ASPICE PAM 4.0

.. all changes in one view

Part I:

SWE.1 - SWE.6

Seitz & Reinhart Consulting

A.SPICE.DE

SWE.1

ASPICE PAM 3.1

Outcomes

1) the software requirements to be allocated to the software elements of the system and their interfaces are defined; (BP.1)

2) software requirements are categorized and analyzed for correctness and verifiability; (BP.2, BP.3, BP.5)

3) the impact of software requirements on the operating environment is analyzed; (BP.4)

4) prioritization for implementing the software requirements is defined; (BP.2)

5) the software requirements are updated as needed; (BP.1)

6) consistency and bidirectional traceability are established between system requirements and software requirements; and consistency and bidirectional traceability are established between system architectural design and software requirements; (BP.6, BP.7)

7) the software requirements are evaluated for cost, schedule and technical impact; (BP.1; BP.3, BP.4, BP.5)

8) the software requirements are agreed and communicated to all affected parties. (BP.8)

ASPICE PAM 4.0

Outcomes

1) Software requirements are specified. (BP.1)

2) Software requirements are structured and prioritized. (BP.2)

3) Software requirements are analyzed for correctness and technical feasibility. (BP.3)

4) The impact of software requirements on the operating environment is analyzed. (BP.4)

5) Consistency and bidirectional traceability are established between software requirements and system requirements. (BP.5)

6) Consistency and bidirectional traceability are established between software requirements and system architecture. (BP.5)

7) The software requirements are agreed and communicated to all affected parties. (BP61)

1 Outcome 1 & 5 merges into Outcome 1

2 Outcome 2 & 4 merges into Outcome 2

3 Outcome 6 splits into Outcome 5 & 6

SWE.1

ASPICE PAM 3.1

Base Practices

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

ASPICE PAM 4.0

Base Practices

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

1 ASPICE 4.0 points into public standards where requirements characteristics are well established. (i.e. IEEE29148, ISO 26262, etc.), rather than defining additional characteristics. Hence verification criteria are implicitly expected in specified software requirements. (SWE.1.BP.1. Note 1&2)

2 Ensure consistency and traceability is merged together in all process areas

SWE.2

ASPICE PAM 3.1

Outcomes

1) a software architectural design is defined that identifies the elements of the software; (BP.1, BP.6, BP.8)

2) the software requirements are allocated to the elements of the software; (BP.2, BP.6, BP.8)

3) the interfaces of each software element are defined; (BP.3, BP.6)

4) the dynamic behavior and resource consumption objectives of the software elements are defined; (BP.4, BP.5, BP.6)

5) consistency and bidirectional traceability are established between software requirements and software architectural design; (BP.6, BP.7, BP.8)

6) the software architectural design is agreed and communicated to all affected parties; (BP.8, BP.9)

ASPICE PAM 4.0

Outcomes

1) A software architecture is designed including static and dynamic aspects. (BP.1, BP2)

2) The software architecture is analyzed against defined criteria. (BP.3)

3) Consistency and bidirectional traceability are established between software architecture and software requirements. (BP.4)

4) The software architecture is agreed and communicated to all affected parties. (corresponding outcome). (BP.5)

1 SW architecture at a glance. ASPICE4.0 moves the architectural elements, interfaces, as well as static and dynamic behavior (*Outcome 1, 3 & 4*) into one outcome

2 Outcome 2 & 5 merges into Outcome 3. Consistency includes the relation to requirements

SWE.2

ASPICE PAM 3.1

Base Practices

- BP1: Develop software architectural design.
- BP2: Allocate software requirements.
- BP3: Define interfaces of software elements.
- BP4: Describe dynamic behavior.
- BP5: Define resource consumption objectives.
- BP6: Evaluate alternative software architectures.
- BP7: Establish bidirectional traceability.
- BP8: Ensure consistency.
- BP9: Communicate agreed software architectural design.

ASPICE PAM 4.0

Base Practices

- **BP1: Specify static aspects of the software architecture.**
- **BP2: Specify dynamic aspects of the software architecture.**
- **BP3: Analyze software architecture.**
- **BP4: Ensure consistency and establish bidirectional traceability.**
- BP5: Communicate agreed software architecture.

1. In ASPICE 4.0 common elements of static and dynamic design are not separated by individual BP's anymore. Hence the number of BP's has reduced.
2. Comparing AS.3.1 BP.6 with AS.4.0 BP.3 indicates a higher relevance of cooperation between design decisions and project management objectives. ***(see also VDA Guideline chapter 3.9.2.1)***
3. Consistent and transparent allocation of software requirements and architecture has merged into one base practice.

SWE.3

ASPICE PAM 3.1

Outcomes

1) a detailed design is developed that describes software units; (BP.1, BP.4)

2) interfaces of each software unit are defined; (BP.2, BP.4)

3) the dynamic behavior of the software units is defined; (BP.3, BP.4)

4) consistency and bidirectional traceability are established between software requirements and software units; and consistency and bidirectional traceability are established between software architectural design and software detailed design; and consistency and bidirectional traceability are established between software detailed design and software units; (BP.4, BP.5, BP.6)

5) the software detailed design and the relationship to the software architectural design is agreed and communicated to all affected parties; (BP.7)

6) software units defined by the software detailed design are produced. (BP.8)

ASPICE PAM 4.0

Outcomes

1) A detailed design is specified including static and dynamic aspects. (BP.1, BP2)

2) Software units as specified in the software detailed design are produced. (BP.3)

3) Consistency and bidirectional traceability are established between software detailed design and software architecture; and consistency and bidirectional traceability are established between source code and software detailed design; and consistency and bidirectional traceability are established between the software detailed design and the software requirements. (BP.4)

4) The source code and the agreed software detailed design are communicated to all affected parties. (BP.5)

1 SW detailed design at a glance. ASPICE4.0 moves the detailed design, interfaces, as well as all static and dynamic behavior (*Outcome 1, 2 & 3*) into one outcome

SWE.3

ASPICE PAM 3.1

Base Practices

- BP1: Develop software detailed design.
- BP2: Define interfaces of software units.
- BP3: Describe dynamic behavior.
- BP4: Evaluate software detailed design.
- BP5: Establish bidirectional traceability.
- BP6: Ensure consistency.
- BP7: Communicate agreed software detailed design.
- BP8: Develop software units.

ASPICE PAM 4.0

Base Practices

- **BP1: Specify the static aspects of the detailed design.**
- BP2: Specify dynamic aspects of the detailed design.
- BP3: Develop software units.
- **BP4: Ensure consistency and establish bidirectional traceability.**
- BP.5: Communicate agreed software detailed design and developed software units.

1 Base practices BP.1 and BP.2 are defined consistent to SWE.2. Hence the number of base practices has been removed.

2 Consistency and transparency are merged into one base practice systematically.

SWE.4

ASPICE PAM 3.1

Outcomes

1) a software unit verification strategy including regression strategy is developed to verify the software units; (BP.1)

2) criteria for software unit verification are developed according to the software unit verification strategy that are suitable to provide evidence for compliance of the software units with the software detailed design and with the non-functional software requirements; (BP.2)

3) software units are verified according to the software unit verification strategy and the defined criteria for software unit verification and the results are recorded; (BP.3, BP.4)

4) consistency and bidirectional traceability are established between software units, criteria for verification and verification results; (BP.5, BP.6)

5) results of the unit verification are summarized and communicated to all affected parties; (BP.5)

ASPICE PAM 4.0

Outcomes

1) Verification measures for software unit verification are specified. (BP.1)

2) Software unit verification measures are selected according to the release scope, including criteria for regression verification. (BP.2)

3) Software units are verified using the selected verification measures, and results are recorded. (BP.3)

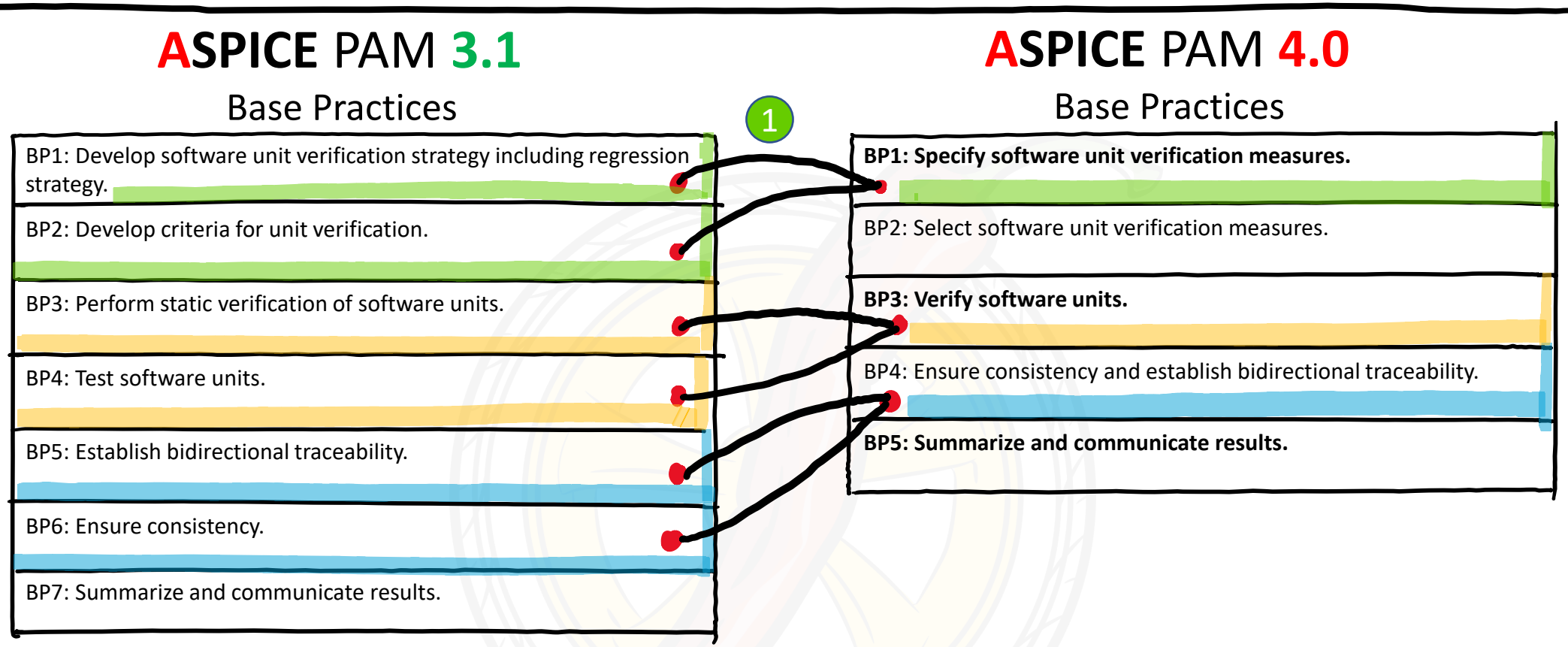
4) Consistency and bidirectional traceability are established between verification measures and software units; and bidirectional traceability is established between verification results and verification measures. (BP.4)

5) Results of the software unit verification are summarized and communicated to all affected parties. (BP.5)

1

1 The structure looks very similar between PAM 3.1 and 4.0, **but** on level 1, ASPICE4.0 does not explicitly mention the term "**strategy**" for the right side of the V-model anymore. A deeper look into the notes shows, that "**specify measures**" still requires a kind of "**verification-tactic**" to ensure meaningful verification measures and characteristics for a release.

SWE.4



1 ASPICE4.0 does not explicitly mention the term "**strategy**" on the right side of the V-model anymore. A deeper look into the notes shows, that "**specify measures**" still requires a kind of documented "**verification-tactic**" to ensure meaningful verification measures and characteristics for a release.

1 The term "**regression**" did not disappear but is an implicit criteria for the selection of the appropriate verification measures.

SWE.5

ASPICE PAM 3.1

Outcomes

1) a software integration strategy consistent with the project plan, release plan and the software architectural design is developed to integrate the software items; (BP.1)

2) a software integration test strategy including the regression test strategy is developed to test the software unit and software item interactions; (BP.2)

3) a specification for software integration test according to the software integration test strategy is developed that is suitable 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; (BP.3)

4) software units and software items are integrated up to a complete integrated software according to the integration strategy; (BP.4)

5) Test cases included in the software integration test specification are selected according to the software integration test strategy, and the release plan; (BP.5)

6) integrated software items are tested using the selected test cases and the results of software integration test are recorded; (BP.6)

7) consistency and bidirectional traceability are established between the elements of the software architectural design and the test cases included in the software integration test specification and between test cases and test results; (BP.7, BP.8)

8) results of the software integration test are summarized and communicated to all affected parties. (BP.9)

ASPICE PAM 4.0

Outcomes

1) Verification measures are specified for software integration verification of the integrated software elements based on the software architecture and detailed design, including the interfaces of, and interactions between, the software components. (BP.1)

2) Verification measures for software components are specified to provide evidence for compliance of the software components with the software components' behavior and interfaces. (BP.2)

3) Software elements are integrated up to a complete integrated software. (BP.4)

4) Verification measures are selected according to the release scope considering criteria, including criteria for regression verification. (BP.3)

5) Software components are verified using the selected verification measures, and the results of the integration verification are recorded. (BP.5)

6) Integrated software elements are verified using the selected verification measures, and the results of the integration verification are recorded. (BP.4)

7) Consistency and bidirectional traceability are established between verification measures and the software architecture and detailed design; and bidirectional traceability is established between verification results and verification measures. (BP.6)

8) The results of software component verification and software elements integration verification are summarized and communicated to all affected parties. (BP.7)

1 Same as in SWE.4, the term "**strategy**" disappeared. Anyhow, a certain "**verification-tactic**" is still useful to ensure meaningful verification measures and characteristics according to ASPICE PAM 4.0.

2 Introduction of the term "Software Components", in order to enable a more context driven rating of the integration procedure.
(***recommendation to read VDA Guideline chapter 2.4***)

SWE.5

ASPICE PAM 3.1

Base Practices

BP1: Develop software integration strategy.

BP2: Develop software integration test strategy including regression test strategy.

BP3: Develop specification for software integration test.

BP4: Integrate software units and software items.

BP5: Select test cases.

BP6: Perform software integration test.

BP7: Establish bidirectional traceability.

BP8: Ensure consistency.

BP9: Summarize and communicate results.

ASPICE PAM 4.0

Base Practices

BP1: Specify software integration verification measures.

BP2: Specify verification measures for verifying software component behavior.

BP3: Select verification measures.

BP4: Integrate software elements and perform integration verification.

BP5: Perform software component verification.

BP6: Ensure consistency and establish bidirectional traceability.

BP7: Summarize and communicate results.

1 The circumstance that the term "**strategy**" disappeared from the level 1 terminology shall not be misunderstood in the way to skip any kind of strategy. Same as in SWE.4, meaningful verification measures and characteristics are still required.

A **tactic** of how to achieve these measures and characteristics must still be documented.

SWE.6

ASPICE PAM 3.1

Outcomes

1) a software qualification test strategy including regression test strategy consistent with the project plan and release plan is developed to test the integrated software; (BP.1)

2) a specification for software qualification test of the integrated software according to the software qualification test strategy is developed that is suitable to provide evidence for compliance with the software requirements; (BP.2)

3) test cases included in the software qualification test specification are selected according to the software qualification test strategy and the release plan; (BP.3)

4) the integrated software is tested using the selected test cases and the results of software qualification test are recorded; (BP.4)

5) consistency and bidirectional traceability are established between software requirements and software qualification test specification including test cases and between test cases and test results; (BP.5, BP.6)

6) results of the software qualification test are summarized and communicated to all affected parties; (BP.7)

1

ASPICE PAM 4.0

Outcomes

1) Verification measures are specified for software verification of the software based on the software requirements. (BP.1)

2) Verification measures are selected according to the release scope considering criteria, including criteria for regression verification. (BP.2)

3) The integrated software is verified using the selected verification measures and the results of software verification are recorded. (BP.3)

4) Consistency and bidirectional traceability are established between verification measures and software requirements; and bidirectional traceability is established between verification results and verification measures. (BP.4)

5) Results of the software verification are summarized and communicated to all affected parties. (BP.5)

1 Same as in SWE.4 and SWE.5, the term “**strategy**” disappeared. Anyhow, a certain “**verification-tactic**” is still useful to ensure meaningful verification measures and characteristics according to ASPICE PAM 4.0.

SWE.6

ASPICE PAM 3.1

Base Practices

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

ASPICE PAM 4.0

Base Practices

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

1 The circumstance that the term "**strategy**" disappeared from the level 1 terminology shall not be misunderstood in the way to skip any kind of strategy. Same as in SWE.4 and SWE.5 meaningful verification measures and characteristics are still required.

A **tactic** of how to achieve these measures and characteristics must still be documented.

Conclusion:

- A lot of simplifications and a novelty for SWE.5.
  - No more use of the term “strategy” on the right side of the V-model but the expectations towards the doing didn’t really change.
  - Less BP’s: Elements which are common because of other ISO norms and standards are not represented by individual BP’s anymore. (e.g. characteristics of requirements)
  - Better structure: almost 1:1 mapping of outcomes to BP’s.

Seitz & Reinhart Consulting

A.SPICE.DE