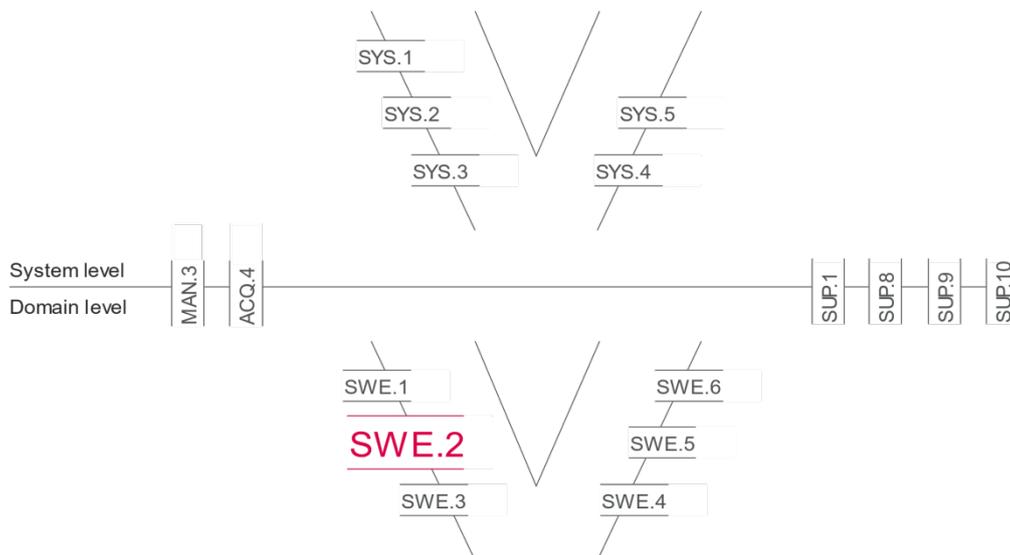


SOFTWARE ARCHITECTURAL DESIGN – SWE.2 in Automotive SPICE®

an introduction

Dr. Bhaskar Vanamali



Learn Automotive SPICE® with Kugler Maag Cie GmbH

→ 1st version – January 2021



Competence from A to Z: Training courses and workshops

Whether you need training to gain a certificate or an individual workshop, we can set up and provide the training you need, anything from courses for experts to exclusively in-house training at your place of business. Workshops or training courses? With us, you'll find it a lot easier!



Who to contact?

Smilja Mateja
Training Coordination
+49 7154 1796 229

information@kuglermaag.com

About this white paper

This white paper extends the materials I've covered in my Automotive SPICE[®] tutorial for Beginners on YouTube.

<https://www.youtube.com/qA-D10IWv38>

Both the YouTube tutorial and this document cover the core concepts and they are not complete by any means. This publication has been prepared for general guidance only. Please do not act according to any information given in this document without receiving specific professional consultancy. The publisher, KUGLER MAAG CIE GmbH, shall not be liable for any damages resulting from any use of the information contained in this report.

If you want to learn everything Automotive SPICE[®] has to offer and become an Automotive SPICE[®] expert, check out our ASPICE Training:

<https://www.kuglermaag.com/training/automotive-spice.html>



Competence from A to Z: Training courses and workshops

Whether you need training to gain a certificate or an individual workshop, we can set up and provide the training you need, anything from courses for experts to exclusively in-house training at your place of business. Workshops or training courses? With us, you'll find it a lot easier!



Who to contact?

Smilja Mateja
Training Coordination
+49 7154 1796 229

information@kuglermaag.com

About the Author



Bhaskar Vanamali is Principal and Partner at Kugler Maag Cie GmbH. He has been working on process improvement for nearly 20 years and was secretary of the working group 13 of VDA QMC.

He is Principal Assessor and Trainer for Automotive SPICE[®], and a co-author of books. He has performed more than 140 assessments and trained more than 250 assessors.

Due to his background, he is trying to shed light on new approaches from different perspectives. He is actually a veterinary by profession but is working in IT for 24 years.



Competence from A to Z: Training courses and workshops

Whether you need training to gain a certificate or an individual workshop, we can set up and provide the training you need, anything from courses for experts to exclusively in-house training at your place of business.

Workshops or training courses? With us, you'll find it a lot easier!



Who to contact?

Smilja Mateja
Training Coordination
+49 7154 1796 229

information@kuglermaag.com

Short summary of the YouTube tutorial

<https://www.youtube.com/qA-D10IWv38>

<https://www.kuglermaag.com/swe2>

The Software Architectural Design process in Automotive SPICE[®] (also known as SWE.2) helps your organization structure and document the internal logic of the software product.

What is the goal of the Software architecture? The expectation is that you already have Software requirements, which describe what the software shall do. The purpose of the software architecture is to define how the functionality documented in the software requirements is going to be implemented. In short, the requirements describe the “what”, the architecture the “how”.

A lot of organizations and projects have problems understanding how to document the architecture and which elements are required.

Three aspects of the Software Architecture:

Aspect One: **Architectural views**

Often, the architecture comprises of a physical view, a block diagram, of the software only. Especially in complex projects which applies to most projects nowadays, this is not enough. Surely you want a hierarchical breakdown of the software which demonstrates and explains how the



Competence from A to Z: Training courses and workshops

Whether you need training to gain a certificate or an individual workshop, we can set up and provide the training you need, anything from courses for experts to exclusively in-house training at your place of business. Workshops or training courses? With us, you'll find it a lot easier!



Who to contact?

Smilja Mateja
Training Coordination
+49 7154 1796 229

information@kuglermaag.com

functionality and non-functional requirements are going to be implemented in the different components and sub-components.

Other views are dynamic views, specific functional views which show a break-down of a specific feature, state-flow diagrams, interfaces and so on.

Typically, the more complex a system, the more different views are required.

As the different views have to be kept consistent an appropriate UML- or SysML-tool should be used. The tool will support consistency checks.

Aspect two: **Interfaces**

A pitfall often encountered in assessments is the lack of detailed description of the interfaces. Expected content of interface documentation is:

- Name
- Type
- Unit
- Resolution
- Range
- Default-value
- Etc.



Competence from A to Z: Training courses and workshops

Whether you need training to gain a certificate or an individual workshop, we can set up and provide the training you need, anything from courses for experts to exclusively in-house training at your place of business.

Workshops or training courses? With us, you'll find it a lot easier!



Who to contact?

Smilja Mateja
Training Coordination
+49 7154 1796 229

information@kuglermaag.com

Without this information, correct testing of the interfaces in the integration test is impossible. Again, describing these interfaces in an appropriate UML- or SysML-tool will support consistency between the different views.

Supplementing the definition in the system requirements analysis SYS.2, software-specific interfaces between the SW components are considered here in terms of interprocess communication mechanisms and bus communication mechanisms.

Aspect three: **Traceability**

This process also requires that you ensure traceability between your Software architecture and the Software requirements.

Normally, there is a tool break between the requirements and the architecture which makes the traceability difficult.

The purpose of traceability is that it

- a) supports consistency checks, i.e. checking the completeness and accuracy of the coverage of Software requirements.
- b) supports the impact assessment in case of change requests or bugs.
- c) supports the report of stakeholder expectations and identifies whether the requirements have been implemented in the architecture.



Competence from A to Z: Training courses and workshops

Whether you need training to gain a certificate or an individual workshop, we can set up and provide the training you need, anything from courses for experts to exclusively in-house training at your place of business. Workshops or training courses? With us, you'll find it a lot easier!



Who to contact?

Smilja Mateja
Training Coordination
+49 7154 1796 229

information@kuglermaag.com

Software Architectural Design – the process according to Automotive SPICE®

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.

BP1: Develop software architectural design. Develop and document the software architectural design that specifies the elements of the software with respect to functional and non-functional software requirements.

NOTE 1: The software is decomposed into elements across appropriate hierarchical levels down to the software components (the lowest level elements of the software architectural design) that are described in the detailed design.

BP2: Allocate software requirements. Allocate the software requirements to the elements of the software architectural design.

BP3: Define interfaces of software elements. Identify, develop and document the interfaces of each software element.

BP4: Describe dynamic behavior. Evaluate and document the timing and dynamic interaction of software elements to meet the required dynamic behavior of the system.



Competence from A to Z: Training courses and workshops

Whether you need training to gain a certificate or an individual workshop, we can set up and provide the training you need, anything from courses for experts to exclusively in-house training at your place of business. Workshops or training courses? With us, you'll find it a lot easier!



Who to contact?

Smilja Mateja
Training Coordination
+49 7154 1796 229

information@kuglermaag.com

NOTE 2: Dynamic behavior is determined by operating modes (e.g. start-up, shutdown, normal mode, calibration, diagnosis, etc.), processes and process intercommunication, tasks, threads, time slices, interrupts, etc.

NOTE 3: During evaluation of the dynamic behavior the target platform and potential loads on the target should be considered.

BP5: Define resource consumption objectives. Determine and document the resource consumption objectives for all relevant elements of the software architectural design on the appropriate hierarchical level.

NOTE 4: Resource consumption is typically determined for resources like Memory (ROM, RAM, external / internal EEPROM or Data Flash), CPU load, etc.

BP6: Evaluate alternative software architectures. Define evaluation criteria for the architecture. Evaluate alternative software architectures according to the defined criteria. Record the rationale for the chosen software architecture.

NOTE 5: Evaluation criteria may include quality characteristics (modularity, maintainability, expandability, scalability, reliability, security realization and usability) and results of make-buy-reuse analysis.

BP7: Establish bidirectional traceability. Establish bidirectional traceability between software requirements and elements of the software architectural design.



Competence from A to Z: Training courses and workshops

Whether you need training to gain a certificate or an individual workshop, we can set up and provide the training you need, anything from courses for experts to exclusively in-house training at your place of business. Workshops or training courses? With us, you'll find it a lot easier!



Who to contact?

Smilja Mateja
Training Coordination
+49 7154 1796 229

information@kuglermaag.com

NOTE 6: Bidirectional traceability covers allocation of software requirements to the elements of the software architectural design.

NOTE 7: Bidirectional traceability supports coverage, consistency and impact analysis.

BP8: Ensure consistency. Ensure consistency between software requirements and the software architectural design.

NOTE 8: Consistency is supported by bidirectional traceability and can be demonstrated by review records.

BP9: Communicate agreed software architectural design.

Communicate the agreed software architectural design and updates to software architectural design to all relevant parties.

Output Work Products: Software architectural design, Communication record, Review record, Traceability record, Interface requirement specification



Competence from A to Z: Training courses and workshops

Whether you need training to gain a certificate or an individual workshop, we can set up and provide the training you need, anything from courses for experts to exclusively in-house training at your place of business. Workshops or training courses? With us, you'll find it a lot easier!



Who to contact?

Smilja Mateja
Training Coordination
+49 7154 1796 229

information@kuglermaag.com

Advanced tutorial about Software Architectural Design

What is the benefit of Software Architectural Design?

It is impossible to develop automotive software without defining an architecture first. It provides a systematic structuring of the software into its elements which is consistent with the requirements. It also defines the dynamic behavior of the software and the resource consumption of its elements. The quality of the architecture highly impacts the efficiency, maintainability, testability, reusability, and the maintenance cost of the product.

The architecture is also a must to satisfy the requirements of the ISO 26262 standard. Safety requirements need to be traceable to software elements. This is the basis for the safety concept (required at project start) and the safety case (required at the end of the project).

What is the content of the Software Architectural Design process?

- Based on the functional and non-functional software requirements (and safety requirements, if applicable) the software architecture is developed. (BP1, BP2) State of the art is:
 - a physical breakdown into elements and down to the components (BP1)



Competence from A to Z: Training courses and workshops

Whether you need training to gain a certificate or an individual workshop, we can set up and provide the training you need, anything from courses for experts to exclusively in-house training at your place of business. Workshops or training courses? With us, you'll find it a lot easier!



Who to contact?

Smilja Mateja
Training Coordination
+49 7154 1796 229

information@kuglermaag.com

- modeling of the dynamic behavior (BP4)
- optionally the modeling of the functionalities (particularly if a completely new product is developed)
- Alternative architectures are considered (BP6) based on factors such as maintainability, expandability, scalability, reliability, security, and usability. Rationales for architecture decisions are documented.
- The interfaces and the dynamic behavior (BP3, BP4) are specified, as well as resource consumption targets (e.g., regarding ROM, RAM, EEPROM, flash memory, CPU load, etc.) for the architecture elements (BP5).
- Bidirectional traceability and consistency between software requirements and architecture elements is established (BP7, BP8). Traceability ensures that one knows which requirements shall be implemented in which elements and vice versa. Traceability is a prerequisite for consistency. Consistency means that the links between requirements and architecture elements are correct and complete.
- The architecture is communicated to all relevant parties (BP9).

Experiences, problems and hints:

- Software architecture design is typically supported by design tools. There is also a trend towards automatic code generation from the design models created within the design tool.
- State of the art tools support the structural decomposition into elements, dynamic modeling (e.g., task sequence diagrams) and



Competence from A to Z: Training courses and workshops

Whether you need training to gain a certificate or an individual workshop, we can set up and provide the training you need, anything from courses for experts to exclusively in-house training at your place of business. Workshops or training courses? With us, you'll find it a lot easier!

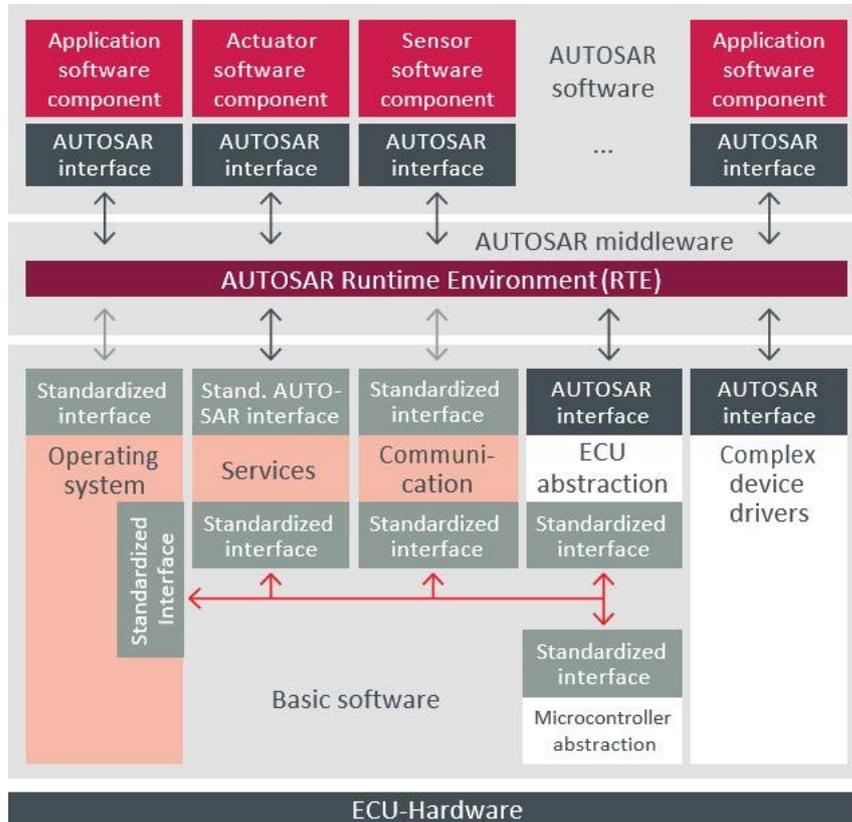


Who to contact?

Smilja Mateja
Training Coordination
+49 7154 1796 229

information@kuglermaag.com

functional modeling. UML, SysML and XML tools typically provide consistency checks between different representations of the architecture thereby avoiding manual effort.



Example of a software architecture

- Functional modeling is relatively new in automotive. It took many projects huge efforts which seemed to be not always justified, especially when the functionality of existing legacy software was modeled.



Competence from A to Z: Training courses and workshops
 Whether you need training to gain a certificate or an individual workshop, we can set up and provide the training you need, anything from courses for experts to exclusively in-house training at your place of business.
 Workshops or training courses? With us, you'll find it a lot easier!



Who to contact?
 Smilja Mateja
 Training Coordination
 +49 7154 1796 229

information@kuglermaag.com

- A significant goal of the architecture is to provide simplified views into the software system. It is meant as an abstraction, hiding details so that a higher order understanding of the software pieces and their purposes can be described and understood easily.
- When organizations build platforms, they often intend it to be a basis architecture for which a customer/application team can then tune aspects to their specific needs. This cannot be successfully achieved unless both parties, the platform and customer/application team both understand and maintain a common understanding of the underlying software architecture.
- Simply put, the architecture is often too complicated, and the architect unable to express it concisely and simply. This impacts the whole project negatively, as the team is unable to work from a common foundation of understanding.
- Considering alternative architectures is new to this Automotive SPICE version. This is a great tool when developing completely new software. Probably 99% of automotive projects, however, implement changes to existing software often without modifying the architecture. Providing evidence of alternative architectures can be challenging for this reason. However, configuring the platform in a structured way can be considered as an evaluation of alternative architectures. If the configuration is done by means of application parameters, for instance, alternative parameter combinations can be analyzed and the rationale for choosing one particular set can be documented.



Competence from A to Z: Training courses and workshops

Whether you need training to gain a certificate or an individual workshop, we can set up and provide the training you need, anything from courses for experts to exclusively in-house training at your place of business. Workshops or training courses? With us, you'll find it a lot easier!



Who to contact?

Smilja Mateja
Training Coordination
+49 7154 1796 229

information@kuglermaag.com

- If the architecture was a carry-over from another project or a platform one can provide their documentation of alternative design decisions.
- While traceability has improved a lot in the last years, consistency is still troublesome. The root causes are that consistency has not been understood well for a long time and that it requires substantial review effort.



Competence from A to Z: Training courses and workshops

Whether you need training to gain a certificate or an individual workshop, we can set up and provide the training you need, anything from courses for experts to exclusively in-house training at your place of business. Workshops or training courses? With us, you'll find it a lot easier!



Who to contact?

Smilja Mateja
Training Coordination
+49 7154 1796 229

information@kuglermaag.com

Want to become an Automotive SPICE® expert?

If you're serious about learning Automotive SPICE® I highly encourage you to participate to one of our Automotive SPICE® Courses.

Visit the following page:

<https://www.kuglermaag.com/training/automotive-spice.html>



Competence from A to Z: Training courses and workshops

Whether you need training to gain a certificate or an individual workshop, we can set up and provide the training you need, anything from courses for experts to exclusively in-house training at your place of business. Workshops or training courses? With us, you'll find it a lot easier!



Who to contact?

Smilja Mateja
Training Coordination
+49 7154 1796 229

information@kuglermaag.com