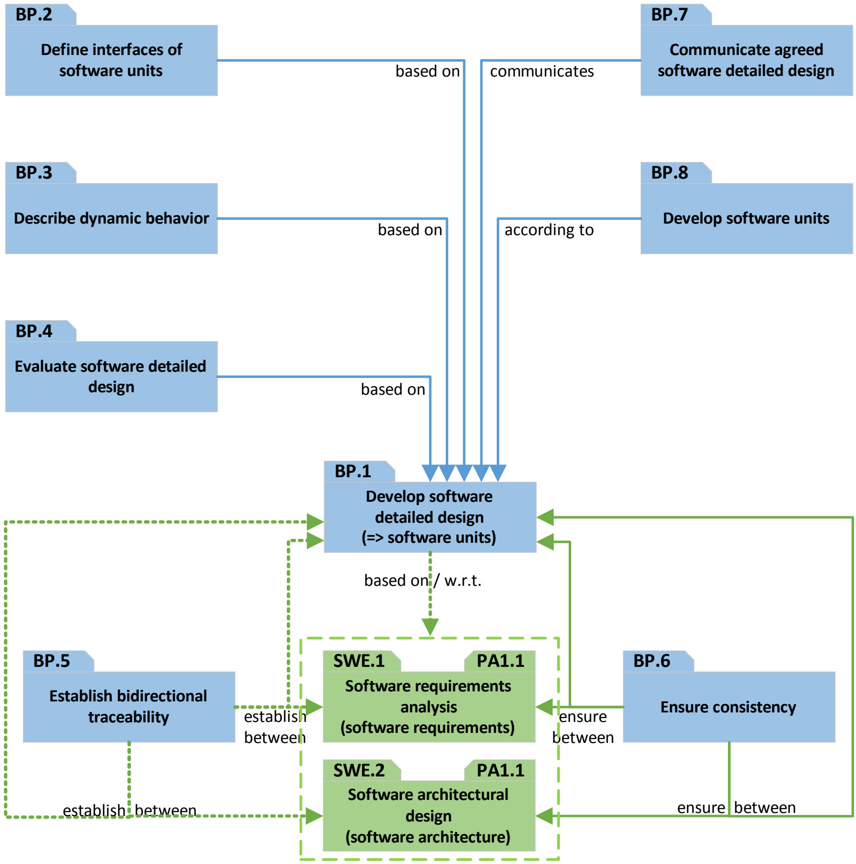
Software units must not contain content that is not described in the detailed design as e.g. this supports maintainability or defect analysis.

Recommendations and rules:

> **[SWE.3.RL.6]** If software units contain content which is not described in the detailed design, the indicator BP8 shall be downrated.

3.8.2 Rating consistency

The following figure shows the relationships between SWE.3 base practices as well as their relationships to other processes:



These relationships are used as the basis for the rating rules and recommendations defined in the following subchapters.

Generic aspects regarding traceability and consistency (2.1.1), and communication (2.1.2) shall also be considered for rating.

3.8.2.1 Rating consistency within SWE.3

The following rating rule is related to the development of the software detailed design and thus influences several base practices of the process:

[SWE.3.RL.7] If the development of the software detailed design (BP1) is downrated, PA 1.1 shall be downrated as all indicators (BP2, BP3, BP4, BP5, BP6, BP7, and BP8) are affected.

3.8.2.2 Rating consistency to other processes at level 1

The following base practices of SWE.3 have relationships to other processes:

BP1 Develop software detailed design

[SWE.3.RC.2] If PA 1.1 for SWE.1 is downrated, this should be in line with the rating of the indicator BP1.

[SWE.3.RC.3] If PA 1.1 for SWE.2 is downrated, this should be in line with the rating of the indicator BP1.

BP5 Establish bidirectional traceability

[SWE.3.RC.4] If PA 1.1 for SWE.1 is downrated, this should be in line with the rating of the indicator BP5.

[SWE.3.RC.5] If PA 1.1 for SWE.2 is downrated, this should be in line with the rating of the indicator BP5.

BP6 Ensure consistency

[SWE.3.RC.6] If PA 1.1 for SWE.1 is downrated, this should be in line with the rating of the indicator BP6.

[SWE.3.RC.7] If PA 1.1 for SWE.2 is downrated, this should be in line with the rating of the indicator BP6.

3.9 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 software unit verification process covers not only software unit testing aspects but also unit verification aspects e.g. static verification of units.

3.9.1 Rating recommendations

3.9.1.1 Software unit verification strategy

In general, all software unit verification activities follow a software unit verification strategy documented in a test plan.

The expectations for a software unit verification strategy cover these aspects:

a) A definition of the verification objects
b) A definition of how specific requirements regarding verification and testing (e.g. test-specific stakeholder requirements, ISO 26262, Metrics, MISRA) are covered.
c) A definition of the methods for test case and test data development derived from the detailed design and the non-functional requirements (e.g. development of positive/negative tests, equivalence partitioning).
d) A definition of the methods and tools for static verification and for reviews
e) A definition of the test environment regarding each test method
f) A definition of the test coverage in relation to project and release plan
g) A definition of entry and exit criteria for the software unit verification
h) A documentation of sufficient test coverage of each test level, if the test levels (e.g. software qualification test, software integration test and unit test) are combined
i) An approach for the handling of failed tests, failed static verifications (e.g. justification for failed MISRA-check or compiler warnings) and review findings.

   Note: This aspect of the software unit verification strategy should refer to SUP.9 Problem Resolution Management strategy

Recommendations and rules:

> **[SWE.4.RL.1]** If the software unit verification strategy does not cover all aspects above, the indicator BP1 must not be rated F.

> **[SWE.4.RL.2]** If the software unit verification strategy does not cover aspect b), c) or d), the indicator BP1 must not be rated higher than P.

Related to:

- BP1 “Develop software unit verification strategy including regression strategy”
- Output WP 08-52 “Test plan”

3.9.1.2 Verification logs as evidence for verification results

By verifying the software units, large amount of logged data may be generated, which have to be documented in verification logs. This is especially true for automated tests and static verification. Also if verification is performed manually the results may be provided in different levels of detail.

The expectations for a successful implementation cover this aspect:

- Verification logs supplying a meaningful summary of the logged data as an adequate evidence for each verification result.

Recommendations and rules:

> **[SWE.4.RL.3]** If the verification logs of static verification do not cover the aspect above, the indicator BP3 must not be rated F.

> **[SWE.4.RL.4]** If the verification logs of unit test do not cover the aspect above, the indicator BP4 must not be rated F.

> **[SWE.4.RL.5]** If the verification results of static verification contain only a pure passed/failed information without a supporting verification log, the indicator BP3 must not be rated higher than P.

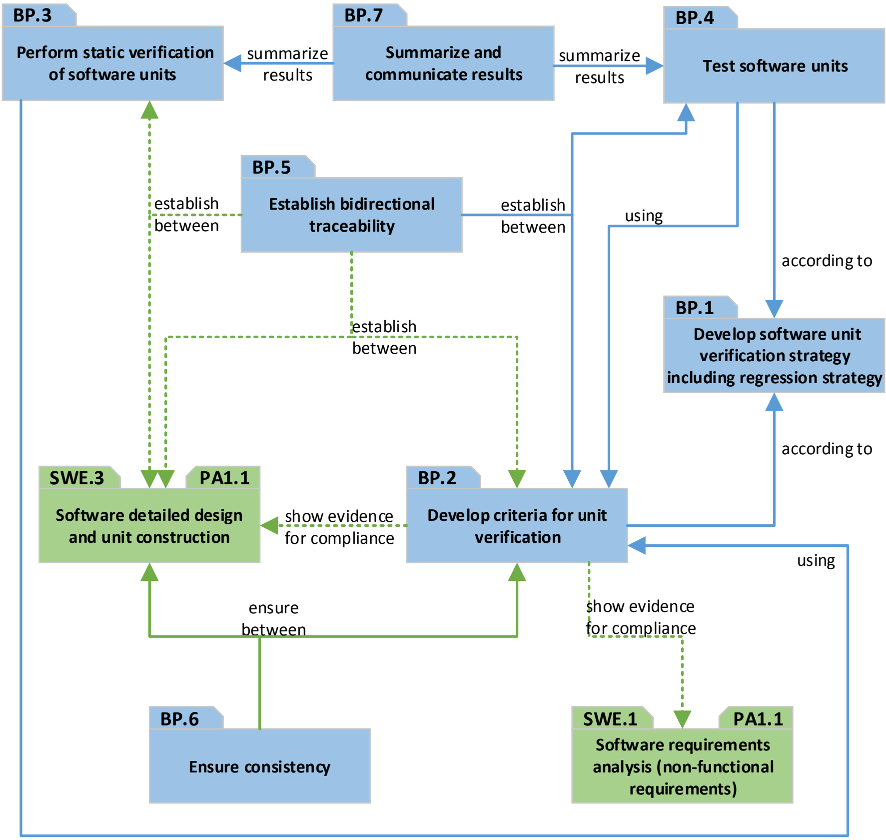
> **[SWE.4.RL.6]** If the verification results of unit test contain only a pure passed/failed information without a supporting verification log, the indicator BP4 must not be rated higher than P.

Related to:

- BP3 “Perform static verification of software units”
- BP4 “Test software units”
- Output WP 13-25 “Verification result”
- Output WP 13-50 “Test result”

3.9.2 Rating consistency

The following figure shows the relationships between SWE.4 base practices as well as their relationships to other processes:



These relationships are used as the basis for the rating rules and recommendations defined in the following subchapters.

Generic aspects regarding traceability and consistency (2.1.1), summarize and communication (2.1.2), and strategy and plan (2.1.4) shall also be considered for rating.

3.9.2.1 Rating consistency within SWE.4

The following rating rule is related to the software unit verification strategy and thus covers several base practices of the process:

> **[SWE.4.RL.7]** If the strategy-related activities are not performed according to the defined strategy (BP1), the indicators BP2 and BP4, respectively, shall be downrated.

Within SWE.4, the following base practices have relationships to each other:

BP3 Perform static verification of software units

> **[SWE.4.RL.8]** If developing criteria for unit verification (BP2) is downrated due to missing or inadequate definitions of criteria for static verification of software units, the indicator BP3 shall be downrated.

BP4 Test software units

> **[SWE.4.RL.9]** If the indicator for developing criteria for unit verification (BP2) is downrated due to missing or inadequate definitions of criteria for unit test specification, the indicator for testing software unitsBP4 shall be downrated.

BP7 Summarize and communicate results

> **[SWE.4.RL.10]** If the indicator for performing static verification of software units (BP3) is rated P or N, the indicator for summarizing and communicating the results BP7 shall be downrated.

> **[SWE.4.RL.11]** If the indicator for testing software units (BP4) is rated P or N, the indicator for summarizing and communicating the results BP7 shall be downrated.

3.9.2.2 Rating consistency to other processes at level 1

The following base practices of SWE.4 have relationships to other processes.

BP2 Develop criteria for unit verification

[SWE.4.RC.1] If the PA 1.1 for SWE.1 is downrated due to missing or inadequate non-functional requirements, this should be in line with the rating of the indicator BP2.

[SWE.4.RC.2] If the PA 1.1 for SWE.3 is downrated due to missing or inadequate detailed design, this should be in line with the rating of the indicator BP2.

BP5 Establish bidirectional traceability

[SWE.4.RC.3] If PA 1.1 for SWE.3 is downrated, this should be in line with the rating of the indicator BP5.

BP6 Ensure consistency

[SWE.4.RC.4] If PA 1.1 for SWE.3 is downrated, this should be in line with the rating of the indicator BP6.

3.10 SWE.5 Software Integration and Integration Test

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.

3.10.1 Rating recommendations

3.10.1.1 Integration Strategy

In general, the planning and extent of all integration activities is achieved by following a documented integration strategy.

The expectations for an integration strategy cover these aspects:

- A definition of the intended approach for the integration of software elements (bottom-up, top-down, according to availability, according to criticality level, items on a critical path etc.) which leads to:
  - A definition of the integration steps and their sequence in relation to the project plan and release plan.
  - A definition of which items, defined in the software architecture, need to be ready for the defined integration steps.
- A definition of how the level of complexity regarding product and organization is handled (e.g. Multi-site development, technical complexity of the software).
- A definition of the preconditions for software items to be ready for integration (e.g. predefined test steps or quality criteria).

Recommendations and rules:

[SWE.5.RL.1] If the integration strategy does not cover all aspects above, the indicator BP1 must not be rated F.

Related to:

- BP1 “Develop software integration strategy”
- Output WP 08-52 “Test plan”

3.10.1.2 Test strategy

In general, all testing activities follow a test strategy documented in a test plan.

The expectations for a test strategy cover these aspects:

a) A definition of the test scope
b) A definition how specific requirements regarding testing (e.g. test-specific stakeholder requirements, ISO 26262) are covered.
c) A definition of the methods for test case and test data development (e.g. development of positive / negative tests, test of static and dynamic behavior, equivalence partitioning).
d) A definition of the criteria to select test cases including

   - the coverage of new or changed requirements
   - the coverage of changes in the architecture or interface specifications
   - the coverage of change requests
   - the coverage of item changes
   - the consideration of dependencies, based on the analysis of changes (e.g. causal chain analysis) and
   - the selection of appropriate test cases for regression testing including a set of test cases selected as a basis set to be executed.

e) A definition of the test environment regarding each test method
f) The assignment of test methods to project phases (e.g. stress test, smoke test and fault injection test).
g) A definition of the test coverage in relation to the project plan and release plan.
h) A definition of entry and exit criteria for the test
i) A documentation of sufficient test coverage of each test level, if the test levels (e.g. software qualification test, software integration test and unit test) are combined
j) An approach for the handling of failed tests

   Note: This aspect of the test strategy should refer to SUP.9 Problem Resolution Management strategy

Recommendations and rules:

> **[SWE.5.RL.2]** If the test strategy does not cover all aspects above, the indicator BP2 must not be rated F.
>
> **[SWE.5.RL.3]** If the test strategy does not cover aspect b), c) or d), the indicator BP2 must not be rated higher than P.

Related to:

- BP2 "Develop software integration test strategy including regression test strategy"
- Output WP 08-52 "Test plan"

3.10.1.3 Develop specification for integration test

The test specification has to be developed according to the strategy. For details see chapter 3.10.1.2, *"Test strategy"*, aspect b), c) and e).

Recommendations and rules:

> **[SWE.5.RL.4]** If the test specifications are not based on the architecture and interface specifications, the indicator BP3 must not be rated higher than P.

Related to:

- BP3 "Develop specification for software integration test"
- Output WP 08-50 "Test specification"

3.10.1.4 Select test cases

Test cases are selected from the integration test specification.

The expectations for a successful implementation of the test case selection cover the following aspects:

a) The selection of the test cases has to be performed according to the defined strategy
b) The selection of test cases has to consider the intended use of the deliverable item (test bench, test track, use on public road, …)
c) The selection criteria used (as defined in the strategy) have to be documented
d) The selection of the test cases has to be documented

Recommendations and rules:

> **[SWE.5.RL.5]** If the test case selection does not cover the aspects a) and b), the indicator BP5 must not be rated F.
>
> **[SWE.5.RC.1]** If the test case selection does not cover the aspects c) and d), the indicator BP5 should not be rated F.

Related to:

- BP5 “Select test cases”
- Output WP 08-50 “Test specification”

3.10.1.5 Test implementation using automation

On the different levels of testing the execution of the test cases shall follow a test plan including the test strategy. According to the test cases specified this can be done by manual testing or by an automated approach using test scripts processed by a test automation tool or specific programmed test routines.

The expectations for a successful implementation cover these aspects:

- Completeness of test scripts and programs with respect to the test cases assigned to an automated test in the test specification
- Consistency of test scripts and programs with respect to each test case.

Recommendations and rules:

> **[SWE.5.RL.6]** If the test implementation is not complete in terms of all aspects above, the indicator BP6 must not be rated F.

Related to:

- BP6 “Perform software integration test”
- Output WP 13-50 “Test result”

3.10.1.6 Test logs as evidence for test results

By testing the system and the software on different levels a large amount of logged data may be generated, which have to be documented in test logs. This is especially true for automated tests. Also in tests performed manually the results may be provided in different levels of detail.

The expectations for a successful implementation cover this aspect:

- Test logs supplying a meaningful summary of the logged data as an adequate evidence for each test result.

Recommendations and rules:

[SWE.5.RL.7] If the test logs do not cover the aspect above, the indicator BP6 must not be rated F.

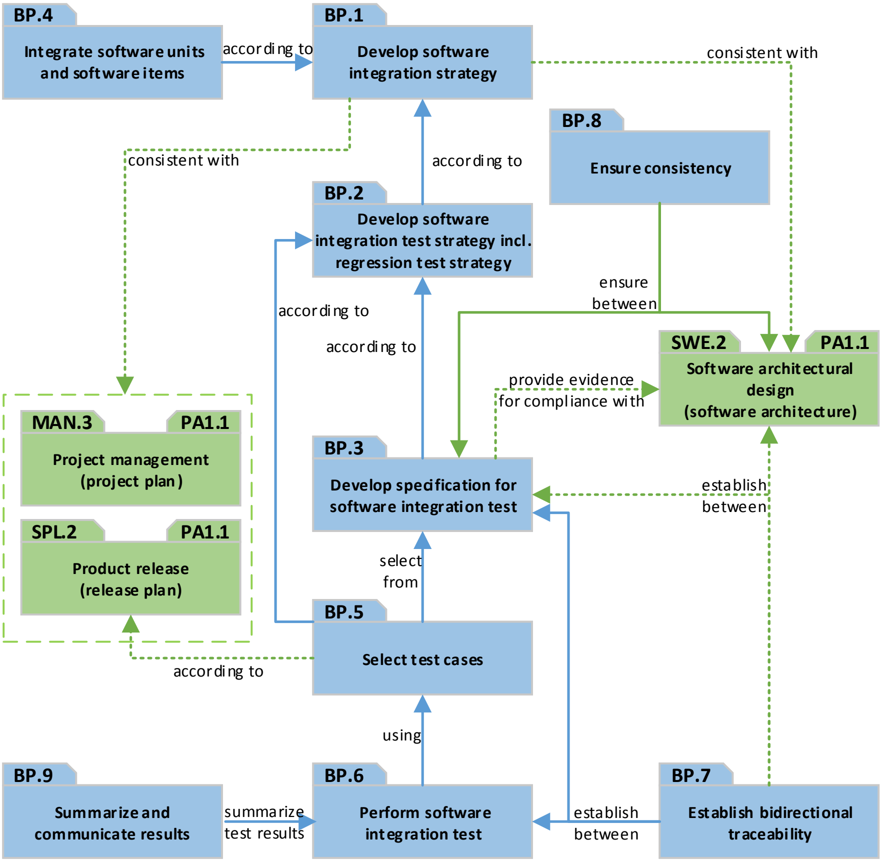
[SWE.5.RL.8] If the test results contain only a pure passed/failed information without a supporting test log, the indicator BP6 must not be rated higher than P

Related to:

- BP6 “Perform software integration test”
- Output WP 13-50 “Test result”

3.10.2 Rating consistency

The following figure shows relationships between SWE.5 base practices as well as their relationships to other processes:



These relationships are used as the basis for the rating rules and recommendations defined in the following subchapters.

Generic aspects regarding traceability and consistency (2.1.1), summarize and communication (2.1.2), and strategy and plan (2.1.4) shall also be considered for rating.

3.10.2.1 Rating consistency within SWE.5

The following rating rule is related to the software integration test strategy and thus covers several base practices of the process:

> **[SWE.5.RL.9]** If the strategy-related activities are not performed according to the defined strategy (BP2), the indicators BP3 and BP5, respectively, shall be downrated.

Within SWE.5, the following base practices have relationships to each other:

BP2 Develop software integration test strategy including regression test strategy.

> **[SWE.5.RL.10]** If the test strategy is not developed according to the defined integration strategy (BP1), the indicator BP2 shall be downrated.

BP3 Develop specification for software integration test

> **[SWE.5.RL.11]** If the indicator for developing the test strategy (BP2) is downrated due to missing or inadequate definitions of methods for test case and test data development, the indicator for development of the test specification BP3 shall be downrated.

BP4 Integrate software items

> **[SWE.5.RL.12]** If the strategy-related activities are not performed according to the defined strategy (BP1), the indicator BP4 shall be downrated.

BP5 Select test cases

> **[SWE.5.RL.13]** If the indicator for developing the test specification (BP3) is downrated, the indicator BP5 must not be rated higher.

> **[SWE.5.RL.14]** If the indicator for developing the test strategy (BP2) is downrated due to a missing or inadequate definition of the test case selection criteria, the indicator select test cases (BP5) shall be downrated.

BP6 Test integrated software

> **[SWE.5.RL.15]** If the indicator for selecting test cases (BP5) is rated P or N, the indicator (BP6) shall be downrated.

3.10.2.2 Rating Consistency to other processes at level 1

The following base practices of SWE.5 have relationships to other processes.

BP1 Develop software integration strategy

[SWE.5.RC.2] If the project plan or release plan are not adequate, this should not be used to downrate the indicator BP1.

[SWE.5.RC.3] If the PA 1.1 for SWE2 is downrated, this should be in line with the rating of the indicator BP1.

BP3 Develop specification for software integration test

[SWE.5.RC.4] If the PA 1.1 for SWE2 is downrated, this should be in line with the rating of the indicator BP3.

BP5 Select test cases

[SWE.5.RC.5] If only the release plan is not adequate, but the test cases are selected according to the strategy, this should not be used to downrate the indicator BP5.

BP7 Establish bidirectional traceability

[SWE.5.RC.6] If PA 1.1 for SWE.2 is downrated, this should be in line with the rating of the indicator BP7.

BP8 Ensure consistency

[SWE.5.RC.7] If PA 1.1 for SWE.2 is downrated, this should be in line with the rating of the indicator BP8.

3.11 SWE.6 Software Qualification Test

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.

3.11.1 Rating recommendations

3.11.1.1 Test strategy

In general, all testing activities follow a test strategy documented in a test plan.

The expectations for a test strategy cover these aspects:

a) A definition of the test scope
b) A definition of how specific requirements regarding testing (e.g. test-specific stakeholder requirements, ISO 26262) are covered.
c) A definition of the methods for test case and test data development (e.g. development of positive/negative tests, equivalence partitioning).
d) A definition of the criteria to select test cases including

- the coverage of new or changed requirements
- the coverage of change requests
- the coverage of item changes
- the consideration of dependencies, based on the analysis of changes (e.g. causal chain analysis) and
- the selection of appropriate test cases for regression testing including a set of test cases selected as a basis set to be executed.

e) A definition of the test environment regarding each test method
f) The assignment of test methods to project phases (e.g. stress test, smoke test and fault injection test).
g) A definition of the test coverage in relation to the project plan and release plan.
h) A definition of entry and exit criteria for the test
i) A documentation of sufficient test coverage of each test level, if the test levels (e.g. software qualification test, software integration test and unit test) are combined
j) An approach for the handling of failed tests

Note: This aspect of the test strategy should refer to SUP.9 Problem Resolution Management strategy)

Recommendations and rules:

[SWE.6.RL.1] If the test strategy does not cover all aspects above, the indicator BP1 must not be rated F.

[SWE.6.RL.2] If the test strategy does not cover aspect b), c) or d), the indicator BP1 must not be rated higher than P.

Related to:

- BP1 “Develop software qualification test strategy including regression test strategy”
- Output WP 08-52 “Test plan”

3.11.1.2 Develop specification for qualification test.

The test specification has to be developed according to the strategy. For details see chapter 3.11.1.1, *“Test strategy”*, aspect b), c and e).

Recommendations and rules:

[SWE.6.RL.3] If the test specifications are not based on the requirement specifications and the verification criteria, the indicator BP2 must not be rated higher than P.

Related to:

- BP2 “Develop specification for software qualification test“
- Output WP 08-50 “Test specification“

3.11.1.3 Select test cases

Test cases are selected from the qualification test specification.

The expectations for a successful implementation of the test case selection cover the following aspects:

a) The selection of the test cases has to be performed according to the defined strategy
b) The selection of test cases has to consider the intended use of the deliverable item (test bench, test track, use on public road, …)
c) The selection criteria used (as defined in the strategy) have to be documented
d) The selection of the test cases has to be documented

Recommendations and rules:

> **[SWE.6.RL.4]** If the test case selection does not cover the aspects a) and b), the indicator BP3 must not be rated F.
>
> **[SWE.6.RC.1]** If the test case selection does not cover the aspects c) and d), the indicator BP3 should not be rated F.

Related to:

- BP3 "Select test cases"
- Output WP 08-50 "Test specification"

3.11.1.4 Test implementation using automation

On the different levels of testing the execution of the test cases shall follow a test plan including the test strategy. According to the test cases specified this can be done by manual testing or by an automated approach using test scripts processed by a test automation tool or specific programmed test routines.

The expectations for a successful implementation cover these aspects:

- Completeness of test scripts and programs with respect to the test cases assigned to an automated test in the test specification
- Consistency of test scripts and programs with respect to each test case.

Recommendations and rules:

> **[SWE.6.RL.5]** If the test implementation is not complete in terms of all aspects above, the indicator BP4 must not be rated F.

Related to:

- BP4 "Test integrated software"
- Output WP 13-50 "Test result"

3.11.1.5 Test logs as evidence for test results

By testing the system and the software on different levels a large amount of logged data may be generated, which have to be documented in test logs. This is especially true for automated tests. Also in tests performed manually the results may be provided in different levels of detail.

The expectations for a successful implementation cover this aspect:

- Test logs supplying a meaningful summary of the logged data as an adequate evidence for each test result.

Recommendations and rules:

[SWE.6.RL.6] If the test logs do not cover the aspect above, the indicator BP4 must not be rated F.

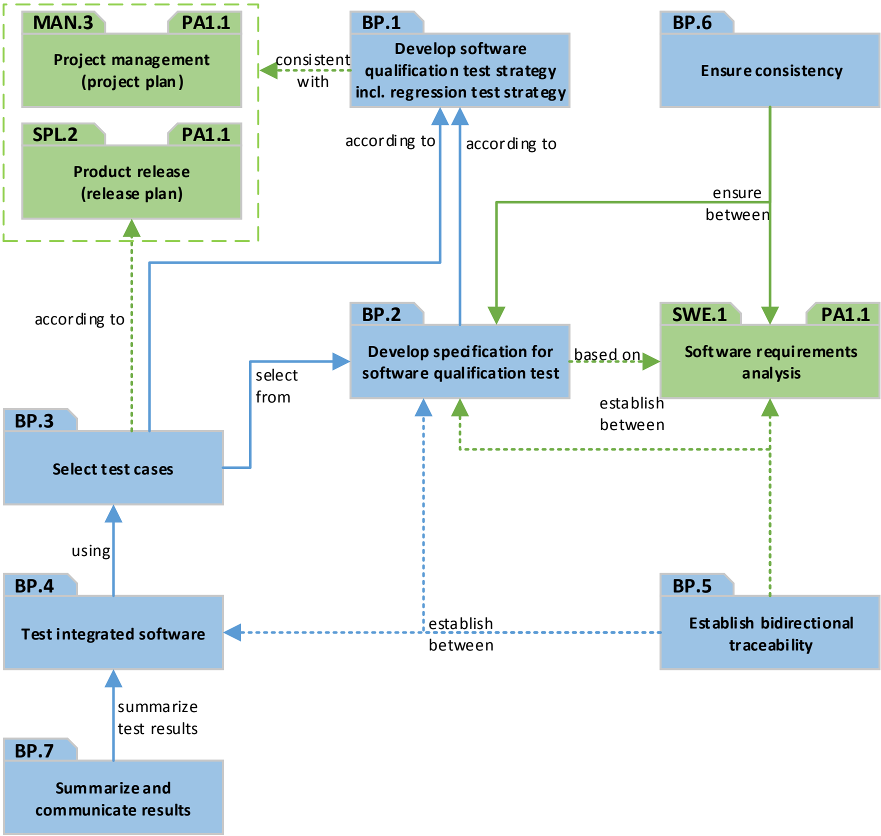
[SWE.6.RL.7] If the test results contain only a pure passed/failed information without a supporting test log, the indicator BP4 must not be rated higher than P.

Related to:

- BP4 “Test integrated software”
- Output WP 13-50 “Test result”

3.11.2 Rating consistency

The following figure shows relationships between SWE.6 base practices as well as their relationships to other processes:



These relationships are used as the basis for the rating rules and recommendations defined in the following subchapters.

Generic aspects regarding traceability and consistency (2.1.1), summarize and communication (2.1.2), and strategy and plan (2.1.4) shall also be considered for rating.

3.11.2.1 Rating consistency within SWE.6

The following rating rule is related to the software qualification test strategy and thus covers several base practices of the process:

> **[SWE.6.RL.8]** If the strategy-related activities are not performed according to the defined strategy (BP1), the indicators BP2 and BP3, respectively, shall be downrated.

Within SWE.6, the following base practices have relationships to each other:

BP2 Develop specification for software qualification test

> **[SWE.6.RL.9]** If the indicator for developing the test strategy (BP1) is downrated due to missing or inadequate definitions of methods for test case and test data development, the indicator for development of the test specification BP2 shall be downrated.

BP3 Select test cases

> **[SWE.6.RL.10]** If the indicator for developing the test specification (BP2) is downrated, the indicator BP3 must not be rated higher.

> **[SWE.6.RL.11]** If the indicator for developing the test strategy (BP1) is downrated due to a missing or inadequate definition of the test case selection criteria, the indicator select test cases BP3 shall be downrated.

BP4 Test integrated software

> **[SWE.6.RL.12]** If the indicator for selecting test cases (BP3) is rated P or N, the indicator BP4shall be downrated.

3.11.2.2 Rating consistency to other processes at level 1

The following base practices of SWE.6 have relationships to other processes.

BP1 Develop software qualification test strategy including regression test strategy

> **[SWE.6.RC.2]** If the project plan or release plan are not adequate, this should not be used to downrate the indicator BP1.

BP2 Develop specification for software qualification test

[SWE.6.RC.3] If the PA 1.1 for SWE.1 is downrated, this should be in line with the rating of the indicator BP2.

BP3 Select test cases

[SWE.6.RC.4] If only the release plan is not adequate, but the test cases are selected according to the strategy, this should not be used to downrate the indicator BP3.

BP5 Establish bidirectional traceability

[SWE.6.RC.5] If PA 1.1 for SWE.1 is downrated, this should be in line with the rating of the indicator BP5.

BP6 Ensure consistency

[SWE.6.RC.6] If PA 1.1 for SWE.1 is downrated, this should be in line with the rating of the indicator BP6.

3.12 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.

3.12.1 Rating recommendations

3.12.1.1 Quality assurance strategy

As stated in the purpose, the predefined provisions have to be included in the quality assurance strategy. This may include

- Quality Management
- Customer Requirements
- Stakeholder Requirements
- Internal quality criteria
- Compliance with project-relevant quality standards

These provisions lead to project specific quality criteria which are documented in the quality strategy.

From the identified project-specific criteria, methods are derived which ensure the quality of all work products (i.e. not just software source code) and processes for the project. Measures should cover review methods, audits, assessments, lessons learned workshops, frequency, review coverage, and review participants for all relevant work products and processes. Definitions of review frequencies, review coverage, and review methods in the quality assurance strategy also support GP 2.2.1 of all the other processes.

The quality assurance strategy does not contain schedule or effort information as this is addressed at capability level 2.

> **[SUP.1.RL.1]** If predefined provisions are not considered in the quality strategy, the indicator BP1 must not be rated higher than P.

> **[SUP.1.RL.2]** If there are no quality criteria defined, the indicator BP1 must not be rated higher than P.

[SUP.1.RL.3] If the quality assurance strategy includes criteria for software source code only, the indicator BP1 must not be rated higher than P.

[SUP.1.RL.4] If review methods, review criteria, review frequency, review coverage, or involvement of relevant parties are not part of the quality assurance strategy, or not documented in the review evidence, the indicator BP1 shall be downrated.

Note: An organizational quality management strategy, according to e.g. ISO 900x or ISO 16949, must not be mistaken for a standard quality assurance strategy in a SUP.1 context. Quality management addresses customer satisfaction resulting from a company's business processes; in contrast, Automotive SPICE addresses the product development sub-context only, but provides much more details. Furthermore, quality assurance refers to a timely guarantee that criteria for work products and processes are met.
Note: The quality assurance strategy may be documented in a quality assurance plan or can be included in any other appropriate document.

Related to:

- SUP.1.BP1 “Develop a project quality assurance strategy”
- Output WP 08-13 “Quality plan”
- Output WP 18-07 “Quality criteria”

3.12.1.2 Independence and objectivity

Objectivity and independence of quality assurance can be reached using different approaches:

- Individuals from a different project or team, department or business area.

  Note: In the case of e.g. small organizations with people having close relations to each other organizational independency may not be sufficiently effective. Furthermore, the more organizationally independent the reviewer is, the less competent in the subject matter he probably is.

- External services

  Note: External reviewers might have less knowledge about the subject matter under review. Furthermore, external contractors are not necessarily fully independent as they might strive for follow-up contracts.

- Internal heterogeneous team. A mix of internal representatives of different teams, departments or business areas.

- External heterogeneous team. A mix of external parties and internal representatives of different teams or departments.

Quality assurance also has to cover the quality of supplier deliveries.

> **[SUP.1.RL.5]** If quality assurance strategy does not cover the quality assurance of supplier deliveries, the indicator BP1 must not be rated higher than L.

> **[SUP.1.RC.1]** If the approach for guaranteeing objectivity is in conflict with subject matter competence, the indicator BP1 should not be rated higher than P.

> **[SUP.1.RC.2]** If Quality Assurance is not organized in terms of distinct organizational departments or separate independent persons, the indicator BP1 should not be downrated.

> *Rationale: QA may be organized, or realized, by means of project-independent parties.*

Related to:

- SUP.1.BP1 “Develop a project quality assurance strategy”
- Output WP 08-13 “Quality plan”
- Output WP 18-07 “Quality criteria”

3.12.1.3 Assure quality of work products and processes

Work products have to be reviewed based on predefined review methods, review criteria, review frequency, review coverage, and all relevant review participants. All review participants have to be identified which have an interest in the work product (e.g. testers have to review the requirements).

Process quality assurance may include process assessments and audits, problem analysis, regular check of methods, tools, documents and the adherence to defined processes, reports and lessons learned that improve processes for future projects.

> **[SUP.1.RL.6]** If process quality assurance is based on performing process assessments (either by a customer or internally) only, the indicator BP3 must not be rated higher than P.

[SUP.1.RL.7] If work product quality assurance is done based on checking for pure work product existence only, the indicator BP2 must not be rated higher than P.

Related to:

- SUP.1.BP2 “Assure quality of work products”
- SUP.1.BP3 “Assure quality of process activities”
- Output WP 13-18 “Quality record”
- Output WP 13-19 “Review record”
- Output WP 13-07 “Problem record”

3.12.1.4 Escalation

Based on the established independence (see chapter 3.12.1.2) an escalation mechanism has to be established. The mechanism should cover all relevant stakeholders (e.g. technical and quality management, management, customer, suppliers). After escalations, these stakeholders shall drive corrective actions.

[SUP.1.RL.8] If escalations are not followed up by corrective actions, the indicator BP6 must not be rated higher than P.

Related to:

- SUP.1.BP6 “Implement an escalation mechanism”
- Output WP 13-04 “Communication record”
- Output WP 14-02 “Corrective action register”
- Output WP 13-07 “Problem record”

3.12.1.5 Resolution of non-conformances

Non-conformances identified in reviews of work products and processes have to be resolved. The strategy should cover how non-conformances should be tracked and who is responsible for such tracking.

[SUP.1.RL.9] If non-conformances are not tracked, not resolved in a timely manner, or not escalated, the indicator BP5 must not be rated higher than P.

Related to:

- SUP.1.BP5 “Ensure resolution of non-conformances”
- Output WP 13-04 “Communication record”
- Output WP 14-02 “Corrective action register”
- Output WP 13-07 “Problem record”

3.12.1.6 Non-conformances not found in review

Work products have to be reviewed based on predefined review methods, review frequency, review coverage, and by all relevant review participants. However, sometimes during assessments it becomes apparent that a defect was not detected, or documented, even though quality assurance has taken place.

Example: the review report on the software architectural design showed no findings even though a dynamic design was missing.

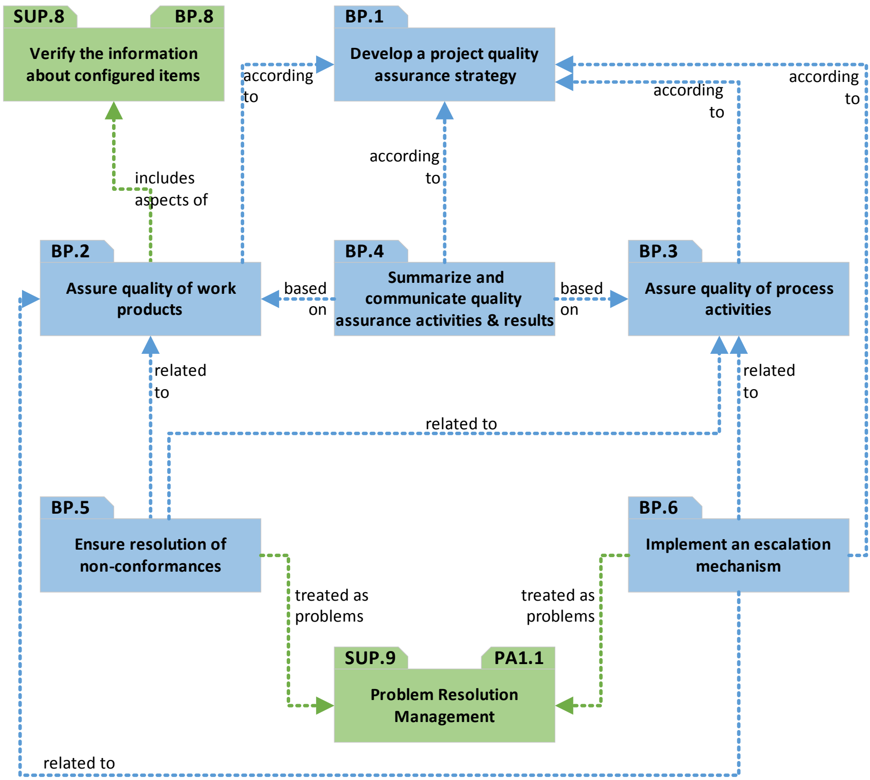
> **[SUP.1.RC.3]** If work product or process non-conformances are not identified or documented even if the defined quality assurance methods were applied, the indicators BP2 or BP3, respectively, should be downrated.

Related to:

- SUP.1.BP2 “Assure quality of work products”
- SUP.1.BP3 “Assure quality of process activities”
- Output WP 13-18 “Quality record”
- Output WP 13-19 “Review record”
- Output WP 13-07 “Problem record”

3.12.2 Rating consistency

The following figure shows the relationships between SUP.1 base practices as well as their relationships to other processes:



These relationships are used as the basis for the rating rules and recommendations defined in the following subchapters.

Generic aspects regarding summarize and communication (2.1.2), and strategy and plan (2.1.4) shall also be considered for rating.

3.12.2.1 Rating consistency within SUP.1

Within SUP.1, the following base practices have relationships to each other:

> **[SUP.1.RC.4]** If the quality of work products (BP2) is downrated, the indicators BP4, BP5, and BP6, respectively, should be downrated.

[SUP.1.RL.10] If the quality of work products (BP2) is rated N or P, PA 1.1 must not be rated higher than L.

[SUP.1.RC.5] If the quality of process activities (BP3) is downrated, the indicators BP4, BP5, and BP6, respectively, should be downrated.

[SUP.1.RL.11] If the quality of process activities (BP3) is rated N or P, PA 1.1 must not be rated higher than L.

BP2 Assure quality of work products

[SUP.1.RC.6] If the strategy (BP1) is downrated because of a missing verification approach for work products, or because of missing reporting methods, or because of an inappropriate escalation mechanism, or because of an inadequate objectivity and independence approach, the indicator BP2 should be downrated.

BP3 Assure quality of process activities

[SUP.1.RC.7] If the strategy (BP1) is downrated because of a missing verification approach for processes, or because of missing reporting methods, or because of an inappropriate escalation mechanism, or because of an inadequate objectivity and independence approach, the indicator BP3 should be downrated.

3.12.2.2 Rating consistency to other processes at level 1

The following base practices of SUP.1 have relationships to other processes:

> **[SUP.1.RC.8]** If quality non-conformances are to be treated as problems according to the problem resolution strategy, the indicator BP5 and BP6, respectively, this should be in line with the rating of PA 1.1 of process SUP.9.

BP2 Assure quality of work products

> **[SUP.1.RC.9]** If the indicator verifies the information about configured items (SUP.8.BP8) is downrated because of missing or inadequate activities such as baseline audits, baseline reproduction checks, or check-in comments of configuration items, this should be in line with the rating of the indicator BP2.

3.13 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 configuration management process covers the identification of configuration items, i.e. inputs and work products of all relevant processes (e.g., project plan, quality assurance status report, requirement specification, test cases, source code, tools). It does not cover the actual planning and versioning of those configuration items. This is allocated at level 2 of the processes to which the configuration items belong (for details refer to GP 2.2.2 / GP 2.2.3 and chapter 5.2 *"Interpretation on process capability, PA 2.2"*).

Furthermore, the configuration management process covers the definition, creation and verification of baselines. This includes the definition of events that trigger a baseline. It does not cover the actual planning of baselines (e.g., the actual date for creation of a baseline). This is allocated to MAN.3 on level 1.

Product releases are defined in the process Product Release (SPL.2). Therefore, the configuration management strategy defined in SUP.8 has to consider the requirements regarding the definition of releases according to SPL.2. The baselines created according to SUP.8 are used for deliveries as defined in SPL.2.

3.13.1 Rating recommendations

3.13.1.1 Strategy

Generic aspects, rules and recommendations regarding the strategy are given in chapter 2.1.4 and shall also be considered for rating of SUP.8.

The expectations for a successful strategy cover these aspects:

a) All organizational and/or project-specific aspects like disciplines (e.g., system, software, and electronics), sites, and processes (including engineering processes, management processes, and supporting processes) are included.
b) An overall strategy is developed, especially if different solutions are defined for different disciplines, sites, or processes.

c) The definition of access rights.
d) The definition of required activities and tools, (e.g., infrastructure, resources like file shares, repositories, or dedicated configuration management systems) in accordance to the complexity of the product to be developed.
e) The criteria for the identification of configuration items, including naming convention (for e.g., items, folder structures). Examples for criteria are categories such as documents, requirements, source code, development tools, third-party software.
f) The conditions to create a revision of a configuration item.
g) The definition of the approach for the creation of baselines, including the event that creates the baseline (required or optional), the procedures used to establish the baseline, their naming convention, and their relationship to revisions of items.
h) The definition for handling of variants, creation and merging of branches for items and sets of items (e.g., requirements for variants). This includes in which cases branching is permissible, whether authorization is required, and how branches are merged.
i) The revision history approach of for configuration items.

Recommendations and rules:

[SUP.8.RL.1] If the strategy does not include all aspects above, the indicator BP1 must not be rated F.

[SUP.8.RL.2] If there is no dedicated configuration management system defined in the strategy but the procedure is adequate for the complexity of the product to be developed this must not be used to downrate the indicator BP1.

[SUP.8.RL.3] If major configuration management aspects (according to d) or e)) are missing in the strategy the indicator BP1 must not be rated higher than P.

[SUP.8.RL.4] If major baselining aspects (according to g)) are missing in the strategy the indicator BP1 must not be rated higher than P.

[SUP.8.RL.5] If major branching and merging aspects (according to h)) are missing in the strategy the indicator BP1 must not be rated higher than P.

[SUP.8.RC.1] If there is only an adequate generic strategy but no project specific implementation, the indicator BP1 should not be downrated.

Related to:

- BP1 “Develop a configuration management strategy”
- Output WP 08-04 “Configuration management plan”
- Output WP 08-14 “Recovery plan”
- Output WP 16-03 “Configuration management system”

3.13.1.2 Baselines

The expectations for establishing baselines cover these aspects:

a) Definition of the items that are to be controlled in which kind of baseline.
b) Internal and external baselines are created for all events as defined in the strategy (required or optional).
c) Overall baselines are created over different disciplines, sites, processes etc. and have to be consistent.
d) The baselines contain complete and consistent sets of items necessary to reproduce the work products.
e) The baselines are created according to the naming convention defined in the strategy.

Recommendations and rules:

[SUP.8.RL.6] If it is not defined for each kind of baseline which configuration items are to be controlled, the indicator BP6must not be rated higher than P.

[SUP.8.RL.6] If required baselines do not exist for events defined in the strategy, the indicator BP6 shall be downrated.

[SUP.8.RL.7] If established baselines for different disciplines, sites, processes etc. (according to c) are not consistent or if overall baselines do not exist, the indicator BP6 shall be downrated.

[SUP.8.RL.8] If content of a baseline is not verified (by e.g., a baseline or configuration management audit), the indicator BP8 shall be downrated.

[SUP.8.RC.2] If the defined naming convention for baselines is not used, the indicator BP6 should be downrated.

Related to:

- BP6 “Establish baselines”
- BP8 “Verify the information about configured items”
- Output WP 01-00 “Configuration item”
- Output WP 08-04 “Configuration management plan”
- Output WP 13-08 “Baseline”
- Output WP 13-10 “Configuration management record”

3.13.1.3 Branching and merging

The expectations for branching and merging activities cover these aspects:

- Branches for configuration items and sets of configuration items are created according to the strategy (i.e., they are only created where required).
- Consistency and completeness has to be ensured for merged configuration items and sets of configuration items.

Recommendations and rules:

[SUP.8.RL.9] If branches are not created according to the strategy, the indicator BP4 shall be downrated.

[SUP.8.RL.10] If consistency and completeness of merged items or sets of items is not ensured, the indicator BP8 must not be rated F.

Related to:

- BP4 “Establish branch management”
- BP8 “Verify the information about configured items”
- Outcome WP 01-00 “Configuration item”
- Outcome WP 08-04 “Configuration management plan”
- Outcome WP 13-08 “Baseline”
- Outcome WP 16-03 “Configuration management system”

3.13.1.4 Configuration management infrastructure

The expectations for a configuration management infrastructure cover these aspects:

a) The established infrastructure is able to support the procedures defined in the strategy, including access rights.
b) The established infrastructure provides the resources needed to support the defined complexity, including aspects like multisite operation, size of projects, multi-project or multi-variant application.
c) The properties of used IT services (e.g., file shares, tools) regarding storage, archiving (long-term storage), and backup are known and compared with the project requirements. Deviations are known and corrective actions are established.

Recommendations and rules:

[SUP.8.RL.11] If the established infrastructure is not able to support the procedures (according to a)) or the complexity (according to b)), the indicator BP3 shall be downrated.

[SUP.8.RL.12] If there is no dedicated configuration management system in place but the established procedure is adequate for the complexity of the product to be developed this must not be used to downrate the indicator BP3.

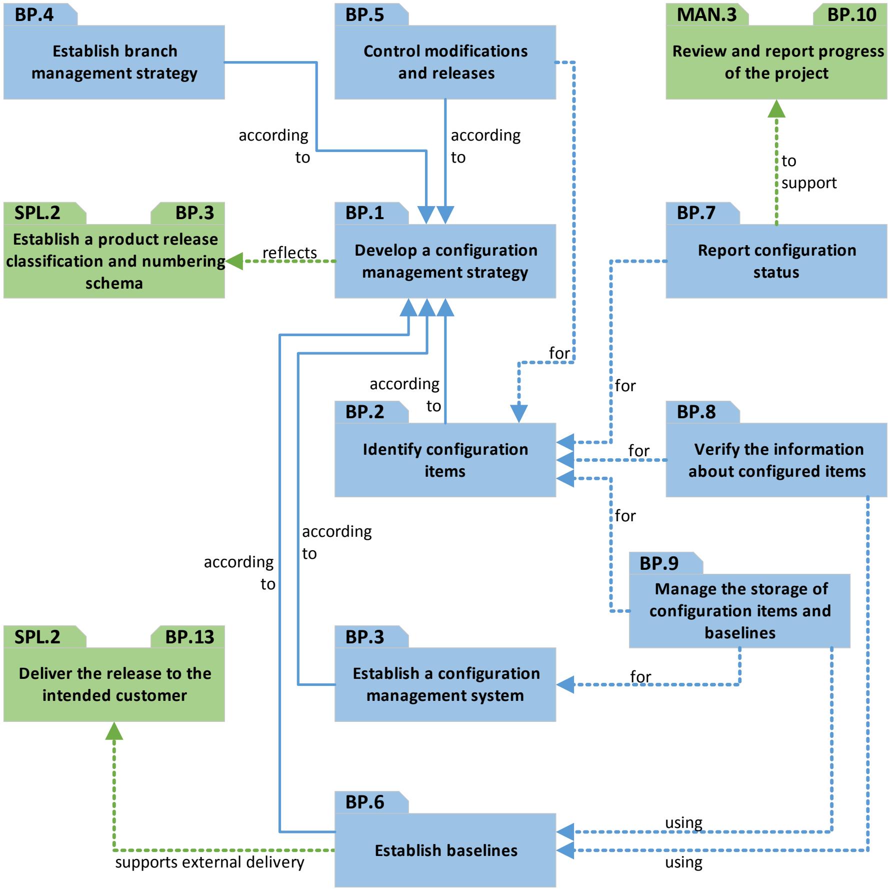
[SUP.8.RL.13] If properties of used IT services are not known, or known but in case of deviations from project requirements no corrective actions are established, the indicator BP9 shall be downrated.

Related to:

- BP3 “Establish a configuration management system”
- BP9 “Manage the storage of configuration items and baselines”
- Outcome WP 01-00 “Configuration item”
- Outcome WP 06-02 “Handling and storage guide”
- Outcome WP 08-04 “Configuration management plan”
- Outcome WP 08-14 “Recovery plan”
- Outcome WP 13-08 “Baseline”
- Outcome WP 13-10 “Configuration management record”
- Outcome WP 14-01 “Change history”
- Outcome WP 16-03 “Configuration management system”

3.13.2 Rating consistency

The following figure shows the relationships between SUP.8 base practices as well as their explicit relationships to other processes:



These relationships are used as the basis for the rating rules and recommendations defined in the following subchapters.

Generic aspects regarding strategy and plan (2.1.4) shall also be considered for rating.

3.13.2.1 Rating consistency within SUP.8

The following rating rule is related to the configuration management strategy (BP1) and thus covers several base practices of the process:

> **[SUP.8.RL.14]** If the strategy-related activities are not performed according to the defined strategy (BP1), the indicators BP2, BP3, BP4, BP5, and BP6 shall be downrated, respectively.

Furthermore, the proper identification of configuration items (BP2) is the basis for several base practices of the process, which leads to the following recommendation.

> **[SUP.8.RC.3]** If the identification of configuration items is not properly done (BP2), the indicators BP5, BP7, BP8, and BP9 should be downrated, respectively.

BP8 Verify the information about configured items

> **[SUP.8.RC.4]** If establishing baselines (BP6) is downrated, the indicator BP8 should be downrated.
> *Rationale: If baselines are not established their consistency cannot be verified.*

BP9 Manage the storage of configuration items and baselines

> **[SUP.8.RC.5]** If establishing a configuration management system (BP3) is downrated, the indicator BP9 should be downrated.

> **[SUP.8.RC.6]** If establishing baselines (BP6) is downrated, the indicator BP9 should be downrated.

3.13.2.2 Rating consistency to other processes at level 1

The following base practices of SUP.8 have relationships to other processes:

BP1 Develop a configuration management strategy

[SUP.8.RC.7] If the indicator BP1 is downrated due to improper naming conventions for baselining, this should be in line with the rating of establishing a product release classification and numbering scheme (SPL.2.BP3).

BP6 Establish baselines

[SUP.8.RC.8] If the indicator BP6 is downrated, this should be in line with the rating of delivering releases to the customer (SPL.2.BP13).

BP7 Report configuration status

[SUP.8.RC.9] If the indicator BP7 is downrated, this should be in line with the rating of reviewing and reporting progress of the project (MAN.3.BP10).

Rationale: If the reported status of configuration items does not reflect the actual status, the reporting of project progress is done on a wrong basis.

3.14 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 Problem Resolution Management Process covers the management of all issues where e.g., more than one stakeholder is involved, or which are not immediately resolved. It does not cover the implementation of necessary actions to solve the problem since this is done according to standard engineering processes.

A problem record may lead to the initiation of change requests (see also SUP.9.BP7), any issues in rating due to improper handling of these change requests have to be considered also when rating the Problem Management Process.

Furthermore, mapping of problem records to change requests, and corresponding baselines has to be ensured over all affected disciplines and all affected domains considering the project-specific complexity.

3.14.1 Rating recommendations

3.14.1.1 Strategy

Generic aspects regarding the strategy are given in chapter 2.1.4 and shall also be considered for rating of SUP.9.

The expectations for a successful strategy cover these aspects:

- A definition of which kind of issues shall be considered as problems and the project phases in which problem records have to be used (e.g., external tickets, failed tests – whether due to specification, implementation, or test environment errors –, review findings, missing resources). This may include different definitions for specific project phases (e.g., during prototype construction, series development, and after start of production)
- All organizational and/or project-specific aspects like affected disciplines (e.g., system, software, electronics), affected domains (e.g., software platform, COTS-Software), or affected sites are included.

- The handling of all relevant interfaces to the customer, the supplier and relevant internal stakeholders (e.g., purchasing, marketing, and management) is included.
- Management and exchange of problem reports across disciplines, domains, sites, stakeholders, and (sub)-projects are defined.
- A life cycle including a status model and workflow for problem records is defined. This life cycle has to cover any aspects and constraints mentioned above (different disciplines, sites, subprojects, stakeholders, etc.).
- A definition of problem categories related to cause and impact.
- A definition of an urgent resolution strategy, including criteria for application and criteria for alert notification.
- A methodology to ensure mapping of problems to change requests, and corresponding baselines where problems are solved.

Recommendations and rules:

[SUP.9.RL.1] If the strategy does not include all aspects above, the indicator BP1 must not be rated F.

[SUP.9.RL.2] If the strategy does not address interfaces between multisite organizations/projects, subprojects, and/or groups in case of correspondingly complex projects, the indicator BP1 must not be rated higher than P.

Related to:

- BP1 “Develop a problem resolution management strategy”
- Output WP 08-27 “Problem management plan”
- Output WP 13-07 “Problem record”

3.14.1.2 Cause determination and impact analysis

The expectations for an adequate cause determination and impact analysis of problem records cover these aspects of the defined strategy:

- The input from all relevant stakeholders (internal and external) is considered including technical aspects and potential side effects.
- The systematic consideration of similar problems in the same application (e.g., in software clones, variants).

- The systematic evaluation of potential effects of detected problems on other systems (e.g., use of base software components in different software projects).
- Identify work products which are affected by the problem.

Recommendations and rules:

[SUP.9.RL.3] If the impact analysis does not adequately address potential side effects due to insufficient involvement of relevant stakeholders, the indicator BP4 must not be rated F.

[SUP.9.RL.4] If the impact analysis is incomplete due to missing consideration of similar problems in the same application or potential effects on other systems, the indicator BP4 must not be rated F.

[SUP.9.RL.5] If affected work products are not identified by the impact analysis, the indicator BP4 must not be rated F.

[SUP.9.RL.6] If there is no evidence for required alert notifications due to missing consideration of potential effects on clones, variants or other systems, the indicator BP6 shall be downrated.

Related to:

- BP4 “Diagnose the cause and determine the impact of the problem”
- BP6 “Raise alert notifications”
- Output WP 13-07 “Problem record”
- Output WP 15-01 “Analysis report”
- Output WP 15-05 “Evaluation report”

3.14.1.3 Status model and workflow

The expectations for a status model and workflow cover these aspects:

- Status model and workflow are defined including criteria for status changes, and relevant stakeholders together with their responsibility and authorization.
- The actual way of working (e.g., possibility to switch back to previous states, use of iterations, etc.) is in line with the definition of the workflow.

- The status model and workflow are applied as defined. The problem record always shows the actual status (e.g., all problem records stated as closed in a release note shall also be in a final state according to the status model).
- The problem record follows the workflow and is tracked to a final status that shall be confirmed by an authorized role related to the initiator of the problem record (e.g., a problem record given by the customer should be confirmed by the customer). There might be more than one final status (e.g., closed, rejected, cancelled), but it has to be ensured that one of them is always reached (e.g., there is a status "solved" but the status model defines an additional step "closed" that will usually not be reached).

Recommendations and rules:

[SUP.9.RL.7] If the strategy does not include the definition of a status model, workflow, criteria for status changes, stakeholder and their authorization, the indicator BP1 shall be downrated.

[SUP.9.RL.8] If the status model and workflow does not fit to the actual way of working or is not applied correspondingly, the indicator BP3 must not be rated higher than P.

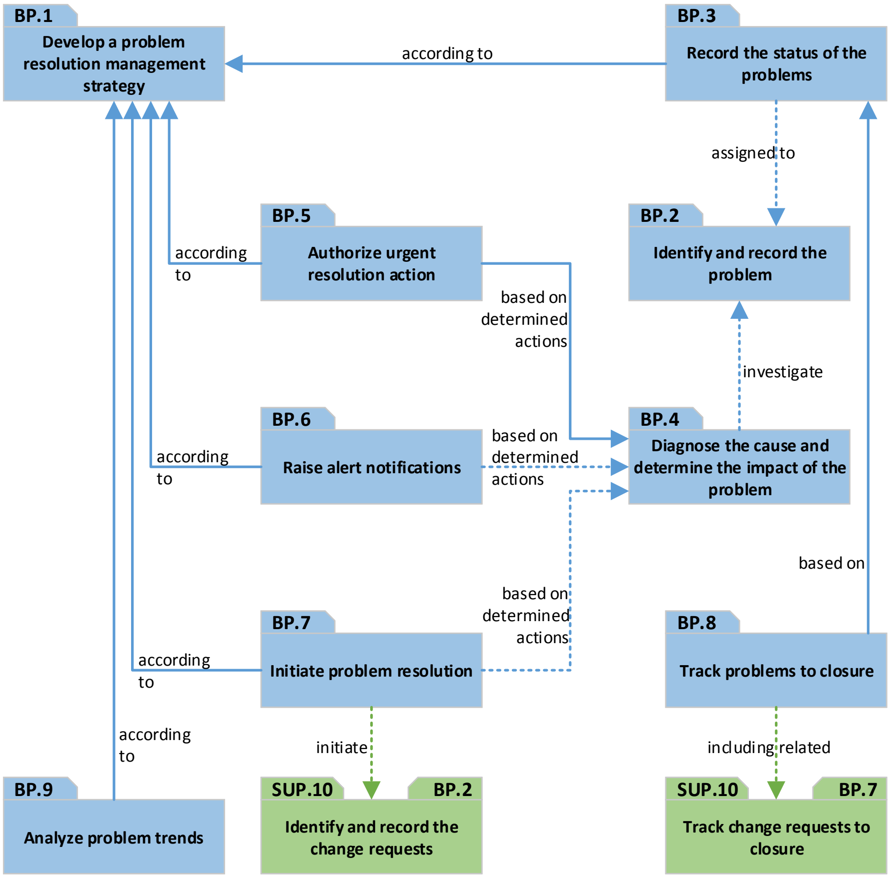
[SUP.9.RC.1] If the initiator of the problem is not also authorizing the closure of the problem, the indicator BP8 should be downrated.

Related to:

- BP1 "Develop a problem resolution management strategy"
- BP3 "Record the status of the problems"
- BP8 "Track problems to closure"
- Output WP 08-27 "Problem management plan"
- Output WP 13-07 "Problem record"
- Output WP 15-12 "Problem status report"

3.14.2 Rating consistency

The following figure shows relationships between SUP.9 base practices as well as their relationships to other processes:



These relationships are used as the basis for the rating rules and recommendations defined in the following subchapters.

Generic aspects regarding traceability (2.1.1), and strategy and plan (2.1.4) shall also be considered for rating.

3.14.2.1 Rating consistency within SUP.9

The following rating rule is related to the problem resolution management strategy and thus covers several base practices of the process:

> **[SUP.9.RL.9]** If the strategy-related activities are not performed according to the defined strategy (BP1), the indicators BP3, BP5, BP6, BP7, and BP9 shall be downrated, respectively.

Within SUP.9, the following base practices have relationships to each other:

BP3 Record the status of the problem

> **[SUP.9.RC.2]** If the degree of problem identification (BP2) is downrated, the indicator BP3 should not be rated higher.

BP4 Diagnose the cause and determine the impact of the problem

> **[SUP.9.RC.3]** If the recording of problems (BP2) is rated P or N due to insufficient content, the indicator BP4 should be downrated.

BP5 Authorize urgent resolution action

> **[SUP.9.RL.10]** If the analysis of the problem (BP4) is rated P or N, the indicator BP5 must not be rated higher.
>
> *Rationale: If the impact analysis is not available the authorization of resolution actions cannot be done on a sound basis.*

BP6 Raise alert notifications

> **[SUP.9.RC.4]** If the analysis of the problem (BP4) is rated P or N, the indicator BP6 should be downrated.
>
> *Rationale: If the impact analysis is not available the raising of alert notifications cannot be done on a sound basis.*

BP7 Initiate problem resolution

> **[SUP.9.RC.5]** If the analysis of the problem (BP4) is rated P or N, the initiation BP7 should be downrated.

BP8 Track problems to closure

> **[SUP.9.RL.11]** If problem status recording (BP3) is rated P or N, the indicator BP8 shall be downrated.
>
> *Rationale: If there are weaknesses in recording the status the tracking cannot be done on a sound basis.*

3.14.2.2 Rating consistency to other processes at level 1

The following base practices of SUP.9 have relationships to other processes:

BP7 Initiate problem resolution

[SUP.9.RC.6] If the rating of the indicator BP7 is downrated due to improper initiation of change requests, this should be in line with the rating of the identification and recording of change requests (SUP.10.BP2).

BP8 Track problems to closure

[SUP.9.RC.7] If the rating of the indicator BP8 is downrated, this should be in line with the rating of tracking change requests to closure (SUP.10.BP7).

The rationale is that if change request closing is not done properly, the closure of corresponding problem records can also not be done properly.

3.15 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 Change Request Management Process does not cover the actual implementation of change requests since this is done according to standard engineering processes (see also SUP.10.BP6). Therefore, any issues in rating this process have to be considered also carefully when rating those engineering processes.

The initiation of change requests (CRs) might come from a problem report (see also SUP.9.BP7). Any issues in rating due to improper handling of these change requests have to be considered also when rating the Problem Management Process.

Furthermore, traceability between change requests, problems, affected work products and corresponding baselines has to be ensured over all affected disciplines and all affected domains considering the project-specific complexity.

3.15.1 Rating recommendations

3.15.1.1 Strategy

Generic aspects, rules and recommendations regarding the strategy are given in chapter 2.1.4 and shall also be considered for rating of SUP.10.

The expectations for a successful strategy cover these aspects:

a) All organizational and/or project-specific aspects like affected disciplines (e.g., system, software, electronics), affected domains (e.g., software platform, COTS-Software), or affected sites are included.
b) The handling of all relevant interfaces to the customer, the supplier and relevant internal stakeholders (e.g., purchasing, marketing, and management) is included.
c) Management and exchange of change requests across disciplines, domains, sites, stakeholders, and (sub)-projects is defined.
d) A status life cycle for the change requests is defined. That life cycle has to cover any aspects and constraints mentioned above (different disciplines, sites, subprojects, stakeholder, etc.).

e) The strategy includes goals for certain activities (e.g., response times, lead times).

f) The strategy has to include guidance on the hierarchical approach for approval of a change request (e.g., up to a certain cost/effort limit the project manager approves, if the limit is exceeded the line manager approves).

g) A definition of the project phases in which change requests have to be used and a definition of potential differences at specific project phases have to be included (e.g., during prototype construction, series development, and after SOP).

h) A methodology to ensure traceability between change requests, problems, affected work products and corresponding baselines.

Recommendations and rules:

[SUP.10.RL.1] If the strategy does not include all aspects above, the indicator BP1 must not be rated F.

[SUP.10.RL.2] If the strategy does not address interfaces between multisite organizations/projects, subprojects, and/or groups in case of correspondingly complex projects, the indicator BP1 must not be rated higher than P.

[SUP.10.RC.1] If the strategy does not include goals according to e) above, the indicator BP1 should be downrated.

[SUP.10.RC.2] If change request handling is actually different over project life cycle phases but not consistent with the defined strategy, the indicator BP1 should be downrated.

[SUP.10.RC.3] If the use of a strategy is obvious by the implementation in a tool but not explicitly documented this should not be used to downrate the indicator BP1 to N or P.

Related to:

- BP1 “Develop a change request management strategy”
- Output WP 08-28 “Change management plan”

3.15.1.2 Decision authority

A change control board (CCB) is a common mechanism used to take decisions on the approval of change requests. Other mechanisms may also be used. For simplification, the term CCB is used.

The expectations of an adequately defined and operating CCB cover these aspects:

- All affected disciplines are appropriately represented
- All required stakeholders are represented (e.g., project manager, tester)
- The participants have the necessary authority to take decisions
- Attendance of all required stakeholders is given
- CCB drives the change management process, (e.g., focuses on decisions, takes decisions in time, re-delegates technical issues if necessary)
- Dependent on the organizational/project structure and/or organizational constraints (e.g., responsibility, budget, effort), there may be different hierarchical or organizational CCBs which have to be described regarding their respective responsibilities.

Recommendations and rules:

[SUP.10.RL.3] If not all relevant disciplines or stakeholders are represented in the actual CCB the indicator BP5 must not be rated F.

[SUP.10.RC.4] If it is apparent that decisions are not taken or not taken in time by the CCB without justification, the indicator BP5 should be downrated.

Related to:

- BP1 “Develop a change request management strategy”
- BP5 “Approve change requests before implementation”
- Output WP 08-28 “Change management plan”
- Output WP 13-16 “Change request”

3.15.1.3 Impact analysis and change confirmation

The expectations for an adequate impact analysis of change requests cover these aspects of the defined strategy:

- The input from all relevant stakeholders (internal and external) is considered including technical aspects and potential side effects.
- Feasibility, risks, complexity and impact regarding the potential changes are systematically evaluated and documented.
- The “defined modification and potential alternatives are unequivocally documented.
- Criteria for confirming implementation are established (e.g., selection of existing regression test case(s), newly developed test case, review of all modified work products).
- A review of the implemented change requests ensures that all relevant processes (e.g., SYS, SWE, MAN, and SUP) are applied and corresponding work products are updated accordingly.

Recommendations and rules:

[SUP.10.RL.4] If the analysis does not adequately address potential side effects due to specific risks and complexity of the potential changes the indicator BP4 must not be rated F.

[SUP.10.RC.5] If the technical content of the change request or in case of alternatives the decision for one alternative is not properly documented the indicator BP4 should be downrated.

[SUP.10.RL.5] If the review of implemented changes fails to detect that relevant processes are not applied; the indicator BP6 shall be downrated.

[SUP.10.RC.6] If the confirmation of a successful implementation of change requests is not based on documented criteria the indicator BP6 should be downrated.

Related to:

- BP4 “Analyze and assess change requests”
- BP6 “Review the implementation of change requests”
- Output WP 13-16 “Change request”
- Output WP 13-19 “Review record”
- Output WP 13-21 “Change control record”

3.15.1.4 Change request status model and workflow

The expectations for a status model and workflow cover these aspects:

- The status model and workflow are defined as part of the strategy, including criteria for status changes, and relevant stakeholder together with their responsibility and authorization, etc.
- The definition is in line with the actual way of working (e.g., possibility to switch back to previous states, use of iterations).
- Status model and workflow are applied as defined. The change requests always show the actual status (e.g., if modified software based on change requests is already released, the status should not still be "in implementation", "in review").
- The change request follows the workflow to a final status and is tracked accordingly. There might be more than one final status (e.g., closed, rejected, cancelled), but it has to be ensured that one of them is always reached (e.g., there is a status "solved" but the status model defines an additional step "closed" that will usually not be reached).

Recommendations and rules:

[SUP.10.RL.6] If the strategy does not include the definition of a status model, workflow, criteria for status changes, stakeholders and their authorization, the indicator BP1 shall be downrated.

[SUP.10.RL.7] If the status model and workflow does not fit to the actual way of working or is not applied correspondingly, the indicator BP3 must not be rated higher than P.

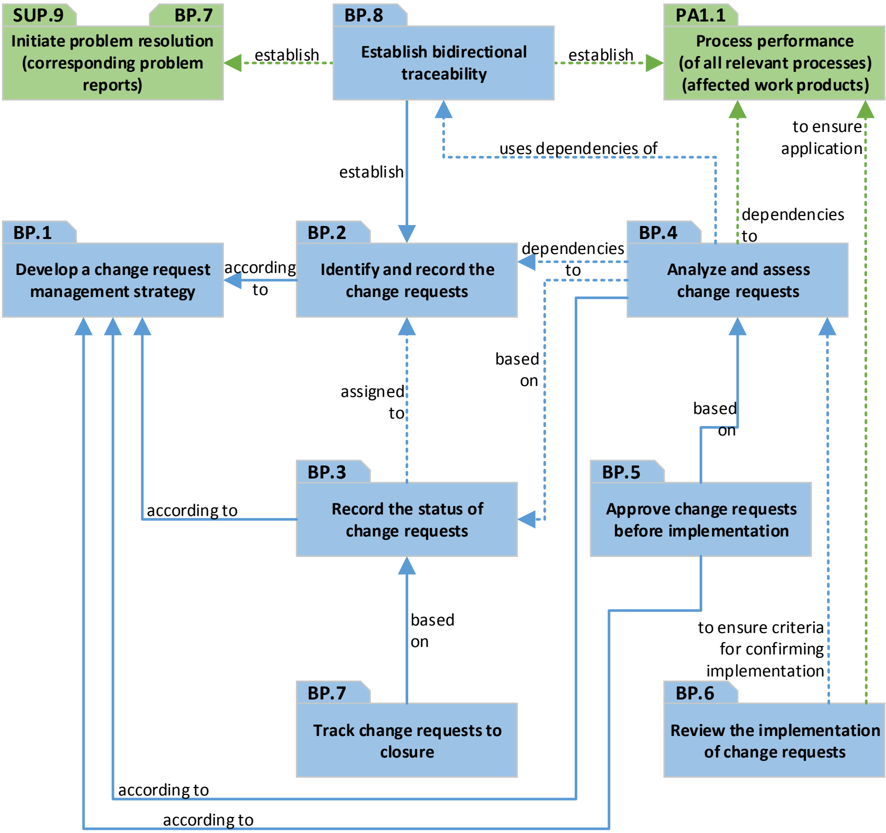
[SUP.10.RC.7] If closed CRs do not reflect a final state according to d) above, the indicator BP7 should be downrated.

Related to:

- BP1 "Develop a change request management strategy"
- BP3 "Record the status of change requests"
- BP7 "Track change requests to closure"
- Output WP 08-28 "Change management plan"
- Output WP 13-16 "Change request"
- Output WP 13-21 "Change control record"

3.15.2 Rating consistency

The following figure shows the relationships between SUP.10 base practices as well as their explicit relationships to other processes:



These relationships are used as the basis for the rating rules and recommendations defined in the following subchapters.

Generic aspects regarding traceability (2.1.1), and strategy and plan (2.1.4) shall also be considered for rating.

3.15.2.1 Rating consistency within SUP.10

The following rating rule is related to the change request management strategy and thus covers several base practices of the process:

> **[SUP.10.RL.8]** If the strategy-related activities are not performed according to the defined strategy (BP1), the indicators BP2, BP3, BP4, and BP5 shall be downrated, respectively.

Within SUP.10, the following base practices have relationships to each other:

BP3 Record the status of change requests

> **[SUP.10.RC.8]** If the strategy (BP1) is downrated due to not reflecting the complexity of the organization or project in the status flow, BP3 should be downrated.
>
> **[SUP.10.RC.9]** If the degree of CR identification (BP2) is downrated, the indicator BP3 "CR status recording" should not be rated higher.
>
> **[SUP.10.RC.10]** If the review of the implementation of the CRs (BP6) is downrated, it should have no influence on the rating BP3.

BP4 Analyze and assess change requests

> **[SUP.10.RC.11]** If the recording of CRs (BP3) is rated P or N due to insufficient content, the indicator BP4 should be downrated.
>
> **[SUP.10.RC.12]** If the rating of establishing bidirectional traceability (BP8) is downrated due to missing dependencies between CRs and affected work products, the indicator BP4 should be downrated.
>
> *Rationale: If the dependencies are missing, the impact analysis regarding the expected changes cannot be done on a sound basis.*

BP5 Approve change requests before implementation

> **[SUP.10.RL.9]** If the analysis of the change request (BP4) is rated P or N, the indicator BP5 must not be rated higher.
>
> *Rationale: If the analysis is not available the approval can not be done on a sound basis.*

BP6 Review the implementation of change requests.

> **[SUP.10.RC.13]** If the analysis of CRs (BP4) is rated P or N due to missing confirmation criteria, the indicator BP6 shall be downrated.

BP7 Track change requests to closure

[SUP.10.RL.10] If CR status recording (BP3) is rated P or N, the indicator BP7 shall be downrated.

Rationale: If there are weaknesses in recording the status the tracking can not be done on a sound basis.

BP8 Establish bidirectional traceability

[SUP.10.RL.11] If the initial recording of CRs (BP2) is rated P or N due to missing information about origin and/or reason, the indicator BP8 shall be downrated.

Rationale: If origin and/or reason are not recorded, the traceability cannot be established on a sound basis.

3.15.2.2 Rating consistency to other processes at level 1

The following base practices of SUP.10 have relationships to other processes:

BP4 Analyze and assess change requests

[SUP.10.RC.14] If the indicator BP4 is downrated due to missing analysis of dependencies to affected work products, this should be in line with the rating of PA 1.1 of the processes relevant to the maintenance of work products affected by the CR.

Rationale: If these dependencies are missing in the impact analysis, it can neither be ensured that all affected work products are updated as required, nor that relevant practices of the affected processes are performed.

BP6 Review the implementation of change requests

[SUP.10.RC.15] If the indicator BP6 is downrated due to not properly applying relevant processes during CR implementation, this should be in line with the rating of PA 1.1 of the processes, relevant to the maintenance of work products affected by the CR.

BP8 Establish bidirectional traceability

[SUP.10.RC.16] If the indicator BP8 is downrated due to missing dependencies between CRs and corresponding problem reports, this should be in line with the rating of the initiation of problem resolution activities (SUP.9.BP7).

[SUP.10.RC.17] If the indicator BP8 is downrated due to missing traceability between CRs and affected work products, this should be in line with the rating of PA 1.1 of the processes relevant to the maintenance of work products affected by the CR.

Additionally to the explicit relationships of SUP.10 BPs to other processes (as shown in the figure above), the following implicit relationship exists.

BP5 Approve change requests before implementation

[SUP.10.RC.18] If the indicator BP5 is downrated, this should be in line with the rating of the definition of the content of a release (SPL.2.BP1).

Rationale: If the approval of change requests does not include the allocation of change requests to releases, the content of a release is not fully defined.

3.16 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 of the process Project Management is to cover all aspects of planning, monitoring and tracking. In the previous version of the Process Assessment Model (V2.5) processes like e.g. SUP.1 Quality Assurance or SUP.10 Change Request Management included the planning of the activities of the respective processes.

In Automotive SPICE 3.1 all planning activities are either covered in MAN.3 Project Management or PA 2.1 Performance management process attribute of the respective processes.

Release planning is part of project management. However, the management of the release-baselines is part of SUP.8 Configuration management and the management of releases is covered in SPL.2 Product release.

3.16.1 Rating recommendations

3.16.1.1 New concept in Automotive SPICE 3.1 (Identify, monitor and adjust)

The formulation "*Identify, Monitor and adjust*" are used for the base practices BP3 (activities), BP4 (estimates and resources), BP7 (interfaces and commitments) and BP8 (schedule).

In the former versions, the respective base practices only addressed the setup of these artifacts or work products. Now, with the current version of the process assessment model, the base practices are also including the monitoring and adjustment aspects as depicted above.

For this reason, the former adjustment base practice BP.12 *"Act to correct to correct deviations*" is no longer necessary.

3.16.1.2 Scope of work

The scope of work has to cover the content, the boundaries, and the constraints of the project, including project and product scope. Describing the product only is not sufficient.

[MAN.3.RL.1] If the scope of work (BP1) is a product description only, the indicator BP1 must not be rated higher than L.

Specific scenario: If the organization has full responsibility for the system but a part of the application SW is developed by the customer, and it is provided by means of a SW library. As a result, the developing organization cannot be entirely responsible for the software requirements and the corresponding SW testing of this SW. This has to be explicitly documented as part of the scope of the project.

[MAN.3.RC.1] If the scope of work (BP1) does not address the responsibilities of all affected parties regarding the project and product, the indicator BP1 should not be rated higher than L.

[MAN.3.RL.2] If the scope of work (BP1) is not appropriately documented at project start, the indicator BP1 must not be rated higher than L.

Related to:

- BP1 “Define the scope of work”
- Output WP 08-12 “Project plan”

3.16.1.3 Neglecting commitments

Often projects do not fulfill their commitment by delaying timelines or cancelling functionality etc.

[MAN.3.RC.2] If the commitment is not fulfilled by delaying the timeline of the project or by cancelling functionality etc., the indicators BP1 and BP3 should not be rated higher than L.

[MAN.3.RL.3] If the commitment is not fulfilled by delaying the timeline of the project or by cancelling functionality etc., the indicator BP5 and BP8 must not be rated higher than L.

Rationale: The goal of the BPs for feasibility, estimates and schedule (BP3, BP5 and BP8) are covering “define, monitor and adjust…”. That means that monitoring and adjusting only is not enough. The initial definition of feasibility, estimates and scheduling should already be reliable.
Obviously delays agreed with the stakeholders should not lead to downrating.

Related to:

- BP1 “Define the scope of work”
- BP3 “Evaluate feasibility of the project”
- BP5 “Determine, monitor und adjust project estimates and resources”
- BP8 “Define, monitor and adjust project schedule”
- Output WP 08-12 “Project plan”
- Output WP 14-06 “Schedule”

3.16.1.4 Definition of activities

In general, a work break down structure is required. The dependencies of as well as the activity itself with input and output should be described. Another aspect which has to be considered is an adequate size of the work packages. Depending on the monitoring cycles of the detailed planning (as a rule at least one to two releases) work packages should not exceed the time of one, max. two monitoring cycles.

Rationale: If a project manager checks the schedule once a week (monitoring cycle), work packages should not exceed one week. The maximum size of work packages can extend up to two weeks but that should be the exception. Only with a detailed planning and monitoring on the basis of such work packages it is possible to ensure to control the project and detect deviations early. A 4-week-package is difficult to monitor in a weekly check (did we achieve 25% after the first week?).

[MAN.3.RC.3] If the activities are not described with input and output artifacts, the indicator BP4 should not be rated higher than P.

[MAN.3.RC.4] If the dependencies between activities are not identified, the indicator BP4 should not be rated higher than L.

[MAN.3.RC.5] If the work packages are too big (e.g. longer than the update cycle for the schedule), the indicator BP8 should be downrated.

Related to:

- BP4 “Define, monitor and adjust project activities”
- BP8 “Define, monitor and adjust project schedule”
- Output WP 14-06 “Schedule”
- Output WP 14-09 “Work breakdown structure”

3.16.1.5 Effort and resource estimation

The method for effort estimation must be comprehensible. This would be the case if for instance a database derived from an analysis of past projects would be used. Another possibility would be using widely accepted methods like wide-band Delphi, or independent estimates by several experts which are then to be consolidated.

Not acceptable are e.g. estimates by a single person only without any further review, or without involvement of affected parties.

The necessary resources should include e.g. people, development tools, hardware samples, infrastructure & test equipment.

> **[MAN.3.RC.6]** If the estimation method used is not comprehensible, the indicator BP5 should not be rated higher than P.
>
> **[MAN.3.RC.7]** If the estimates are too high level, e.g. based on high-level packages rather than on actual activities, the indicator BP5 should not be rated higher than P.
>
> **[MAN.3.RC.8]** If there are not sufficient resources to cover the estimated effort, the indicator BP5 should not be rated higher than P.
>
> **[MAN.3.RC.9]** If the resources are sufficient to cover the estimates but a monitoring of actual effort versus the estimates is missing, the indicator BP5 should not be rated higher than L.
>
> **[MAN.3.RC.10]** If the rationale for the estimates is missing, the indicator BP5 should not be rated higher than L.

Related to:

- BP5 “Determine, monitor und adjust project estimates and resources”
- Output WP 08-12 “Project plan”

3.16.1.6 Estimation of change requests and problem resolution

In practice an increasing number of change requests, reported problems, and verification issues towards later sample stages can be anticipated. This needs to be reflected during project definition and project planning.

[MAN.3.RC.11] If the definition of activities, effort and resource estimation, and the preparation of schedule(s) do not sufficiently reflect expectable change requests and problem resolution, the indicators BP4, BP5 and BP8 should be downrated.

[MAN.3.RC.12] If the project lifecycle does not contain phases that allow for addressing change requests and problem resolution, the indicator BP2 should be downrated.

Related to:

- BP2 “Define project life cycle”
- BP4 “Define, monitor and adjust project activities”
- BP5 “Determine, monitor und adjust project estimates and resources”
- BP8 “Define, monitor and adjust project schedule”
- Output WP 08-12 “Project plan”
- Output WP 13-16 “Change request”
- Output WP 14-02 “Corrective action register”
- Output WP 14-06 “Schedule”

3.16.1.7 Schedule and tracking

The tracking of action items and corrective actions has to be checked in the following BPs:

- All “…monitor, and adjust...” base practices (BP4, BP5, BP7 and BP8)
- BP10 Review and Report Progress of the Project

See also 3.16.1.1 here.

Tracking of corrective actions may also be linked to SUP.9 Problem Resolution Management.

[MAN.3.RC.13] If action items or corrective actions are not properly tracked to closure, the corresponding indicators BP4, BP5, BP7, BP8 and/or BP10 should be downrated.

[MAN.3.RL.4] If the schedule is not based on the defined activities (BP4) and estimations (BP5), the indicators BP8 and BP9 must not be rated higher than P.

[MAN.3.RL.5] If the schedule does not contain all of the following
- a start and end date,
- duration,
- degree of fulfillment (for monitoring purposes),
- resources,
- dependencies

the indicator BP8 must not be rated higher than L.

[MAN.3.RL.6] If any of the following:
- start and end date,
- effort,
- degree of fulfillment

is missing, the indicator BP8 must not be rated higher than P.

[MAN.3.RL.7] If the schedule is changed without a documented reason, or the change is not documented, the indicator BP8 shall be downrated.

[MAN.3.RL.8] If the degree of activity fulfillment as tracked in the schedule is not up to date (at least biweekly depending on the project scope and release plan), the indicator BP8 shall be downrated.

[MAN.3.RL.9] If the critical path in a schedule is not determined, the indicator BP8 shall be downrated.

Related to:

- BP4 “Define, monitor and adjust project activities”
- BP5 “Determine, monitor und adjust project estimates and resources”
- BP7 “Identify, monitor and adjust project interfaces and agreed commitments”
- BP8 “Define, monitor and adjust project schedule”
- BP9 “Ensure consistency”
- BP10 “Review and report progress of the project”
- Output WP 14-02 “Corrective action register”
- Output WP 14-06 “Schedule”
- Output WP 14-09 “Work breakdown structure”

3.16.1.8 Actual project progress

In practice, it often occurs that content progress is not aligned with resource consumption or chronology, e.g. progress of 20% but already 80% of the allocated budget consumed one week before a planned delivery. Such a situation implies failure of project management in terms of controlling all aspects to achieve the project goals within constraints and estimates.

> **[MAN.3.RL.10]** If monitoring does not assess the correlation of actual consumption of resources, meeting of deadlines and fulfillment of activities (i.e. progress of content), the indicator BP10 must not be rated higher than P.

Related to:

- BP10 “Review and report progress of the project”
- Output WP 14-06 “Schedule”
- Output WP 15-06 “Project status report”

3.16.1.9 Release management

Releases and their management are not dealt within a single process only but represent a topic distributed across several processes:

- Generally, the project has to define which information, work products, and products have to be delivered to, or received from all relevant stakeholders (MAN.3.BP7).
- The planning of releases is based on the estimates (BP5), and schedules (BP8) in MAN.3 as well as in SUP.8 Configuration Management and SPL.2 Product Release Management
- The release must be built from configured items (SPL.2.BP5) which relates to configuration management that ensures integrity (SUP.8). Deadline information of product releases will be part of schedules (MAN.3.BP8).
- Release planning is also covered in the requirements processes (SYS.2.BP2 and SWE.1.BP2) which expect a mapping of requirements to specific releases (see Note of those BPs).

Therefore, the following rules apply:

[MAN.3.RL.11] If product release recipients are not considered as stakeholders, the indicator BP7 must not be rated higher than P.

[MAN.3.RL.12] If product release deadlines or milestones are not reflected in schedules (consider also consistency across different schedules), the indicator BP8 must not be rated higher than P.

[MAN.3.RL.13] If the scope of the current and next release is not identified in detail (features and/or functions per release), the indicators BP7 and BP8 must not be rated higher than P.

[MAN.3.RL.14] If the mid and long-term planning of the releases does not at least cover a latest release/milestone for features and/or functions, the indicators BP7 and BP8 must not be rated higher than L.

[MAN.3.RL.15] If for the current and next release not all the expected activities are planned and tracked (without a good reason), the indicators BP4, BP7 and BP8 shall not be rated higher than L. If less than 50% of the expected activities are planned, BP4, BP7, BP8 must not be rated higher than P.

Related to:

- BP4 “Define, monitor and adjust project activities”
- BP7 “Identify, monitor and adjust project interfaces and agreed commitments”
- BP8 “Define, monitor and adjust project schedule”
- Output WP 08-12 “Project plan”
- Output WP 14-06 “Schedule”
- Output WP 14-50 “Stakeholder groups list”
- Output WP 15-06 “Project status report”

3.16.1.10 Consistency of planning information

The outputs from BP1 to BP8 are amongst others the description of the scope of work, analysis of the feasibility, description of the work packages, the estimates, the master schedule and detailed schedules, documentation of the skills and training needs, communication plan and stakeholders, reports on the status of the project, and a project plan. All these artifacts have to be consistent which means that the content of the different work products is free of contradiction and can be mapped.

Examples:

- Estimates are often done for high level activities. Usually these high-level activities are refined in the schedule. These low-level activities have to map to the high-level activities which were the basis for the estimates, e.g. by naming convention or structure. No tool supported links are required.
- Activities of the master project and subprojects have to be aligned and consistent, e.g. project plans for the different engineering domains. Dependencies between these plans have to be easily identified and mapped.

For project management, explicit links between e.g. plans and schedules are not required. Consistency can be reached by comparing and matching planning information.

[MAN.3.RC.14] If links between different types of planning information are not supported by tools, this should not be used to downrate the indicator BP9.

[MAN.3.RL.15] If the correlation between different plans or between estimates and plans is too high level or weak, the indicator BP9 shall be downrated.

Related to:

- BP9 “Ensure consistency”
- Output WP 08-12 “Project plan”
- Output WP 13-16 “Change request”
- Output WP 14-02 “Corrective action register”
- Output WP 14-06 “Schedule”
- Output WP 14-09 “Work breakdown structure”
- Output WP 14-50 “Stakeholder groups list”
- Output WP 15-06 “Project status report”

3.16.1.11 Project risks

Project management should consider risks for the project scope, feasibility, estimates, skills etc.

[MAN.3.RC.15] If risks regarding feasibility are not considered, the indicator BP3 should be downrated.

[MAN.3.RC.16] If risks regarding estimates or resources are not considered, the indicator BP5 should be downrated.

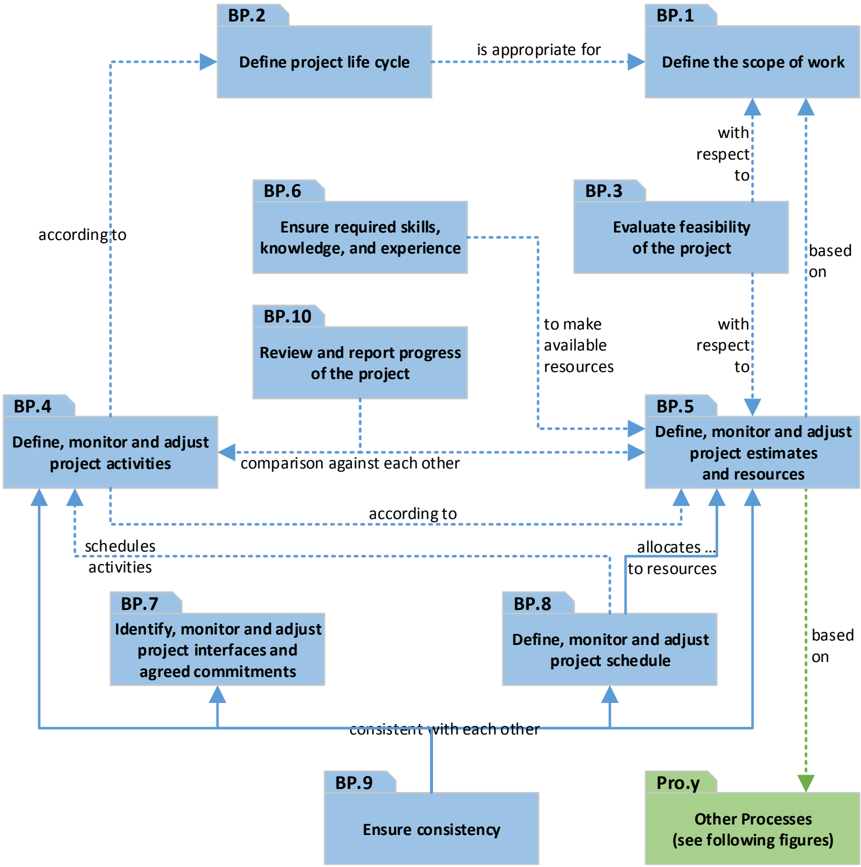
[MAN.3.RC.17] If risks regarding skills or knowledge are not considered, the indicator BP6 should be downrated.

Related to:

- BP3 “Evaluate feasibility of the project”
- BP5 “Determine, monitor und adjust project estimates and resources”
- BP6 “Ensure required skills, knowledge, and experience”
- Output WP 08-12 “Project plan”
- Output WP 14-06 “Schedule”
- Output WP 14-09 “Work breakdown structure”

3.16.2 Rating consistency

The following figure shows relationships between MAN.3 base practices as well as their relationships to other processes:



These relationships are used as the basis for the rating rules and recommendations defined in the following subchapters.

Generic aspects regarding consistency (2.1.1) shall also be considered for rating.

3.16.2.1 Rating consistency within MAN.3

BP2 Define project life cycle

[MAN.3.RC.18] If the definition of the scope of work (BP1) is downrated, then the indicator BP2 should be downrated.

BP3 Evaluate feasibility of the project

[MAN.3.RC.19] If the definition of the scope of work (BP1) is downrated, then the indicator BP3 should be downrated.

BP4 Define, monitor and adjust project activities

[MAN.3.RC.20] If the project lifecycle (BP2) is downrated, the indicator BP4 should be downrated.

BP5 Determine, monitor und adjust project estimates and resources

[MAN.3.RC.21] If the definition of the scope of work (BP1) is downrated, then the indicator BP5 should be downrated

[MAN.3.RC.22] If the feasibility of the project (BP3) is not evaluated, the indicator BP5 should be downrated.

[MAN.3.RC.23] If the activities defined as a result of BP4 are not mapped to the estimates, the indicator BP5 should be downrated.

[MAN.3.RC.24] If the estimates are not correlated with the available skills of the project (BP6), the indicator BP5 should be downrated.

Example: The estimate is based on the availability of a senior engineer but only a junior engineer is available. Hence, the estimate should be updated as a junior engineer will need more time to implement the same task than a senior engineer.

BP8 Define, monitor and adjust project schedule

[MAN.3.RL.17] If the estimates are not developed systematically (BP5) the indicator BP8 shall be downrated.

[MAN.3.RC.25] If the definition of work packages is weak, dependencies between work packages are not captured or the activities of the project are not properly broken down and documented (BP4), the indicator BP8 should be downrated.

BP9 Ensure consistency

See chapter 3.16.1.10.

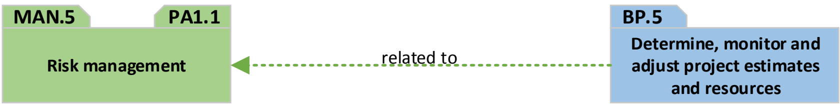
BP10 Review and report progress of the project

[MAN.3.RC.26] If the activities of the project are not properly broken down and documented (BP4), the indicator BP10 should be downrated.

[MAN.3.RC.27] If the estimates are not properly documented (BP5), the indicator BP10 should be downrated.

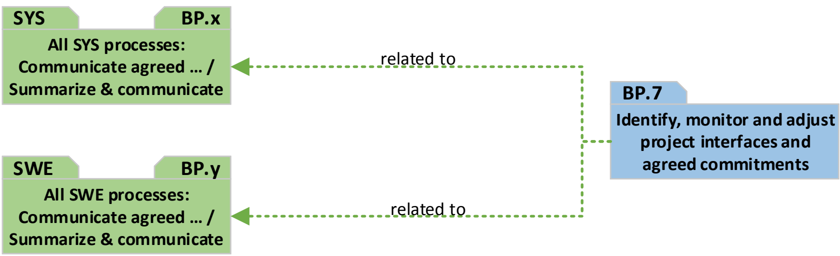
3.16.2.2 Rating consistency to other processes at BP level

Relation to Risk Management process:



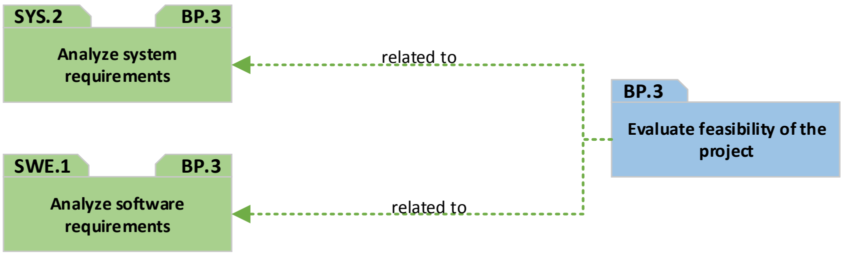
[MAN.3.RC.28] If Risk Management (MAN.5 PA 1.1) is downrated then the indicator BP5 (control of estimates and resources) should not be rated higher than L.

Relation to communication of results of engineering processes:



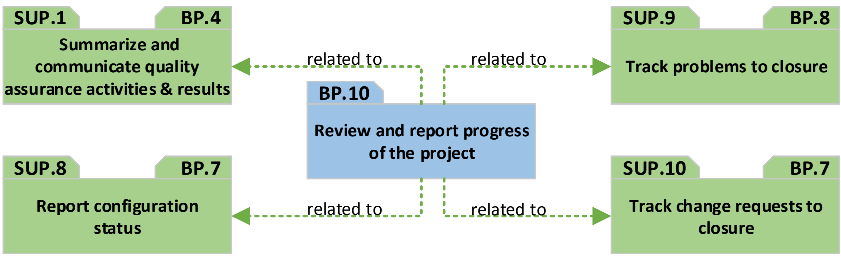
[MAN.3.RC.29] If the relevant BP of the engineering processes regarding communication (last BP of all engineering processes) is downrated, this should be in line with the rating of the indicator BP7.

Relation to analysis in requirements processes:



[MAN.3.RC.30] If the relevant BP of the requirement processes on system level (SYS.2.BP3) or software level (SWE.1.BP3) is downrated due to a missing or weak analysis regarding technical feasibility, this should be in line with the rating of the indicator BP3.

Relation to tracking in supporting processes:



[MAN.3.RC.31] If the relevant BP regarding status of quality assurance (SUP.1.BP4), status of configuration items (SUP.8.BP7), status of problems (SUP.9.BP8) and status of change requests (SUP.10.BP7) is downrated due to a missing or weak report, this should be in line with the rating of the indicator BP10.

4 Rating guidelines on process capability level 2

The previously described performed process is now implemented in a managed fashion (planned, monitored and adjusted) and its work products are appropriately established, controlled and maintained.

On capability level, one process-specific indicators are used to evaluate the extent to which the outcomes of the process are achieved. Assessors regularly use the base practices to assess a project's capability. These are activity-based indicators. In addition, there are output work products which are result-oriented indicators. Guidance on possible content of the output work products is documented in Annex B of Automotive SPICE.

On higher levels of capability generic practices and generic resources are available as indicators. As the names imply these indicators are not process-specific and have to be used for all processes. Hence, they have to be interpreted for each single process individually. Assessors focus on generic practices which cover the generic resources implicitly.

On capability level one the goal is to achieve the purpose of the process. Therefore, the assessor judges whether the result of the process is appropriate with respect to the context of the project including achievement of all outcomes.

On capability level 2 all activities which lead to the purpose of the process and capability level 2 itself (like e.g. reviews) have to be planned and controlled and all resulting work products have to be considered regarding configuration management and quality assurance.

Additionally, on capability level 2, objectives (e.g., planning goals) for the activities which have to be planned for the assessed process have to be documented. Also, requirements for all relevant work products of each process have to be defined. These requirements include such information as content and structure (e.g., as table of contents), history, layout, etc. Very often, the requirements for a work product are documented as a work product template including instructions for the usage of the template. If tools are used it should be documented how the tools have to be used, e.g. which fields are mandatory and which optional.

There is a strong dependency between project management (MAN.3) and process attribute 2.1 Performance Management. Regarding the process attribute 2.2 Work Product Management there is a strong dependency to quality assurance (SUP.1) and configuration management (SUP.8). For details refer to the chapters on PA 2.1 and PA 2.2 below.

4.1 Dependency between process attributes of level 1 and 2

[CL2.RC.1] At least one of the ratings of PA 2.1 or PA 2.2 should not be greater than the rating of PA 1.1.

Rationale: If only little of the expected process outcomes are established then only this little can be planned, monitored, and adjusted. Also, if the activities were well planned and controlled and the work products are well-managed, there could not be weaknesses on level 1.

4.2 Performance Management (PA 2.1)

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

The objectives for the performance of the process including required activities, tasks, responsibilities, resources, and involved stakeholders have to be defined on level 2 for the project in order to ensure a proper planning, monitoring and adjusting of the activities of the corresponding process. This includes also the planning, monitoring, and adjusting of all activities related to work product management as required by PA 2.2, e.g. work product reviews (see chapter 4.3). An explicit process description is not necessarily required for fulfilling PA 2.1 as long as all generic practices are accomplished.

Organizations do not need to structure the activities to be planned and monitored in the same way as it is done in the PAM and can use their own process naming conventions. Process assessors are in charge of mapping planning and monitoring related data to the right processes. It is up to the project to define its own structure, and consequently, uses this structure for its planning, monitoring, and adjusting activities (which might also cover more than one PAM process). Furthermore, it might even not be reasonable to plan all single activities explicitly (e.g., requiring explicitly planned check-in and check-out tasks in the project plan is not reasonable when assessing the configuration management process SUP.8).

Important for process attribute 2.1 is also the identification of objectives (e.g. planning goals or milestone conditions) for the planning. It is not required that this is described on an organizational level. However, if the objectives are described on an organizational level this may support the practices on capability level 2.

Generic practices of PA 2.1 are used to evaluate the capability of a project to plan and monitor activities related to a certain process, and not the degree to which planning and monitoring of particular processes are consistent regarding the overall project (which is the main focus of the MAN.3 process, see also chapter 3.16). However, there is a strong relationship between PA 2.1 and MAN.3 (see also chapter 4.3.2.2, *“Rating consistency to processes at level 1”*).

However, all interpretation and rating guidelines defined in chapter 3.16 (“MAN.3 Project Management”) have to be applied correspondingly for all GPs of PA2.1 (e.g., granularity of activities, frequency of monitoring activities).

Furthermore, all individuals and groups involved in the process performance have to be determined (GP 2.1.7), which means that all relevant stakeholders have to be considered also for all other generic practices of PA 2.1.

4.2.1 Rating rules and recommendations

4.2.1.1 Identify the objectives for the performance of the process (GP 2.1.1)

Process performance objectives are defined:

a) requirements regarding necessary activities and tasks in order to fulfill the process purpose are considered. This may include:

- milestones and/or due dates to be kept
- effort
- process cycle time or frequency
- metrics
- usage of qualified human and defined infrastructure resources
- quality criteria regarding the process

b) assumptions and constraints are considered, e.g.:

- adherence to internal standards
- adherence to customer standards, norms, or laws

c) stakeholder requirements are considered

Process performance objectives can either be quantitative (e.g., percentage of implemented requirements) or qualitative (e.g., adherence to Automotive SPICE capability level).

In the case that a standard process exists (required on level 3), the standard process might already include performance objectives. However, if the performance objectives for the project (on level 2) are based on or derived from the standard process, it has to be ensured that the standard process fits to the project’s purpose when using those objectives as evidence for rating the indicator GP 2.1.1.

Even though note 2 of GP 2.1.1 in the PAM states that as a minimum, project performance objectives for resources, effort and schedule should be stated, it must be judged by the assessor whether the argumentation for the selected performance objectives is appropriate and properly reflects customer requirements (e.g., detailed schedule might not be needed for project management meetings, or configuration management activities like check-in or check-out).

This leads to the following rating rules and recommendations for the indicator GP 2.1.1:

> **[CL2.RL.1]** If process performance objectives do not cover aspect a) above, the indicator GP 2.1.1 must not be rated higher than P.
>
> **[CL2.RL.2]** If process performance objectives do not cover aspect b) and c) above, the indicator GP 2.1.1 shall be downrated.
>
> **[CL2.RL.3]** If process performance objectives do not include KPIs but consider aspect a) above, the indicator GP 2.1.1 must not be downrated.
>
> **[CL2.RL.4]** If a standard process does not exist, but all aspects above are fulfilled, the indicator GP 2.1.1 must not be downrated.

4.2.1.2 Plan the performance of the process to fulfill the identified objectives (GP 2.1.2),

In order to ensure a proper plan for performing the process, the following aspects must be covered while considering the identified process performance objectives (from GP 2.1.1) adequately:

a) all required activities in order to fulfill the process purpose are defined
b) estimates for the defined process performance attributes are defined (e.g., effort, duration, size of work products, etc.) and are reproducible
c) the sequence of required activities is defined
d) a schedule including key milestones and required activities is defined and in line with the stakeholder requirements. Additionally, time for bug fixing, vacation, and planning buffer should be considered.
e) the planning / schedule of defined activities must

   1) either include due date, effort, assigned resources, and responsibility for each required activity (typically done for engineering activities)

2) or as percentage or absolute number of a full-time-equivalent's available time for a certain period of time (typically done for project management (MAN.3) or supporting activities (e.g., quality assurance activities (SUP.1))

f) work product reviews (as required by GP 2.2.1, see also chapter 4.3.1.1) are part of the planning

g) evidence of the planning must be available, e.g.:

- as part of the project plan,
- as process-specific document (e.g., meeting plan, audit plan),
- as backlog, task board, Kanban board, etc.
- as part of an open-item list

Even though GP 2.1.1 and GP 2.1.2 require the definition of activities and tasks to be performed to satisfy the objectives of the process, it is not mandatory to have a process description in place (on level 2), as long as the information regarding process objectives and scope is available elsewhere.

This leads to the following rating rules and recommendations for the indicator GP 2.1.2:

[CL2.RL.5] If the planning of the performance of the process does not cover all aspects above, the indicator GP 2.1.2 shall be downrated.

[CL2.RL.6] If the planning of the performance of the process does not cover the aspects d) and e) above, the indicator GP 2.1.2 must not be rated higher than P.

[CL2.RL.7] If required activities of the process are not separately planned, but cover aspects e) and g) above, the indicator GP 2.1.2 must not be downrated.

[CL2.RL.8] If supporting activities as mentioned in e.2) above are not planned as explicit activities, but are planned as percentage or absolute number of hours over a certain period of time, the indicator GP 2.1.2 must not be downrated.

[CL2.RL.9] If no process description including required activities and tasks is available, but all aspects above are covered, the indicator GP 2.1.2 must not be downrated.

4.2.1.3 Monitor the performance of the process against the plans (GP 2.1.3)

In order to monitor the performance of the process against the plans (as defined according to GP 2.1.2), the following aspects have to be monitored:

a) the process is performed as planned
b) data regarding the defined process performance attributes is continuously collected
c) actual data is continuously compared with planned values (this means also that the granularity of planned and actual data is similar):
   - by comparing actual results in given time/duration/effort (regarding aspect e.1) of GP 2.1.2)
   - by comparing booked effort per cost center to planned values (regarding aspect e.2) of GP 2.1.2)
d) the comparison between planned and actual data should:
   - show the current state of progress,
   - ensure that planned results are achieved, or
   - identify deviations from the plan,
   - be performed in an adequate frequency (e.g., in case of delivery every eight weeks and monitoring and comparison every four weeks, a higher frequency would be adequate)
e) documentation of monitoring activities, e.g., as:
   - status report
   - status meeting minutes

This leads to the following rating rules and recommendations for the indicator GP 2.1.3:

> **[CL2.RL.10]** If the monitoring of the process does not cover all aspects above, the indicator GP 2.1.3 shall be downrated.
>
> **[CL2.RL.11]** If the level of detail of planned and actual values does not fit together (or if there is no mapping available) (aspect d), the indicator GP 2.1.3 shall be downrated.
>
> **[CL2.RL.12]** If the frequency of monitoring activities does not fit to the project duration (aspect d), the indicator GP 2.1.3 shall be downrated.

4.2.1.4 Adjust the performance of the process (GP 2.1.4)

In order to adjust the performance of the process, the following aspects have to be covered:

a) performance issues have to be identified on the basis of deviations (as ascertained from continuous monitoring of the continuously monitored process as required by GP 2.1.3)
b) in case of identified deviations regarding the defined process performance attributes (e.g., due dates, effort estimations, resource usage)
   - deviations are analyzed and causes determined, and
   - either corrective measures to align performance with plans have to be taken or
   - plans have to be adapted in such way that plan changes are still in line with the stakeholder requirements.

This leads to the following rating rules and recommendations for the indicator GP 2.1.4:

> **[CL2.RL.13]** If adjusting the performance of the process does not cover aspect a), the indicator GP 2.1.4 shall be downrated.
>
> **[CL2.RL.14]** If adjusting the performance of the process does not cover aspect b), the indicator GP 2.1.4 must not be rated higher than P.

4.2.1.5 Define responsibilities and authorities for performing the process (GP 2.1.5)

The following aspects need to be covered in the project and adequately documented:

a) Responsibilities (e.g. RACI-Definition for activities), commitments and authorities (e.g. access rights, budget release, release of work products) to perform the process activities of the project need to be defined, assigned, communicated, and agreed.
b) Responsibilities and authorities to verify process work products need to be defined, assigned, communicated, and agreed (e.g., for every work product with a defined verification measure it should be defined, who is responsible for the verification and who has the corresponding authority (e.g. senior engineer, independent quality assurance, management)).

c) Needs for process performance experience, knowledge and skills are defined. Needs can either be process-specific (e.g. method or tool trainings) or project-specific (e.g. customer flash tool).

In distinction to GP 3.1.3 all definitions made can be project-specific without considering a standard process or roles.

This leads to the following rating rules and recommendations for the indicator GP 2.1.5:

> **[CL2.RL.15]** If the definitions do not cover all aspects above, the indicator GP 2.1.5 shall be downrated.
>
> **[CL2.RL.16]** If the definition of the responsibilities does not cover aspect a), the indicator GP 2.1.5 must not be rated higher than P.
>
> **[CL2.RL.17]** If all aspects above are adequately covered, without considering the role definition in a standard process, the indicator GP 2.1.5 must not be downrated.

4.2.1.6 Identify, prepare, and make available resources to perform the process according to the plan (GP 2.1.6)

These cover the following aspects:

a) The human and infrastructure resources, necessary for performing the process (according to GP 2.1.2) are identified, made available, allocated and used. Resource planning is comprehensible (e.g. rate of utilization is transparent, vacation and trainings are considered, procedures for planning in matrix organization or distributed development are defined). A comparison of actual used and target resources should be available.

b) The human and infrastructure resources, necessary for performing the process are re-planned if objectives or constraints of the process changed during the defined project life cycle

c) The individuals performing and managing the process are prepared by training, mentoring, or coaching to execute their responsibilities (according to GP 2.1.5). A qualification fit/gap analysis should be performed. Necessary qualification measures are planned in time, according to the needs of the project.

The information necessary to perform the process is identified and made available for all individuals performing and managing the processes (e.g. project hand book, project wiki).

This leads to the following rating rules and recommendations for the indicator GP 2.1.6:

> **[CL2.RL.18]** If identification, preparation, and availability of resources do not cover all aspects above, the indicator GP 2.1.6 shall be downrated.
>
> **[CL2.RL.19]** If identification, preparation, and availability of resources do not cover aspects a) and b), the indicator GP 2.1.6 must not be rated higher than P.

4.2.1.7 Manage the interfaces between involved parties (GP 2.1.7)

The individuals and groups involved in the process performance are determined.

Managing the interfaces should cover the exchange of information and work products and should include the following aspects:

a) Responsibilities of the involved parties are assigned. It should be defined
   - who delivers what and
   - who is the receiver.

b) Interfaces between the involved parties are managed. Evidence is that
   - meetings are planned or set up on a regular basis
   - participants for the meetings are defined (depending on responsibilities, tasks or processes)
   - the communication path is defined (e.g. protocol, link to a baseline)
   - the trigger is defined
   - distribution lists are established (e.g. for minutes)

c) Communication between the involved parties is assured and effective. Evidence is that

- regular or planned meetings take place as planned
- interfaces are used as defined
- active (e.g. by mail or status transition) and/or passive (information just made available) communication is defined
- communication is documented (Agenda, meeting minutes, open item lists)
- follow-up on open items is assured

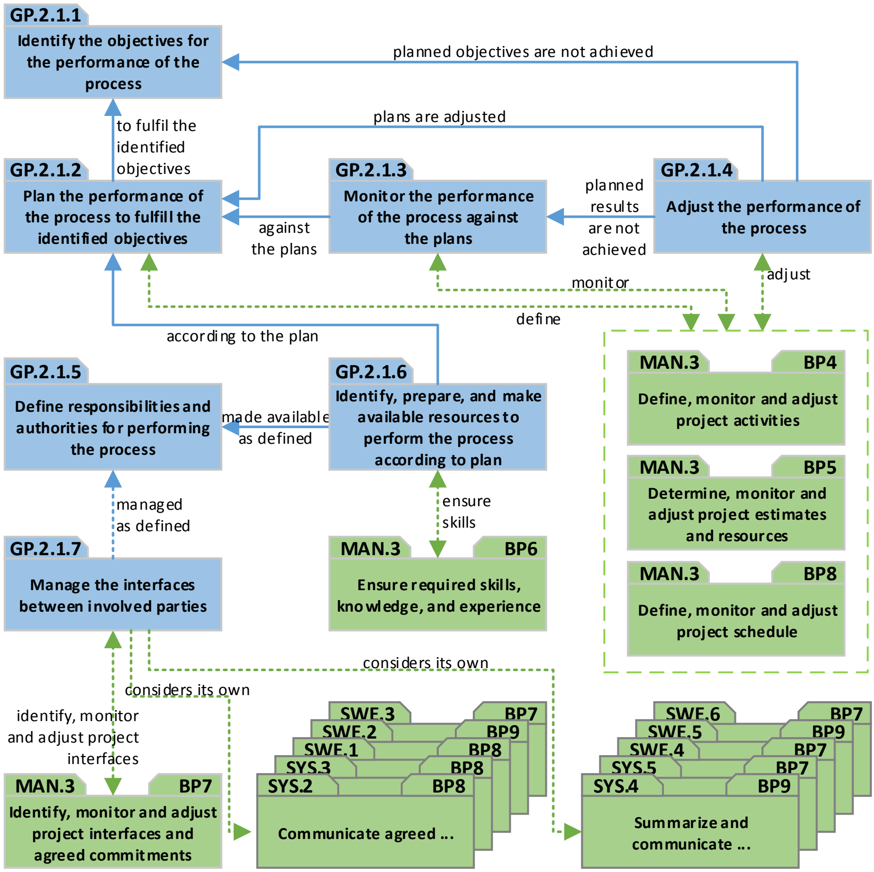
This leads to the following rating rules and recommendations for the indicator GP 2.1.7:

[CL2.RL.20] If managing the interfaces between involved parties does not cover all aspects above, the indicator GP 2.1.7 shall be downrated.

[CL2.RL.21] If communication between involved parties is not assured and effective (aspect c) above), the indicator GP 2.1.7 must not be rated higher than P.

4.2.2 Rating consistency

The following figure shows relationships between GP 2.1.x generic practices as well as their relationships to base practices of certain processes:



There is a strong dependency between project management (MAN.3) and process attribute PA 2.1 (“Performance Management”).

> **[CL2.RC.2]** If PA 2.1 is downrated for several processes, this should be in line with the rating of PA 1.1 of MAN.3.

Further relationships exist additionally to the dependencies shown above.

The relationship between the objectives (GP 2.1.1) and the strategy-related base practices is handled in chapter 2.1.4. The relationship between “involved parties” (GP 2.1.7) and the communication-related base practices is handled in chapter 2.1.2.

4.2.2.1 Rating consistency within PA2.1

The following rating rules are derived from the figure:

GP 2.1.2 Plan the performance of the process to fulfill the identified objectives

> **[CL2.RL.22]** If the indicator for identifying the objectives for the performance of the process (GP 2.1.1) is downrated, the indicator GP 2.1.2 shall be downrated.

GP 2.1.3 Monitor the performance of the process against the plans

> **[CL2.RL.23]** If the indicator for planning the performance of the process (GP 2.1.2) is downrated, the indicator GP 2.1.3 must not be rated higher.

GP 2.1.4 Adjust the performance of the process

> **[CL2.RL.24]** If the indicator for identifying the objectives of the process (GP 2.1.1) is downrated, the indicator GP 2.1.4 shall be downrated.
>
> **[CL2.RL.25]** If the indicator for planning the performance of the process (GP 2.1.2) is downrated, the indicator GP 2.1.4 must not be rated higher.
>
> *Rationale: The standard requires the adjustment to be based on deviations of the plan to reality. Consequently, one can only adjust something to the extent that it is planned.*
>
> **[CL2.RL.26]** If the indicator for monitoring the performance of the process (GP 2.1.3) is downrated, the indicator GP 2.1.4 must not be rated higher.
>
> *Rationale: The standard requires the adjustment to be based on deviations of the plan to reality. Consequently, one can only adjust something to the extent that it is monitored.*

GP 2.1.6 Identify, prepare, and make available resources to perform the process according to plan

[CL2.RL.27] If the indicator for planning the performance of the process (GP 2.1.2) is downrated, the indicator GP 2.1.6 must not be rated higher.

[CL2.RL.28] If the indicator for defining responsibilities and authorities (GP 2.1.5) is downrated due to inadequately defined responsibilities, the indicator GP 2.1.6 shall be downrated.

GP 2.1.7 Manage the interfaces between involved parties

[CL2.RC.3] If the indicator for defining responsibilities and authorities (GP 2.1.5) is downrated due to inadequately defined responsibilities, this should be in line with the rating of the indicator GP 2.1.7.

4.2.2.2 Rating consistency to processes at level 1

The following generic practices of the performance management attribute have relationships to other processes.

GP 2.1.2 Plan the performance of the process to fulfill the identified objectives

[CL2.RC.4] The rating of the indicator GP 2.1.2 of all processes should be in line with the ratings of the indicators MAN.3.BP4, MAN.3.BP5 andMAN.3.BP8, respectively.

GP 2.1.3 Monitor the performance of the process against the plans

[CL2.RC.5] The rating of the indicator GP 2.1.3 of all processes should be in line with the ratings of the indicators MAN.3.BP4, MAN.3.BP5 and MAN.3.BP8, respectively.

GP 2.1.4 Adjust the performance of the process

[CL2.RC.6] The rating of the indicator GP 2.1.4 of all processes should be in line with the ratings of the indicators MAN.3.BP4, MAN.3.BP5 and MAN.3.BP8, respectively.

GP 2.1.6 Identify, prepare, and make available resources to perform the process according to plan

[CL2.RC.7] The rating of the indicator GP 2.1.6 of all processes should be in line with the rating of the indicatorMAN.3.BP6.

GP 2.1.7 Manage the interfaces between involved parties

[CL2.RC.8] The rating of the indicator GP 2.1.7 of all processes should be in line with the rating of the indicatorMAN.3.BP7.

[CL2.RC.9] The rating of the indicator GP 2.1.7 of the considered process should be in line with the rating of its indicator for “Communicate Agreed …” (SYS.2.BP8, SYS.3.BP8, SWE.1.BP8, SWE.2.BP9, SWE.3.BP7).

[CL2.RC.10] The rating of the indicator GP 2.1.7 of the considered process should be in line with the rating of its indicator for “Summarize and Communicate” (SYS.4.BP9, SYS.5.BP7, SWE.4.BP7, SWE.5.BP9, SWE.6.BP7).

4.3 Work Product Management (PA 2.2)

The work product management process attribute is a measure of the extent to which the work products produced by the process are appropriately managed.

Relevant work products of the process are those that are required to fully achieve capability level 1, and additionally, evidence (work products) to prove successful implementation of the process attributes 2.1 and 2.2. A work product may not only be a document but could also be a record or database entry in a tool (e.g., change requests or problem reports implemented in a workflow tool are also work products).

Not included in the term “work product” are all process-related documents like e.g., process descriptions, procedures, method descriptions, or role descriptions. Any weaknesses in handling these process assets that are not related to the content (e.g., improper versioning) must not be reflected in the process attribute 2.2 of the process under investigation. However, if organizational process documents are available they can support the implementation of process attribute 2.2.

Work products are defined as output work products in the Automotive SPICE PAM 3.1. Each of the output work products is associated with one or more outcomes of the process and further detailed by work product characteristics in Annex B of the PAM. These work products and their characteristics can be used as a starting point for considering whether, given the context, they are contributing to the intended purpose of the process.

4.3.1 Rating rules and recommendations

4.3.1.1 Define the requirements for the work products (GP 2.2.1)

Work product requirements include:

a) Criteria defining content and structure, e.g.:

- Information regarding the structure such as layout, history, table of contents
- Technical content (e.g., requirement specifications, architectural descriptions)
- Project content (e.g., plans, minutes, open point lists)

- Guidelines (e.g., programming or modeling guidelines)
- Standards
- Instructions for usage of templates and tools (e.g., mandatory, optional, how to fill in)

b) Appropriate review and approval criteria, e.g.:

- Definition whether the work product needs to be explicitly reviewed or only implicitly reviewed by distributing them and accepting them in case of no feedback (e.g., minutes, open-point-lists, reports etc.).
- Definition regarding review method, review coverage (including justification), review frequency (including justification), and review participants

c) Quality criteria (based on aspects a) or b), or e.g., derived from the ISO/IEC 25010 standard, which includes "Efficiency", "Compatibility", "Reliability", "Maintainability" and "Portability".

Very often, the requirements for a work product are documented as a work product template or checklist. However, defining templates is not necessarily required by the work product management attribute as long as all aspects above are adequately documented.

This leads to the following rating rules and recommendations for the indicator GP 2.2.1:

[CL2.RL.29] If work product requirements do not include all aspects above, the indicator GP 2.2.1 shall be downrated.

[CL2.RL.30] If no template or checklist exists for the work product, but all aspects above are adequately documented, the indicator GP 2.2.1 must not be downrated.

[CL2.RL.31] If standard work product templates provided by a standard process are available, but the project has defined a project-specific solution that is effective, the indicator GP 2.2.1 must not be downrated.

[CL2.RL.32] If standard work product templates provided by a standard process are available and used by the project, but do not fit for the purpose of the project, the indicator GP 2.2.1 shall be downrated.

4.3.1.2 Define the requirements for documentation and control of the work products (GP 2.2.2)

Certain requirements regarding documentation and control have to be defined for all relevant work products. These requirements have to be set-up for each identified work product and must fit the overall configuration and change request management strategy (see also chapters 3.13 and 3.15 *"Rating Guidelines for SUP.8 and SUP.10"* and chapter 4.2.2.2).

The requirements for documentation and control should cover at least:

a) Identification of work products and their dependencies (including traceability between them)
b) Naming convention
c) Ownership
d) Access rights (at least read and write permission)
e) Work product life cycle including status model, approval and release procedure
f) Versioning rules (including baselining mechanisms depending on the work product type)
g) Storage media (e.g., project drive, configuration management tool)
h) Distribution channels (communication mechanisms for releases and changes)

This leads to the following rating rules and recommendations for the indicator GP 2.2.2:

> **[CL2.RL.33]** If the requirements for documentation and control do not cover all aspects above, the indicator GP 2.2.2 shall be downrated.

> **[CL2.RL.34]** If the requirements for documenting and controlling work products do not cover versioning and storage requirements (aspects f) and g) above), the indicator GP 2.2.2 must not be rated higher than P.

4.3.1.3 Identify, document and control the work products (GP 2.2.3)

All identified work products must be documented, and controlled (indicator GP 2.2.3) according to their requirements (indicator GP 2.2.2). Because of this dependency, the corresponding rule is defined in chapter 4.3.2.1, *"Rating consistency within PA 2.2"*.

4.3.1.4 Review and adjust work products to meet the defined requirements (GP 2.2.4)

Work product reviews have to be performed against defined work product requirements (see GP 2.2.1) in accordance with the planning (see PA 2.1). The execution of work product reviews including results has to be demonstrable. This does not necessarily require a formal review including dedicated review record, but can also be a less formal approach like walk-through, or pair-programming according to the quality assurance strategy (see SUP.1). However, the following aspects must be demonstrable:

a) Review information:

   1) Work product under review (including name and version to ensure proper identification)
   2) Date of the review
   3) Name(s) of reviewer(s)
   4) Review findings, if they are not immediately solved in the review (e.g., in pair programming)
   5) Review result (e.g., “Passed”, “Conditionally Passed”, “Failed / Re-review required”)
   6) used review and approval criteria

b) Handling of review findings:

   1) A procedure for handling of review findings has to be defined
   2) Review findings have to be monitored and tracked until resolution

This leads to the following rating rules and recommendations for the indicator GP 2.2.4:

> **[CL2.RL.35]** If the proof of work product reviews does not cover all aspects above, the indicator GP 2.2.4 shall be downrated.
>
> **[CL2.RL.36]** If the proof of work product reviews does not cover aspects a.1), a.4), and a.6) for the most relevant work products, the indicator GP 2.2.4 must not be rated higher than P.
>
> **[CL2.RL.37]** If work product review findings are not resolved for the most relevant work products, the indicator GP 2.2.4 must not be rated higher than P.

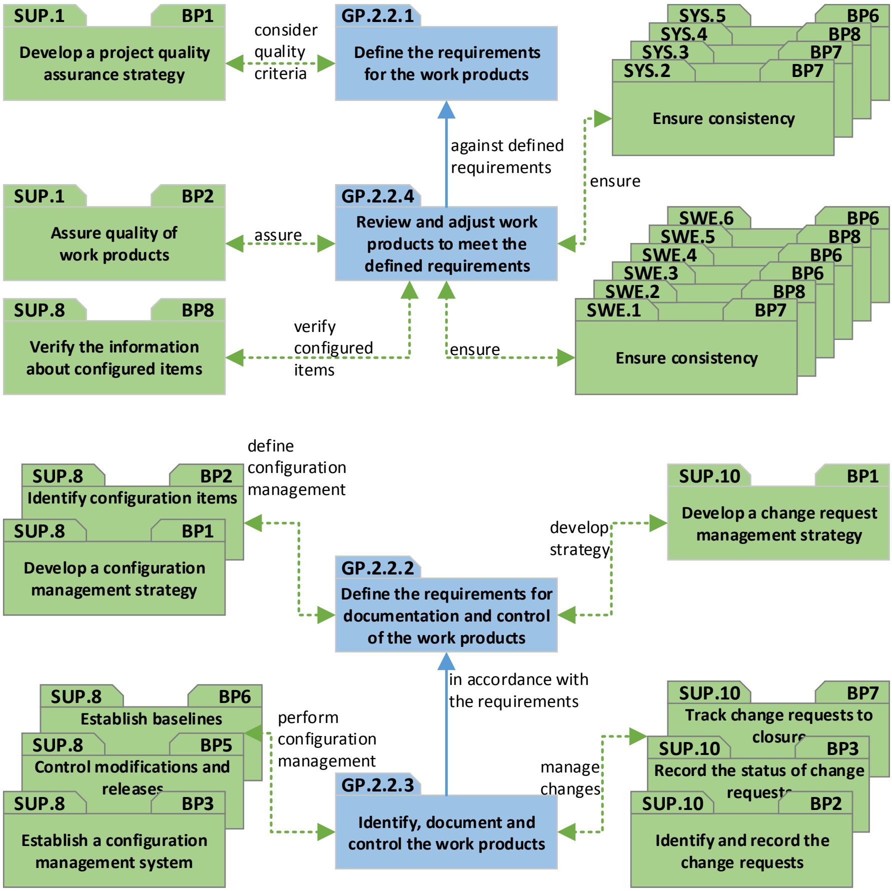
[CL2.RL.38] If the quality of work products was not established in time (i.e. not according to the planning in the context of PA 2.1), the indicator GP 2.2.4 shall be downrated.

[CL2.RL.39] If work product reviews are demonstrable according to all aspects above, but are not explicitly documented in a formal review record, the indicator GP 2.2.4 must not be downrated.

Rationale for not explicitly requesting a formal protocol: Review commenting might be done by using build-in functionality of the corresponding editor (e.g. tracking changes and/or commenting in MS Word).

4.3.2 Rating consistency

The following figure shows relationships among GP 2.2.x generic practices as well as their relationships to base practices of certain processes:



There is a strong dependency between quality assurance (SUP.1) respectively configuration management (SUP.8) and process attribute PA 2.2 "Work product Management".

> **[CL2.RC.11]** If PA 2.2 is downrated for several processes, this should be in line with the rating of SUP.1 and SUP.8.

4.3.2.1 Rating consistency within PA 2.2

The generic practices of capability level 2 can be grouped into two main topics (as visible in the figure above). The first one covers requirements, quality criteria, review, and adjustment of all relevant work products of the corresponding process (GP 2.2.1 & GP 2.2.4), whereas the second one covers the documentation and control of those work products (GP 2.2.2 & GP 2.2.3).

Consequently, the following two rules are derived:

GP 2.2.3 Identify, document and control the work products

> **[CL2.RL.40]** If the indicator for defining requirements for documentation and control of the work products (GP 2.2.2) is downrated, the indicator GP 2.2.3 must not be rated higher.

GP 2.2.4 Review and adjust work products to meet the defined requirements

> **[CL2.RL.41]** If the indicator for defining requirements for the work products (GP 2.2.1) is downrated due to non-appropriate review and approval criteria, the indicator GP 2.2.4 shall be downrated.

4.3.2.2 Rating consistency to processes at level 1

The following generic practices of the work product management attribute have relationships to other processes:

GP 2.2.1 Define the requirements for the work products

> **[CL2.RC.12]** The rating of the indicator GP 2.2.1 of all processes should be in line with the rating of the indicator SUP.1.BP1.

GP 2.2.2 Define the requirements for documentation and control of the work products

> **[CL2.RC.13]** The rating of the indicator GP 2.2.2 of all processes should be in line with the rating of indicator SUP.8.BP1 and the indicator SUP.8.BP2.The rationale for the recommendation is that the definition of the work product life cycle including status model is related to the identification, documentation and control of work products.

[CL2.RC.14] The rating of the indicator GP 2.2.2 of all processes should be in line with the rating of indicator SUP.10.BP1. The rationale for the recommendation is that the change request management strategy covers also the status model of the change requests, which is related to the identification, documentation and control of work products.

GP 2.2.3 Identify, document and control the work products

[CL2.RC.15] The rating of the indicator GP 2.2.3 of all processes should be in line with the ratings of the indicators SUP.8.BP3, SUP.8.BP5, and SUP.8.BP6, respectively.

[CL2.RC.16] The rating of the indicator GP 2.2.3 of all processes should be in line with the ratings of the indicators SUP.10.BP2, SUP.10.BP3, and SUP.10.BP7, respectively.

GP 2.2.4 Review and adjust work products to meet the defined requirements

[CL2.RC.17] The rating of the indicator GP 2.2.4 of all processes should be in line with the rating of the indicator SUP.1.BP2.

[CL2.RC.18] The rating of the indicator GP 2.2.4 should be in line with the rating of the indicator of the corresponding process for ensuring consistency of work products (SYS.2.BP7, SYS.3.BP7, SYS.4.BP8, SYS.5.BP6, SWE.1.BP7, SWE.2.BP8, SWE.3.BP6, SWE.4.BP6, SWE.5.BP8, SWE.6.BP6, SUP.8.BP8).

5 Rating guidelines on process capability level 3

The previously described Managed process is now implemented using a defined process that is capable of achieving its process outcomes.

On capability level 2 all projects may use “their” own process as long as the requirements of Automotive SPICE are fulfilled.

On capability level 3 the projects have to use a standard process. A possibility to cover variations between projects is to describe tailoring guidelines. This derived process is the so-called “defined” process. The defined process has to cover all activities and work products of capability level 1 and 2 for the assessed project.

Large organizations would have problems with only one standard process. The organization may define several different standard processes (e.g. one standard process for each development site, or one standard for each business unit). The other possibility to cover variations between projects is the afore-mentioned description of tailoring guidelines. Based on predefined criteria the process may be tailored to the needs of the project.

Exceptionally waivers for the standard process may be used (which should not be the rule), assessors should check whether these exceptions have a rationale and are approved by appropriate organizational roles.

It has to be kept in mind that the advantage of organizational processes is to standardize the approach to e.g.:

- establish processes known by the stakeholders
- establish interfaces to facilitate cooperation (also between different locations)
- facilitate introduction of new personnel or exchange personnel between projects
- facilitate reuse of assets and work products
- establish benchmarking

The aim of establishing processes might get missed if there are too many variations of the processes. This should be reflected by the assessment result.

5.1 Process Definition (PA 3.1)

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.

The process defined is organization wide and no longer project specific. The process description includes at least

- fundamental process elements (GP 3.1.1)
- sequence and interaction with other processes (GP 3.1.2)
- roles and competencies, responsibilities, and authorities for performing the standard process (GP 3.1.3)
- required infrastructure and work environment for performing the standard process GP 3.1.4)

Each of these 4 aspects have to be rated only in the respective GP.

5.1.1 Rating recommendations

5.1.1.1 Define and maintain the standard process that will support the deployment of the defined process (GP 3.1.1)

GP 3.1.1 covers the definition, the maintenance and the tailoring guidelines of the process.

Define the standard process

A process description should include at least the following elements or descriptions:

a) The scope and the intended use of the process
b) The description of the process activities
c) The definition of the Input and Output Work Products for every process activity
d) Templates or at least detailed requirements for the work products
e) The description of methods and procedures

[CL3.RL.1] If the definition of the standard process does not cover all aspects above, the indicator GP 3.1.1 must not be rated F.

[CL3.RL.2] If one of the aspects b) or c) is missing in the defined standard process, the indicator GP 3.1.1 must not be rated higher than P.

Maintain the standard process

Maintaining the standard process includes

a) continuous improvement of the process description, including the documentation of change requests and implemented changes (input from 3.2.6.)
b) adaptation to new requirements (e.g. infrastructure, new standards)
c) a definition of the responsibilities for the process development (e.g. process owner, process developer)
d) a definition of the valid version
e) a mechanism to ensure the availability of previous process versions since they might still be in use by running projects
f) a procedure for the deployment of new versions of the standard process e.g.

    - Release of a new standard process or new process elements in a late project phase
    - Obligatory use of the latest version of the standard process at project start

[CL3.RL.3] If maintaining the standard process does not cover all aspects above, the indicator GP 3.1.1 shall be downrated.

Deployment of the defined process

Deployment can be done with or without tailoring of the standard process.

Tailoring can be deleting, adding or selection between different elements of the process based on predefined criteria.

Tailoring Guidelines describe

a) criteria for tailoring,
b) proceeding of tailoring and
c) responsibility assignment (e.g. RASIC) for tailoring activities.

[CL3.RL.4] If the tailoring guidelines do not cover all aspects above, the indicator GP 3.1.1 shall be downrated.

If there is no tailoring defined, the following aspects need to be checked:

- The standard process is used unmodified in the project but is not Appropriate (see PA 3.2)

- The standard process cannot be effectively applied by the project
- The standard process is not suitable for the project

[CL3.RL.5] If there is no tailoring defined despite the aspects above, the indicator GP 3.1.1 shall be downrated.

5.1.1.2 Determine the sequence and interaction between processes so that they work as an integrated system of processes (GP 3.1.2)

There should be an applicable way to determine

a) which process activities depend on other process activities or work products

b) the sequence in which process activities need to be performed. Sequence may also include parallel or iterative sequencing of activities which are synchronized by e.g. work product completion.

The integrity of both aspects above needs to be ensured for the defined tailoring criteria, too.

[CL3.RL.6] If the sequence and interaction mentioned above are not defined, the indicator GP 3.1.2 must not be rated higher than P.

5.1.1.3 Identify the roles and competencies, responsibilities, and authorities for performing the standard process (GP 3.1.3)

The standard process should contain

a) the description of the involved roles and assignment of the roles to the process activities,

b) the responsibilities and authorities of roles (e.g. RACI-Definition) and

c) the necessary qualification for performing a role (qualification may include experience, expertise as well as social skills and trainings).

[CL3.RL.7] If the criteria mentioned above are not defined, the indicator GP 3.1.3 must not be rated higher than P.

5.1.1.4 Identify the required infrastructure and work environment for performing the standard process (GP 3.1.4)

Requirements for identifying infrastructure and work environment are

a) the definition and description of the used tools and infrastructure

b) methods and responsibilities to ensure that the needed work environment is available for the projects (e.g. licenses)
c) scope of use and
d) if required, the qualification of the tools (e.g. for safety critical use).

[CL3.RL.8] If the criteria mentioned above are not defined, the indicator GP 3.1.4 must not be rated higher than P.

5.1.1.5 Determine suitable methods and measures to monitor the effectiveness and suitability of the standard process (GP 3.1.5)

These methods and measures should consider:

a) The measurement of defined key figures (e.g. number of review findings, failures found after a dedicated test step, ideal discovery to actual discovery of failures, effort variance)

Examples: Rates of remaining failure after a dedicated test step can be defined for at least all SYS- and SWE-Processes with respect to e.g. the customer satisfaction.

These measures can be observed in relation to industrial standards, other standard processes of the company or as a trend for a single process.
The process in which a product fault was actually identified (i.e. during testing) vs. the process during which the product fault could, or should, have been identified already (e.g. during review of requirements).

b) Audits or assessments by quality assurance or external partner (e.g. process compliance, lack in process compliance)
Process compliance can be an evidence for the suitability of a process, a lack in process compliance for a majority of projects can be evidence that the process is not suitable.
c) Lessons learned reviews
Lessons learned or retrospective meetings can be used to get process feedback. Feedback should be documented in a defined way, analyzed and taking in account for process development
d) Process feedback of project staff
There should be a defined and well known way for project staff to give process feedback to the responsible process development organization.

[CL3.RL.9] If only the feedback methods c) and d) are determined, the indicator GP 3.1.5 must not be rated F.

5.2 Process Deployment (PA 3.2)

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.

The rating of process attribute 3.2 should reflect the degree to which the process is using the standard process under consideration of the tailoring guidelines.

5.2.1 Rating recommendations

5.2.1.1 Deploy a defined process that satisfies the context specific requirements of the use of the standard process (GP 3.2.1)

The deployment of a defined process should include

a) the project specific selection and/or tailoring from the standard process using the defined tailoring criteria. The decisions made and the rationale for the decisions need to be documented.

b) the verification that the defined process is conformant with standard process requirements and accordingly applied in the project. This has to be done by an authorized role, e.g. process owner, process group, quality management or quality assurance. Evidences of the verification or a final release of the defined process need to be documented.

[CL3.RL.10] If the defined process is not documented and verified according the criteria above, the indicator GP 3.2.1 shall be downrated.

5.2.1.2 Assign and communicate roles, responsibilities and authorities for performing the defined process (GP 3.2.2)

According to the defined process

a) the roles for performing the defined process are assigned and communicated and

b) the responsibilities and authorities for performing the defined process are assigned and communicated.

[CL3.RL.11] If the assignment does not cover all aspects above, the indicator GP 3.2.2 must not be rated F.

[CL3.RC.1] If roles, responsibilities and authorities are assigned and the assignment is available for all project members but there is no evidence for an active communication of the assignment, the indicator GP 3.2.2 should not be downrated.

5.2.1.3 Ensure necessary competencies for performing the defined process (GP 3.2.3)

Necessary competencies can either be process-specific (e.g. role or standard tool trainings) or project- specific (e.g. customer flash tool).

Ensuring necessary competencies includes:

a) The assurance of appropriate competencies for assigned personnel. Evidence that the assigned persons have the required qualification (e.g. qualification records) should be available. The qualification has to be in line with the competencies defined in GP 3.1.3 for performing the standard process. If deviations are shown, adequate qualification action should be defined.
b) The availability of suitable qualification for those performing the defined process. Availability ensures that project members are qualified in time, to perform the defined processes in the project.

[CL3.RL.12] If no evidence that the assigned persons have the required qualification is available, the indicator GP 3.2.3 must not be rated higher than P.

[CL3.RL.13] If the necessary competencies are not available in time, the indicator GP 3.2.3 must not be rated F.

Rationale: If a qualification measure is planned for the future, but the qualification is required today, the qualification is still missing.

5.2.1.4 Provide resources and information to support the performance of the defined process (GP 3.2.4)

Provide resources and information includes that

a) the required human resources are made available, allocated and used. In addition to GP 2.1.5:
   - the resources need to be available according the roles and qualification defined in the standard process.

- The availability of the resources needs to be ensured, taking into account that resources may be also used by other projects of the organization.

b) Required information should be
   - made available, allocated and used
   - easy to access for all project members

c) and should include
   - Expert knowledge from previous projects and training materials or
   - Models for resource estimation based on the recording of needed resources of former projects

[CL3.RL.14] If the provided resources and information do not cover all aspects above, the indicator GP 3.2.4 must not be rated F.

5.2.1.5 Provide adequate process infrastructure to support the performance of the defined process (GP 3.2.5)

The provision of adequate process infrastructure includes that:

a) The required infrastructure and work environment, according to standard process and the project specific definition is available.

b) Organizational support to effectively manage and maintain the infrastructure and work environment is available and known by the project members.
   - Resources for the support are planned by the organization.
   - Availability of licenses is checked regularly.
   - Information about anticipated or planned process infrastructure changes, e.g. new tool chains, shall be made available to the projects.

c) Infrastructure and work environment is used and maintained. If updates or new versions of the work environment are available, the handling has to be planned in coordination with the project.

[CL3.RL.15] If the organizational support does not cover the aspect above, the indicator GP 3.2.5 must not be rated F.

5.2.1.6 Collect and analyze data about performance of the process to demonstrate its suitability and effectiveness (GP 3.2.6)

The defined process should ensure that

a) Data required to understand the behavior, suitability and effectiveness of the defined process are identified based on the definitions of GP 3.1.5. Data about process performance may be qualitative or quantitative.
b) Data is collected and analyzed to understand the behavior, suitability and effectiveness of the defined process. Frequency and approach for collecting and analyzing data is defined on project and process level.
c) Results of the analysis are used to identify where continual improvement of the standard and/or defined process can be made. Results should be documented and made available to all affected parties.

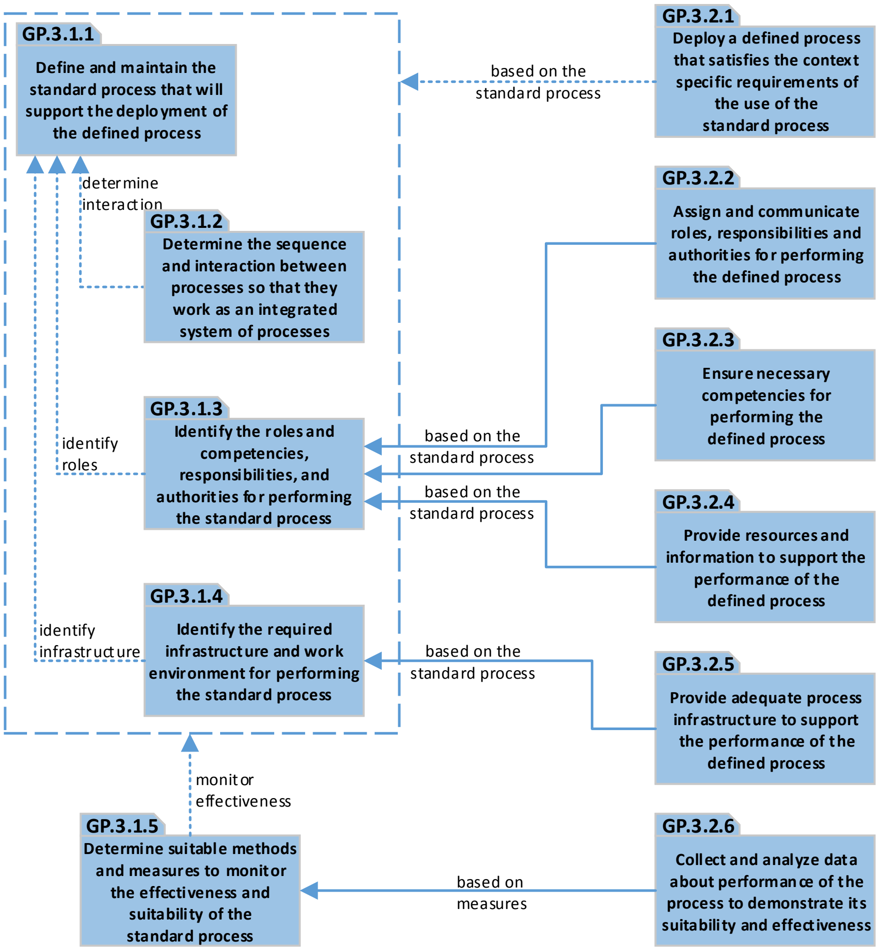
[CL3.RL.16] If the defined process does not ensure all aspects above, the indicator GP 3.2.6 must not be rated F.

[CL3.RC.2] If the collected and analyzed data are only qualitative, this should not be used to downrate the indicator GP 3.2.6.

5.3 Rating consistency

5.3.1 Rating consistency within capability level 3

The following figure shows relationships among generic level 3 practices:



GP 3.1.1 Define and maintain the standard process that will support the deployment of the defined process.

The following rating recommendation is related to the definition and maintenance of the standard process and thus covers several generic practices of level 3:

> **[CL3.RC.3]** If the defining and maintaining of the standard process (GP 3.1.1) is downrated due to an inadequate definition of the standard process, this should be in line with the rating of the indicator GP 3.1.2, GP 3.1.3 and GP 3.1.4, respectively.

GP 3.1.5 Determine suitable methods and measures to monitor the effectiveness and suitability of the standard process.

> **[CL3.RC.4]** The rating of the indicator GP.3.1.5 should be in line with the ratings of the standard process related GP (GP 3.1.1, GP 3.1.2, GP 3.1.3, GP 3.1.4).

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

> **[CL3.RC.5]** The rating of the indicator GP 3.2.1 should be in line with the ratings of the standard process related GP (GP 3.1.1, GP 3.1.2, GP 3.1.3, GP 3.1.4).

GP 3.2.2 Assign and communicate roles, responsibilities and, authorities for performing the defined process.

> **[CL3.RL.17]** If the indicator for identify the roles and competencies, responsibilities, and authorities (GP 3.1.3) is downrated due to missing or inadequate definitions of roles, responsibilities and authorities, the indicator GP 3.2.2 shall be downrated.

GP 3.2.3 Ensure necessary competencies for performing the defined process.

> **[CL3.RL.18]** If the indicator for identify the roles and competencies, responsibilities, and authorities (GP 3.1.3) is downrated due to missing or inadequate definitions of competencies, the indicator (GP 3.2.3) shall be downrated.

GP 3.2.4 Provide resources and information to support the performance of the defined process

> **[CL3.RL.19]** If the indicator for identify the roles and competencies, responsibilities, and authorities (GP 3.1.3) is downrated due to missing or inadequate definitions of roles, responsibilities and authorities, the indicator (GP 3.2.4) shall be downrated.

GP 3.2.5 Provide adequate process infrastructure to support the performance of the defined process.

> **[CL3.RL.20]** If the indicator for Identify the required infrastructure and work environment (GP 3.1.4) is downrated, the indicator (GP 3.2.5) shall be downrated.

GP 3.2.6 Collect and analyze data about performance of the process to demonstrate its suitability and effectiveness.

> **[CL3.RL.21]** If collecting and analyzing the defined data is not performed according to the defined methods and measures (GP 3.1.5), the indicators (GP 3.2.6) shall be downrated.

5.3.2 Rating consistency to other processes

Consistency to PIM.3 and ORG.1.A (ISO/IEC 15504-5:2012) is obvious but not described, because both are not part of the VDA Scope.

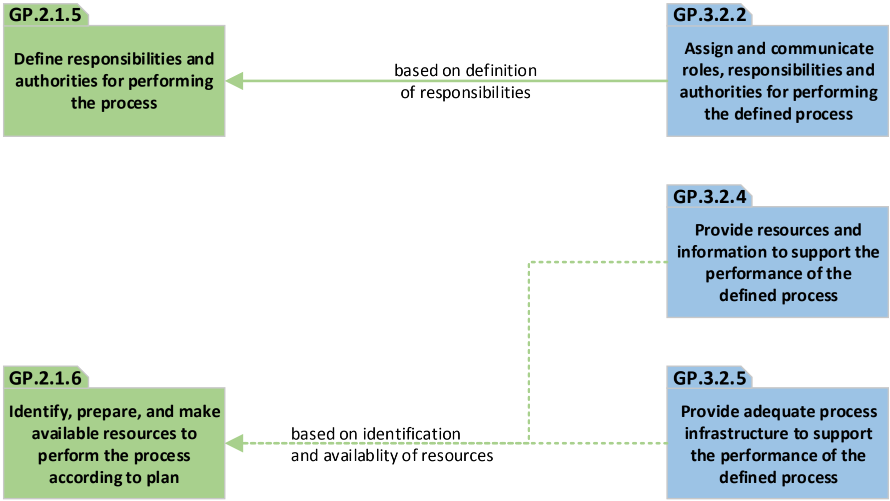
5.3.3 Dependencies between generic practices of capability level 2 and 3

Process attribute 3.1 Process definition is one of the few process attributes which does not have a dependency on the lower process attributes.

The rationale is that whether the lower process attributes are performed well or badly may or may not affect the definition of the standard process.

However, for a capability level 3 the standard process has to cover all aspects of capability level 1 and 2 and a feedback mechanism to regularly check and improve the standard process itself. Therefore, the rating of the process attribute PA 3.1 is relatively independent of the project.

If this standard process is used the following picture shows the dependencies to level 2:



Relationships exist between:

GP 3.2.2 Assign and communicate roles, responsibilities and authorities for performing the defined process

[CL3.RL.22] If the definition of responsibilities and authorities (GP 2.1.5) is downrated, the indicator GP 3.2.2 shall be downrated.

Rationale: If there is a weakness on 2.1.5 regarding definition of the responsibilities and authorities this weakness could be evident in two possible scenarios on level 3:

- *The weakness is also found in the GP 3.1.3, the identification of roles, competencies etc. which in turn would lead to a weakness in the project which is using this standard process. (GP 3.2.2.)*
- *The standard process is F regarding GP 3.1.3 but the project does not use the process properly because the weakness on GP 2.1.5 would not be evident if the standard process had been followed properly.*

GP 3.2.4 Provide resources and information to support the performance of the defined process

[CL3.RC.6] If the identification, preparation and availability of resources (GP 2.1.6) is downrated, due to human resources issues, this should be in line with the rating of indicator GP 3.2.4.

Rationale: If there is a weakness on 2.1.6 regarding identification and availability of the resources, especially human resources this weakness could be evident in two possible scenarios on level 3:

- *The weakness is also found in the GP 3.1.3, the identification of roles, competencies etc. which in turn would lead to a weakness in the project which is using this standard process. (GP 3.2.4.)*
- *The standard process is F regarding GP 3.1.3 but the project does not use the process properly because the weakness on GP 2.1.6 would not be evident if the standard process had been followed properly.*

GP 3.2.5 Provide adequate process infrastructure to support the performance of the defined process

[CL3.RC.7] If the identification, preparation and availability of resources (GP 2.1.6) is downrated due to infrastructure issues, this should be in line with the rating of the indicator GP 3.2.5.

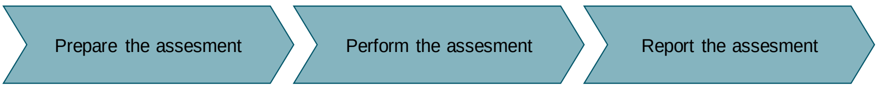
Rationale: If there is a weakness on 2.1.6 regarding identification and availability of the resources, especially technical resources this weakness could be evident in two possible scenarios on level 3:

- *The weakness is also found in the GP 3.1.4, the identification of infrastructure a work environment which in turn would lead to a weakness in the project which is using this standard process. (GP 3.2.5)*
- *The standard process is F regarding GP 3.1.4 but the project does not use the process properly because the weakness on GP 2.1.6 would not be evident if the standard process had been followed properly.*

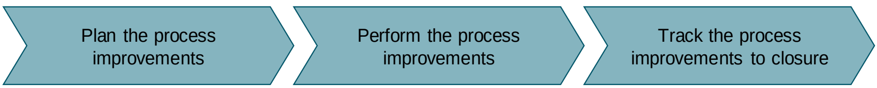
Part 2: Guidelines for performing the assessment

The purpose of the part two of the current publication is to support the assessors in performing an assessment based on the Automotive SPICE process reference and assessment model, considering the requirements of ISO/IEC 33002.

Chapter 6, "Documented assessment process" provides a necessary input for performing the assessment defined in ISO/IEC 33002. It provides the tasks and activities of the so-called evaluation phase, in which the assessment is planned, prepared, performed and documented.



In chapter 7, *"Improvement process"* an overview of the tasks and activities are given, in the case that the assessment results are to serve as an input for subsequent improvement measures. In this so-called improvement phase the assessment results of the evaluation phase are used to plan, execute and track the process improvement actions.



Chapter 8, *"Recommendations for performing an assessment"* provides additional requirements when applying the documented assessment process.

In chapter 9, *"Requirements relating to assessor qualification"* the requirements for assessors to demonstrate the competencies to conduct an assessment and to monitor and verify the conformance of a process assessment are given.

6 Documented assessment process

6.1 Introduction

This chapter provides a documented assessment process (DAP) according to ISO/IEC 33002, clause 4.1:

> *The assessment shall be conducted according to a documented assessment process. The documented assessment process shall be capable of meeting the assessment purpose and shall be structured in a manner that ensures that the purpose for performing the assessment is satisfied, in terms of the rigor and independence of the assessment and its suitability for the intended use.*

The documented assessment process provided was setup to serve the majority of assessments within the automotive domain. It fulfills the requirements of ISO/IEC 33002 under the following preconditions:

- The assessment is using the PRM and PAM Automotive SPICE 3.1 and subsequent versions.
- The assessment is using the process measurement framework defined in ISO/IEC 33020 "*Process measurement framework for assessment of process capability".*
- A defined rating and aggregation method according to ISO/IEC 33020 is used.
- The assessment is classified as "Class 3" according to ISO/IEC 33002 clause 4.6.
- The category of independence of the body performing the assessment, the lead assessor and the other members of the assessment team is A, B or C according to ISO/IEC 33020, Annex A.
- The assessment is not intended to evaluate organizational maturity.

It is the responsibility of the lead assessor to evaluate, whether the assessment provides the given preconditions. In case of deviations, the lead assessor shall take appropriate steps to modify this given DAP or select another suitable one. In this case the lead assessor takes responsibility for the conformity of the DAP to ISO/IEC 33002.

6.2 Assessment input and output

6.2.1 Assessment plan

According to ISO/IEC 33002 an assessment plan shall be setup. Within this DAP the assessment plan shall contain the following elements:

- Required inputs specified in this standard → 6.2.2
- Definition of the class of assessment and the category of independence of the body performing the assessment, the lead assessor and the other members of the assessment team → 6.1
- Communications to the personnel involved in the assessment → 6.3
- Identification of the documented assessment process including:
    - The strategy and techniques for the selection, identification, collection and analysis of objective evidence and data, to satisfy any requirements for coverage of the process scope of the assessment as defined for class 3 assessments → 6.4.1
    - The approach to derive an agreed process attribute rating, where relevant → 6.4.1 and Part 1
    - Activities to be performed in the assessment → 6.4
    - Resources and schedule assigned to these activities → 6.4
- Identification and definition of roles and responsibilities of the participants in the assessment → 6.3
- Criteria to verify that the requirements of ISO/IEC 33002 are met → 6.1
- Description of the planned assessment outputs → 8.4

6.2.2 Assessment inputs

According to ISO/IEC 33002 the necessary assessment input shall be identified. Within this DAP the necessary input shall contain as a minimum the following elements:

- Identity of the sponsor and the sponsor's relationship to the organizational unit(s) being assessed;
- Business context including the organization business's goals and circumstances of the assessment;
- Purpose of the assessment, e.g. process improvement or evaluation of the process capability assigned to a specific product delivery;

- Assessment scope as it applies to the business comprising a defined and declared organization scope, including:
  - The processes to be investigated within the assessment
  - The process quality characteristic to be investigated, including the highest process quality level for each individual process within the assessment scope
  - The organizational unit(s) that deploy the process
  - The boundaries of the assessed organization, including
    - the size of each organizational unit, e.g. number of personnel
    - the application domain of the products or services of each organizational unit and
    - key characteristics (e.g. size, criticality, complexity and quality) of the products or services of each organizational unit
  - The process context including the set of stakeholder requirements and changes which are under investigation
  - The process instances, which have been selected, if applicable

- Identity of the model(s) and process measurement framework used:
  - Automotive SPICE 3.1 or higher
  - ISO/IEC 33020

- Assessment requirements, including:
  - Reference to this documented assessment process
  - Definition of the class of assessment and the category of independence of the body performing the assessment the lead assessor and the other members of the assessment team
  - Rating method(s) to be employed
  - Aggregation method(s) to be employed
  - Assessment constraints considering, at minimum:
    - Availability of key resources
    - Maximum duration of the assessment
    - Specific processes or organizational units to be excluded from the assessment
  - Ownership of the assessment outputs and any restrictions on their use
  - Controls for handling confidential information and non-disclosure
  - Participants and their roles, the assessment team and assessment support staff with specific responsibilities for the assessment
  - Criteria for competence of the lead assessor

6.2.3 Assessment report

The requirements and recommendations for the assessment report are defined in detail in chapter 8.4.

6.2.4 Objective evidence gathered

For evaluating the processes within the assessment scope objective evidence and additional information shall be collected. Each evidence shall be traceable to associated assessment indicators (base practices, WP, generic practices, etc.).

6.3 Parties and roles involved in the assessment

The main parties involved in the assessment are the sponsor, the assessing organization and the assessed organization. The following roles shall be identified:

LAC:Local Assessment Coordinator

Individual or entity, who takes responsibility for the organization of the assessment within the organizational unit assessed.

SP: Sponsor

Individual or entity, internal or external to the organizational unit to be assessed, who requires the assessment to be performed, and provides financial or other resources to carry it out (see ISO/IEC 33001 clause 3.2.9).

AS: Assessor

Individual who participates in the rating of process attributes (see ISO/IEC 33001 clause 3.2.11). Assessors have appropriate education, training and both capability assessment experience and domain experience to perform the required class of assessment and make professional judgments (see ISO/IEC 33001 clause 3.2.11).

LA: Lead Assessor

Assessor who has demonstrated the competencies to conduct an assessment and to monitor and verify the conformity of a process assessment (see ISO/IEC 33001 clause 3.2.12).

PP: Participant

Individual from the organizational unit to be assessed, who takes part in the assessment.

Note: While the role definitions provided above are considered to represent the standard approach to responsibility distribution, it is possible that individual assessments may extend or reduce these role definitions as is appropriate for a given assessment. For example, the SP may be knowledgeable of process assessment and may therefore participate in the detailed aspects of the assessment. The LAC may also be capable of performing a greater role in the process assessment depending on their knowledge and training with respect to process assessment.

For the description of the responsibilities the following abbreviations are used:

R: Responsible

Those who do the work to achieve the task. There is at least one role with a participation type of responsible, although others can be delegated to assist in the work required (see also RACI below for separately identifying those who participate in a supporting role).

A: Accountable (also approver or final approving authority)

The one ultimately answerable for the correct and thorough completion of the deliverable or task, and the one who delegates the work to those responsible [7]. In other words, an accountable must sign off (approve) work that responsible provides. There must be only one accountable specified for each task or deliverable.

C: Consulted (sometimes counsel)

Those whose opinions are sought, typically subject matter experts; and with whom there is two-way communication.

I: Informed

Those who are kept up-to-date on progress, often only on completion of the task or deliverable; and with whom there is just one-way communication.

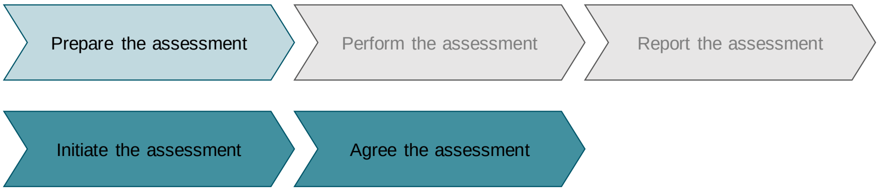
6.4 Assessment activities

The assessment process consists of three tasks:

- Prepare the assessment
- Perform the assessment
- Report the assessment

6.4.1 Prepare the assessment

The preparation for an assessment is split into two sub-tasks:



6.4.1.1 Initiate the assessment

In the initialization phase the assessing organization determines the need for an assessment and determines the framework conditions (scope, time period, team, etc.). All necessary information on the assessed organization is collected.

Brief description	The need for an assessment is determined and the framework conditions for its execution are established.
Process inputs	• Formal or informal assessment enquiry • Information about the organization assessed • Previous audit reports and assessment reports
Process outputs	• Assessment purpose • Assessment agreement • Assessment scope • Time frame • Contact persons in both organizations • Assessment team list • Assessment plan

Activities / Responsibilities	***LA***	***AS***	***SP***	***LAC***	***PP***
Determine the need for an assessment	-	-	A, R	-	-
Establish the assessment agreement	C	C	A, R	C	-
Define the assessment scope	C, R	I	A	C	-
Collecting and evaluating information on the organization assessed	A, R	C	-	C	-
Define the assessment team	A, R	C	-	C	-

Determine the need for an assessment

The need for an assessment must be determined by the sponsor. This may be derived based on different use cases and defines the purpose of the assessment. Examples for use cases are given in chapter 1.2.1. The purpose of the assessment is the base input for setting up the assessment scope.

Establish the assessment agreement

The assessment agreement is established based on the assessment purpose by

- defining the main focus of the assessment. This may be, for example, project management, engineering aspects or other areas of risk. If appropriate, a pre-selection should be made of the processes to be checked.
- By determining the assessing organization, which is responsible for performing the assessment,
- selecting the lead assessor and the assessment team members,
- defining the time-frame, within which the assessment should be carried out, and
- identifying the business divisions or departments and personnel in the organization assessed that are to be involved.

Define the assessment scope

The boundaries of the assessment, provided as part of the assessment input, encompassing

- the boundaries of the organizational unit for the assessment,
- the processes to be included,
- the capability level for each process to be assessed, and
- the context within which the processes operate

is defined.

Collecting and evaluating information on the organization assessed

The information on the organization assessed which is relevant to the assessment must be collected and evaluated. This may include:

- Organizational structure of all those involved in the project, such as
  - Sponsor,
  - Project team,
  - Core/platform development,
  - (independent) quality assurance department,
  - (independent) test department or
  - Sub-suppliers.

- Standard software components/of the shelf items.
- If appropriate, the department responsible for the selection, release and maintenance of tools or the IT department, for example for configuration management.
- Results of other audits and assessments.

  Note: Results from previous audits and assessments can be used for determining the assessment scope. Here, the time has to be considered that has passed since the audit or assessment and whether the results are applicable for the project (assessment method, assessed department, personnel involved).

Define the assessment team

The assessment team is determined and appointed.

6.4.1.2 Agree the assessment

The exact terms of the assessment are agreed between the involved parties.

Brief description	The assessment and its framework conditions are agreed.
Process inputs	• Assessment scope • Time frame • Assessment team list • Assessment plan
Process outputs	• Non-disclosure agreement (NDA) • Assessment time schedule • List of documents to be exchanged in advance • Requirements for the evidence repository • Distribution list for the report • Optional: minutes of the pre-assessment meeting

Activities \ Responsibilities	***LA***	***AS***	***SP***	***LAC***	***PP***
Agree the details of how the assessment shall be performed	R	I	A	C	-
Perform pre-assessment meeting (optional)	A, R	C	-	C	I

Agree the details of how the assessment shall be performed

With the assessment agreement, a consensus regarding the assessment should be achieved by defining details of how the assessment shall be performed and agree them between the parties.

It is essential that the sponsor, the assessing organization and the organization assessed agree on the modalities of the assessment. The agreement can be reached formally by means of a contract and acknowledgement, or in an informal manner. Furthermore, the assessment agreement must consider and specify the following points:

- A non-disclosure agreement (NDA) should be agreed by all parties involved (assessing organization, organization assessed and assessors) and signed (if not already done in the project).
- The final schedule is agreed.
- Contact persons are appointed on both sides for coordination.
- The distribution list for the report is established.
- Requirements relating to the evidence repository for the assessment are established.
- Requirements relating to the infrastructure, e.g. meeting rooms, beamers, printers, flipcharts etc. are established.
- Constraints for the scheduling, e.g. availability due to bank holidays, breaks, local conventions etc. are identified.

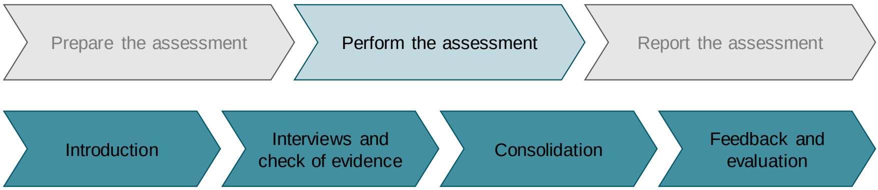
Perform pre-assessment meeting (optional)

If necessary, a pre-assessment meeting can be carried out (on-site, by email or by a telecommunications conference).The purpose is to

- explain the framework and process of the assessment to the personnel involved;
- specify the set of documents to be handed out to the assessment team in advance for study;
- to understand and confirm the assessment context; and
- to perform preliminary document analysis.

6.4.2 Perform the assessment

The execution of the assessment is split into four tasks:



In the introduction task the assessment scope, the project to be assessed and the assessment method are presented. This is followed by the interviews and document reviews, where the actual collection of evidence is done which is the crucial part of the assessment. Once the collection of evidence has been completed, the consolidation task starts and the first evaluation of the results (findings) takes place. Finally, in the feedback and evaluation task, the collected results are stored in the evidence repository, the preliminary process attribute rating results are presented and possible immediate actions are recommended.

6.4.2.1 Introduction

Brief description	The organization to be assessed, the project, the evaluation methodology and the activities of the assessment are presented.
Process inputs	• Information on the organization assessed and the project • Assessment scope • Assessment time schedule • Assessment plan
Process outputs	• Information of the organization assessed and project • Information on Automotive SPICE, the assessment scope and the assessment time schedule

Activities \ Responsibilities	***LA***	***AS***	***SP***	***LAC***	***PP***
Present the organization assessed and the project	I	I	-	A, R	C
Present the assessment activities	A, R	C	I	I	I

The introduction should give all those involved an overview of the organization assessed, the project, the assessment methodology and sequence.

Present the organization assessed and the project

The organization presents itself and the project in the scope to be assessed to the assessment team. The purpose of this activity is to provide the assessment team with an introduction to the project-specific conditions and circumstances.

Present the assessment activities

The assessment team presents the concrete activities of the Automotive SPICE assessment. The purpose of this activity is to inform the organization assessed and the interviewees about the detailed procedure which will be followed during the assessment (for example, the evidence repository).

6.4.2.2 Interviews and checks of evidence

Brief description	The project-related information regarding the selected processes is collected and documented in accordance with the assessment model.					
Process inputs	• Assessment time schedule • Project-related work products					
Process outputs	• Assessment notes regarding results of interviews, documents which have been examined and results of the inspection of the work environment • List of documents which have been examined					
Activities \ Responsibilities	***LA***	***AS***	***SP***	***LAC***	***PP***	
Perform interviews, document checks and inspections of the work environment, if appropriate	A, R	C	-	C	C	
Collect evidence for rating the processes	A, R	C	-	C	C	

Evidence which is relevant to the project in terms of the selected processes is collected and documented.

Perform interviews, document checks and inspections of the work environment, if appropriate

Based on the assessment time schedule, interviews on the individual processes with the key personnel of the organization assessed are carried out and the associated documents/evidence are examined. If necessary, the conditions under which the process is performed can be checked at the workplace.

The results of the interviews are documented in the assessment notes.

Collect evidence for rating the processes

The assessment team collects the evidences to justify and document the findings for the individual processes (for example, with regard to process compliance, the tools used in the project and the quality of existing documents).

6.4.2.3 Consolidation

<table>
<tr><th colspan="1">Brief description</th><td colspan="5">The selected processes are rated by the assessors on the basis of the available evidence.</td></tr>
<tr><th>Process inputs</th><td colspan="5">• Assessment notes</td></tr>
<tr><th>Process outputs</th><td colspan="5">• Consolidated assessment notes
• Provisional process capability profiles</td></tr>
<tr><th>Activities \ Responsibilities</th><th>LA</th><th>AS</th><th>SP</th><th>LAC</th><th>PP</th></tr>
<tr><td>Evaluate the collected evidence</td><td>A, R</td><td>C</td><td>-</td><td>-</td><td>-</td></tr>
<tr><td>Provide a provisional rating</td><td>A, R</td><td>C</td><td>-</td><td>-</td><td>-</td></tr>
<tr><td>Document strengths and potential improvements</td><td>A, R</td><td>C</td><td>-</td><td>I</td><td>I</td></tr>
<tr><td>Establish the traceability of process attribute rating to evidence</td><td>A, R</td><td>C</td><td>-</td><td>-</td><td>-</td></tr>
<tr><td>Document the deviation of rating rules</td><td>A, R</td><td>C</td><td>I</td><td>-</td><td>-</td></tr>
</table>

The evidence collected from interviews and document reviews is consolidated by the assessors.

Note: The consolidation might also be done incrementally after each interview session, see chapter 6.4.2.2.

Evaluate the collected evidence

Following the interviews and the document reviews the assessment team consolidates and documents the analysis results and reaches consensus on the identified strengths and potential improvements of the processes which have been assessed.

Provide a provisional rating

Based on the findings the process attributes are rated and a provisional set of process capability profiles is determined for the assessed processes. The rating is evaluated whether the rating is consistent with the rules and recommendations given in part one of this publication. The rating shall consider the rating rules and recommendations given in Part 1 of this document.

Document strengths and potential improvements

The findings are evaluated in terms of strengths and potential improvements.

Establish the traceability of process attribute rating to evidence

For each process attribute rating the traceability to the collected evidence used in determining that rating is established. The relationship between the assessment indicators for each process attribute rated and the objective evidence is documented.

Document the deviation of rating rules

- The rules not obeyed by the lead assessor are identified. A justification, why the rule is not applicable or why it has no significant impact on the process attribute rating, is provided.

  Note: The purpose of the justification is to document briefly the lead assessor's decision not following a specific rule. It is the clear intention of the authors of this publication not to generate additional effort due to extensive documentation of rule deviations. The provision of a list of all rules, no matter whether they are obeyed or not might make sense for unexperienced assessors and might give an overview, but is not required or intended by the authors of this publication.

6.4.2.4 Feedback and evaluation

<table>
<tr><td>Brief description</td><td colspan="5">A provisional evaluation of the organization assessed is presented and immediate actions are identified.</td></tr>
<tr><td>Process inputs</td><td colspan="5">• Provisional process capability profiles
• List of documents which have been examined
• Consolidated assessment notes</td></tr>
<tr><td>Process outputs</td><td colspan="5">• Provisional process attribute ratings and the process capability profiles
• List of the most important findings (strengths and potential improvements)
• Document archive related to the assessment
• List of immediate actions, if applicable</td></tr>
<tr><td>Activities \ Responsibilities</td><td>LA</td><td>AS</td><td>SP</td><td>LAC</td><td>PP</td></tr>
<tr><td>Present the results</td><td>A, R</td><td>C</td><td>I</td><td>I</td><td>I</td></tr>
<tr><td>Identify immediate actions (optional)</td><td>C</td><td>C</td><td>-</td><td>A, R</td><td>C</td></tr>
<tr><td>Store the evidence in the repository</td><td>I</td><td>-</td><td>-</td><td>A, R</td><td>I</td></tr>
</table>

The purpose of feedback is to provide information on the assessment results and to reach a common understanding of the rating.

The feedback shall contain the following as a minimum:

- The provisional process attribute ratings
- The provisional process capability profiles
- The major strengths and potential improvements
  (for each process assessed).

The feedback should be provided directly following the conclusion of all interviews. The contents of the feedback should be documented in writing as a feedback presentation and afterwards made available as a copy to the assessed party.

Present the results

The provisional process attribute ratings and the capability profiles are prepared and presented to the organization assessed. The most important findings (strengths and potential improvements) are presented.

Identify immediate actions (optional)

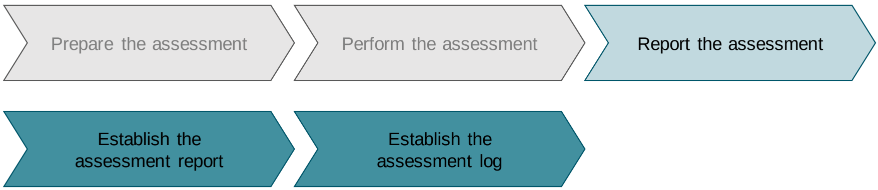
Based on the presented identified potential improvements, immediate actions are recommended to eliminate critical weaknesses.

Store the evidence in the repository

The organization assessed stores the evidence repository including references to the documents which have been analyzed.

6.4.3 Report the assessment

The elaboration and distribution of the report following an assessment is split into two tasks:



The detailed assessment report is drawn up in order to document the results of the assessment. The assessment log is drawn up for submission to the certification body.

6.4.3.1 Establish the assessment report

Brief description	The assessment team compiles the assessment report to be distributed in the assessed organization as defined.				
Process inputs	• Consolidated assessment notes • Provisional process capability profiles • List of the most important findings.				
Process outputs	• Assessment report with the process attribute ratings and the final process capability profiles • An explanation of deviations at the practice level				
Activities \ Responsibilities	***LA***	***AS***	***SP***	***LAC***	***PP***
Consolidate the final process attribute ratings and the final process capability profiles	A, R	C	-	-	-
Compile the assessment report	A, R	C	I	I	I
Distribute the assessment report	-	-	A, R	C	I

Consolidate the final process attribute ratings and the final process capability profiles

The set of final process capability profiles is drawn up. The consolidated findings and observations are documented in detail based on the assessment notes.

Compile the assessment report

The assessment report must be compiled, checked and released by the assessment team. The lead assessor is responsible for drawing up and releasing the assessment report. Deviations from rating rules given in Part 1 of this publication shall be documented in the assessment report. The assessment report is provided to the assessment sponsor for distribution in assessed organization. Please refer to chapter 8.4 for detailed requirements on the assessment report.

Distribute the assessment report

The released version is distributed within the assessed organization.

6.4.3.2 Establish the assessment log

Brief description	The assessment team draws up the assessment log.					
Process inputs	• Template for the assessment log					
Process outputs	• The assessment log					
Activities \ Responsibilities		***LA***	***AS***	***SP***	***LAC***	***PP***
Issue the assessment log		R	C	A	-	-

Issue the assessment log

The assessment log represents the confirmation of the sponsor, the LAC and the assessment team about the performance of the assessment according to the defined assessment process.

The assessment log shall be signed by the lead assessor and the assessment team members. The log shall be approved by the sponsor.

The assessment log shall be drawn up on the basis of the template provided by the certification scheme (see chapter9, *"Requirements relating to assessor qualification"*.

7 Improvement process

7.1 Introduction

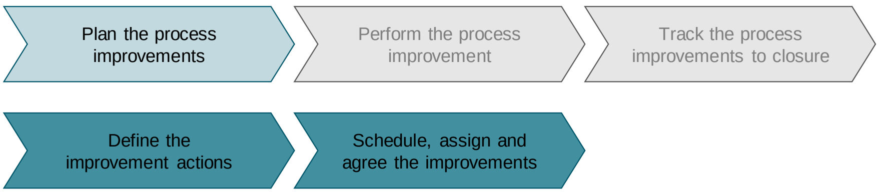
The process improvement phase may follow the evaluation phase and is split into the planning on the process improvement actions, into performing and into tracking these actions.

Since the improvement actions will be in general not be assigned to the roles involved in the evaluation phase, no assignment of responsibilities is given in this chapter.

7.2 Improvement activities

7.2.1 Plan the process improvements

In the agreement task, the process improvement actions are established, together with the monitoring criteria, responsibilities and the time schedule.



7.2.1.1 Define the improvement actions

Brief description	The process improvement actions to be carried out are selected and prioritized.
Process inputs	• Assessment report • List of immediate actions, if applicable
Process outputs	• List of process improvement actions • Monitoring criteria for process improvement actions
Activities	
Specify the process improvement actions	
Prioritize the process improvement actions	
Define the monitoring criteria	

Specify the process improvement actions

A list of process improvement actions is established including the desired improvement result based on the assessment report. A traceability to the identified assessment findings is provided, if applicable.

Prioritize the process improvement actions

Prioritization is performed based on an evaluation of the effectiveness of the improvement actions.

Define the monitoring criteria

Based on the list of process improvement actions monitoring criteria are defined which allow to check whether the implementation of the actions have the desired effects.

7.2.1.2 Schedule, assign and agree the improvements

Brief description	The improvements are scheduled, assigned and a commitment on the improvements is achieved.
Process inputs	• List of process improvement actions
Process outputs	• Responsibilities for process improvement actions • Time schedule for process improvement actions
Activities	
Define the responsibilities	
Define the time schedule for implementation	
Agree on the improvement actions	

Define the responsibilities

The improvement actions are assigned to persons who are responsible for their implementation.

Define the time schedule for implementation

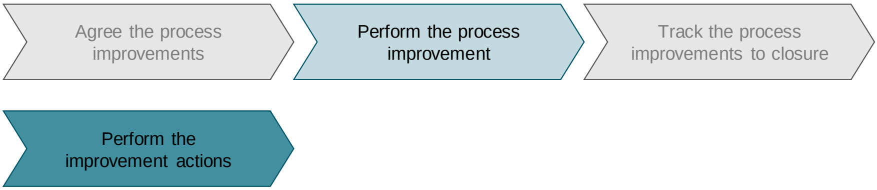
Dates and priorities are assigned to the individual process improvement actions. Based on a risk assessment, the actions from the list are identified which are to be implemented in the project and/or in the organization which has been assessed.

Agree on the improvement actions

An agreement on the improvements is achieved from all affected parties.

7.2.2 Perform the process improvements

Immediate actions should be carried out directly after the assessment. Other process improvement actions are implemented according to the defined schedule.



7.2.2.1 Performing process improvement actions

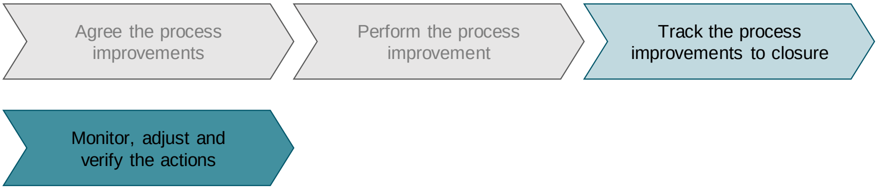
Brief description	The process improvement actions are carried out
Process inputs	• List of process improvement actions • Responsibilities for process improvement actions • Time schedule for process improvement actions.
Process outputs	• Documentation of the improvements which have been carried out
Activities	
Execute the process improvement actions	

Execute the process improvement actions

The process improvement actions should be carried out in due time by those responsible and according to priority.

7.2.3 Track the process improvement to closure

Tracking the process improvement actions represents the completion of the improvement process:



The process improvement actions are monitored and any necessary adjustments are made, taking risks into account.

7.2.3.1 Monitor, adjust and verify the actions

Brief description	The actions are monitored and adjusted if necessary
Process inputs	• List of process improvement actions • Monitoring criteria for process improvement actions • Documentation of the improvements which have been carried out
Process outputs	• Status report of the process improvement actions • Road map for long-term actions exceeding the project scope
Activities	
Monitor the process improvement actions	
Modify improvement actions if deficiencies are detected	
Verify and close improvement actions	
Plan long-term actions exceeding the project scope	

Monitor the process improvement actions

Based on the defined monitoring criteria the process improvement actions are checked regularly regarding their implementation and effectiveness.

Modify improvement actions if deficiencies are detected

If the actions do not achieve the desired effect, modified or new actions are specified.

Verify and close improvement actions

The improvement actions are closed, if they achieved their purpose.

Plan long-term actions exceeding the project scope

Long-term actions exceeding the project scope should be addressed within a road map.

8 Recommendations for performing an assessment

In the current chapter recommendations are provided, which should be considered when following the documented assessment process specified in chapter 6.

8.1 Assessment results

8.1.1 Confidentiality of information

As a fundamental rule, assessment results and the information obtained in the course of an assessment must be treated as confidential by all persons and organizations involved.

8.1.2 Handling the assessment results

The ownership of the assessment results is defined in the initial assessment agreement (see 6.4.1.1); by default the Sponsor is the owner of the results.

If the assessment results are issued to third parties, an additional non-disclosure agreement should be signed where appropriate.

The assessment results and any relevant part of them should be made available to all individuals involved in the assessed project and individuals involved in the performance and monitoring of the improvement actions. The criterion here is their involvement in the project or process development.

The assessment results should be documented and archived by the assessing organization.

8.2 Validity of assessments

8.2.1 Area of validity of the assessment results

Automotive SPICE is predominantly used to assess single projects based on a given scope. In these assessments the focus is always on one particular project. Neither the complete set of all projects in an organization nor a statistically significant selection is investigated. It follows therefore that assessment results are a representative sample of the process capability within the scope of the assessment, but not applicable in general to the assessed organization as whole, the development location or the entire company.

The assessment results may be considered to reflect potential capability of another project with identical characteristics. Here the following criteria should be considered:

- Development locations: As a general rule, assessment results are not transferable from one location to another.
- ECU AUTOSAR domains: If at a large development location ECUs are developed for various AUTOSAR domains, such as powertrain, chassis or body, assessment results are transferable only to a limited degree, given the different development environments.
- Distributed development: Where the development work on ECUs is distributed over several departments or several locations, the assessment results apply only to those locations or departments which have been assessed.

The degree to which assessment results may be transferred will depend on various factors, including the process capability level and must be examined in each individual case.

8.2.2 Period of validity of assessment results

Assessment results have only a limited validity in terms of time. Experience has shown that they allow reliable conclusions to be drawn for 12 months regarding the project which has been assessed.

Changes within the project, such as, for example

- the transfer of the development work to a different location,
- a re-organization in the organization which has been assessed or
- changes to the development processes

can, however, affect significantly the relevance of the assessment results to individual processes even within 12 months. Such changes may cause the actual capability of the development process to be better or worse than indicated by the last assessment result.

On the other hand, where there is a high degree of project stability, the assessment results may permit reliable conclusions regarding the project to be drawn for longer than 12 months. For these reasons, the period of validity must always be considered relative to the specific project circumstances.

8.3 Performing an assessment

The following recommendations should be observed when performing assessments:

8.3.1 Assessment scheduling

When planning the assessment, at a minimum the following conditions should be considered:

- The scope of the assessment, specifically the number of assessed processes, the number of process instances and the highest assessed level
- The process context as defined in chapter 1.2.3.
- The complexity of the assessed project, e.g. in terms of distributed developments, size of the assessment scope, complexity of the developed product
- Results and experiences from previous assessments
- Assessment experience of the assessed party
- Problems associated with different cultures and languages

Based on this sufficient interview and consolidation time frames should be planned.

There should be at least four weeks between agreement on an assessment and its execution.

It is not appropriate to perform interviews for data collection only using phone and/or video conferences.

8.3.2 Individuals involved in the assessment

The assessing organization performing the assessment decides on the composition of the assessment team in agreement with the sponsor.

Participation by observers or other guests in interviews:

- In principle, observers can be present at an interview – e.g., observers from the process development department.
- The number of people taking part in the interview should be kept as small as possible.

- The interviews must not be impaired by observers, whether active or passive.
- The lead assessor decides whether observers may be present at the interviews and can exclude observers (in general or particular individuals) even during the course of the assessment.

8.3.3 Composition of the assessment team

The interviews in the assessment should be carried out by at least two assessors.

The independence of the assessors should be ensured in order to avoid any conflict of interest.

The lead assessor has the final authority for the selection of the assessor(s).

8.4 Assessment report

In the assessment report the organization which has been assessed is given a more detailed feedback of the strengths and potential improvements detected in the assessment. The assessment report should document in particular those points which led to a downrating of the process attribute by referencing to the individual base or generic practices.

The assessment report should contain the following information:

8.4.1 General information

This chapter contains general information on the assessment report.

Item	Required information
Unique identifier	• Document/Version number or equal
Date of issue	• Issue date of the report
Version	• Version identification of the report
Issuer	• Issuer of the report
Change history	• Document change history

8.4.2 Formal information about the assessment

This chapter contains formal information about the assessment.

Item	Required information
Assessment model	• Assessment model and version that has been used (e.g. Automotive SPICE PAM V3.x)
Assessment period	• The period during which the assessment was carried out
Sponsor	• Name of the assessment sponsor
Local assessment coordinator	• Name of the responsible coordinator of the assessed organization
Evidence	• The work products examined for each process.
Distribution list	• Distribution list of the report
Assessment class	• Class of the assessment according to ISO/IEC 33002

8.4.3 Scope of the assessment

This chapter contains information about the assessment scope. Refer also to chapter 1.2.2, *"Defining the assessment scope"*.

Item	Required information
Process scope	• Selection of processes in the assessment • In case of derivation of the VDA scope: A rationale for the selection of the processes
Capability level	• Target capability level for each process assessed
Assessed project	• Project Name / description
Organization	• Company name • Organizational / Business unit • Assessed sites • Assessed Departments
Process context	• Identification of the set of stakeholder requirements considered for the assessment • Identification of the set of changes considered for the assessment *Note: It is sufficient to identify the sets by suitable criteria, please refer to chapter 1.2.3, "Defining the process context in the assessment scope"*

8.4.4 Participants of the assessment

This chapter contains information about the assessment team, the interview persons and other participants of the assessment.

Item	Required information
Lead assessor	• Name of the lead assessor • Lead assessor grade (e.g. Competent, Principal) • License number of Lead Assessor • Expiry date of the assessor license
Assessors	• Name of the Assessor(s) • Assessor(s) grade (e.g. Provisional, Competent, Principal) • License number of Assessor(s) licenses • Expiry date of the assessor license
Local assessment coordinator	• Name of local assessment coordinator
Interviewed persons	• Names of interviewed individuals incl. • their role in the project • mapping to the processes for that they have been interviewed (Project manager e.g. could be interviewed for more than one process)
Guests (optional)	• Names of persons attending the assessment without having one of the defined roles, e.g. observers, assessor candidates… *Note: To gather experience assessor candidates may participate in the process attribute rating, but should not be involved in the rating decision.*

8.4.5 Constraints

This chapter contains information about constraints that have to be considered to understand the assessment results.

Item	Required information
Constraints (if applicable)	e.g. • Somebody was not available (e.g. off, sick) • Separated development areas have been included via Video/WebEx (no on-site assessment) • Disclaimer (e.g. that the assessment results does not allow conclusions to the complete organization or other departments of the organization that has been not assessed) • Confidentiality constraints, e.g. access to evidence or to infrastructure and sites may be subject to legal access rights.

8.4.6 Overview about the assessment results

This chapter contains the process capability and attribute profile, general strengths and weaknesses and general immediate actions.

Item	Required information
Set of process capability profiles	• Determined capability level and process attribute ratings of each process
Findings	• Findings related to process attributes not rated as “fully”.
General strengths and weaknesses	• List of general strengths and weaknesses
General immediate actions (if applicable)	• Proposals for general immediate actions using prioritization, if applicable

9 Requirements relating to assessor qualification

It is essential that Automotive SPICE assessments are conducted by appropriate and trained specialists. The lead assessor entrusted with the leadership of the assessment, who also accepts responsibility for the result of the evaluation, plays a special role.

The training of assessors shall be carried out by registered training organizations on the basis of a published certification scheme.

The personal certification of assessors shall be carried out by a certification body on the basis of a published certification scheme. In this, conformance with ISO/IEC 17024 is a fundamental requirement for acceptance as an Automotive SPICE training scheme.

The certification scheme shall cover the guidance, the rules and the recommendations given within this publication.

Acceptance of valid qualification schemes for assessors is carried out by the quality management board of the VDA QMC. Currently, the intacs scheme is a valid and accepted qualification scheme.

9.1 Requirements for lead assessors

According the definitions provided in ISO/IEC 33001, clause 3.2.12, the term “lead assessor” is defined as:

> *Assessor who has demonstrated the competencies to conduct an assessment and to monitor and verify the conformance of a process assessment.*

A valid personal Automotive SPICE Competent or Principal SPICE Assessor license issued by the VDA QMC is required as evidence for the qualification and experience of the lead assessor.

9.2 Requirements for non-lead assessors

According the definitions provided in ISO/IEC 33001, clause 3.2.11, the term “assessor” is defined as:

individual who participates in the rating of process attributes

A valid personal Automotive SPICE Provisional, Competent or Principal SPICE Assessor license issued by the VDA QMC is required as evidence for the qualification and experience of any other assessor who is member of the assessment team.

ANNEX: Documentation of the assessment scope

One major root cause for diverging assessment results of the same project has been identified as differing assessment scopes. I.e. in case of equivalent assessment scope definitions with same process context, assessment results should be equal.

The process context (see chapter 1.2.3 *"Defining the process context in the assessment scope*") defines the boundaries in which the processes operate in terms of a set of stakeholder requirements and a set of change requests. E.g. if a software development project has to be assessed, the process context defines which software requirements are within and which software requirements are out of the assessment scope. Especially if legacy, platform or third-party software is part of the process context, the assessor has to define together with the sponsor, how these elements will be assessed:

a) Assess only the management of interfaces to the platform/legacy and/or third-party software. I.e. the development of platform/legacy or third-party software is not assessed.
   For the platform/legacy software RL/RCs of chapter 2.2.5 *"Management of platform and legacy software"* and for third-party software the ACQ processes together with the RL/RCs of chapter 2.2.4 *"Third-party software"* are used.

b) Assess development processes (SWE.1, SWE.2, etc.) of platform/legacy software and/or third-party software, applying RL/RCs for SWE processes.
   For the platform/legacy software the RL/RCs of chapter 2.2.5 are not used. For the third-party software the ACQ process are not in the assessment scope and the RL/RCs of chapter 2.2.4 are not used.

In case no platform and legacy software are assigned to software requirements and then platform and legacy software are not part of the process context, chapter 2.2.5 is not applicable, analog for third-party software.

It is highly recommended to document within the process context which legacy/platform software or third-party software will be assessed by assessing the interfaces and by assessing the development processes.

Chapter 8.4.3 requires that the assessment scope including the assessment content and boundaries has to be documented in the assessment report. The table below gives an example for documenting the assessment scope in the assessment report covering the major assessment scope elements on one page.

<table>
<tr><td>PAM name and version</td><td colspan="3">e.g. Automotive SPICE 3.1</td></tr>
<tr><td>VDA guideline version</td><td colspan="3">e.g. 1st edition 2017</td></tr>
<tr><td>Company name / organizational unit</td><td colspan="3">Name(s) of the assessed companies and the assessed organizational unit(s)</td></tr>
<tr><td>Project name</td><td colspan="3">Name(s) of the assessed project(s)</td></tr>
<tr><td>Locations</td><td colspan="3">e.g. Santa Barbara, Berlin, St. Tropez</td></tr>
<tr><td colspan="4">Assessment purpose</td></tr>
<tr><td colspan="4">e.g.
Starting point for process improvement, process improvement progress check, supplier evaluation, process related risk determination</td></tr>
<tr><td>Assessed processes</td><td colspan="3">e.g. VDA scope including MAN.5</td></tr>
<tr><td>Target capability level</td><td colspan="3">e.g. Level 3</td></tr>
<tr><td>Assessment class</td><td>1/2/3</td><td>Category of independence</td><td>A/B/C/D</td></tr>
<tr><td colspan="4">Process context</td></tr>
<tr><td>Process context category</td><td colspan="3">A (Part of product/delivery) or
B (Entire product/delivery)</td></tr>
<tr><td colspan="4">e.g.
A subset of stakeholder requirements valid for a specific product release or
All stakeholder requirements valid for a specific product release or
All changes between two defined project milestones or
All software requirements implemented by changed processes.</td></tr>
<tr><td colspan="4">Application of chapter 2.2: Assessing specific application environments</td></tr>
<tr><td>2.2.1 Model based development</td><td>YES/NO</td><td>2.2.4 Management of third party software</td><td>YES/NO</td></tr>
<tr><td>2.2.2 Agile environments</td><td>YES/NO</td><td>2.2.5 Management of platform and legacy software</td><td>YES/NO</td></tr>
<tr><td>2.2.3 Distributed development</td><td>YES/NO</td><td>2.2.6 Application parameters</td><td>YES/NO</td></tr>
</table>

The elements of the scope table above based on the following definitions in

- chapter 1.2.2 *“Defining the assessment scope”*,
- chapter 1.2.3 *“Defining the process context in the assessment scope”*,
- chapter 2.2 *“Assessing specific application environments”* and
- ISO/IEC 33002 for assessment class and category of independence.

If one process is applied in two or more process instances within the same project, it is recommended to document the process instances in a second table containing all instances of assessed process as defined in chapter 1.2.4 *“Defining instances when setting up the assessment scope”*.

In case all processes have exactly one process instance with the same capability level, no separate instance table is required.

Process	**Location 1**		**Location 2**		**Location 3**
	Instances of (sub) project A	Instances of (sub) project B	Instances of (sub) project C		Instances of (sub) project D
MAN.3	Overall project - CL2	SW project - CL1	Platform - CL3		---
SWE.1	ASILD requirements - CL2	Security requirements - CL3	OEM requirements - CL1	Platform requirements - CL2	Requirements for release 42.3 - CL3
SWE.4	Hand coded - CL2	Model based - CL3			Hand coded - CL2
SUP.8	Overall project - CL2	---	---	---	---
SUP.9	Overall project - CL2	---	---	---	Sub project - CL2
SUP.10	Overall project - CL2	---	---	---	Sub project - CL2

The following examples for process context category A and B show use cases of assessment requests with different assessment scopes and give guidance how the assessment scope and the process instances can be documented in an assessment report.

Process context category A “Part of product / delivery”

First example: “Process improvement for project and organization”

Within the software development project new features and changes are developed on top of an existing system or software which was developed in previous projects.

Product consists of

- development of the current project
- legacy software from a former or platform projects
- third-party software

The assessment purpose is to improve processes up to level 3 of the current development project which is done in one location. The sponsor does not want that legacy and platform software is included in the assessment scope but the management of third-party software.

The project doesn’t use model based, agile or distributed development or application parameters.

No instances are present when assessing the project.

PAM name and version	Automotive SPICE 3.1		
VDA guideline version	1st edition 2017		
Company Name / organizational unit	TIERX AG		
Project Name	Product enhancement		
Locations	One location of TIERX AG		
Assessment Purpose			
Starting point for process improvement for the project and the organization.			
Assessed processes	MAN.3, SWE.1-6, SUP.1, SUP.8-10, ACQ.4		
Target capability level	Level 3		
Assessment class	3	**Category of independence**	A
Process context			
Process context category	A (Part of product/delivery)		
All changes and affected stakeholder requirements in the delta project developing additional functionalities based on the existing software. All third-party used in the project. Legacy and platform software is out of the scope of the assessment.			
Application of chapter 2.2: Assessing specific application environments			
2.2.1 Model based development	NO	**2.2.4** Management of third party software	YES
2.2.2 Agile environments	NO	**2.2.5** Management of platform and legacy software	NO
2.2.3 Distributed development	NO	**2.2.6** Application parameters	NO

Second example: "Analysis of supplier's potential"

The sponsor wants to identify the capability of processes and the performing project team for an upcoming software project.

The capability should be determined by choosing former projects in two locations in which the project team was involved using the same or very similar development processes as in the upcoming project:

- TIERX project with former customer excluding legacy software in location 1.
- TIERX platform project including management of interfaces to legacy software and third-party software in location 1.
- TIERX previous sub project excluding legacy, platform software, third-party software is included in location 2.

The selected processes should be assessed up to level 2.

The management of the interfaces to legacy, platform and third-party software is part of the assessment scope. MAN.5 "Risk Management" is included in the assessment scope.

None of the projects did use model based, agile or distributed development or application parameters.

Setting of instances:

Process	TIERX location 1		TIERX location 2
	Instances of TIERX projects with former customer	Instances of TIERX platform project	Instances of TIERX previous sub-project
MAN.3	Project management of former customer project - CL2	Project management of platform project - CL2	Project management of software sub-project - CL2
MAN.5	- CL2	---	---
ACQ.4	- CL2	---	---
SWE.1-2	- CL2	- CL2	---
SWE.3-4	- CL2	- CL2	- CL2
SWE.5-6	- CL2	---	---
SUP.1	- CL2	- CL2	- CL2
SUP.8	- CL2	---	---

<table>
<tr><td>PAM name and version</td><td colspan="3">Automotive SPICE 3.1</td></tr>
<tr><td>VDA guideline version</td><td colspan="3">1st edition 2017</td></tr>
<tr><td>Company name / organizational unit</td><td colspan="3">TIERX AG</td></tr>
<tr><td>Project name</td><td colspan="3">Project A (former customer)
Project B (platform development)
Project C (sub-project)</td></tr>
<tr><td>Locations</td><td colspan="3">Location 1: Project A and B
Location 2: Project C</td></tr>
<tr><th colspan="4">Assessment purpose</th></tr>
<tr><td colspan="4">Supplier evaluation („Analysis of supplier’s potential“)</td></tr>
<tr><td>Assessed processes</td><td colspan="3">MAN.3, MAN.5, SWE.1-6, SUP.1, SUP.8-10, ACQ.4</td></tr>
<tr><td>Target capability level</td><td colspan="3">Level 2</td></tr>
<tr><td>Assessment class</td><td>3</td><td>Category of independence</td><td>A</td></tr>
<tr><th colspan="4">Process context</th></tr>
<tr><td>Process context category</td><td colspan="3">A (Part of product/delivery)</td></tr>
<tr><td colspan="4">All changes and affected stakeholder requirements of TIERX project with former customer excluding legacy software.
All changes and affected stakeholder requirements of TIERX platform project including the management of legacy and third-party software.
All changes and affected stakeholder requirements of TIERX previous sub-project excluding legacy, platform and third-party software.</td></tr>
<tr><th colspan="4">Application of chapter 2.2: Assessing specific application environments</th></tr>
<tr><td>2.2.1 Model based development</td><td>NO</td><td>2.2.4 Management of third-party software</td><td>YES</td></tr>
<tr><td>2.2.2 Agile environments</td><td>NO</td><td>2.2.5 Management of platform and legacy software</td><td>YES</td></tr>
<tr><td>2.2.3 Distributed development</td><td>NO</td><td>2.2.6 Application parameters</td><td>NO</td></tr>
</table>

Process Context Category B “Entire Product / Delivery”

First example: “Entire product with managed legacy and platform software”

The complete product software development valid for a specific product release should be assessed up to level 2. The aim is to identify possible process-related product risks and to be the starting point for process improvement.

The product consists of

- the software developed in the current project,
- legacy and platform software from former projects and
- third-party software.

The management of the interfaces to legacy, platform and third-party software is part of the assessment scope.

The project uses model based development and an agile approach. Distributed development and application parameters are not used.

Note: The process context category B has to be chosen in this example because all stakeholder requirements and change requests are within the process context. The project realization by choosing legacy, platform and third-party software is an architecture decision to fulfill the stakeholder requirements.

PAM name and version	Automotive SPICE 3.1		
VDA guideline version	1st edition 2017		
Company name / organizational unit	TIERX AG		
Project name	OEM project		
Locations	One location		
Assessment purpose			
Determine process-related risks for the quality of the product. Set a starting point for process improvement to reduce the identified risks.			
Assessed processes	MAN.3, SWE.1-6, SUP.1, SUP.8-10, ACQ.4		
Target capability level	Level 2		
Assessment class	3	**Category of independence**	A
Process context			
Process context category	B (Entire product/delivery)		
All changes and affected stakeholder requirements valid for the recent release to the custom-er. The management of platform, interface and third-party software.			
Application of chapter 2.2: Assessing specific application environments			
2.2.1 Model based development	YES	**2.2.4** Management of third-party software	YES
2.2.2 Agile environments	YES	**2.2.5** Management of platform and legacy software	YES
2.2.3 Distributed development	NO	**2.2.6** Application parameters	NO

Second example: "Meet customer quality requirements for entire product"

The complete product system development, valid for a specific product release, is assessed up to level 2 to identify possible process-related product risks. The development is located at one location and no third-party software is included.

The product consists of

- the development of current project,
- the development of legacy software from a former project and
- the development of a former platform project.

The sponsor wants to identify whether the complete product which was developed by the organization satisfies all stakeholder requirement. 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, i.e. platform and legacy software rules of chapter 2.2.5 will not be used ("Platform and Legacy Software" is marked with no).

The projects don't use model based, agile or distributed development and application parameters.

Setting of instances:

Process	**TIERX location 1**		
	Instances of TIERX new developed OEM project	Instances of legacy software	Instances of platform project
MAN.3	Project management of OEM project - CL2	Project management of legacy project - CL2	Project management of platform project - CL2
ACQ.4	- CL2	---	---
SYS.2-5	- CL2	---	---
SWE.1-4	- CL2	- CL2	- CL2
SWE.5-6	- CL2	---	---
SUP.1	- CL2	- CL2	- CL2
SUP.8	- CL2	---	- CL2
SUP.9-10	- CL2	- CL2	- CL2

PAM name and version	Automotive SPICE 3.1		
VDA guideline version	1st edition 2017		
Company name / organizational unit	TIERX AG		
Project name	OEM project		
Locations	One location		
Assessment purpose			
Determine process-related risks for the quality of the product.			
Assessed processes	VDA scope		
Target capability level	Level 2		
Assessment class	3	**Category of independence**	A
Process context			
Process context category	B (Entire product/delivery)		
All changes and affected stakeholder requirements valid for the recent release to the customer. This includes the legacy and platform software developed.			
Application of chapter 2.2: Assessing specific application environments			
2.2.1 Model based development	NO	**2.2.4** Management of third-party software	NO
2.2.2 Agile environments	NO	**2.2.5** Management of platform and legacy software	NO
2.2.3 Distributed development	NO	**2.2.6** Application parameters	NO

Bibliography

[ISO12207] ISO/IEC 12207:2008, Systems and software engineering — Software life cycle processes, 2008-02

[ISO33001] ISO/IEC 33001:2015, Information technology — Process assessment — Concepts and terminology, 2015-03-01

[ISO33020] ISO/IEC 33020:2015, Information technology — Process assessment — Process measurement framework for assessment of process capability, 2015-03-01

[ISO24765] ISO/IEC/IEEE 24765:2010, Systems and software engineering — Vocabulary, 2010-12

[ISO19011] ISO 19011:2011, Guidelines for auditing management systems

[Metz2016] Dr. Pierre Metz, Automotive SPICE — Capability level 2 und 3 in der Praxis, August 2016, dpunkt.verlag, ISBN 978-3-86490-360-1

[IntAgile] Frank Besemer, Dr. Pierre Metz, Joachim Pfeffer, Intacs white paper, Clarifying Myths with Process Maturity Models vs. Agile, Aug 6th 2014, www.intacs.info

[intacs] International Assessor Certification Scheme, www.intacs.info

[AS31] Automotive SPICE® Process Reference Model, Process Assessment Model, Version 3.1, 2017-09-06, www.automotivespice.com

[Oxford] Oxford Dictionaries, Oxford University Press (OUP), University of Oxford, https://www.oxforddictionaries.com

Quality management in the automotive industry

The current position regarding VDA publications covering quality management in the automotive industry (QAI) is shown in the Internet under http://www.vda-qmc.de.

NOTE: The English edition of the process assessment model Automotive SPICE can be obtained free of charge via http://www.automotivespice.com.

Available from:

Verband der Automobilindustrie e. V. (VDA)
Quality Management Centre (QMC)
10117 Berlin, Behrenstraße 35
Germany

Tel.: +49 (0) 30 897842-0. Fax: +49 (0) 30 897842-605
Email: info@vda-qmc.de, Internet: www.vda-qmc.de

VDA | QMC

Qualitäts Management Center
im Verband der Automobilindustrie