Improving Systems Engineering capabilities with Automotive SPICE and PREEvision

Dr.-Ing. Oliver Plan, Vector Consulting Services
PREEvision User Day, Stuttgart, 20.03.2019
Motivation

"Improving Systems Engineering capabilities with Automotive SPICE and PREEvision"

Automotive SPICE (ASPICE)
- Process capabilities correlate with product maturity
- ASPICE Level 3 is de-facto standard in Automotive Engineering
- OEMs and Tier-1 require ASPICE by their supplier
- ASPICE practices can be used for process improvement

Systems Engineering (SE)
- High complexity and distributed functions in today’s automotive customer features
- SE provides methods and systematic approach to design complex features
- Trend: Document driven -> Model Based Systems Engineering (MBSE)
- Trend: Plan driven -> AGILE Methods

How can the use of PREEvision the implementation of Automotive SPICE and Systems Engineering?
What is SPICE?

**SPICE:** Software Process Improvement and Capability dEtermination

- Maturity model for system and software development
- Includes industrial “best practices” for ensuring a mature, systematic and well-documented system and software development

- SPICE is typically applied in two scenarios:

  - Review of process maturity within the company and at suppliers
  - Capability determination
  - Guideline for in-house process optimization
  - Process improvement
Use Automotive SPICE for Process Improvement

Example: SYS.3 System architectural design (Level 1 only)

BP 1: Develop system architectural design
BP 2: Allocate System Requirements
BP 3: Define interfaces of system elements
BP 4: Describe dynamic behavior
BP 5: Evaluate alternative system architectures
BP 6: Establish bidirectional traceability
BP 7: Ensure consistency
BP 8: Communicate agreed system architectural design […] to all relevant parties
Use Automotive SPICE for Capability Determination

- Level 3 Established
  - PA.3.1 Process Definition
  - PA.3.2 Process Deployment
  - A set of standard processes exists for the organization
  - The standard processes are applied in the projects
  - Feedback to improve standard processes is collected

- Level 2 Managed
  - PA.2.1 Performance Management
  - PA.2.2 Work Product Management
  - Processes are planned and tracked, responsibilities are defined, resources are provided
  - Results are identified, reviewed (QA) and under configuration management
  - Implement base practices, create work products
  - Process outcomes are achieved and results are created (somehow)

- Level 1 Performed
  - PA.1.1 Process Performance
  - Process outcomes are not or only partially achieved

- Level 0 Incomplete
  - Process outcomes are not or only partially achieved
Automotive SPICE and Systems Engineering use similar Methods and Concepts

Example: Methods of abstraction, e.g. system levels, modularity,

- Requirements and allocated architecture on each level
- “Every requirement is a part of someone’s design” (S. Halligan)
- Supported by Model Driven Systems Engineering (i.e. Functional analysis using SysML)

How can we implement those concepts in projects using PREEvision?
System Requirements Engineering

SYS.2 System reqts analysis (Level 1)
BP 1: Specify system requirements
BP 2: Structure system requirements
BP 3: Analyze system requirements
BP 4: Analyze the impact on the operating environment
BP 5: Develop verification criteria
BP 6: Establish bidirectional traceability
BP 7: Ensure consistency
BP 8: Communicate agreed system requirements

- **Benefits**
  - Definition, analysis and alignment is supported by attributes, mappings to other model artefacts in the EE backbone
  - Process is supported by metrics and workflows
  - All needed information can be stored in the model without tool-barriers
**SYS.3 System architect. design (Level 1)**

BP 1: Develop system architectural design
BP 2: Allocate System Requirements
BP 3: Define interfaces of system elements
BP 4: Describe dynamic behavior
BP 5: Evaluate alternative system architectures
BP 6: Establish bidirectional traceability
BP 7: Ensure consistency
BP 8: Communicate agreed system architectural design [...] to all relevant parties

---

**Benefits**

- Consistency of requirements and architecture can be easily applied and managed using mappings
- Requirements, architecture and interfaces can be consistently reused and updated as needed
SW Engineering and Detailed Design

SWE.1 SW reqts analysis (Level 1)
SWE.2 SW architectural design (Level 1)

Same methodology as on System level
Base practices mostly identical

SWE.3 SW detailed design and unit construction (Level 1)
BP 1: Develop software detailed design
BP 2: Define interfaces of software units
BP 3: Describe dynamic behavior
BP 4: Evaluate software detailed design
BP 5: Establish bidirectional traceability
BP 6: Ensure consistency
BP 7: Communicate agreed SW det. design [...]
BP 8: Develop software units

SWE.4 SW unit verification
Dynamic and static verification of SW units

Benefits
- Consistent and reusable definition of interfaces and architecture elements
- Close coupling of System and SW engineering

Coding acc. to Coding guidelines
Interfaces to MATLAB/SIMULINK
Unit Test, e.g. VectorCAST
Integration and Test

SWE.4 SW unit verification
SWE.5 SW integration and int. test
SWE.6 SW qualification test

SYS.4 System integration and int. test
BP 1: Develop system integration strategy
BP 2: Develop system integration test strategy incl. regression test strategy
BP 3: Develop specification for system integration test
BP 4: Integrate system items
BP 5: Select test cases
BP 6: Perform system integration test
BP 7: Establish bidirectional traceability
BP 8: Ensure consistency
BP 9: Summarize and communicate results

SYS.5 System qualification test

Test strategy with test levels
Aligned definition of:
- Test objects
- Test goals
- Test methods
- Test environment
- Test end criteria
- Integration strategy ...

Benefits
- Requirements, architecture and their model attributes can easily and consistently reused during testing
Traceability is one of the biggest challenges in ASPICE projects.

In many projects, traceability is applied manually and costly after engineering is finished. This brings no value to the project!
Applying consistency and traceability

Benefits
- Traceability benefits most of integrated E/E backbone
- Integrated requirements architecture with abstraction is biggest advantage compared to heterogeneous tool-chains
**Project Management**

**MAN.3 Project management**
BP 1: Define the scope of work  
BP 2: Define project life cycle  
BP 3: Evaluate feasibility of the project  
BP 4: Define, monitor and adjust project activities  
BP 5: Define, monitor and adjust project estimates and resources  
BP 6: Ensure required skills, knowledge, and experience  
BP 7: Identify, monitor and adjust project interfaces and agreed commitments  
BP 8: Define, monitor and adjust project schedule  
BP 9: Ensure consistency  
BP 10: Review and report progress of the project

---

**ASPICE does not explicitly require V-Model, Waterfall etc.; AGILE methods are allowed, too!**

<table>
<thead>
<tr>
<th>“Classic” Project Management</th>
<th>Agile Project Management</th>
</tr>
</thead>
<tbody>
<tr>
<td>• The target definition is iteratively developed using best practice methodology.</td>
<td>• The target is defined and evolving during development based on iterations.</td>
</tr>
<tr>
<td>• Driven by the process.</td>
<td>• Driven by the product.</td>
</tr>
</tbody>
</table>

---

**“Model-driven Systems Engineering using AGILE methods”**

- The model is the Systems Engineer’s “code”.  
- The model is the integrated work product of all processes  
- Documentation shall be derived from it as much as possible  
- The model can be automatically verified for consistency and traceability and enables high re-use and quality  
- strongly supported by tools

---

Source: F. Kirschke-Biller (Ford): "Agile meets Automotive Systems Engineering", Vector Forum, 2018
The Future of the Management and Supporting Processes

MAN.3 Project management

"Model-driven Systems Engineering using AGILE methods"

SUP.1 Quality assurance
SUP.8 Configuration Management
SUP.9 Problem resolution management
SUP.10 Change request management

Each of those Management and Supporting Processes benefit from a systematic task-based planning, i.e. using PREEvision’s ticket functionality
Quality Assurance

SUP.1 Quality assurance
BP 1: Develop a project quality assurance strategy
BP 2: Assure quality of work products
BP 3: Assure quality of process activities
BP 4: Summarize and communicate QA activities and results
BP 5: Ensure resolution of non-conformances
BP 6: Implement an escalation mechanism

Benefits
- QA activities are supported by workflow management and review functionalities which can be applied on any artefact (work products)
- QA team is supported also by reporting functionalities
Configuration Management

SUP.8 Configuration Management
BP 1: Develop a configuration management strategy
BP 2: Identify configuration items
BP 3: Establish a configuration management system
BP 4: Establish branch management
BP 5: Control modifications and releases
BP 6: Establish baselines
BP 7: Report configuration status
BP 8: Verify the information about configured items
BP 9: Manage the storage of configuration items and baselines

Benefits
- Configuration Management more focused on model configurations and baselines of artefacts instead of storing files
- Plan and generate required reports for stakeholders as needed.
SUP.10 Change request management
BP 1: Develop a change request management strategy
BP 2: Identify and record the change request
BP 3: Record the status of change requests
BP 4: Analyze and assess change requests
BP 5: Approve change requests before implementation
BP 6: Review the implementation of change requests
BP 7: Track change requests to closure
BP 8: Establish bidirectional traceability

Benefits
► Change and issue management benefits from the integrated model, because most of the needed information can be easily accessed in the model.
Observations from more than 10 years of process improvement consulting projects at customers

- SW development is often disconnected from other Engineering and management activities
- “We build the software and document and test it later”
- Requirements, architecture and safety/security concepts are not systematically considered and tested
- In 10 of our last Code quality assessment at customers we didn’t see one single sufficient architecture description

Our approach

In our consulting approach we want to overcome this:

- Close coupling of processes, methods and tools
- ASPICE Engineering processes benefit a lot from an integrated tool solutions with backbone as shown
- ASPICE Management and Supporting processes benefit from AGILE methods and integrated tool functionality, i.e. ticketing, review workflows, configuration management, reporting and metrics
- Document driven development maybe be more and more replaced by model-driven/reporting methods and close the gap of SW engineering.
- Hopefully this will be more accepted among engineers who don’t like documentation!

Next steps

- All the needed functionality and methodology is there. Let’s make it happen!
- We would like to support and officially assess projects using PREEvision
Thank you for your attention. For more information please contact us.
Conclusions and Outlook

More Information...

Trainings and Media

- Free cybersecurity Webinar (1 hour, continuously updated)  
  [www.vector.com/webinar-security](http://www.vector.com/webinar-security)
- Free Functional Safety Webinar (1 hour, continuously updated)  
  [www.vector.com/webinar-safety](http://www.vector.com/webinar-safety)
- In-house trainings tailored to your needs are worldwide available

Vector White Papers

- [www.vector.com/media-consulting](http://www.vector.com/media-consulting)