Who we are

Founded in 2020, we have built a corporate startup from scratch attracting the best Tech Talents worldwide

6,000+

CARIAD employees worldwide today

80+

Nationalities working at CARIAD

360

teams at CARIAD

Join our mission and become part of one of the biggest endeavors in the automotive industry.

CARIAD

The right software is the basis for Volkswagen's success. CARIAD provides it

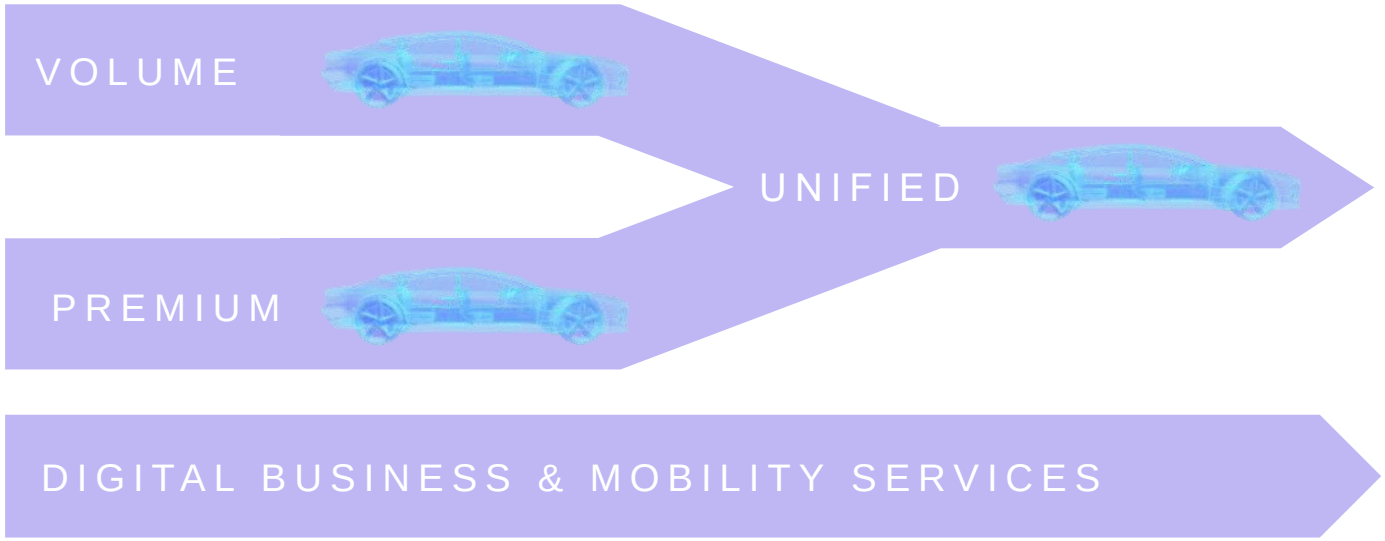
Our Solution: One unified and scalable tech stack for the entire Volkswagen Group

CARIAD's end-to-end electronic architecture platforms for Volkswagen's Volume and Premium brands are technological frontrunners.

// Building one tech stack scalable for all segments

// Full over-the-air update (OTA) capability to increase customer value over time

// By 2030, up to 40 million Volkswagen Group vehicles will run on CARIAD software platforms



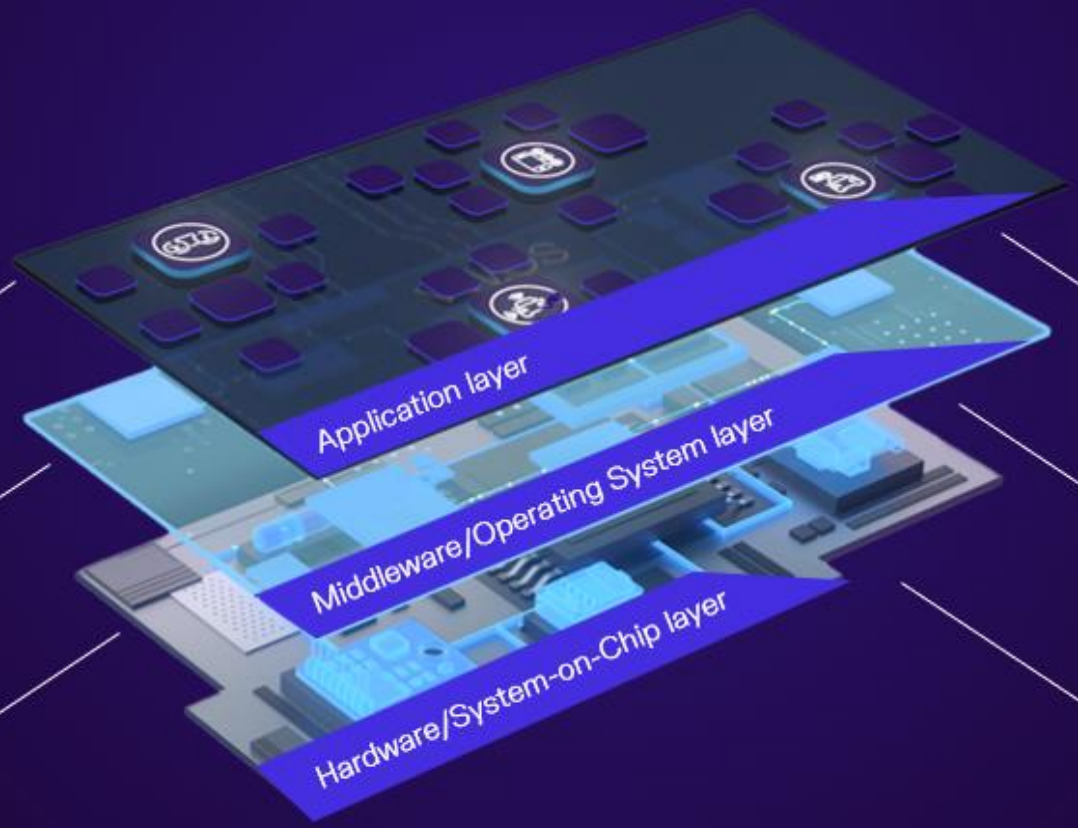
How do we do it? With our Technology Stack, one scalable platform provided by CARIAD with renowned partners

The "Driver"

Automated Driving Functions
CARIAD

Middleware/Operating System
CARIAD | BOSCH

Hardware/System-on-Chip
CARIAD | Qualcomm | Horizon Robotics



The Digital Experience

Vehicle Experience
CARIAD | 3rd Party

Middleware/Operating System
CARIAD | AOSP

Hardware/System-on-Chip
CARIAD | 3rd Party

All the assessment answers we did not wanted to get in the last 75 assessments

Flavia Povirnaru

Andreas Gasch

Warning!
Intent of this talk:

- Be informative
- Give food for thought
- Inspire discusion and curiosity
- Help forming a common understanding

Some quotes from assessment sessions…

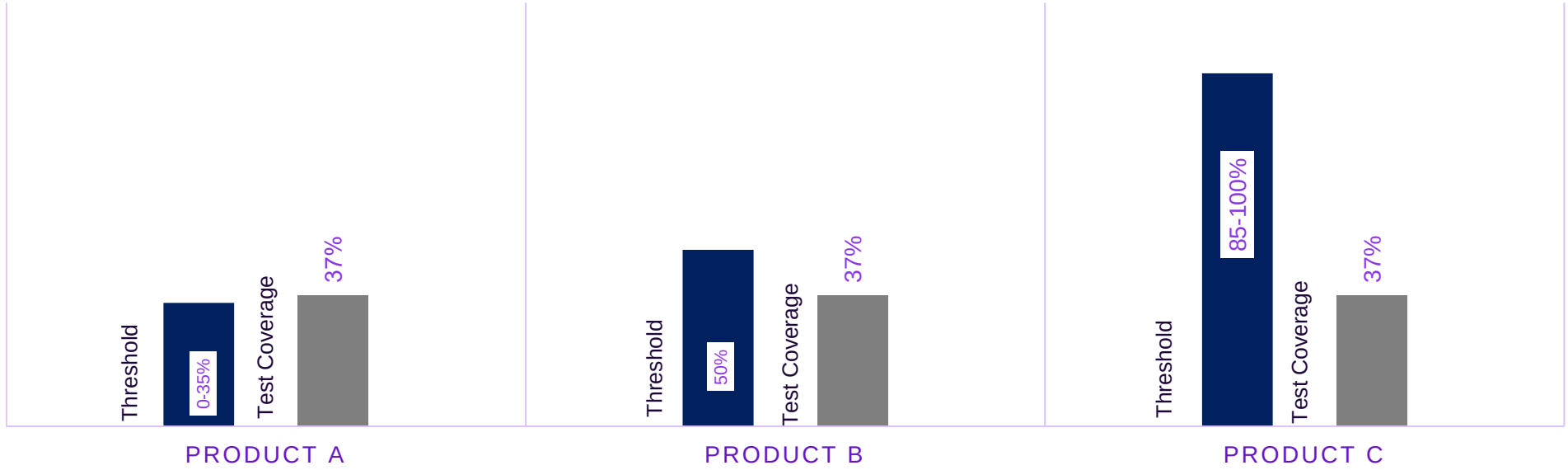
- “**This code is not prepared to be assessed.**“
- “**In the design we cannot identify if one component is riskier than another one. Risks can only be identified on requirements.**”
- “**The goal is to get the picture into the picture**.” (talking about the objectives for the Sys. Arch process)
- “**And this is our MAN.3 KPIs for the reporting...**” (using template values to fill a template report just for the sake of having a report)
- “**The tool is checking something at the end of the project**.” (FOSS strategy)

Some quotes from assessment sessions…

- **“We just copied it from the previous project.”-** Risk of outdated, irrelevant, or non-tailored artifacts.
- **“Nobody ever asked for metrics before.”** → Indicates metrics are not seen as a management or quality tool.
- **“We send the reports to management, but we never get feedback."**→ Indicates poor communication loop and low to no management involvement.
- **“We report what the customer is asking for...”**

- **“The tool collects data automatically.” ->** Data without understanding = no informed decisions.
- **“We collected some numbers but they’re just for reporting.”** → No link between metrics and actions.

A classic of what we usually see in assessments…



=> Report = Green (within limits)

=> Report = Yellow (medium risk)

=> Report = Red (blocker)

Are these template **thresholds** and the actual achievement enough to have an insight into the testing status?

Or **the product context, the timing of deviations and their actual risk impact should be also looked at?** Experience shows that although many projects claim to apply risk-based interpretations, reporting often fails to reflect such practices.

What is it really measured?

🟧 Numbers alone don’t tell *why* something matters — but context does!

➡️ To understand if a metric truly reflects risk, maturity, or readiness, we must ask:

> What is the product supposed to be good at?
>
> -> Is it meant to delight, last, adapt, to be portable etc..?

→ **That’s where for examples quality properties come in.**

They help interpret what raw numbers mean — in a way that matches the *real expectations* from the product.

From Quality Priorities to Sample Products — Context Shapes Meaning

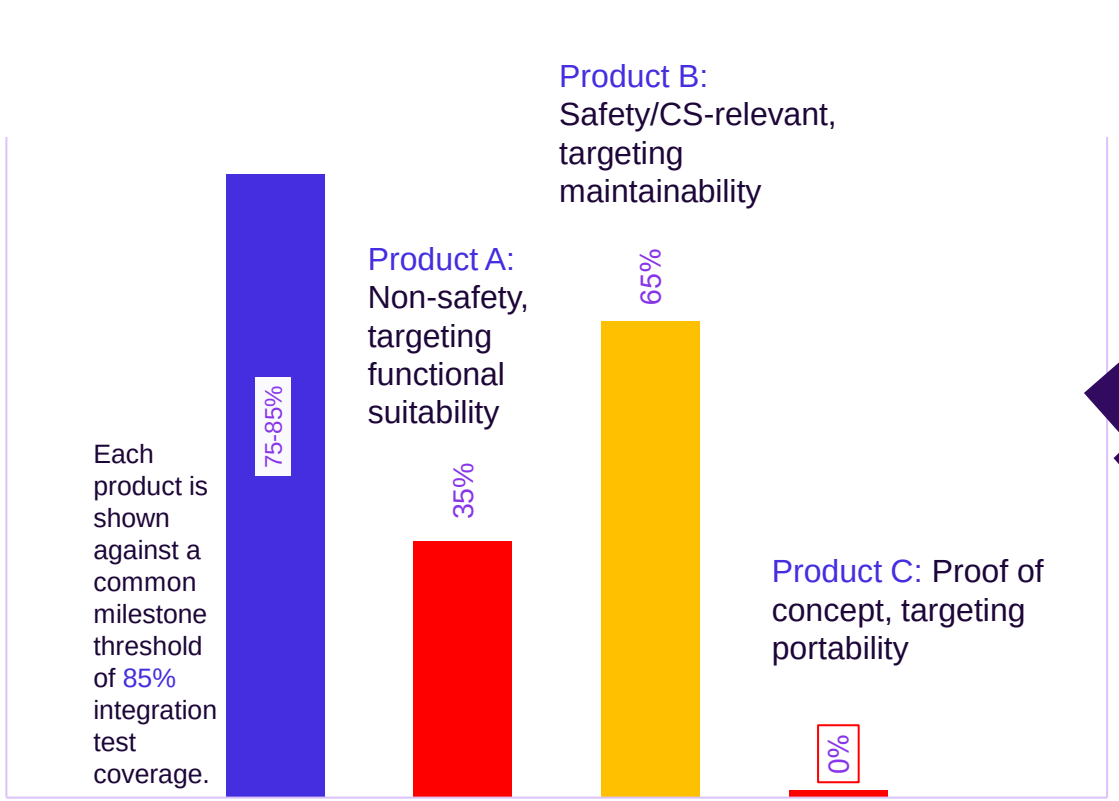
Product	Sample Type	Quality Priority	What It Means
Product A	HMI features (e.g. media player UI, ambient lighting)	Functional Suitability	• Light set of requirements (key excitement feature) - focus on what is appealing to the customer • Quick results, requirements coverage-oriented testing, smoke tests • Customers determine if it is acceptable and if they want „more“
Product B	Operating systems Middleware, Airbag	Maintainability	• Modularity & Documentation • Focus on long life, add ability of features • Integration testing, individual releases may fail, OTA can heal
Product C	Applications	Portability	• Enable multi platform usage - Product usable in different environments • Hardware agnostic design • Hardware integration testing

Non-safety

Safety/CS

Proof of concept

Not all deviations are equal — context matters!



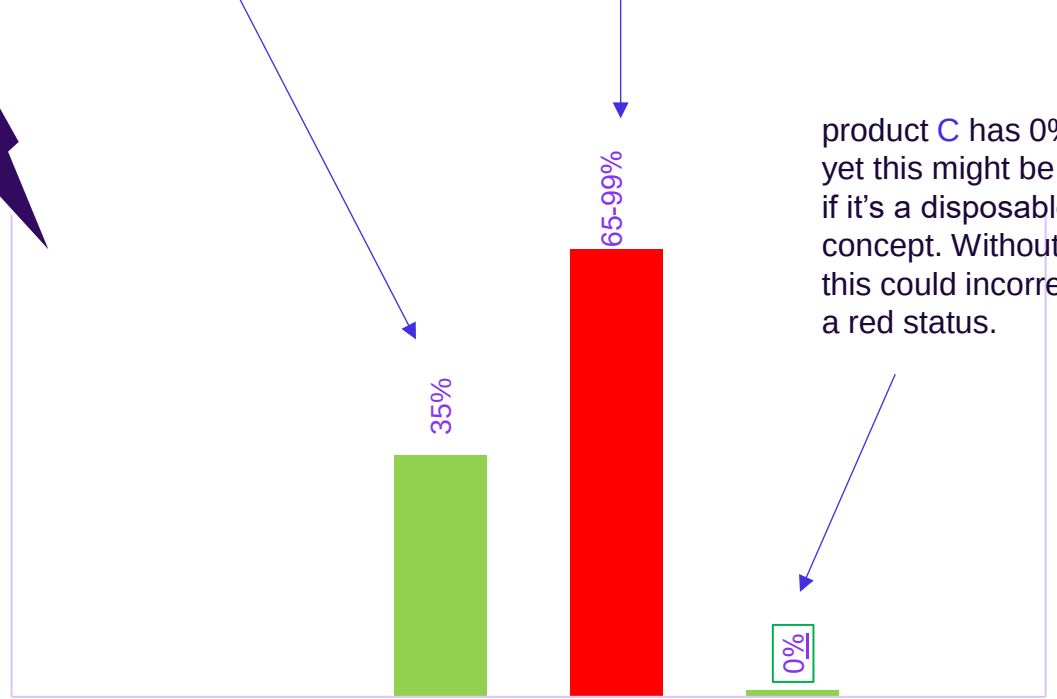
Report using „default“ thresholds

Same numbers — different meaning

product A shows 35% coverage but might be acceptable due to low criticality. May not pose significant risk, especially if the product is One-Time Feature ("Easter Egg").

product B may show 65-99% coverage, but in a safety/CS context, the remaining (even 1%) gap could include critical paths - making the deviation highly relevant.

product C has 0% coverage, yet this might be acceptable if it’s a disposable proof of concept. Without context, this could incorrectly trigger a red status.

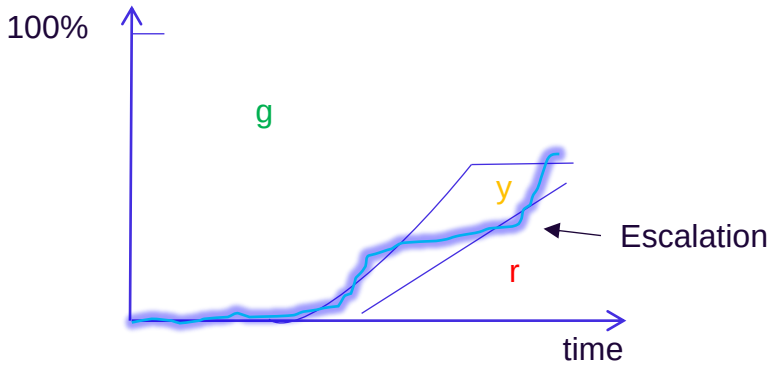


Weighted Report considering the product context

Raw numbers can be easier set in context with e.g. quality properties and **project timelines**…

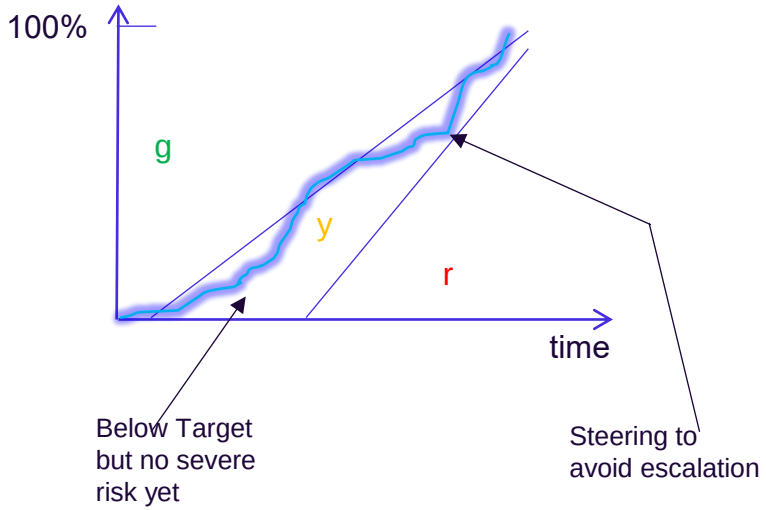
Example: Integration Testing

Product A: „Functional Suitability“



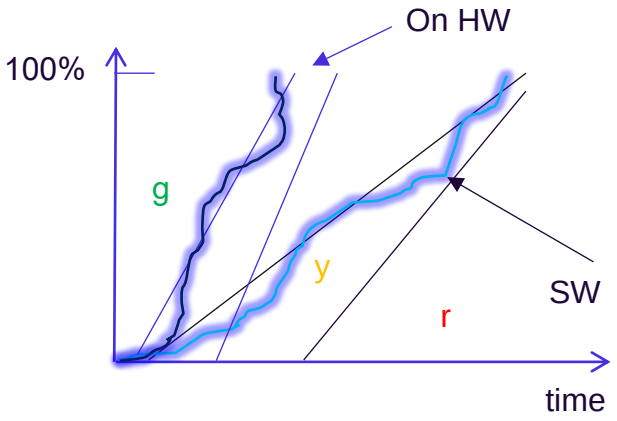
Focused on quick delivery, integration tests are **not urgent early on**
more important to run **demo-based end-to-end tests** early for customer acceptance

Product B: „Maintainability“



Early focus on integration testing.
100% is required to have an argument that the implementation suits the architecture

Product C: „Portability“

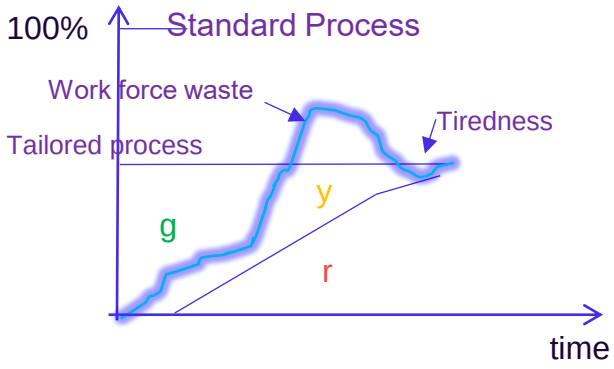


Metrics must be tailored to reflect reality
e.g: track coverage per platform, not only per function.
Early coverage on the different HW

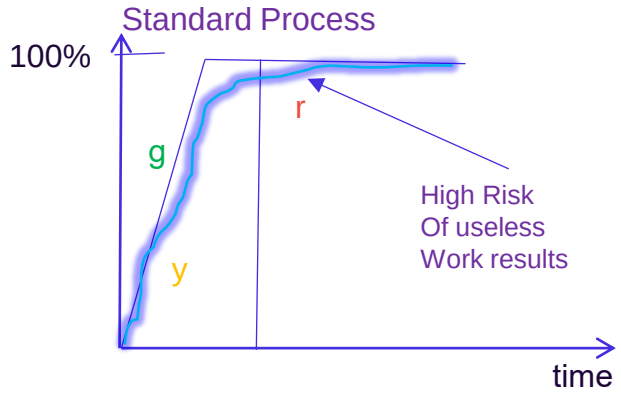
In all three, **context defines meaning**. The same metric means different things in each product.

Let’s look at another example:

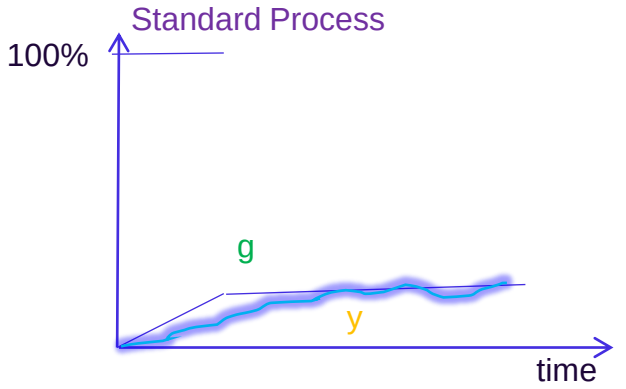
Product A
Functional suitability
Non-Safety Product



Product B
Maintainability
Safety/Cybersecurity
(ISO 26262/(Internal tailored CS))



Product C
Portability
Proof of concept



When interpreting thresholds, it is essential to consider the project context, the timing of deviations and their actual risk impact.
A 1–2% deviation may signal a serious issue in a functional safety context, whereas even a 10% deviation might be tolerable in other domains.

From Insight to Oversight — When meaning gets lost in translation (and in reporting)

All this focus and emphasis on numbers and thresholds exists for a reason…

—> measurement enables reporting, and reporting empowers those **who must act**

Management Report Damages - Red, Yellow, Green — But What's Actually at Risk?

Many reports look structured and neat:

Commonly observed in assessments:

- <75% of tasks done = red
- 75-85% = yellow
- 85-99+ = green

- fixed metrics & thresholds from templates or experience without any context or historic argument behind
- no or red reports as the project during development never achieve 100% of something
- unclear status and uncertainty in outlook
- nearly no decision derivable from those overviews

But do they tell what's broken, what's urgent, and what needs a decision?

-> reporting with no clarity about what is most urgent and what to do...

→ Time to move from status cosmetics to decision-enabling clarity.

From status cosmetics to decision-enabling clarity

Weighted Reporting

- ✓ **26 project on track**
- ✓ 17 projects with findings under monitoring

- ➢ 3 projects to be looked at:
  - risk identified
  - solution recommendation
  - required decision / support request

-> that's actionable.

Assessors are encouraged to challenge one-size-fits-all and instead seek clarity on what reported values *actually* reveal about product readiness and risk.

- ➢ Are all critical tasks done?
- ➢ Was a sufficient method applied?
- ➢ What is the real risk implied by all tasks not done?

Assessment Reports…. Fully Rated. Barely Relevant.

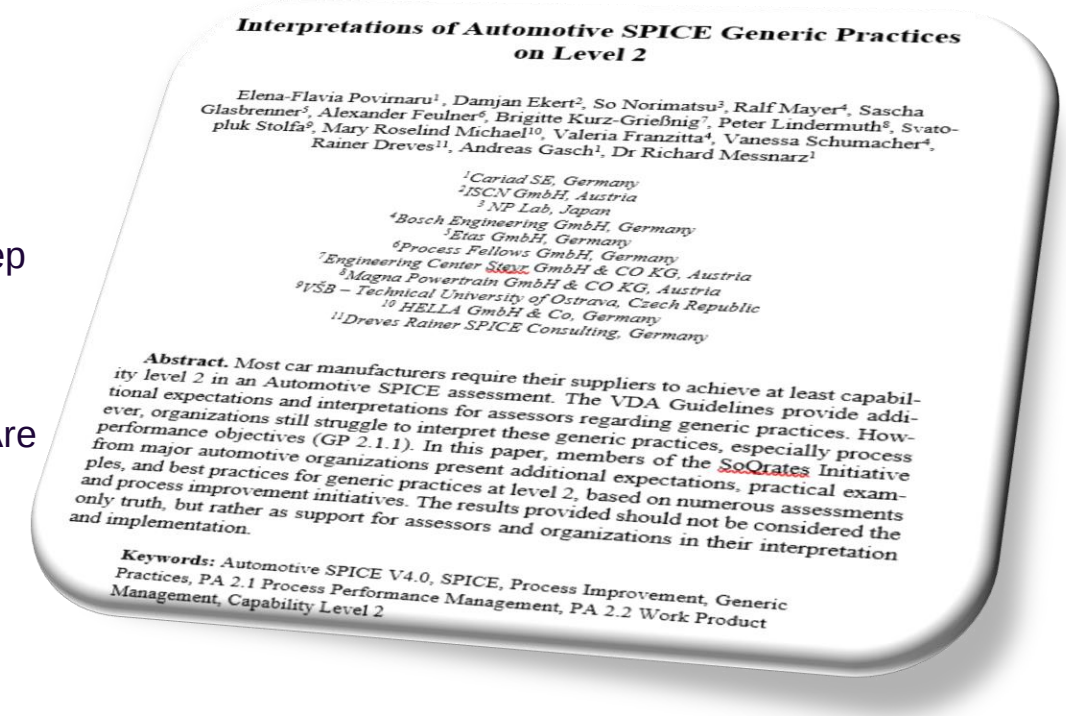
We’ve often seen in the assessment report statements like**: “100% traceability is given” → SWE.5/6 Fully.**

->> Instead, the report should clarify whether the numbers truly reflect the actual agreements and demands (e.g.: Additional OEM Agreements, or Safety require more than 1:1) — and whether the reports are trustworthy or not.

Numbers in assessments should not be taken at face value - they must be examined to determine whether they make sense and are truly used for steering.

What assessors should critically examine :

- if teams have an applicable reference to understand where they are supposed to be vs. status today: Is it clear how teams assess their current status against expected progress? Look for whether meaningful baselines or planning targets are defined and understood.
- predictability for goal achievement (within / outside corridors): Does the reporting allow to identify trends or forecast goal achievement realistically (e.g., deviation corridors)?
- **red** / **yellow** / **green** get their original meaning back and enable decision making: Are red/yellow/green indicators clearly defined and used consistently to trigger decisions, or do they mask issues?
- if projects can be compared without deep knowledge of life-cycle phase and context: Can project performance be interpreted without needing deep knowledge of each lifecycle phase or context, or is interpretation too subjective?
- if raw numbers can be easier set in context with e.g. quality properties: Are numerical indicators (e.g., test coverage, bug rates) contextualized with quality attributes like robustness or reliability to make them meaningful?



Interpretations of Automotive SPICE Generic Practices on Level 2

Elena-Flavia Povirnaru[1], Damjan Ekert[2], So Norimatsu[3], Ralf Mayer[4], Sascha Glasbrenner[5], Alexander Feulner[6], Brigitte Kurz-Grießnig[7], Peter Lindermuth[8], Svatopluk Stolfa[9], Mary Roselind Michael[10], Valeria Franzitta[4], Vanessa Schumacher[4], Rainer Dreves[11], Andreas Gasch[1], Dr Richard Messnarz[1]

[1]Cariad SE, Germany
[2]ISCN GmbH, Austria
[3] NP Lab, Japan
[4]Bosch Engineering GmbH, Germany
[5]Etas GmbH, Germany
[6]Process Fellows GmbH, Germany
[7]Engineering Center Steyr GmbH & CO KG, Austria
[8]Magna Powertrain GmbH & CO KG, Austria
[9]VŠB – Technical University of Ostrava, Czech Republic
[10] HELLA GmbH & Co, Germany
[11]Dreves Rainer SPICE Consulting, Germany

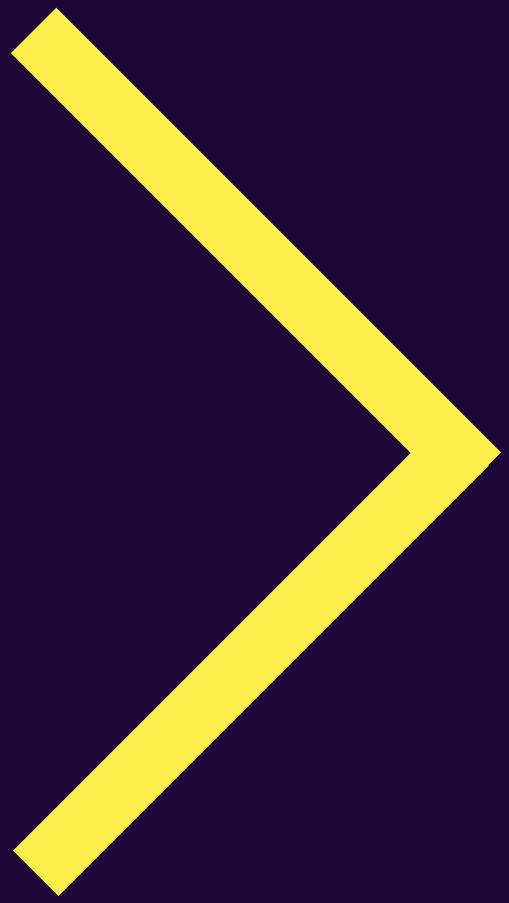
Abstract. Most car manufacturers require their suppliers to achieve at least capability level 2 in an Automotive SPICE assessment. The VDA Guidelines provide additional expectations and interpretations for assessors regarding generic practices. However, organizations still struggle to interpret these generic practices, especially process performance objectives (GP 2.1.1). In this paper, members of the SoQrates Initiative from major automotive organizations present additional expectations, practical examples, and best practices for generic practices at level 2, based on numerous assessments and process improvement initiatives. The results provided should not be considered the only truth, but rather as support for assessors and organizations in their interpretation and implementation.

Keywords: Automotive SPICE V4.0, SPICE, Process Improvement, Generic Practices, PA 2.1 Process Performance Management, PA 2.2 Work Product Management, Capability Level 2

Key Take Away

- Each project, product has different boundaries…
- **Don't Be Misled by Numbers**
  Assessors must evaluate the **sufficiency** of the evidence
  ✔️ **100% traceability** may look impressive,
  ❌ but does it reflect what is truly needed for the product?
- ➡️ Challenge whether reported KPIs are meaningful, used for decision-making, and aligned with project risks and context — not just green for the sake of being green.
- Standard-sets help as a starting point, not tailoring them to the boundaries may lead to waste (measuring what is not needed or misunderstandings e.g. project red while it is in reality on track)

Feedback welcome



Get in touch:

andreas.gasch@cariad.technology
elena-flavia.povirnaru@cariad.technology


We transform
automotive mobility
THE SOFTWARE
POWERHOUSE
SEAT
ŠKODA
CARIAD

FOSS, Safety, Metrics

A Very Common KPI in Standard Sets is „**Comment Code Density**“ highly supporting „Maintainability“ by advocating to have an explanatory Comment each 10 lines of code.

But what do you do if…..

- The code is created by a community distributed around the globe
- There is no contractual obligations to bind someone to
- The measurement says that there are barely comments
- The Standard (e.g. OEM specific requirements, CPM) expects the metric for releases
- The License model is in a way to prohibit commercial change of the Code

=>> Forbidding usage or Establish new metrics to manage the emerging risk ?

Quality Properties