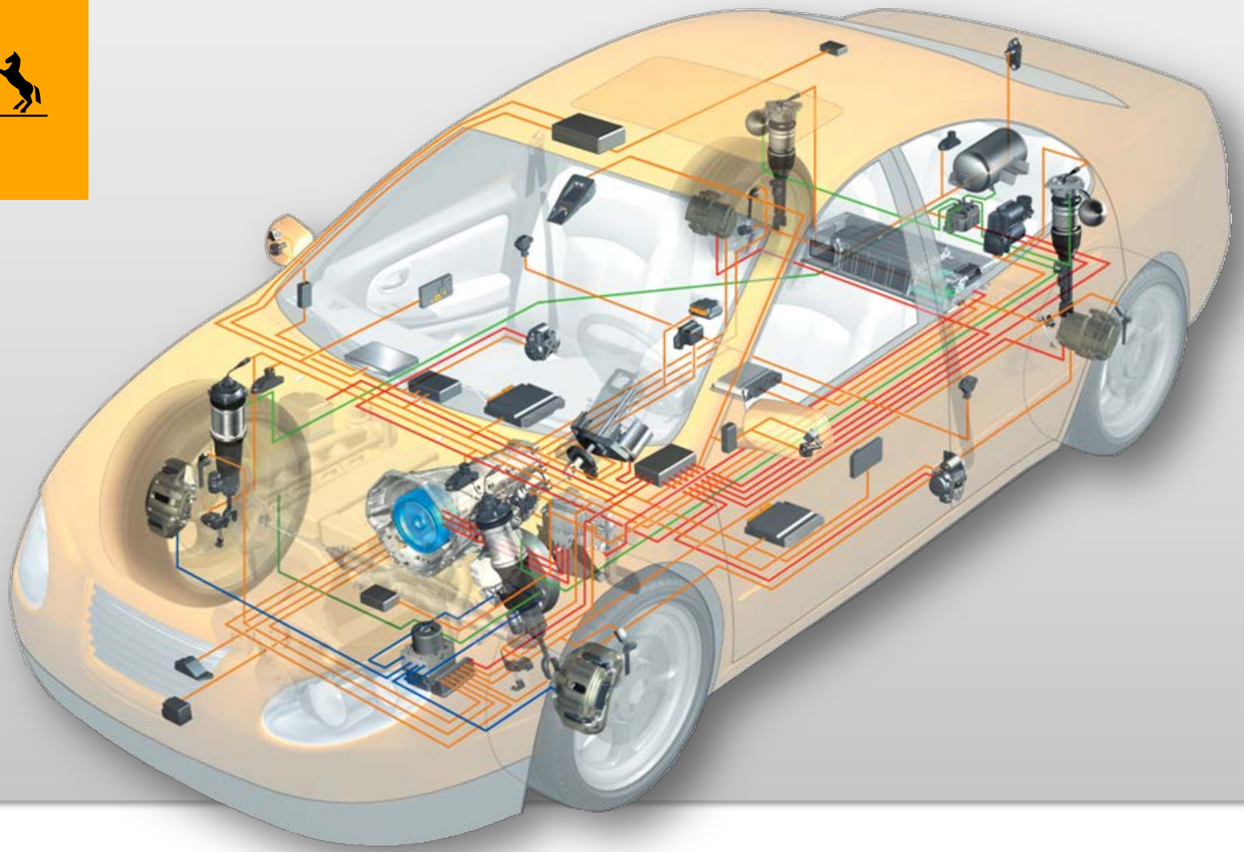


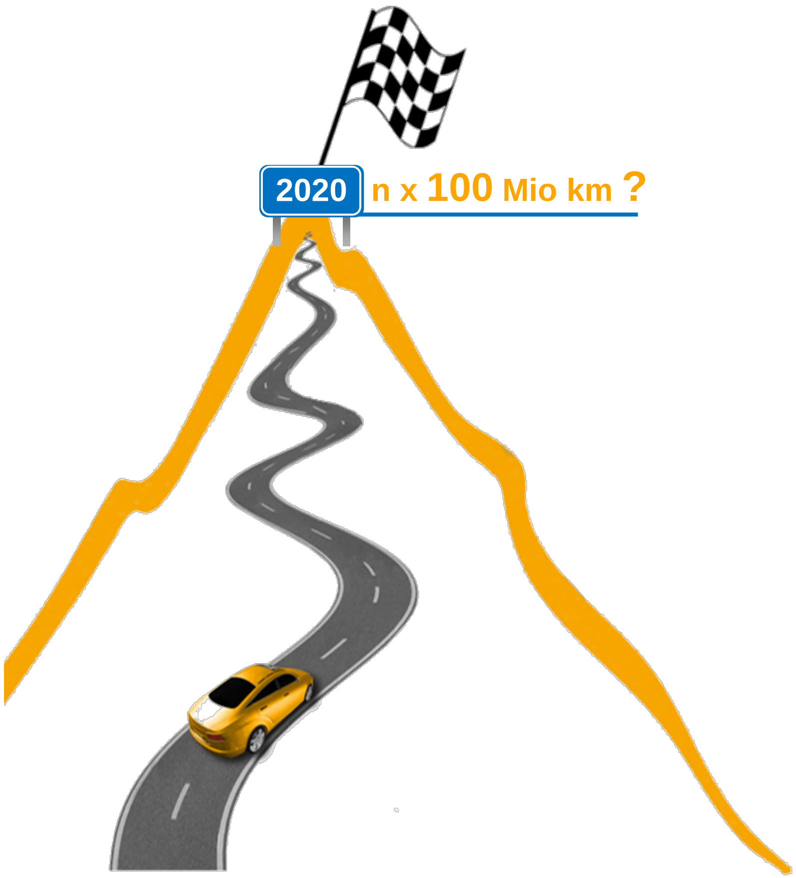
Validation of systems for (highly) automated driving

Matthias Stiller

Head of Testing Cross Divisional Systems

Integration and Validation of AD functions

4 Main Challenges



- The **driver** is (temporarily) **out of the Loop**!
  Validation of hand over scenarios.
- The **number of situations** / **scenarios** an automated driving vehicle will be able to handle correctly is **immense**!
- The resulting Integration & Testing effort **cannot be managed economically** by using traditional testing methods and processes!
- The assumption is that Automated Driving Functions will need to reduce the number of accidents by 50%
  → The amount of testing kilometers to prove correct functionality is estimated with **several 100Mio Kilometers** (Prof. Winner)!

Background: Current Driver Assistance Systems
Testing Strategy

Function shall react to …

	TRUE (Function reaction correct!)	FALSE (Function reaction incorrect!)
Function intervention **POSITIVE**	**Event happened as expected** e.g. real obstacles are reported Traffic Sign correctly reported Lane tracked correctly	**Event happened unexpected** e.g. non existing objects are reported Traffic Signs are incorrectly reported Lane tracking is incorrect
No Function intervention **NEGATIVE**	**No event, as expected** e.g. empty road	**Event not happened unexpected** e.g. a real obstacle is not reported Traffic Signs are not detected Lane tracking is not done

1 False Positive during a vehicle lifetime ≙ 1 incorrect event per 200.000km!

→ 1Mio test kilometers!

Integration & Validation Concept
for Automated Driving

Approach

- Analyse Human behaviour / system in interaction with the environment. What is their interface?
- How does it come to critical / accident situations? What is the probability of an accident?
- How does a human driver react in critical / accident situations?
- All above mentioned questions can be summarized to:
  - **What is the reference for automated driving functions?**
- What is the consequence for system development and the resulting validation concept?

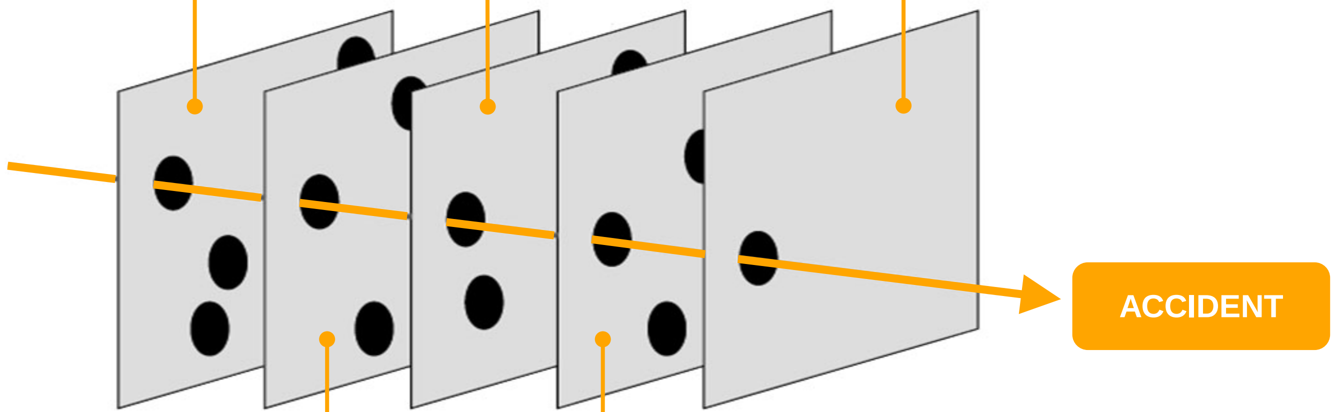
How does it come to accident situations in general?

Swiss Cheese Model (D. Orlandella, J. Reason)

Characteristics of environment
For instance: weather conditions, road conditions, unclear road intersections

Driver characteristics
For instance: age (young & old), high readiness to assume risk, overestimation of one's own capabilities

Error of Driver
For instance: to low concentration, distraction by secondary tasks, conversation with fellow passengers



Vehicle characteristics
For instance: no ESP, worn tires, oversized engines

Violation of driver
For instance: drinking & driving , speeding, not enough safety distance, overtaking, tiredness

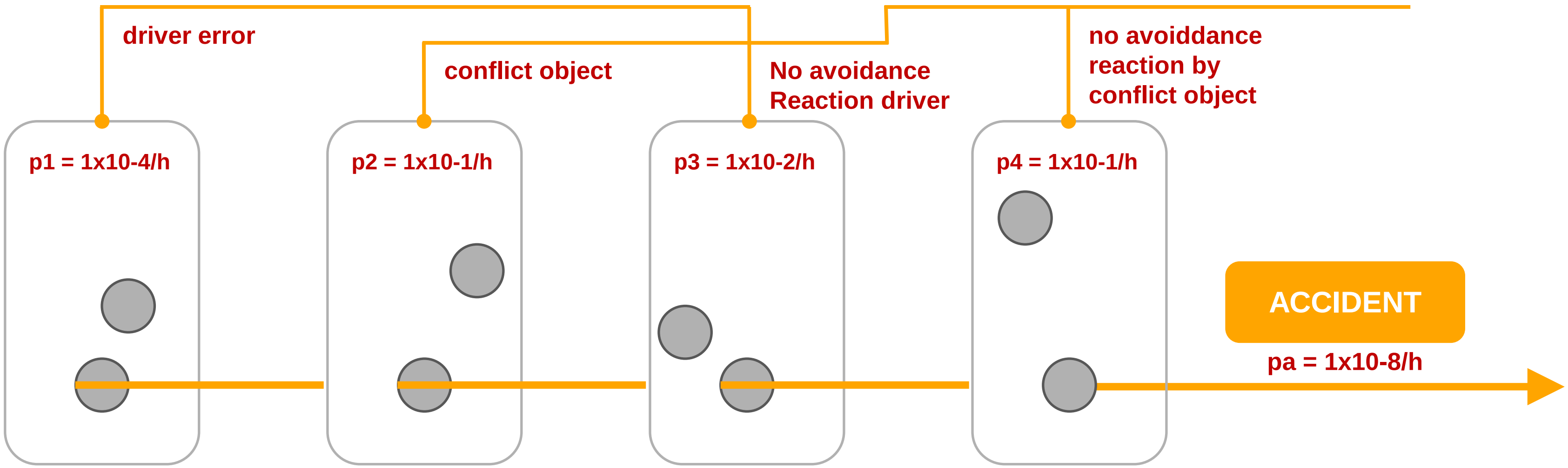
Conclusion: **Accidents are a chain of unfortunate circumstances!**

Swiss Cheese Model

Example

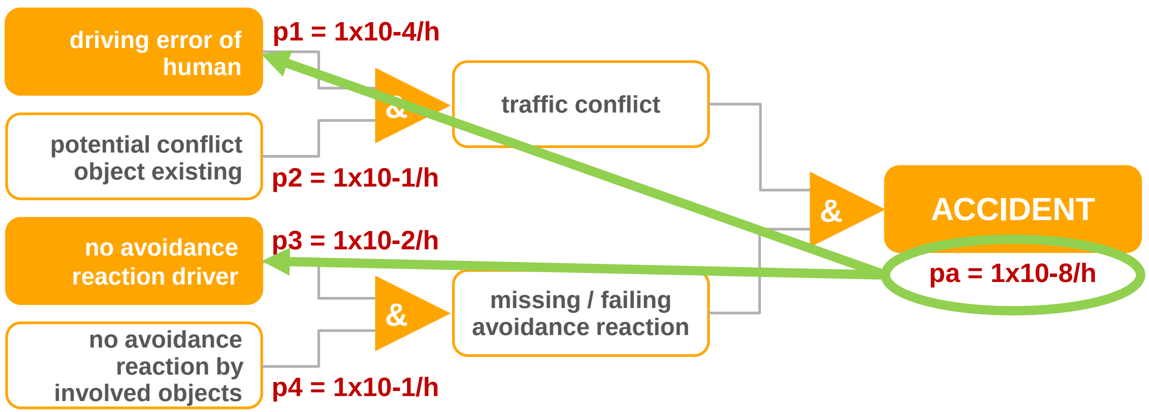
Ego Vehicle

Potential Conflict Object



Failure tree model of an accident and
its probabilities

The probability of an accident with a human driver on a freeway is **about** 1x10-8/h!



Idea

- The overall accident probability (pa) can only be decreased / improved by the system by improving / decreasing the probabilities p1 and p3.
- Acceptable failure probabilities: **p1 ~ 1x10-5, p3 ~ 1x10-3 → 100!**

Two major issues

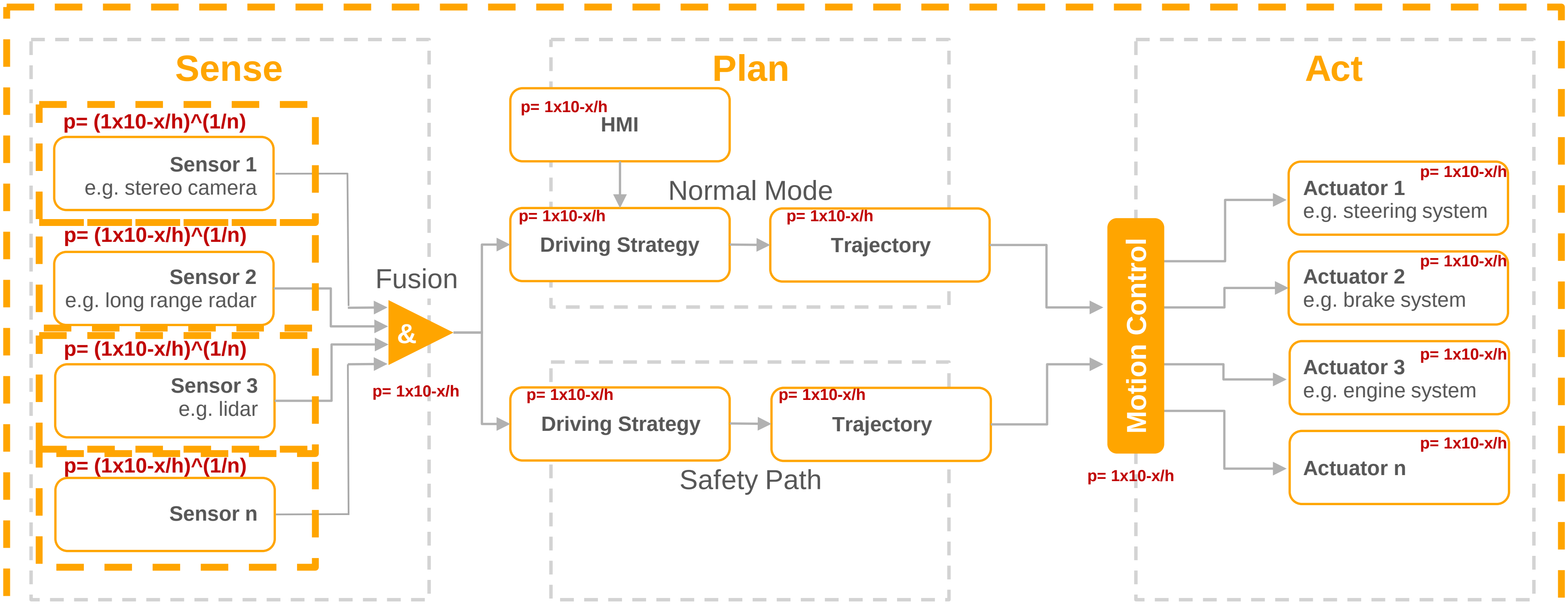
Ensure handling of loss of function <u>**and**</u> loss of integrity

→**Loss of function / fail operational: Redundancy necessary e.g. two EPS, two brake systems**

→**Loss of integrity: Ensure that the performance of the function is correct.**

<u>Example: Loss of integrity</u>

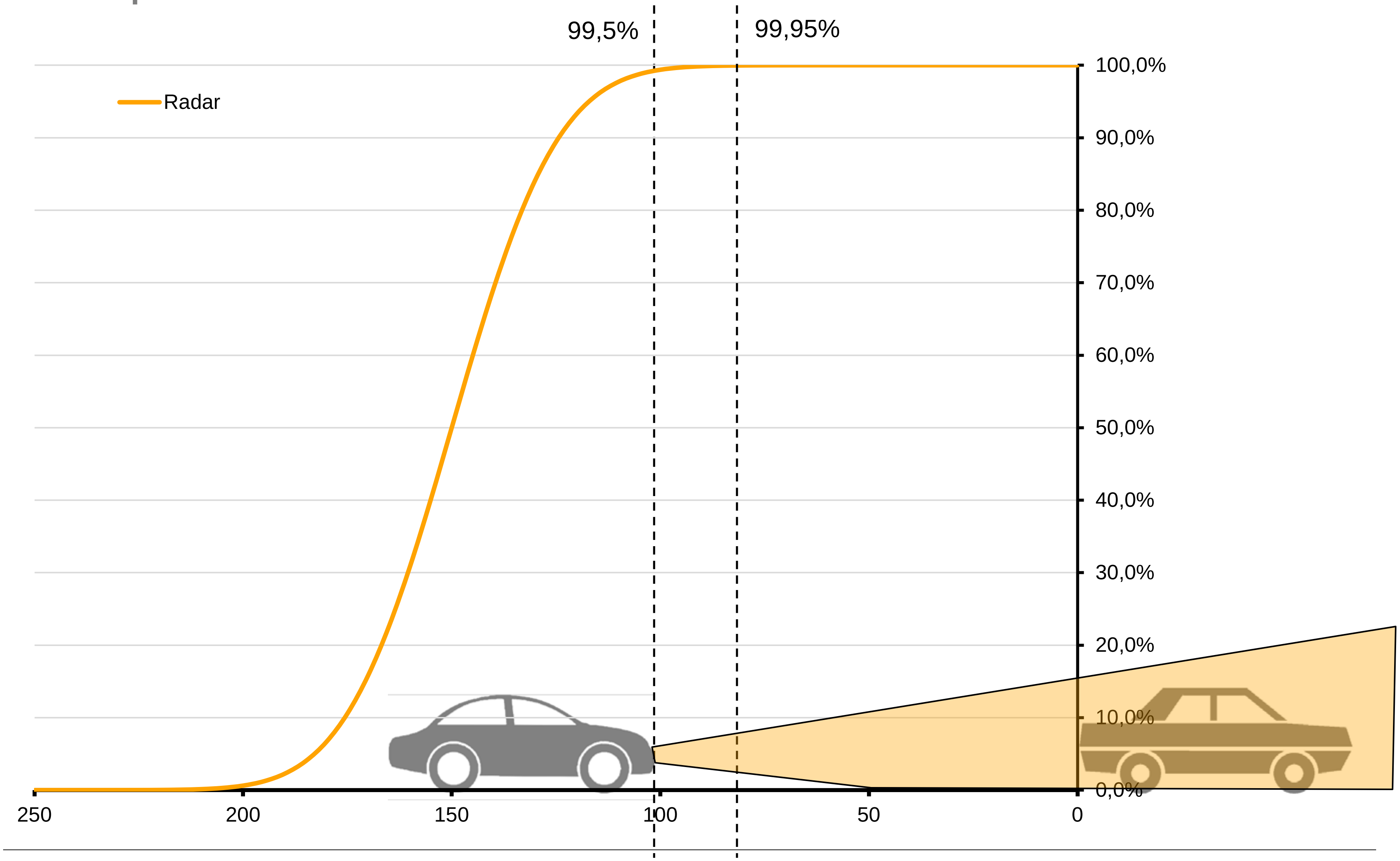
Function AD, failure probability => 1x10-x/h



→<u>**Validation Effort simimlar to current driver assistance systems!**</u>

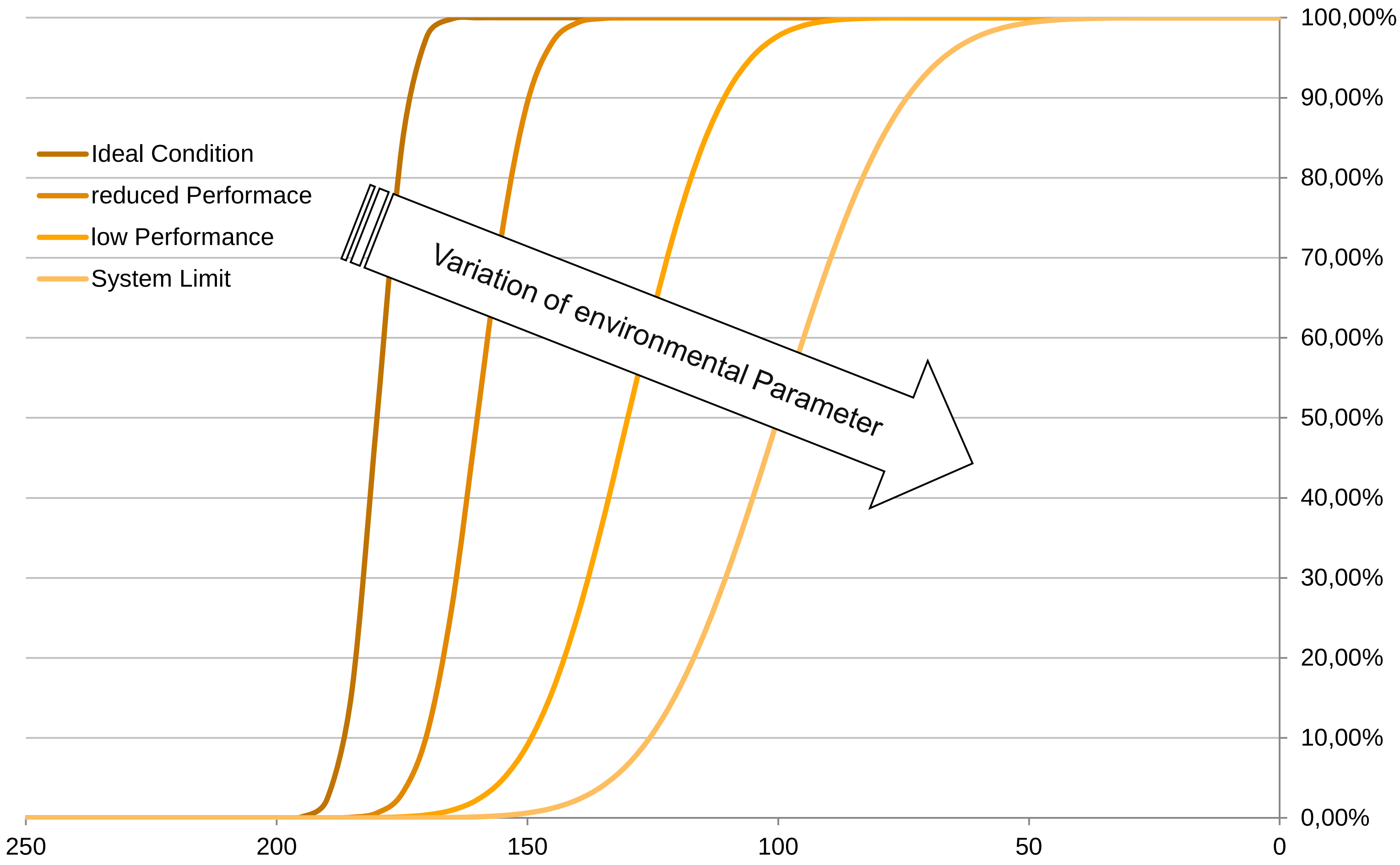
Deduced Sensor Performance

Example: Radar



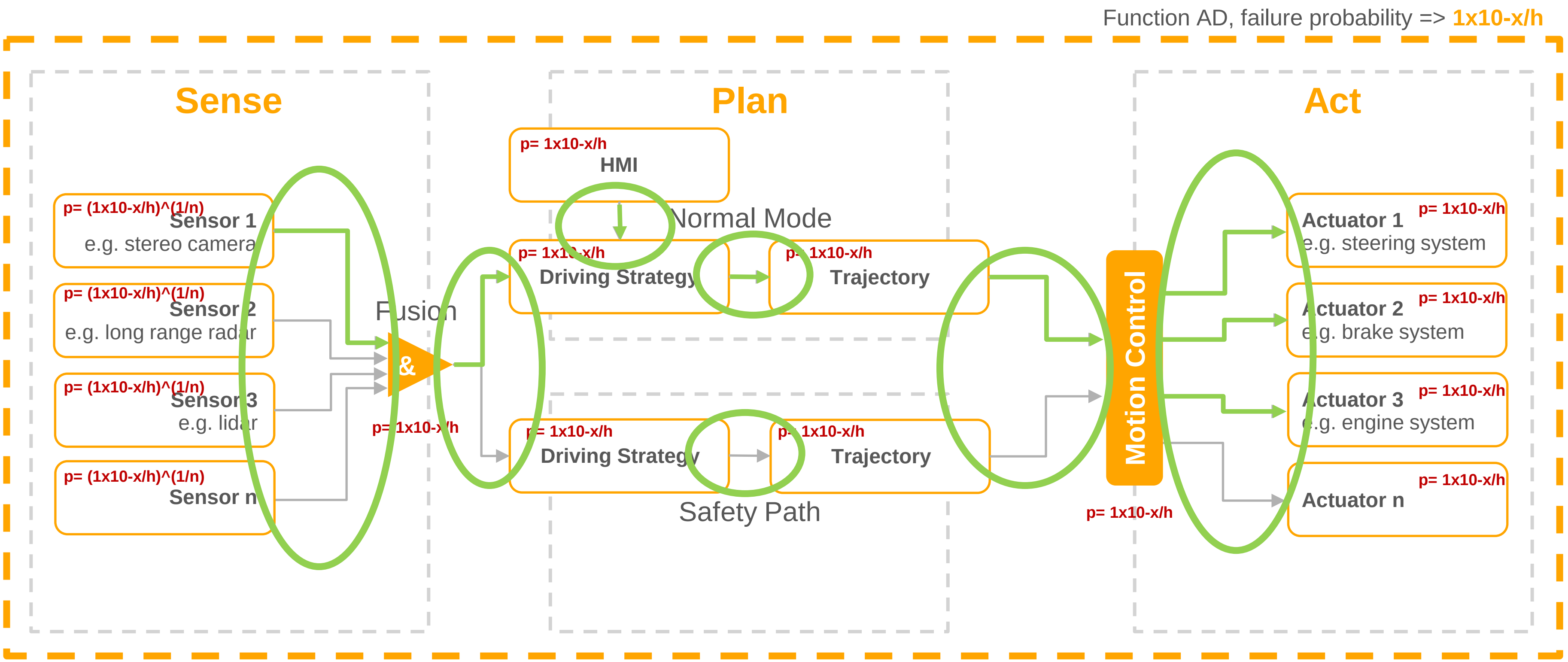
Sensor Performance of radar

and impact of influencing parameters



Bottom Up approach after component validation

focusing on System Integration and validation on vehicle level



Focus on vehicle level

→ **Intercomponent failure! Interface failure! Simulation environments and vehicle!**

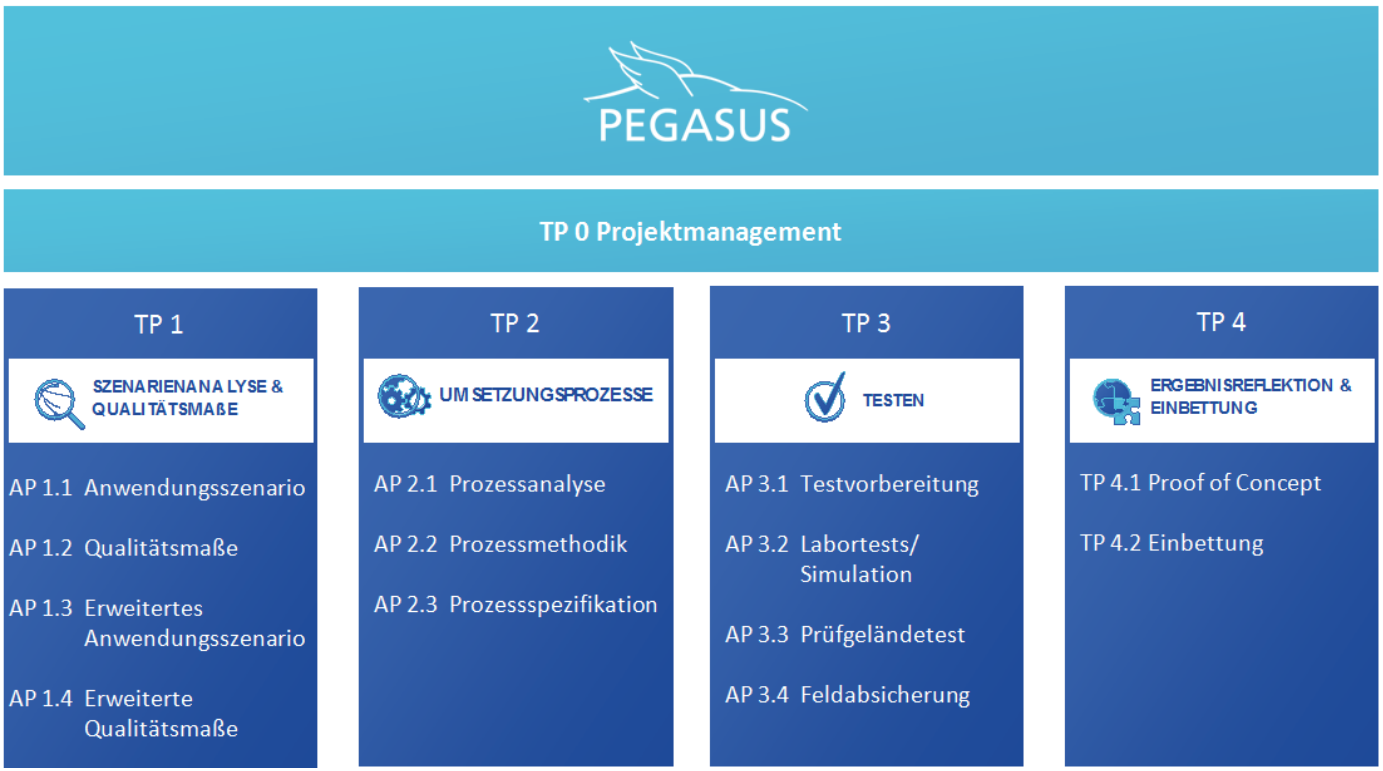
→ **Full chain of effects mainly with simulation methods but also vehicle tests! Focus critical scenarios!**

PEGASUS

(Projekt zur Etablierung von generell akzeptierten Gütekriterien, Werkzeugen und Methoden sowie Szenarien und Situationen zur Freigabe hochautomatisierter Fahrfunktionen)

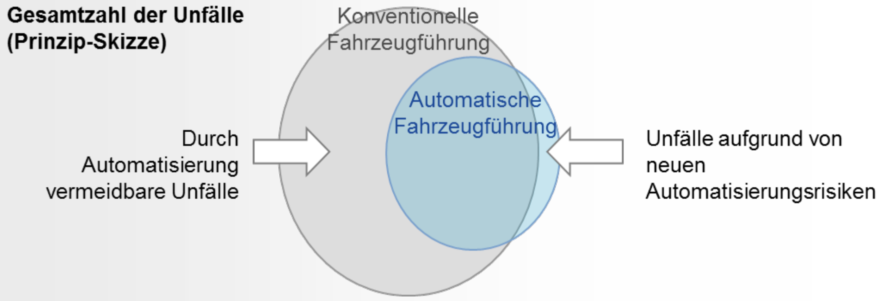
Objective:

- How good is good enough?
- How can we prove that the function works correctly?

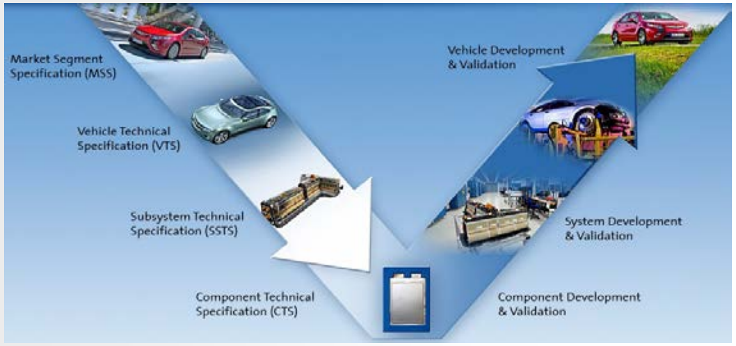


PEGASUS
Objective of subprojects

- **TP 1: Szenarienanalyse und Qualitätsmaße**
  - Description of the Autobahn Chauffeur, determination of critical scenarios, analysis of human behavior in critical scenarios, definition of system limits, definition of quality and acceptance criteria



- **TP 2: Umsetzungsprozesse**
  - Identification and definition of general development and testing processes and methods for automated driving functions



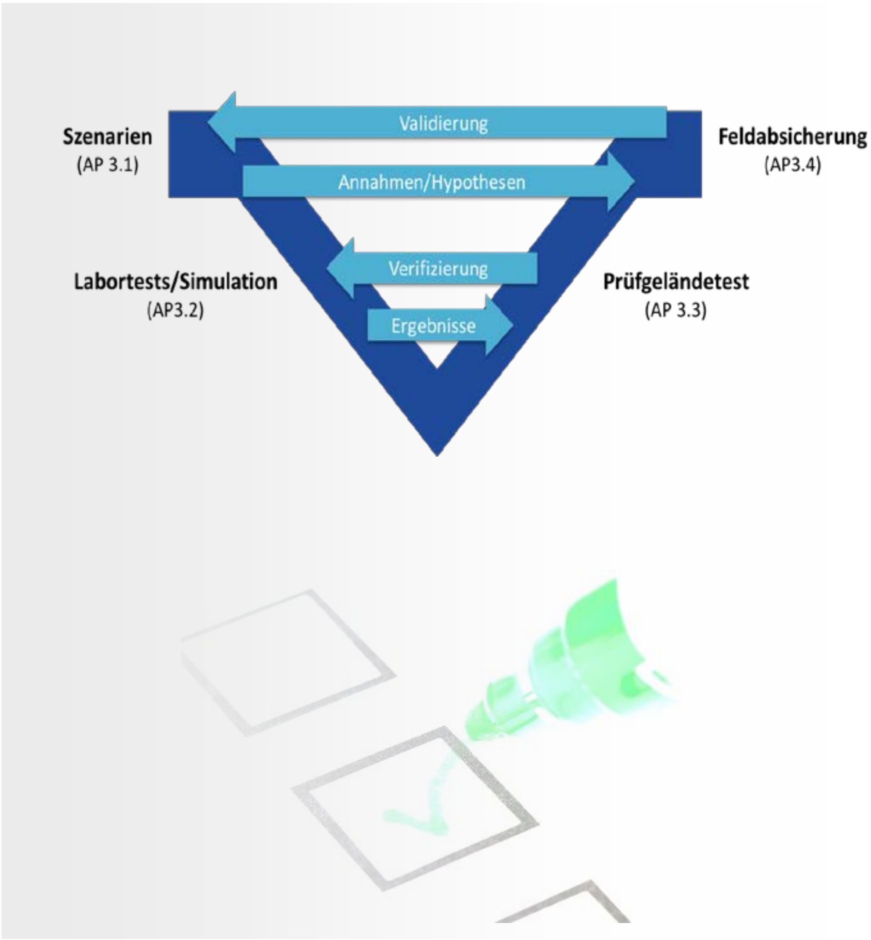
PEGASUS
Objective of subprojects

› **TP 3: Testen**

  › Definition of tool chain simulation, vehicle test on test tracks as well as field operational tests/endurance runs, prove that critical scenarios can be handled correctly by the automation based on acceptance criteria defined in TP1

› **TP 4: Ergebnisreflektion und Einbettung**

  › Proof of concept, critical reflection of the working packages of the other subprojects



Thank you
for your attention!