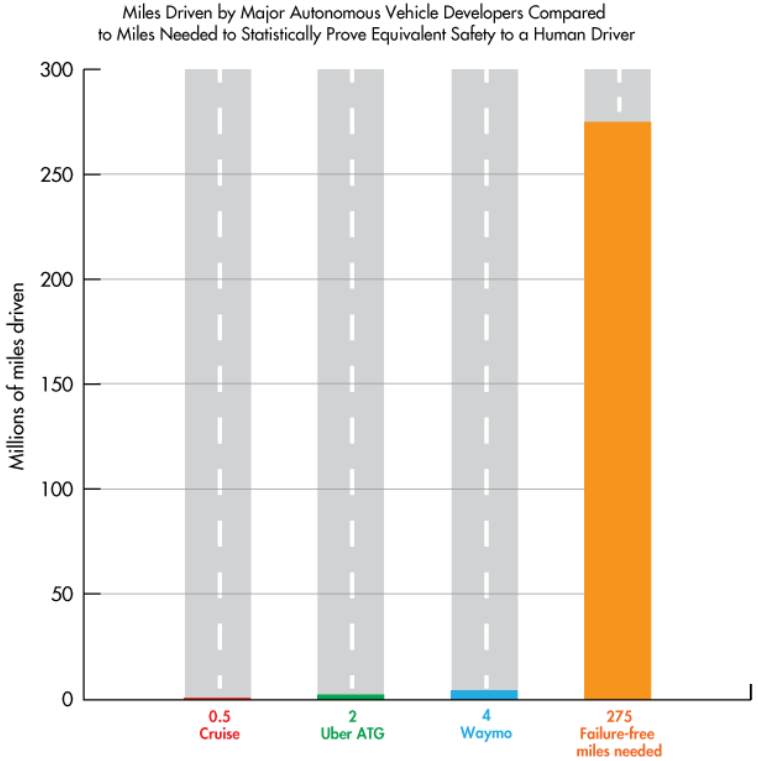
Perception et contrôle : stratégies de tests et moyens de simulation pour le véhicule autonome

Xavier FORNARI
ANSYS Systems

Physical miles driven necessary for L2-L5 validation

Akio Toyoda, President of Toyota @ Paris Auto Show
"It is estimated that some 8.8 billion miles of road testing, are required"

RAND – Driving to Safety Study
275M fault free miles needed to achieve equivalent safety to human driver



https://www.electronicdesign.com/automotive/testing-unknown-real-problem-autonomous-vehicles

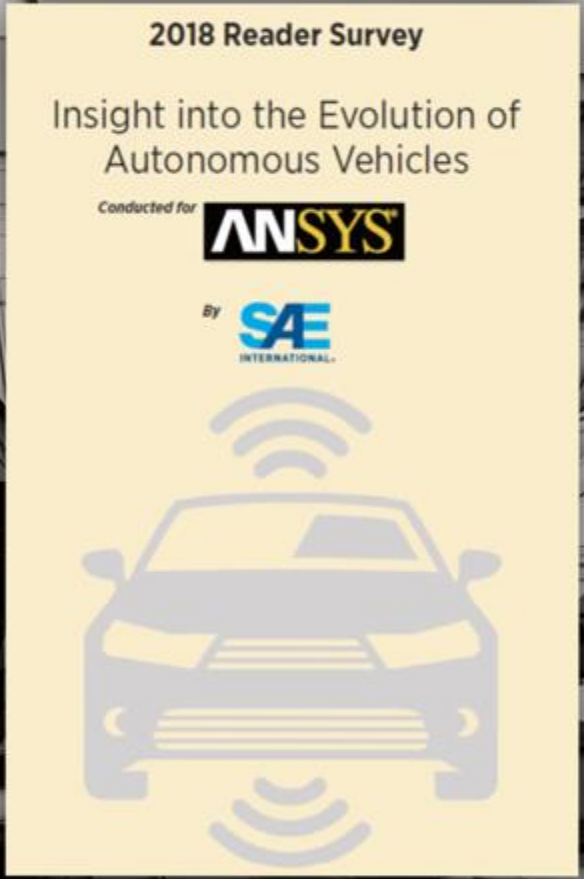
Demonstrating Safety is the Critical Engineering Challenge

Exclusive research by ANSYS and SAE

The SAE Automotive Industry Survey Confirmed:

"Public Confidence and Adoption" is the number 1 barrier to the widespread adoption of fully autonomous vehicles

Demonstrating Safety is the Critical Engineering Challenge



ANSYS addresses all key elements of autonomous vehicles

Sensors

Automated Driving Software

Computing Platform

Human-Machine Interfaces

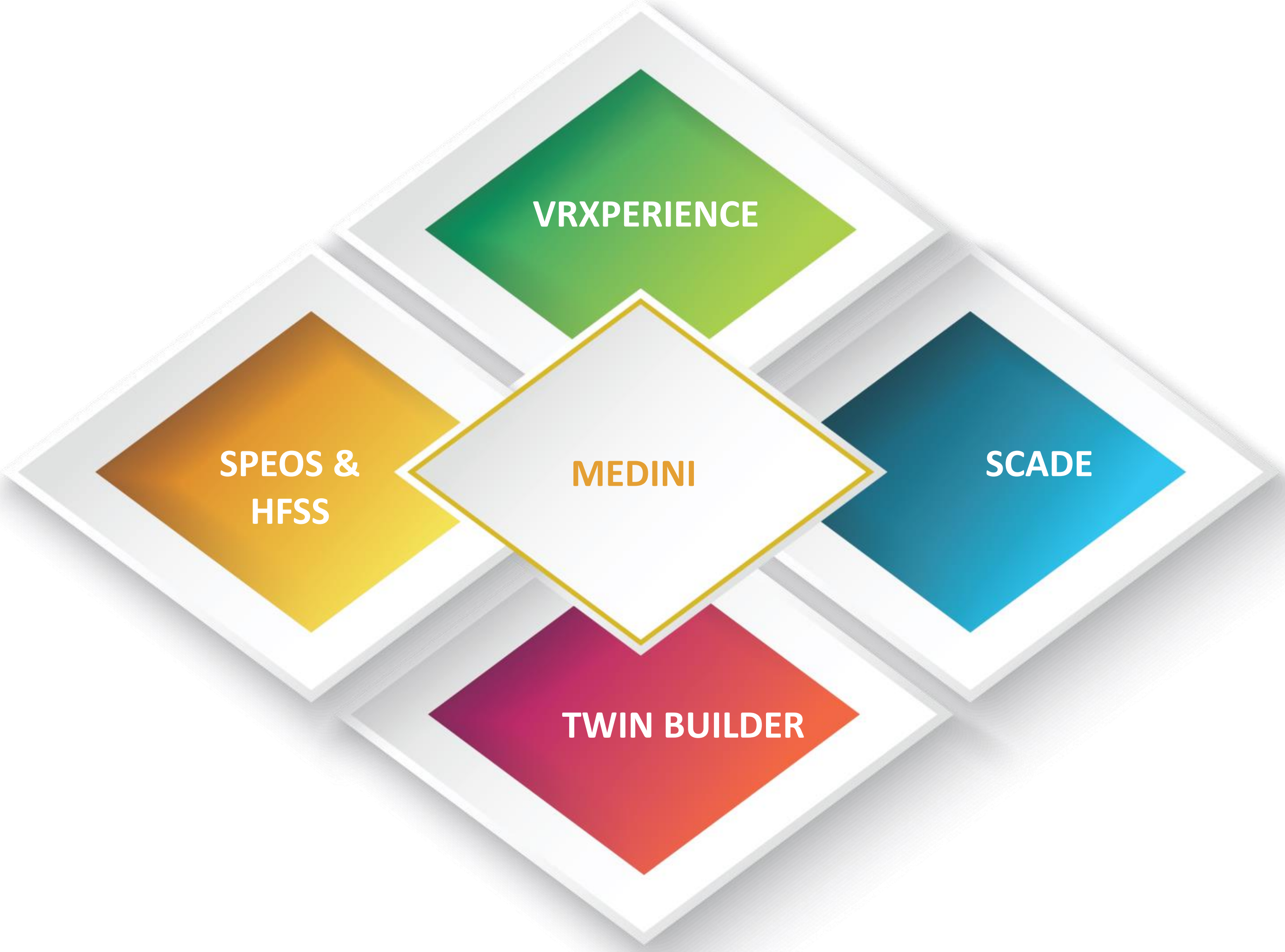
Control Software

Vehicle Platform

ANSYS Digital Safety Solutions for ADAS/AD

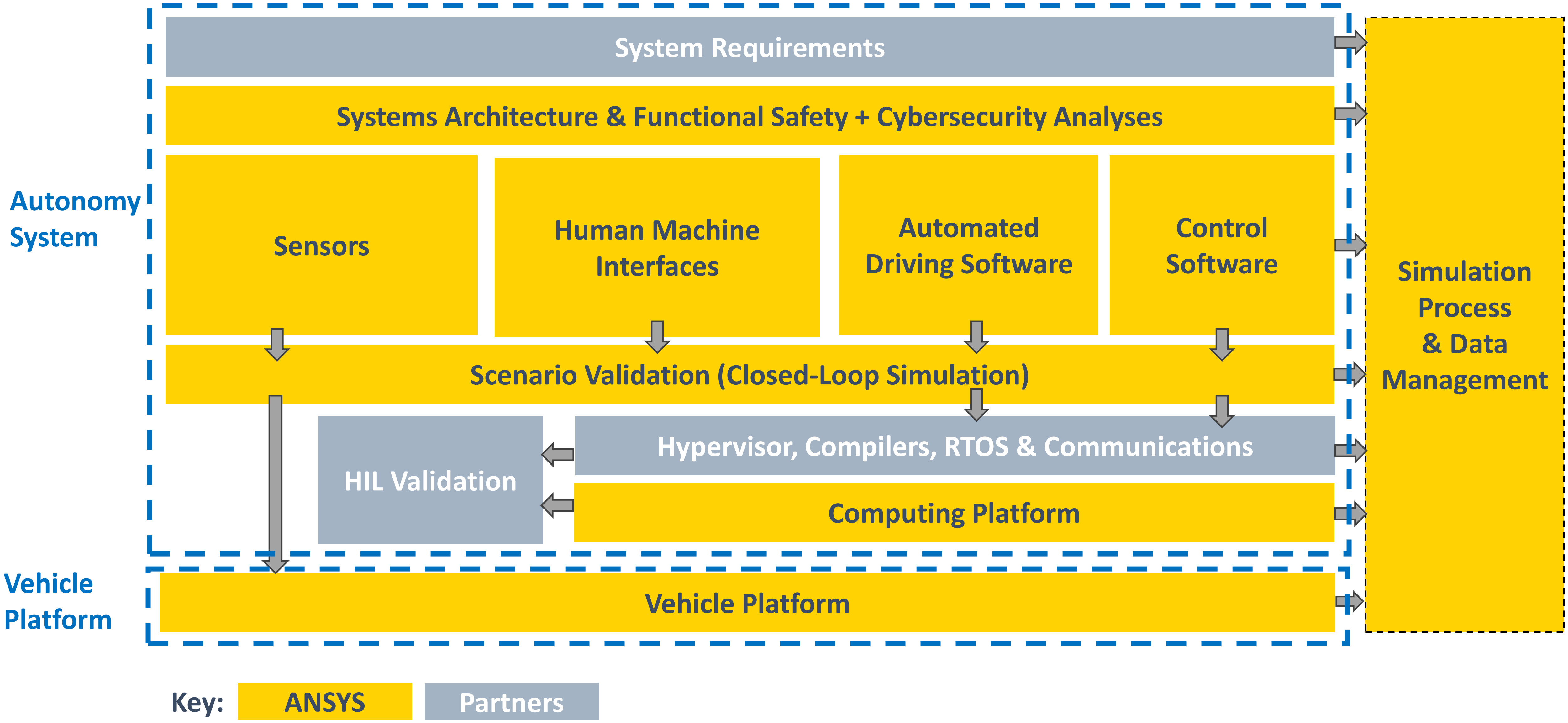
SIMULATION PROCESS AND DATA MANAGEMENT

ANSYS Digital Safety Solutions for ADAS/AS



MINERVA and OPTISLANG

ANSYS AV Technology Stack



1. Functional Safety & Cybersecurity Analysis
2. Sensors
3. Closed-Loop Simulation
4. Control Software
5. Automated Driving Software
6. Vehicle Platform

Safety of ADAS/AD Systems

01 Perception

02 Motion Planning

03 Motion Execution

Safety of the Intended Functionality (SOTIF)

Functional Safety Analysis (FuSa)

FuSa *vs.* SOTIF

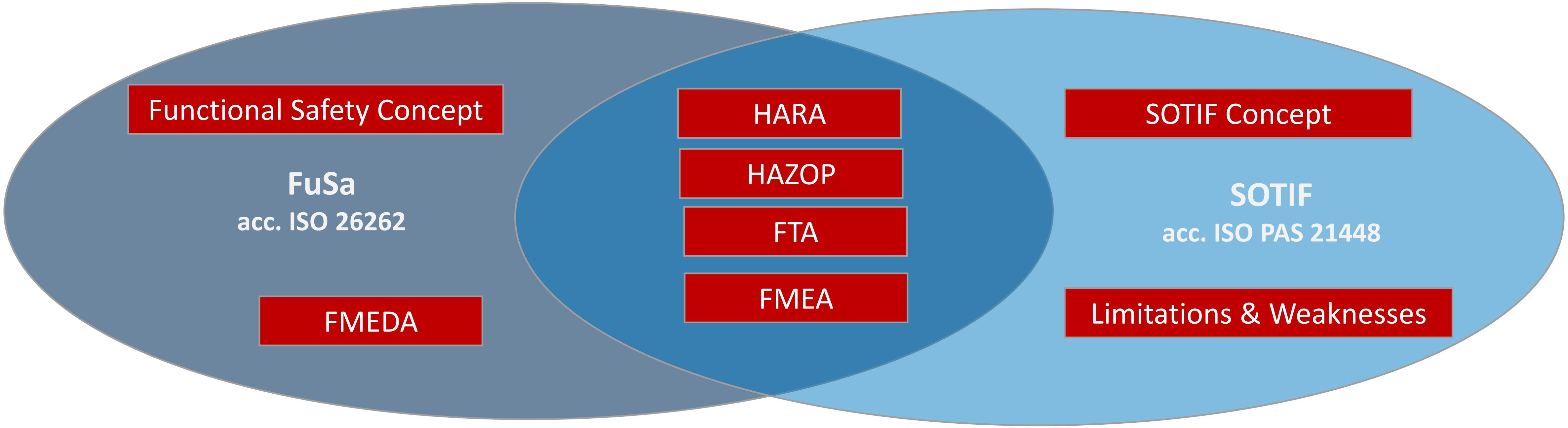
FuSA

- Addresses safety of the E/E control system
- Hazards induced by system failures (e.g. control software bug, bit flip in memory, etc.)

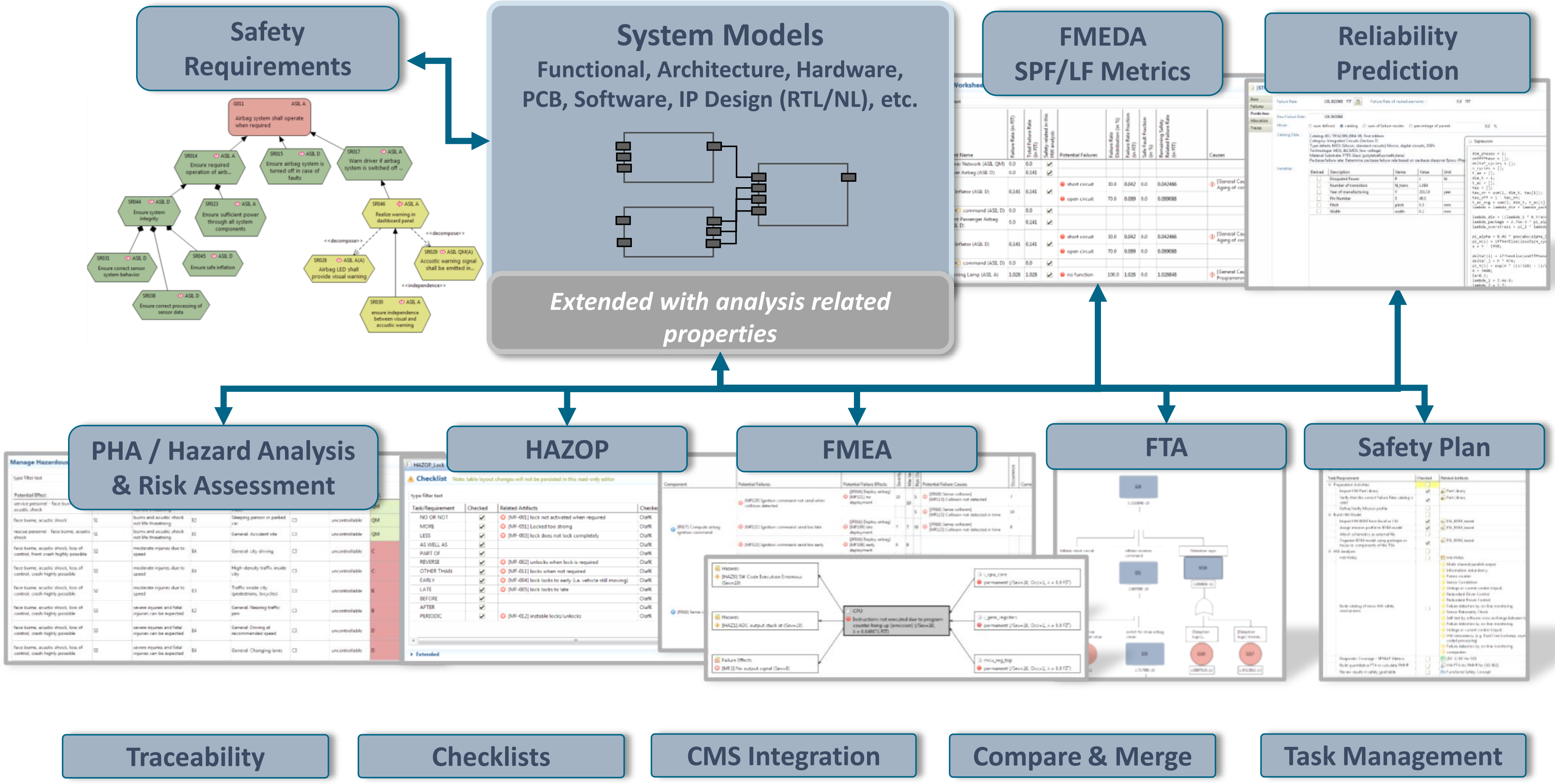
SOTIF

- Addresses safety of the complete ADAS/AD System *incl.* sensors and (AI-based) perception software
- Specific interest in Hazards due to limitations (e.g. weather conditions, radar echoes due to metallic bridge, etc.)

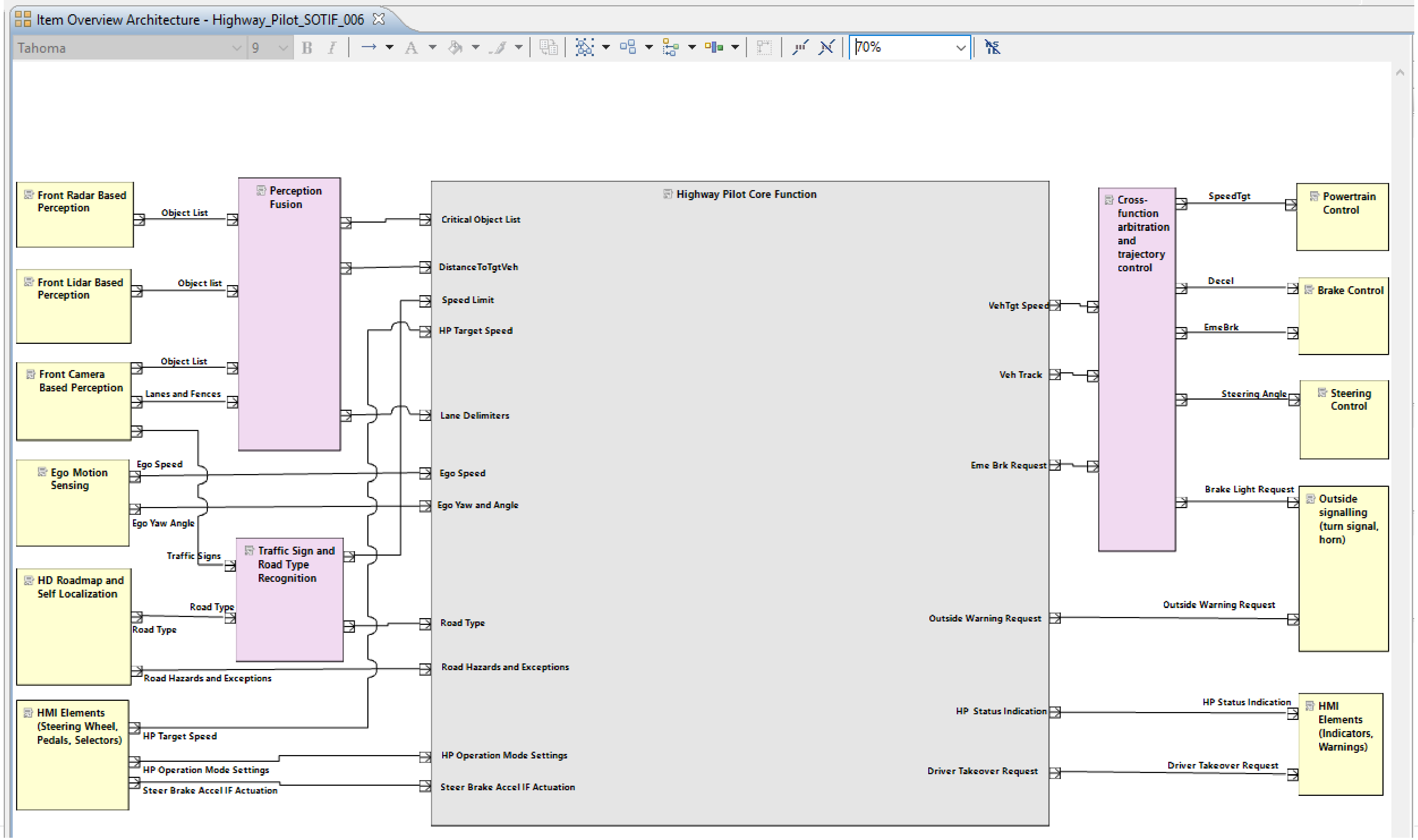
Methods in the FuSa and SOTIF Processes



ANSYS provides a model-based, system oriented solution for functional safety analysis (FuSa)



Architecture Modeling is common for FuSa and SOTIF: Highway Autopilot example



SOTIF Hazard Analysis and Risk Assessment (HARA): Establishing functions and malfunctions for the Highway Autopilot example

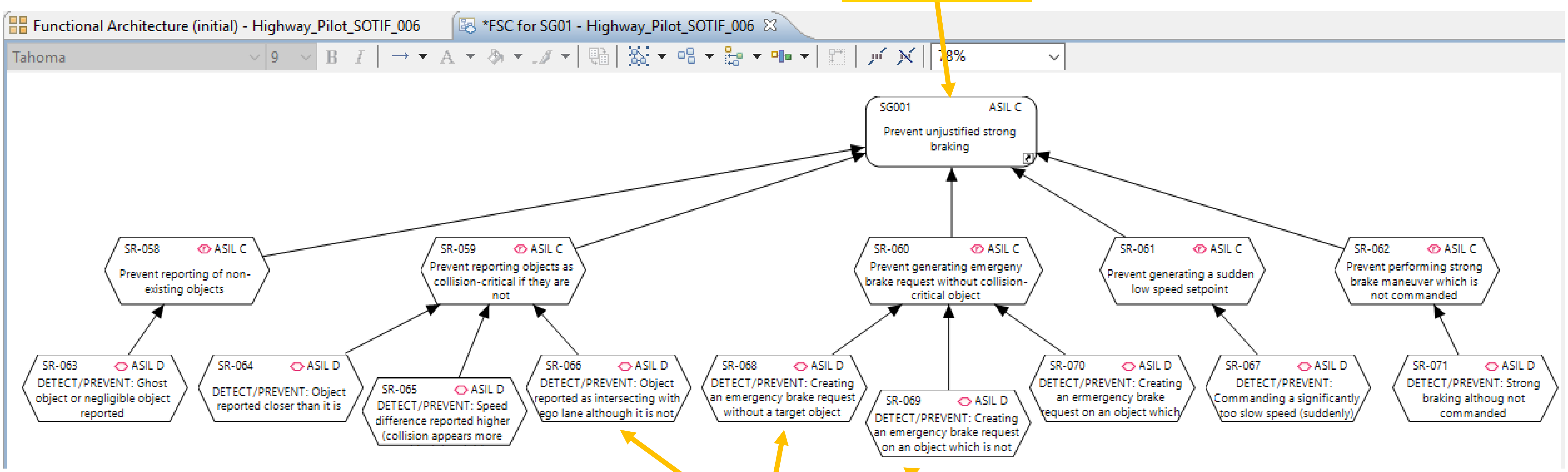
Scenario Analysis

type filter text

Location	Environment	Operation Mode of Item	Traffic and People	E (Combined Exposure)	Malfunctioning Behaviour	Hazard	Severity
Motorway	Daytime, dry and sunny	Highway Pilot active, speed controlled (free running)	Other car overtakes, merges in and immediately after brakes strong	E2	[MF-049] No emergency braking reaction on close motor vehicle.	Crashing into passenger car from behind (high delta speed)	S3
Motorway	Daytime, dry and sunny	Highway Pilot active, speed controlled (free running)	Other car overtakes, merges in and immediately after brakes strong	E2	[MF-056] Emergency braking comes late or is too weak	Crashing into passenger car from behind (high delta speed)	S2
Motorway	Daytime, dry and sunny	Highway Pilot active, speed controlled (free running)	Other car, truck or motorcycle following closely behind	E4	[MF-110] Unjustified strong braking	Following car crashes into ego car from behind (high delta speed)	S3
Motorway	Night time, heavy rain	Highway Pilot active, speed controlled (free running)	Motorcycle on ego lane	E2	[MF-049] No emergency braking reaction on close motor vehicle.	Crashing into motorcycle	S3
Motorway	Night time, heavy rain	Highway Pilot active, speed controlled (free running)	Motrocycle on ego lane	E2	[MF-049] No emergency braking reaction on close motor vehicle.	Crashing into motorcycle	S3

Safety Concept is improved; Requirements are refined

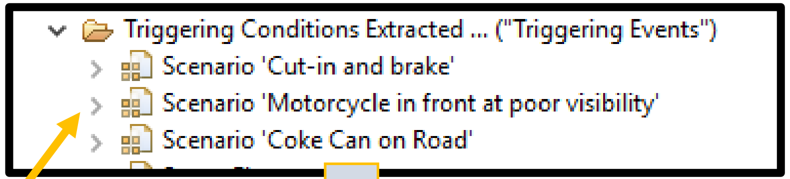
Safety Goal (from HARA)



Functional Requirements:
Initial iteration: imported from normal function development (medini offers interfaces to all major requirements engineering tools)
Following iteration (system improvement): derived from safety analysis (e.g. FTA events)

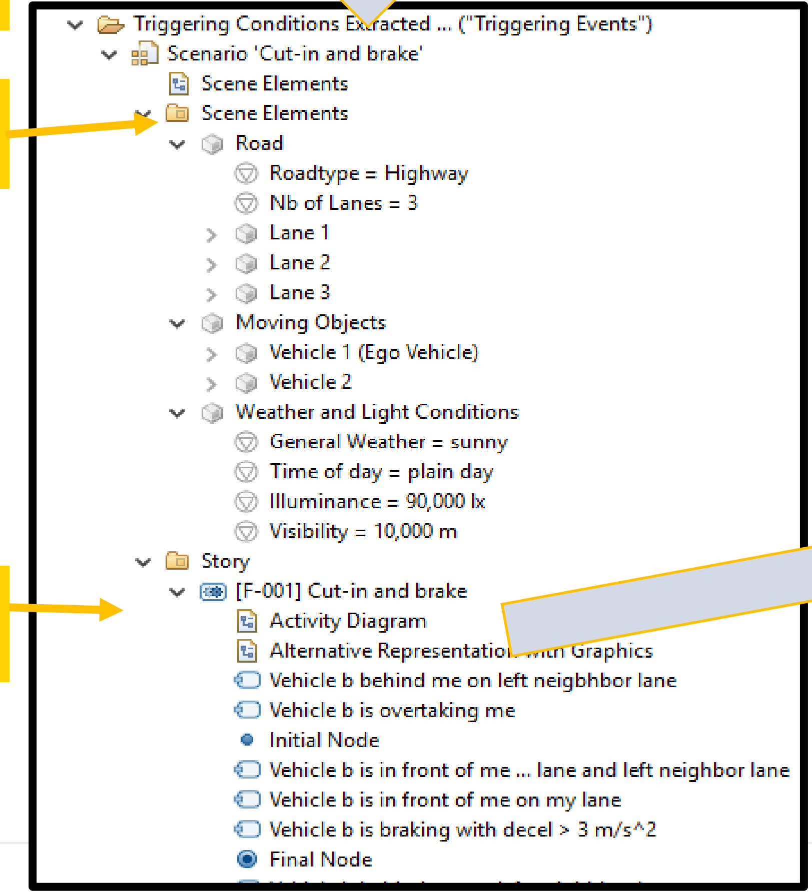
SOTIF Triggering Conditions Analysis:
Building in medini Scenarios to be simulated in VRXPERIENCE

Import/Export to OpenSCENARIO standard under preparation

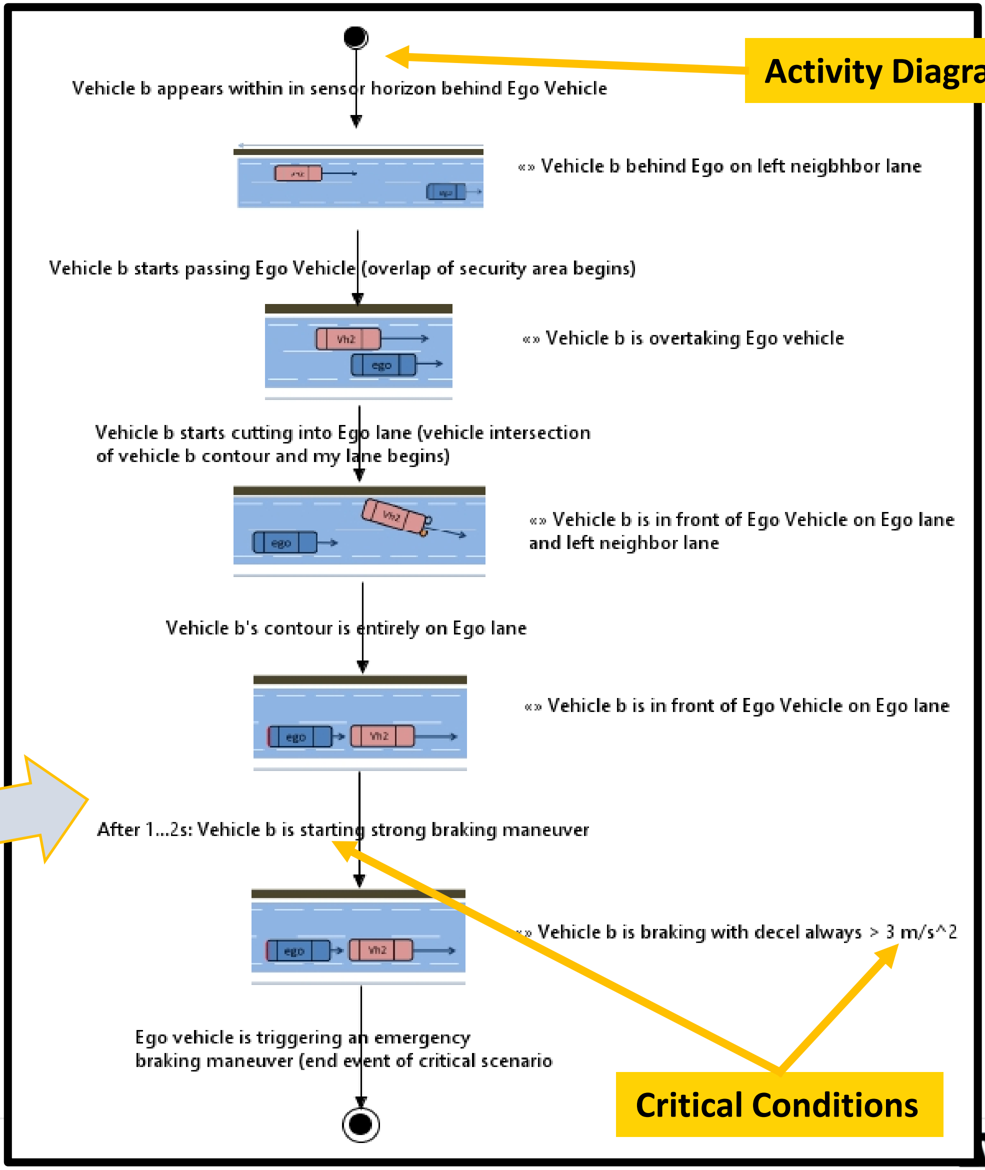


Scene Elements: What is around

Story: What is happening



Activity Diagram



Critical Conditions


Running SOTIF Driving Scenarios in VRXPERIENCE

1 Functional Safety & Cybersecurity Analysis

2 Sensors

3 Closed-Loop Simulation

4 Control Software

5 Automated Driving Software

6 Vehicle Platform

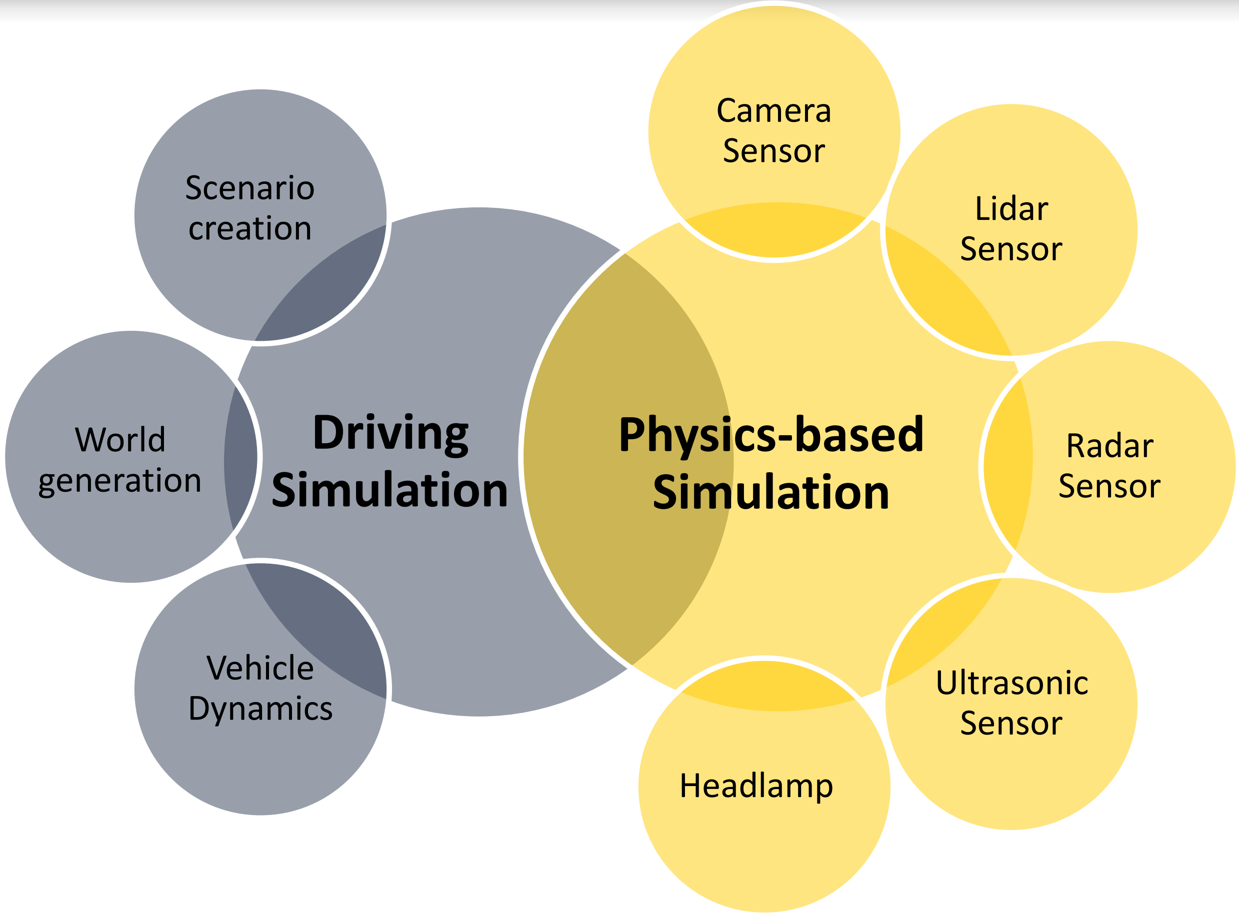
ANSYS AV open and customizable simulation environment

A comprehensive simulation software

Providing:

- ✓ **Sensors & light models**
- ✓ **3D world**
- ✓ **Scenarios**
- ✓ **Vehicle dynamics**

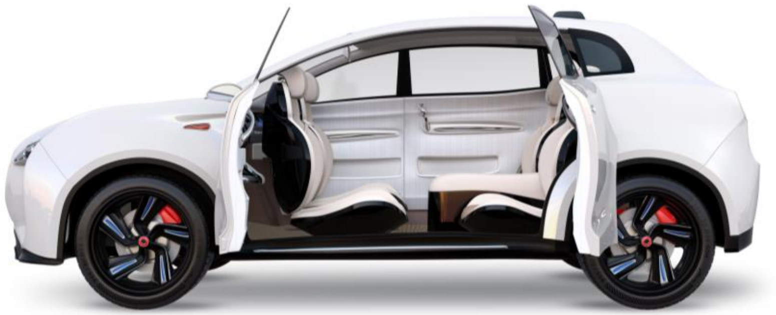
+ closed-loop platform
+ development tools



ANSYS addresses all key AV sensors

Radar

Camera



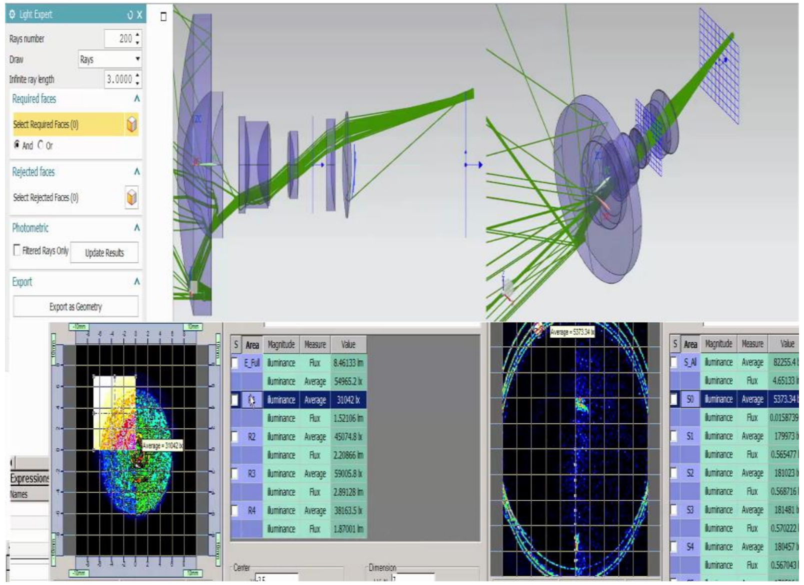
Lidar

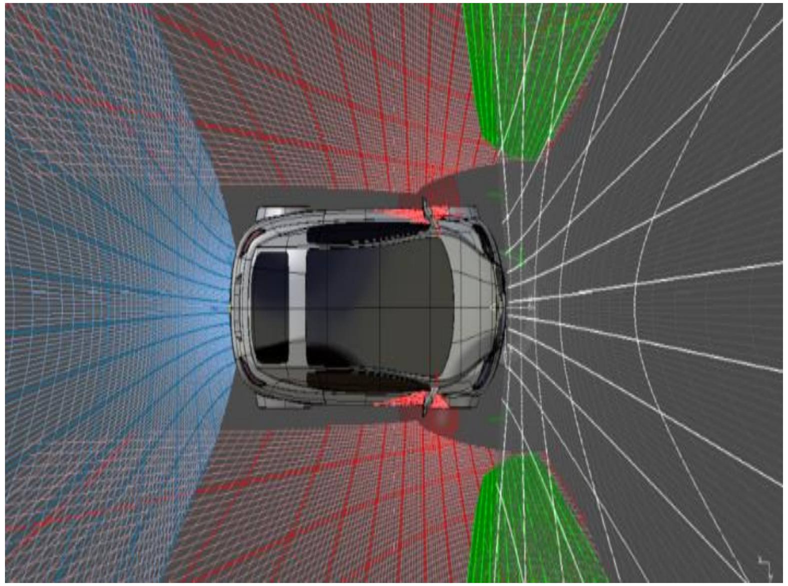
Ultrasonic

Three phases for each sensor

Component Development	Vehicle Integration	Scene Simulation

Camera: Simulation from component design to full scenarios





Component Development

Optical, Thermal, Structural

Design & Analysis

Vehicle Integration

Vision Performance Analysis

Position Optimization

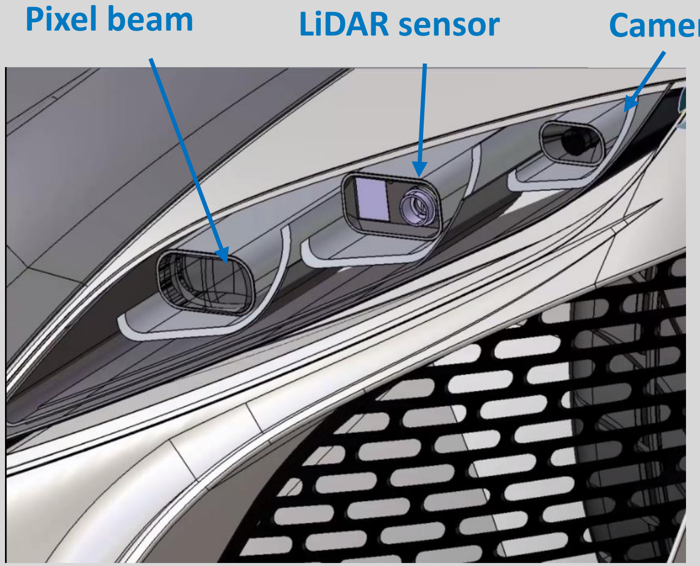
Scene Simulation

Vision System

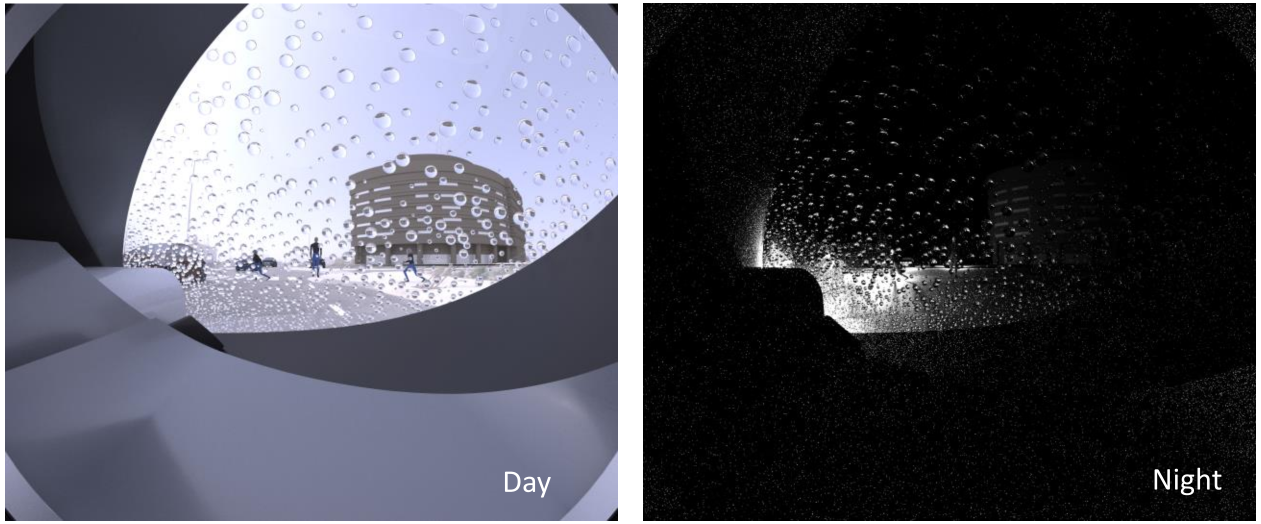
Test & Validation

Camera: Simulations in adverse weather conditions

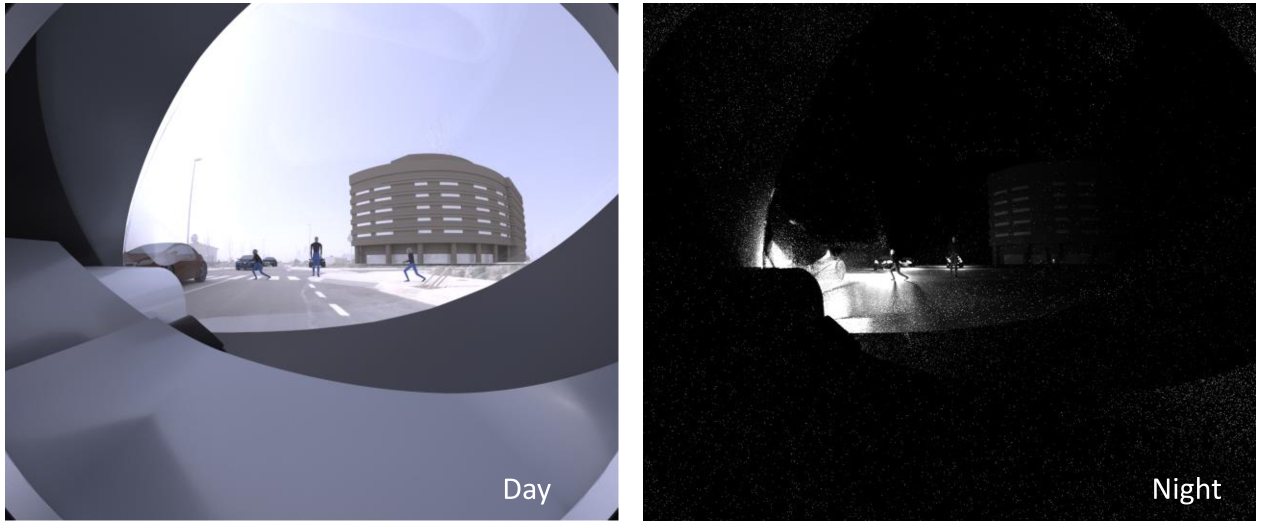
Simulations are performed using a headlamp model with an integrated camera and LiDAR sensor



Headlamp outer lens with water droplet build-up

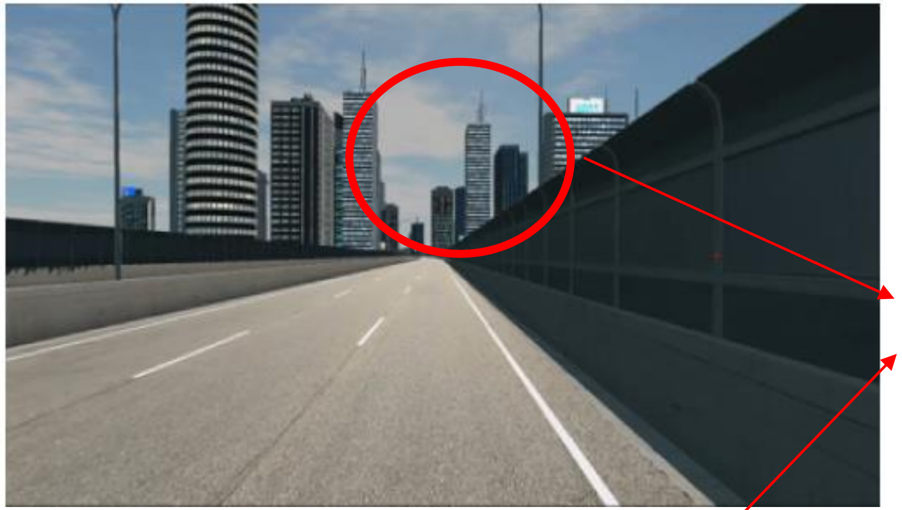


Headlamp outer lens with 3M hydrophobic film



Camera: Real world fault detection - solar glare

Traditional Rendering Engine

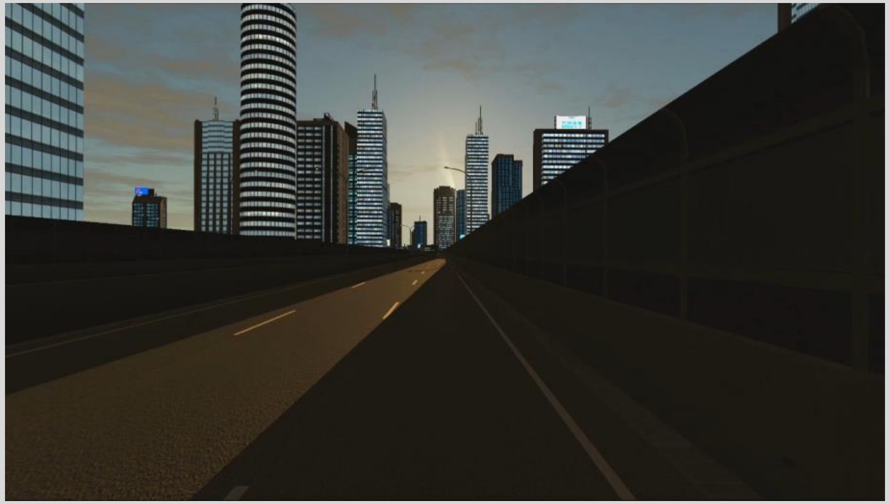


Sensors Fails

No Sun Glare Detected

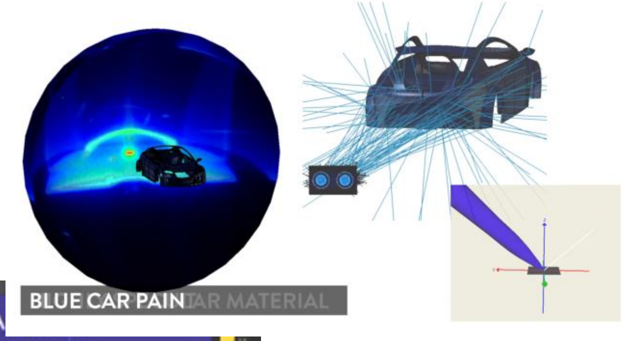
Requires further physical testing on road

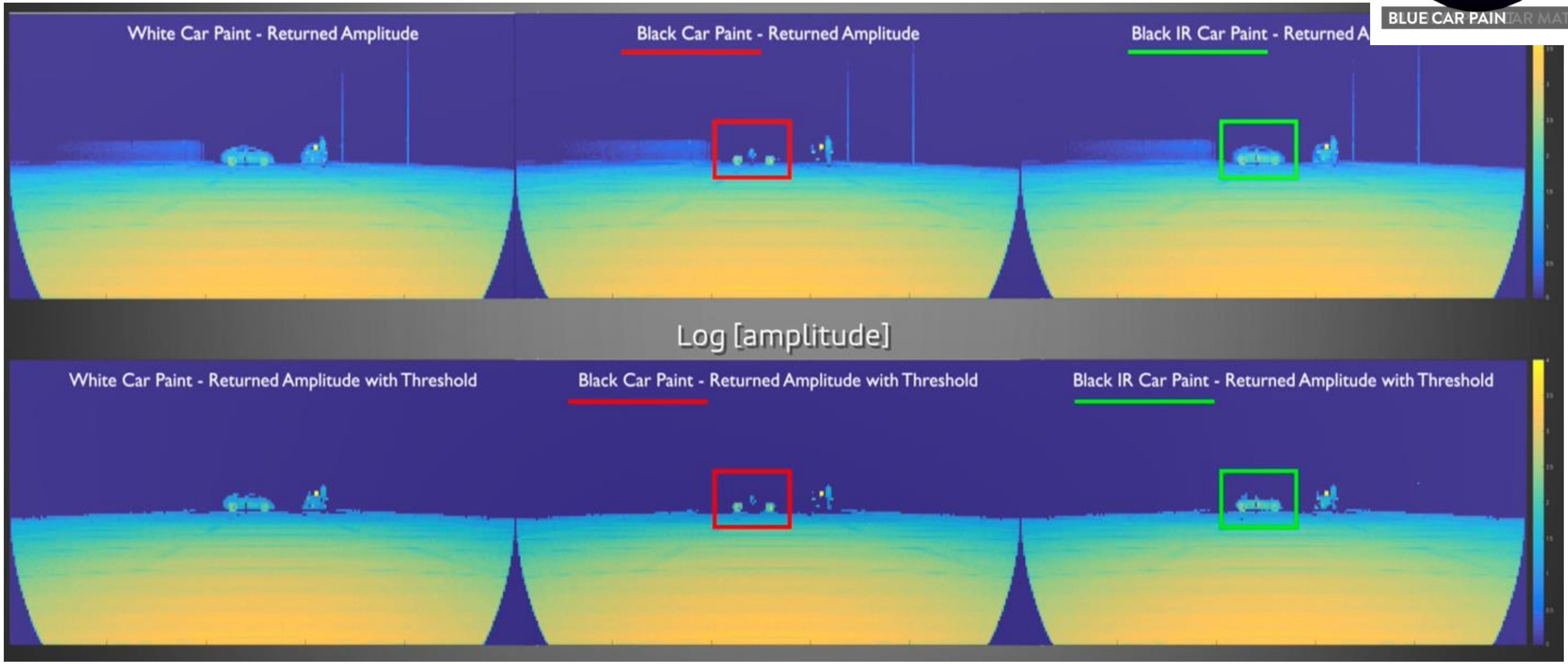
ANSYS's Physically Accurate Simulation



LiDAR: Real world fault detection - paint colors

Simulate LiDAR performance taking into account the IR reflectivity of car paints

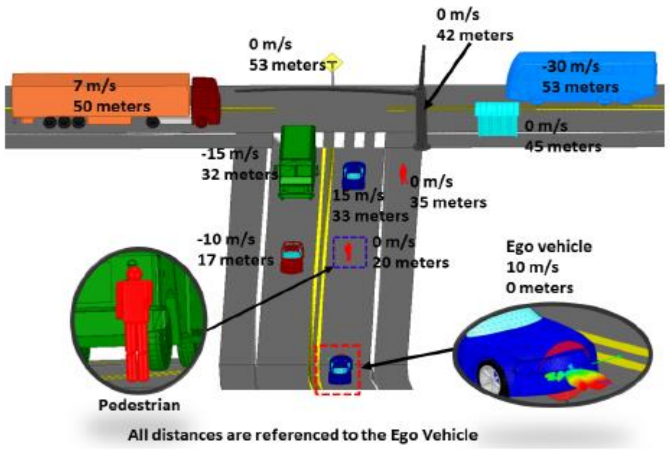


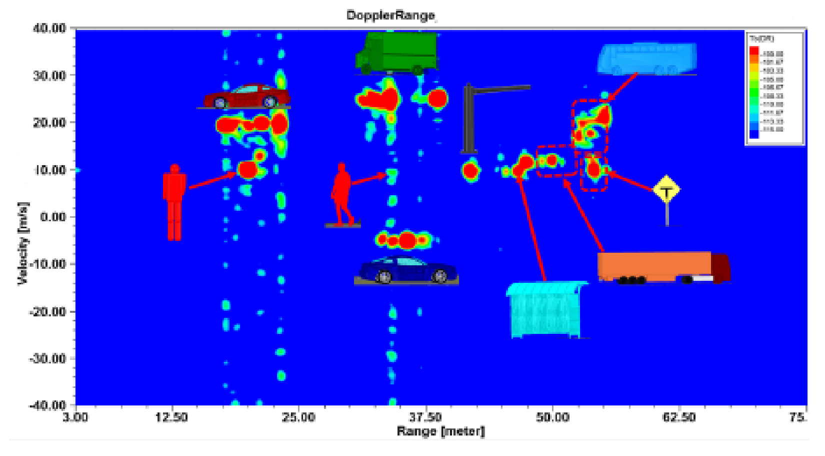


Radar: Real world fault detection - guardrails

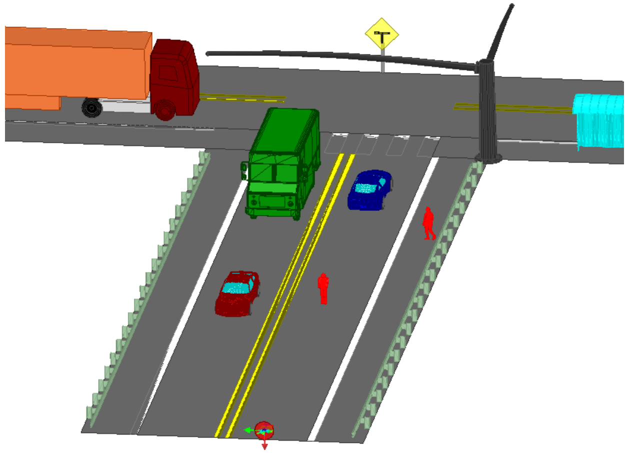
Radar Sensor Perception

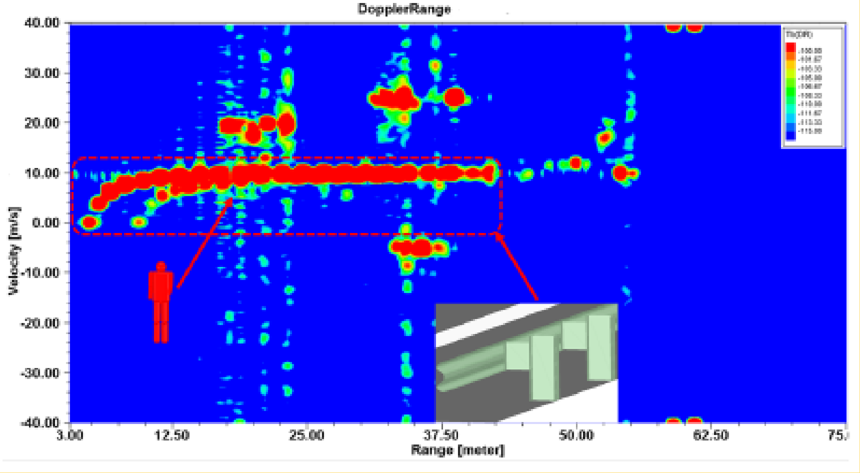
Base Scene





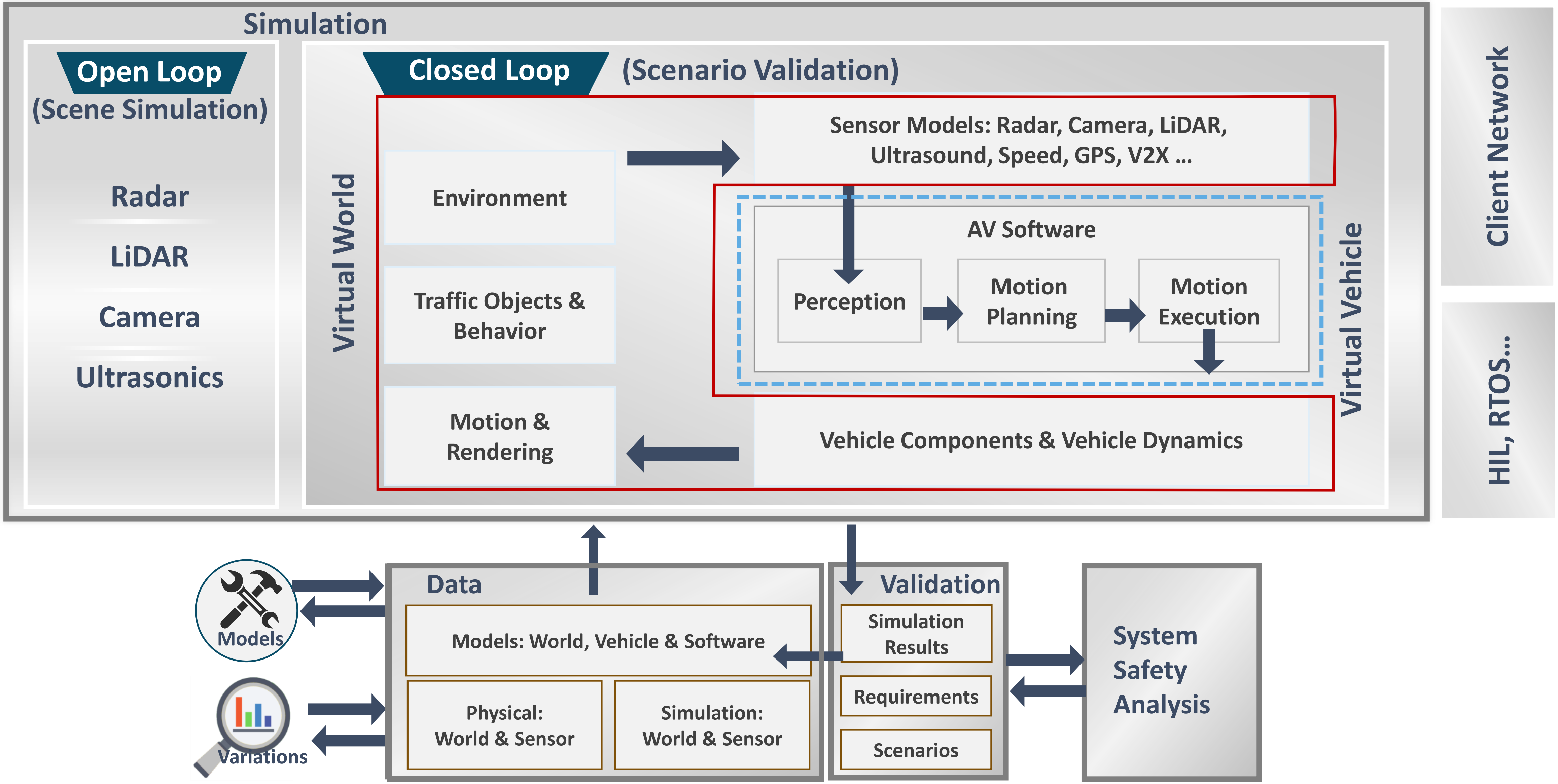
Adding Guardrails





1. Functional Safety & Cybersecurity Analysis
2. Sensors
3. **Closed-Loop Simulation**
4. Control Software
5. Automated Driving Software
6. Vehicle Platform

ANSYS AV open and customizable simulation environment



3D world model & preparation

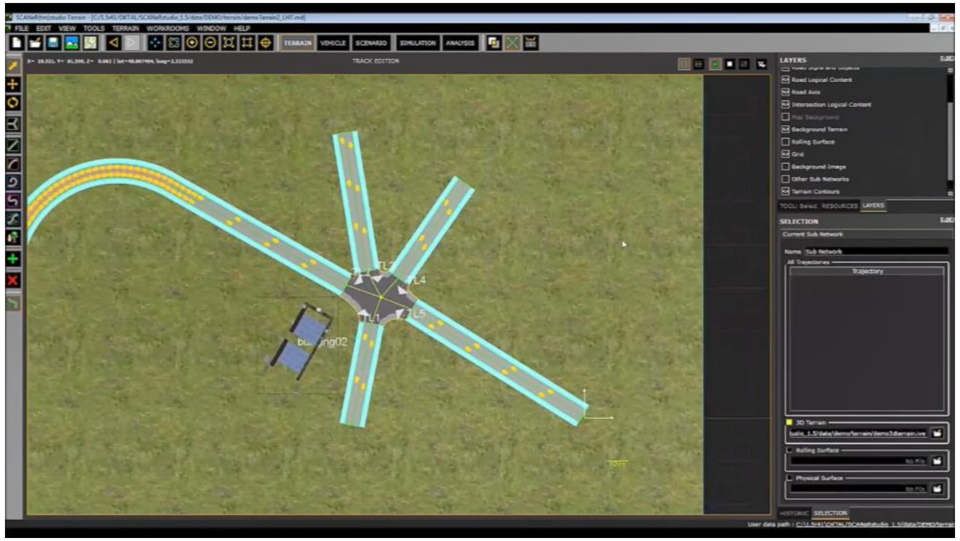
Support any process able to capture real world into simulation with very high fidelity road database and photorealism.

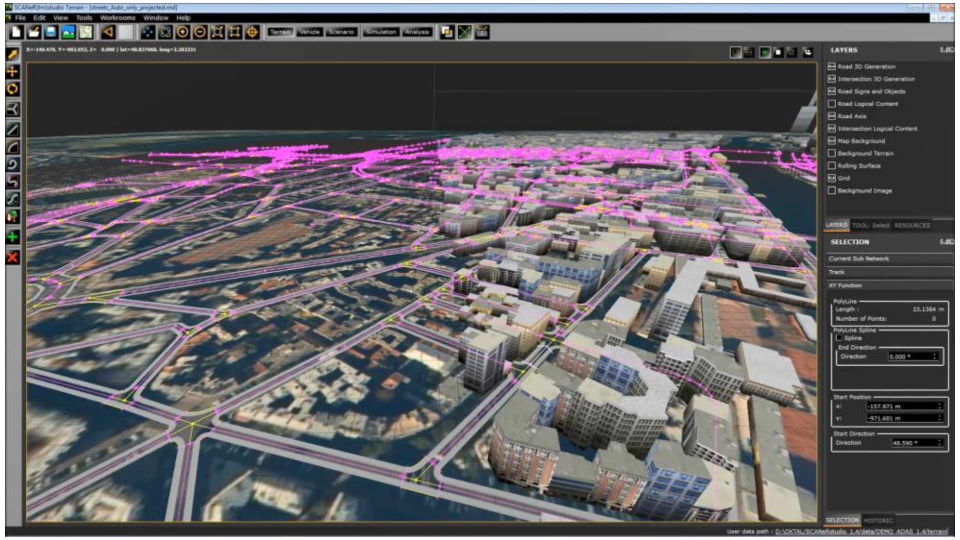
Key Features

- Ease the creation of 3D road environment
- openDrive compliant
- Import map data : OpenStreetMap, Here…
- Trim the world from libraries
- Set physics-based materials from libraries

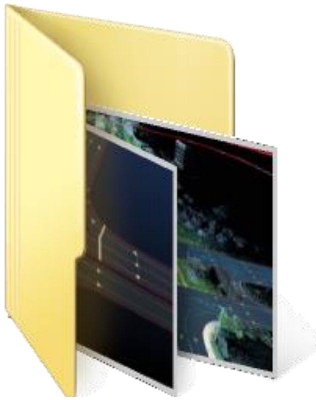
Use case

- Create high fidelity 3D world model
- Automate 3D world model creation for quick and fast simulation test



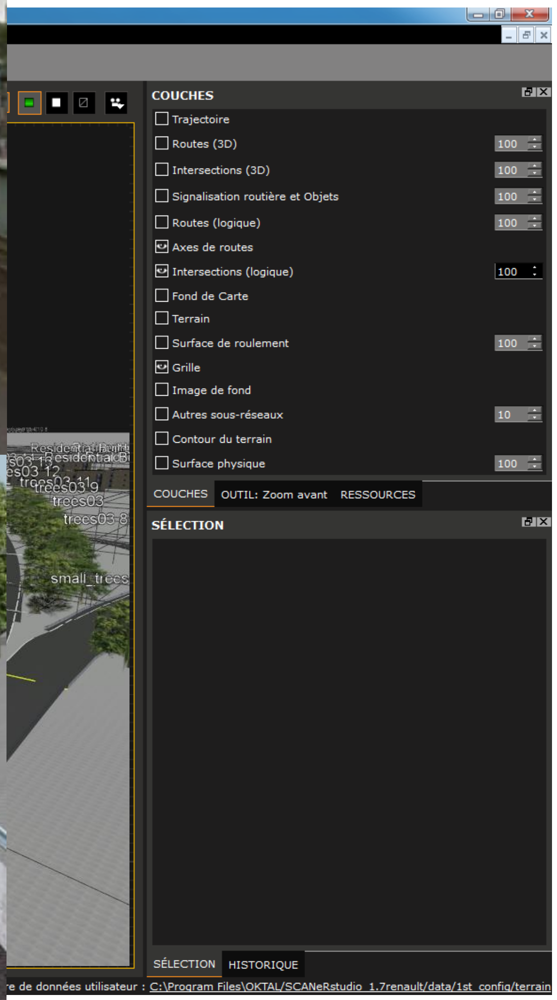


Example: Tomtom HD map import



HD Map

Courtesy of Renault



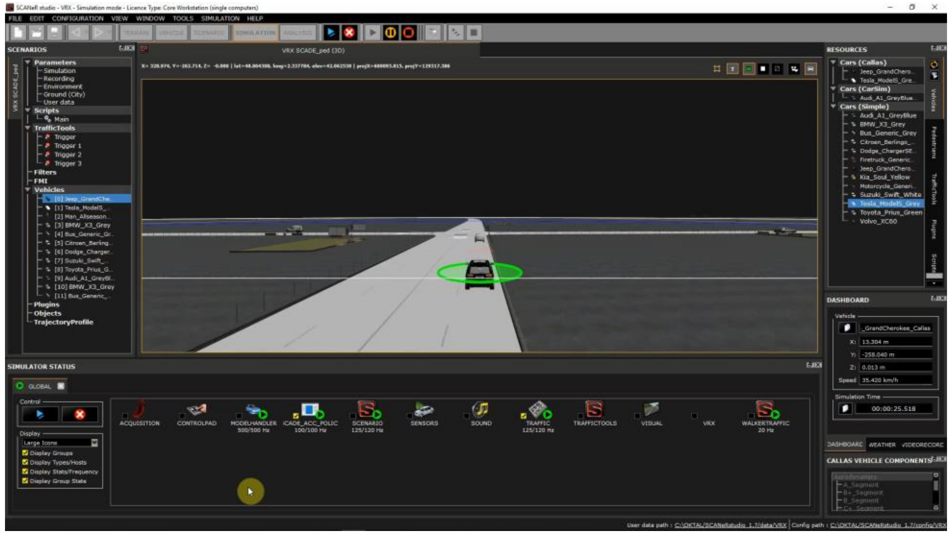
Scenario & Traffic

Key Features

- Bring ego car into a multi-agent simulated traffic model
- Traffic model based on AI able to generate any kind of traffic situations.
- Create scenario via script or GUI
- Automation of scenario from Test Plan.
- Large asset of car, trucks, motorbike, pedestrians, animals...

Use case

- Create dynamics driving scenario
- Create variability of scenario
- Automate scenario creation for massive simulation test



Sensors

Key Features

- **Ideal or physics-based model of Camera, Radar, Lidar, Ultrasonic sensors**

Use case

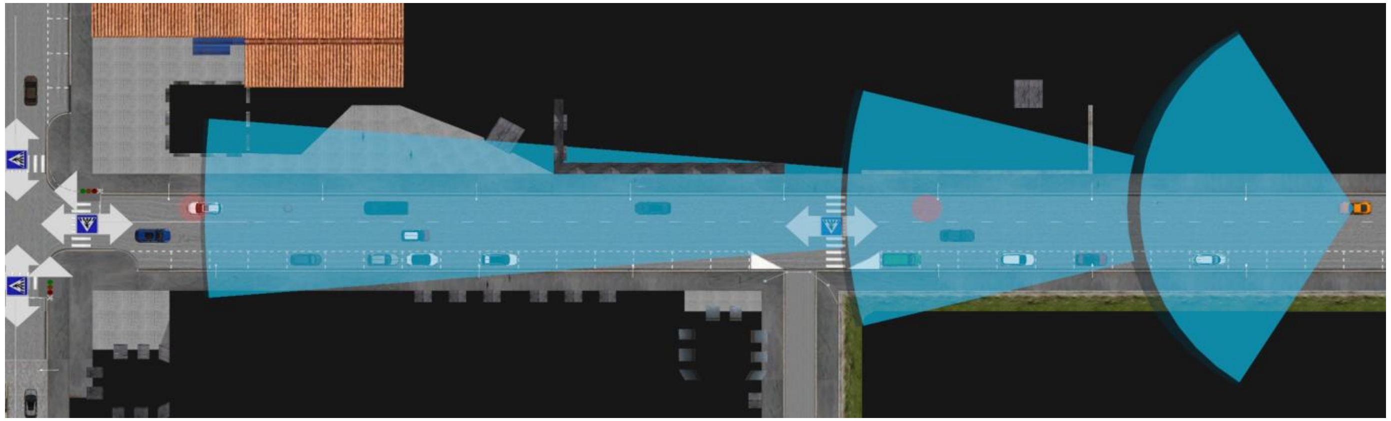
- **Model the ideal or physics-based behavior of sensors**

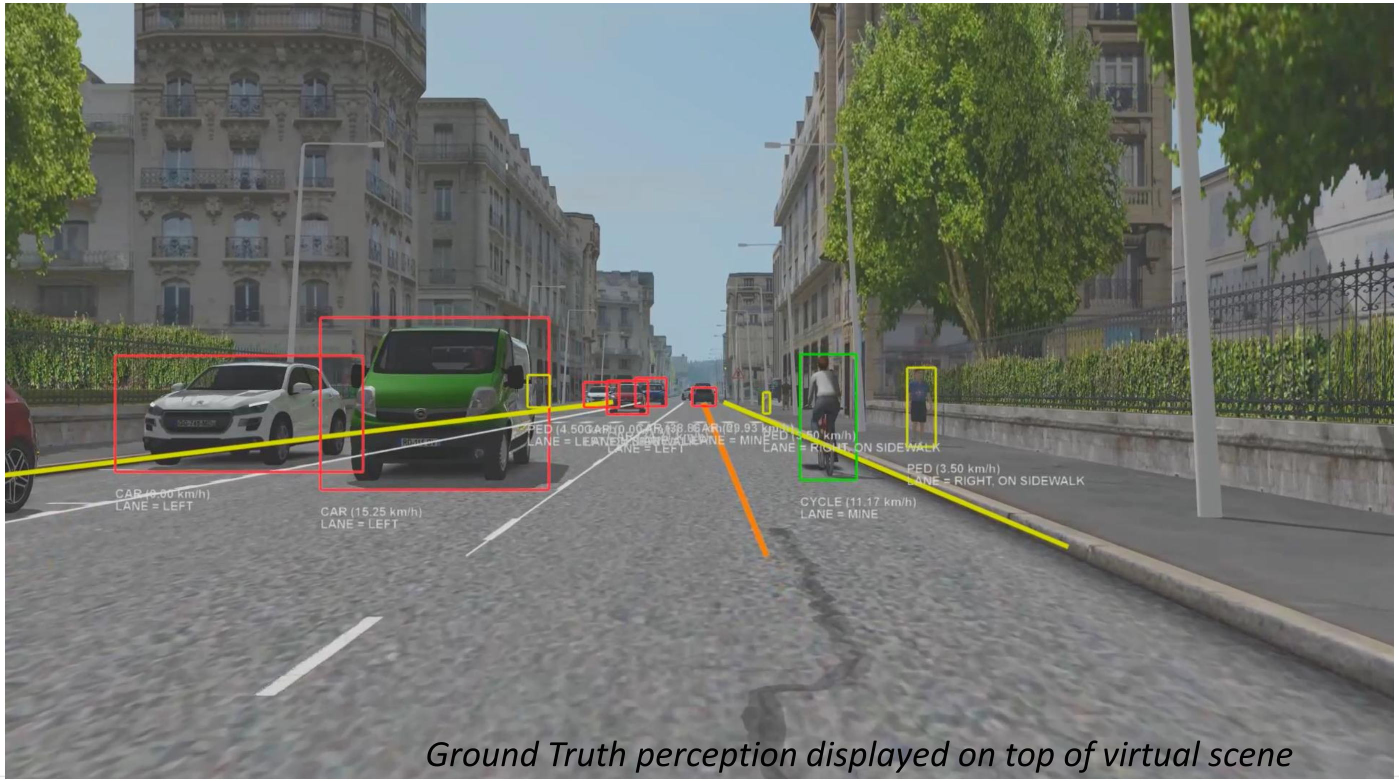
- **Develop and test:**
  - **perception, planning and control algorithm (physics-based)**

  or

  - **planning and control isolated from perception (ideal)**

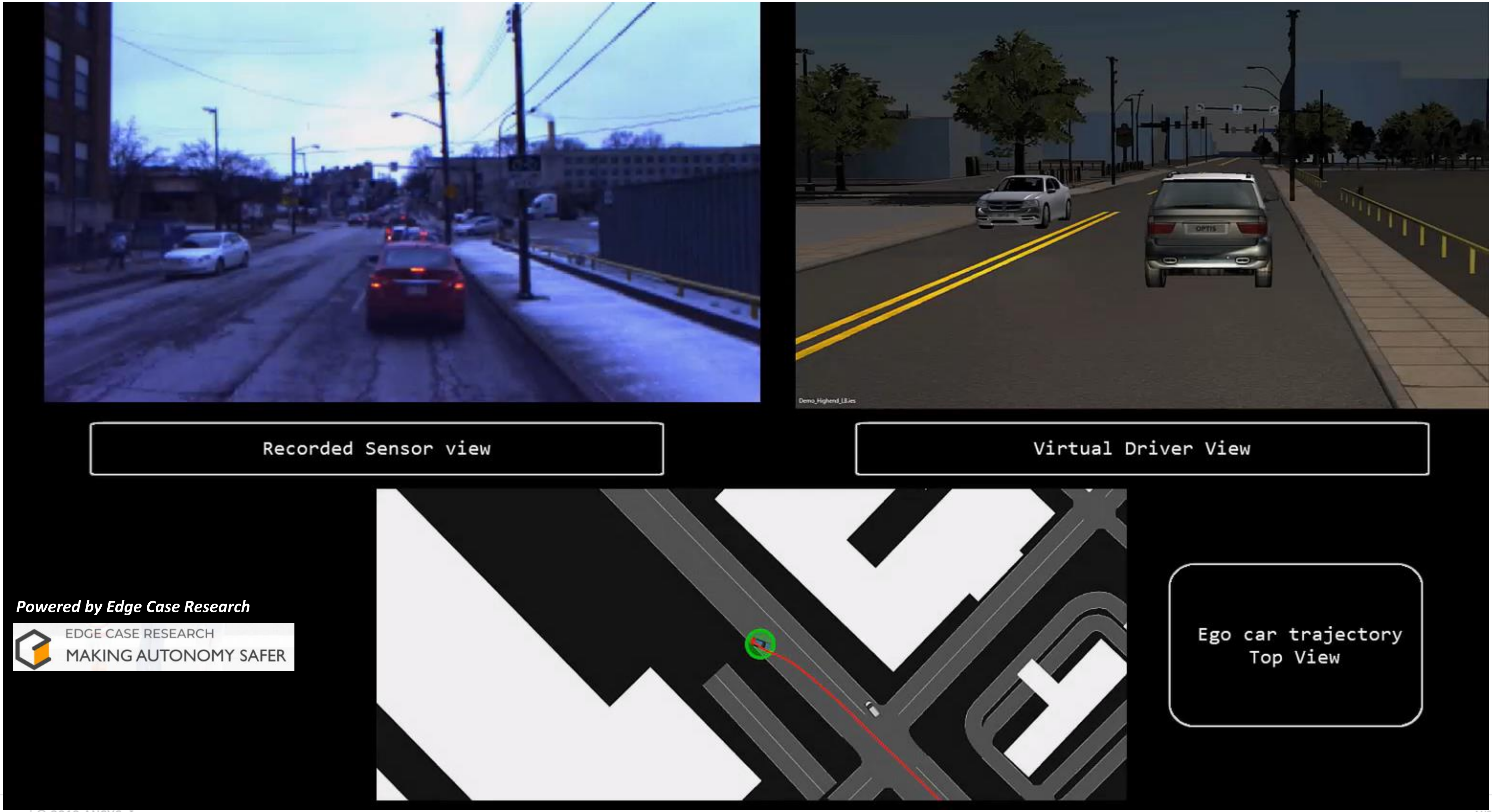
- **Test ADAS feature robustness**





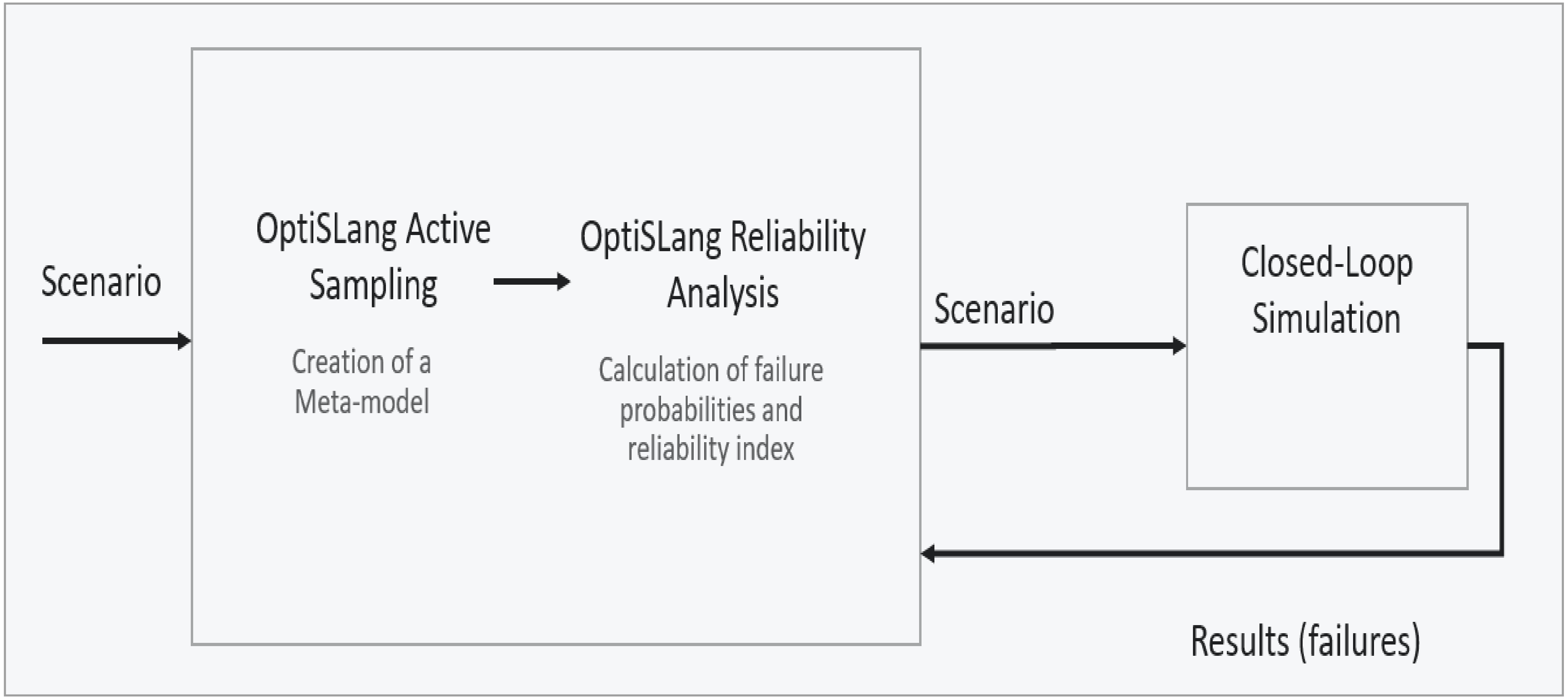
Ground Truth perception displayed on top of virtual scene

Scenario Re-Creation From Real Data : Left turn in Pittsburgh

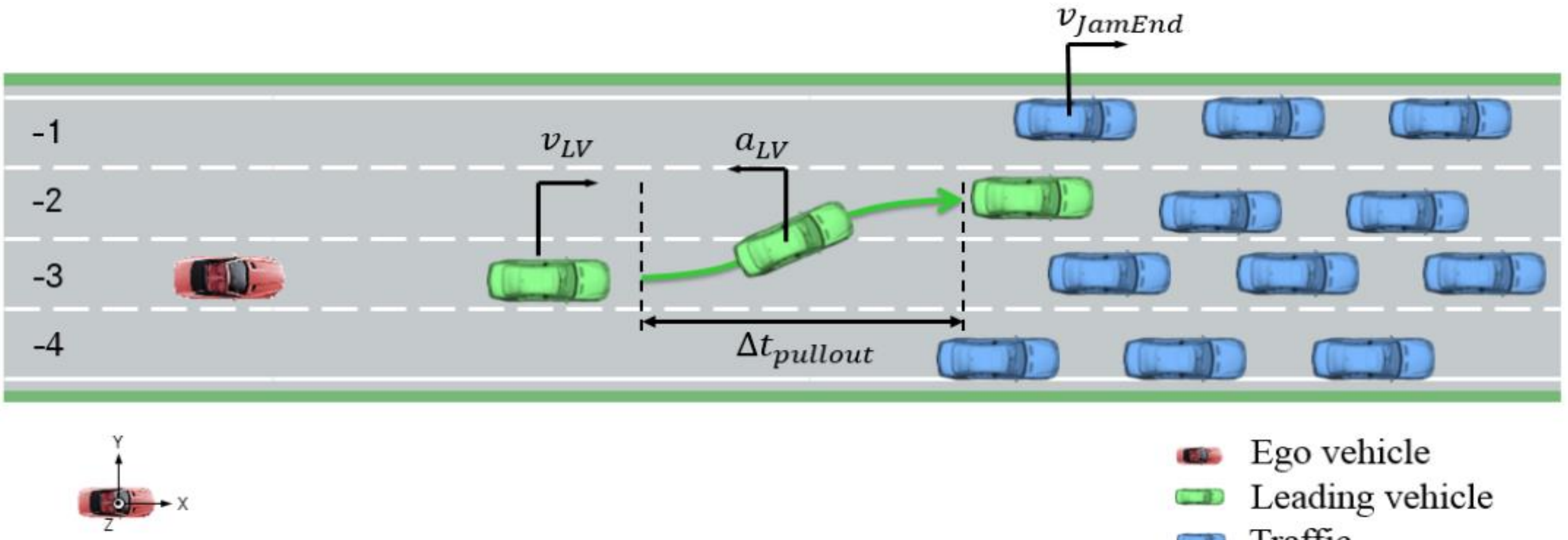


Scenario Variation using ANSYS optiSLang (Dynardo optiSLang®)

The goal is to perform robustness and reliability analysis for <u>parameterized driving scenarios</u> in a way that is much more efficient than Monte-Carlo Simulation.



Example of Scenario Variations (Jam-End, 9 parameters)



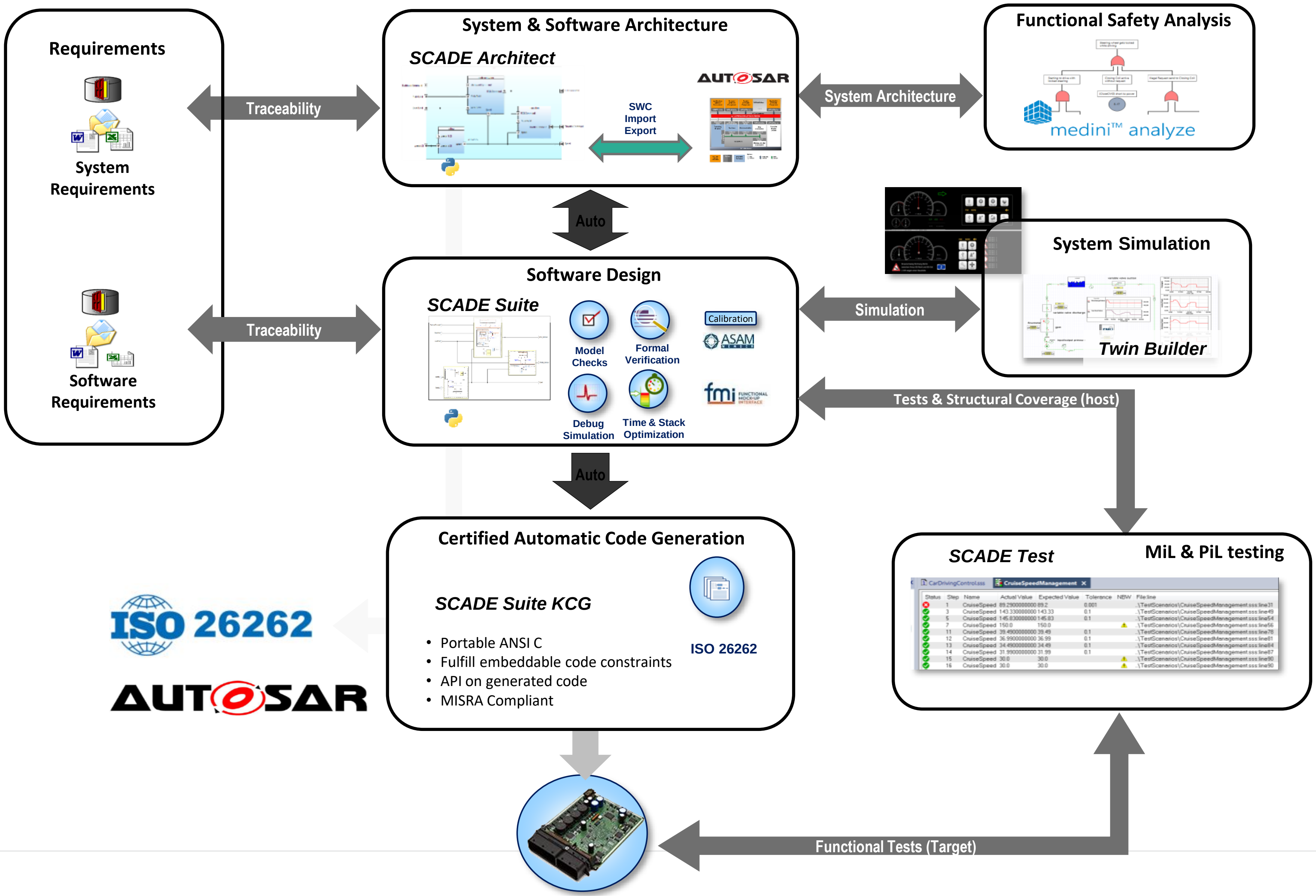
	Number of samples	Failure probability	Coeff. of variation	Reliability index
Limit TTC = 0.4				
MCS	39.420.000	$2.54 \cdot 10^{-6}$	10.0%	4.56
AS	16.000	$2.81 \cdot 10^{-6}$	9.1%	4.54
ISPUD+FORM	7.000+5.500	$2.31 \cdot 10^{-6}$	9.5%	4.58

28,500 simulation runs using optiSLang
vs.
39.420.000 using Monte-Carlo simulation


Massive Scenario Variation & Distributed Computing

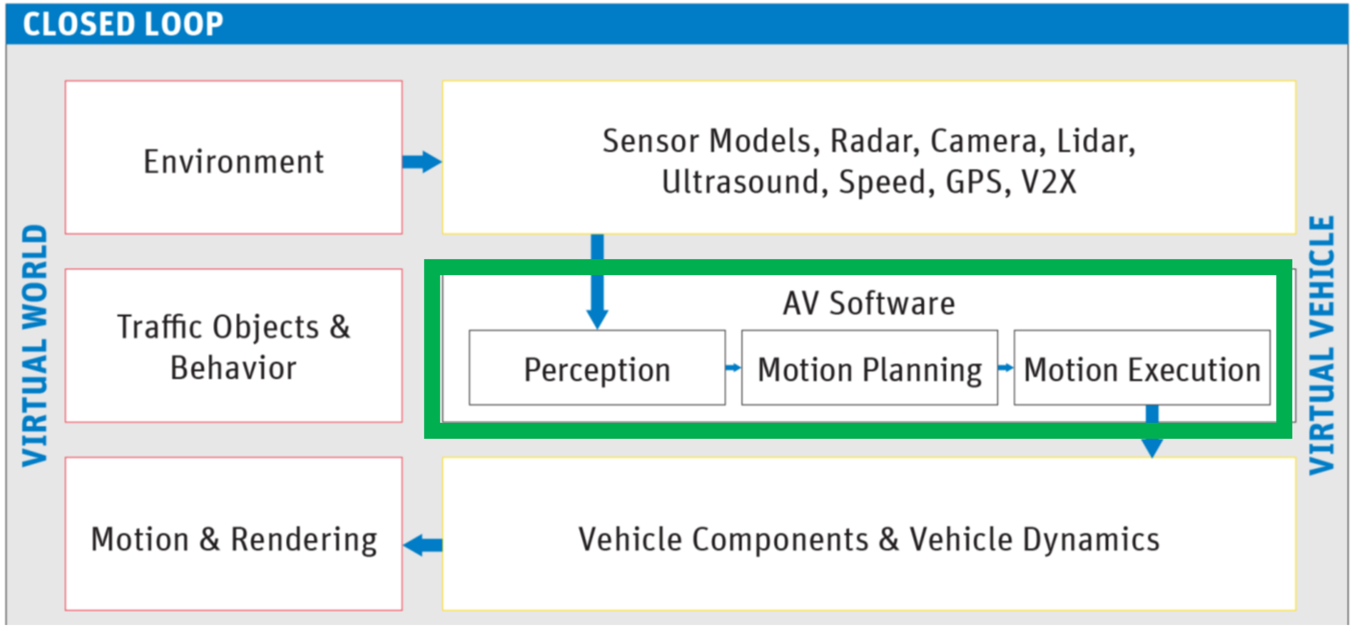
1. Functional Safety & Cybersecurity Analysis
2. Sensors
3. Closed-Loop Simulation
4. Control Software
5. Automated Driving Software
6. Vehicle Platform

ANSYS SCADE provides a model-based software development flow with ISO 26262 certified code generation and AUTOSAR compliance



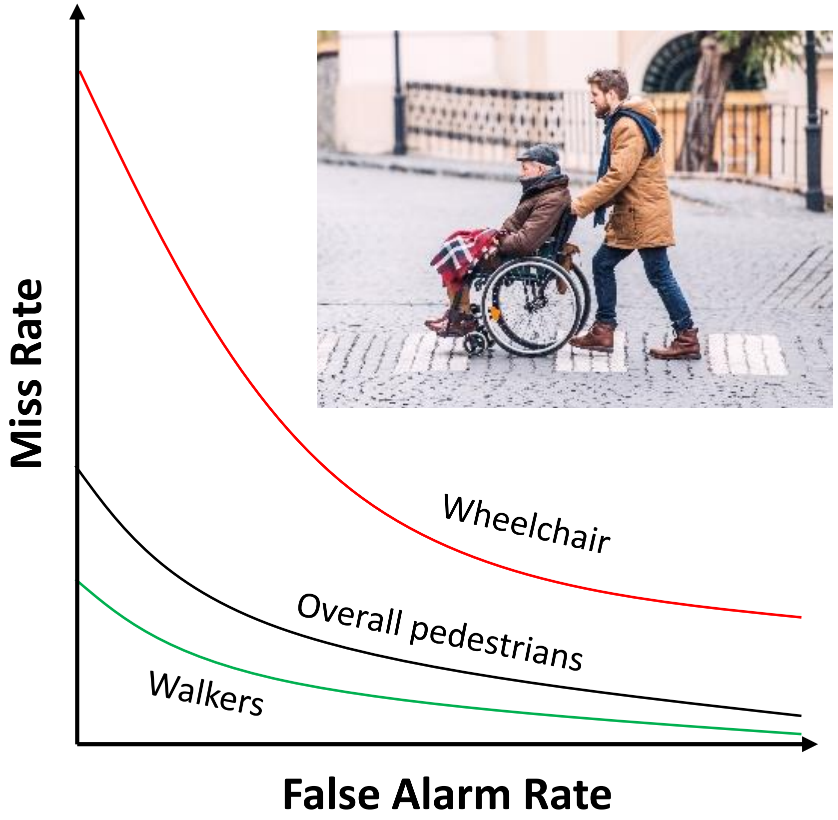
Closed-loop simulation of full AV stack (Software-in-the-loop)

- Physics-based sensor models allow testing of full AV software stack
- Run real-time asynchronous or externally synchronized simulation
- Distribute computing and rendering node on several CPU or GPU
- Massive simulation on HPC



1. Functional Safety & Cybersecurity Analysis
2. Sensors
3. Closed-Loop Simulation
4. Control Software
5. **Automated Driving Software**
6. Vehicle Platform

Why are edge cases a problem?



Perhaps your autonomy can detect 999 out of every 1,000 images with pedestrians that walk on two legs.

But what if it only detected 700 out of every 1,000 images with pedestrians that use wheelchairs?

P (accident | wheelchair) should be the same as P (accident | walker)

There are many more edge cases ☹ !

Finding and identifying the root causes of these edge cases

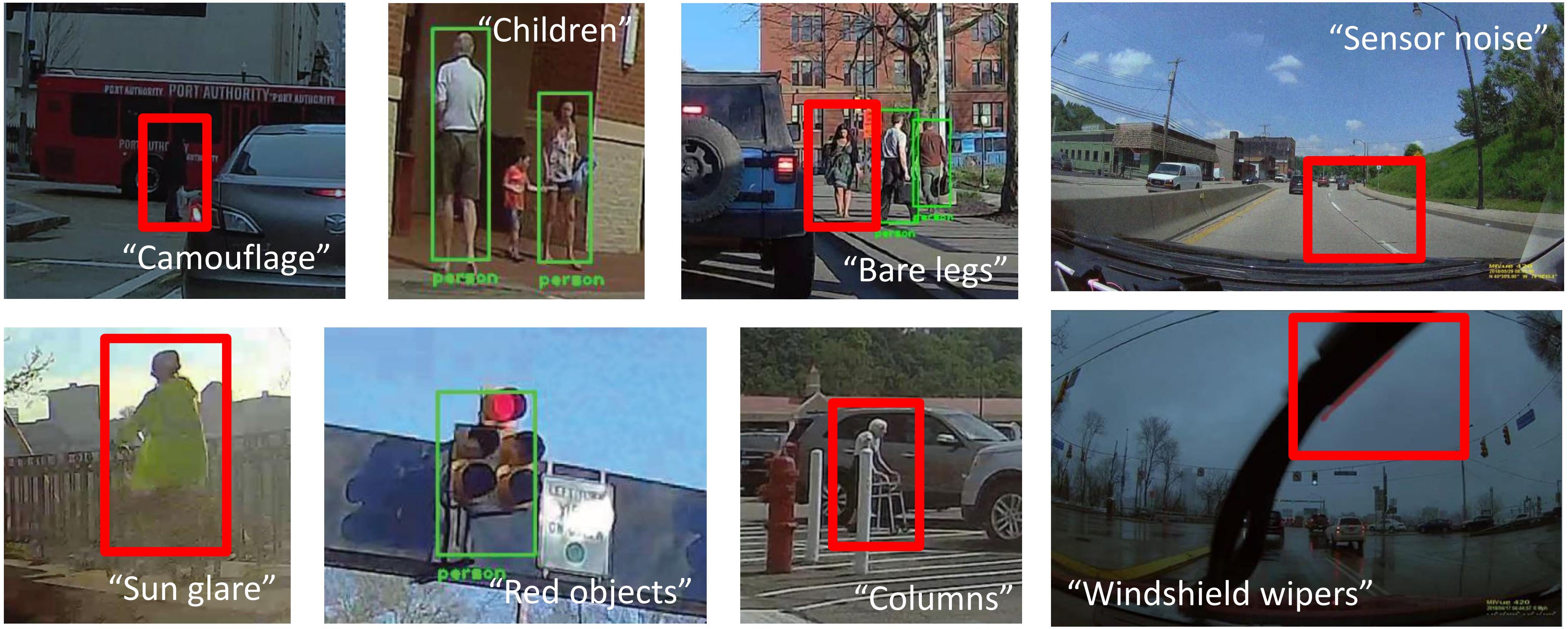
{ “**sun glare**”, “guardrail” }

{ “**sun glare**”, “fence”, “high-visibility vest” }

{ “**sun glare**”, “guardrail” }

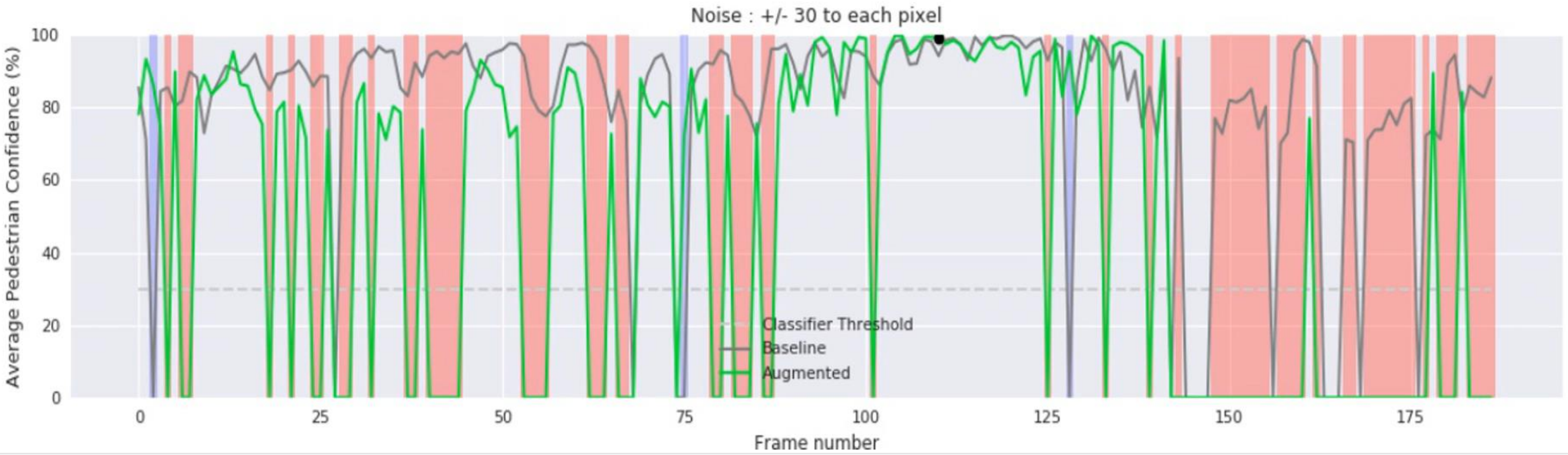
Root causes (“triggering events” per SOTIF) can be hypothesized, validated, mitigated, and verified.

Some Root causes can be surprising

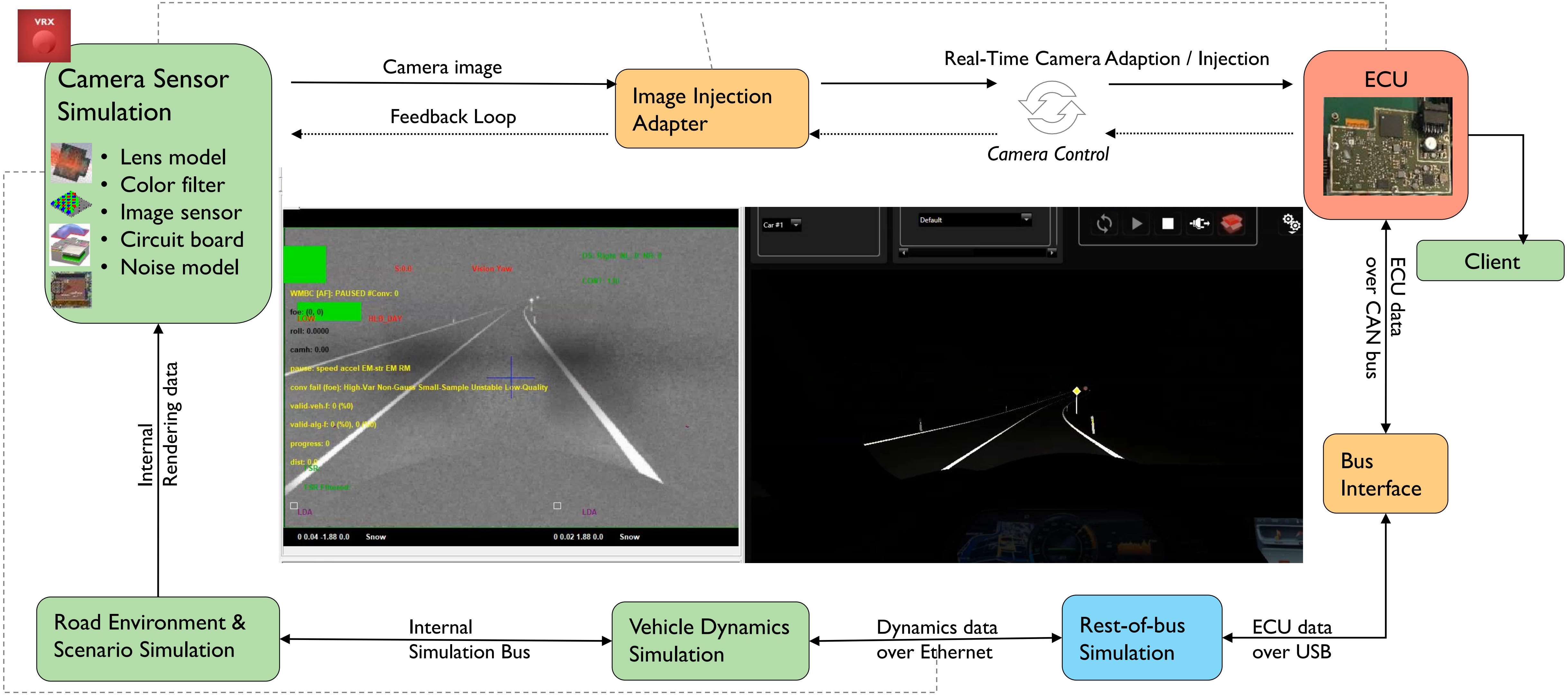


These results are from open-source neural networks. Your mileage may vary.

SCADE Vision (Powered by Edge Case research) filters through huge data sets to identify real-world edge cases and safety risks



Perception testing (HiL Simulation/Closed loop)

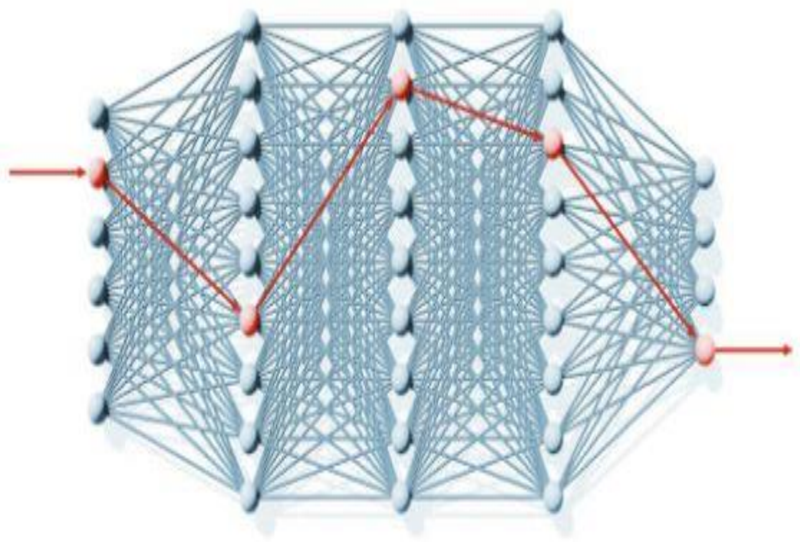


Safe Software Architecture must be designed for Integrating Neural Networks (Planning)

Can safety be guaranteed when using neural nets and machine learning?

No

Lack of controllability implies a high ASIL level and demands changes to safety concept



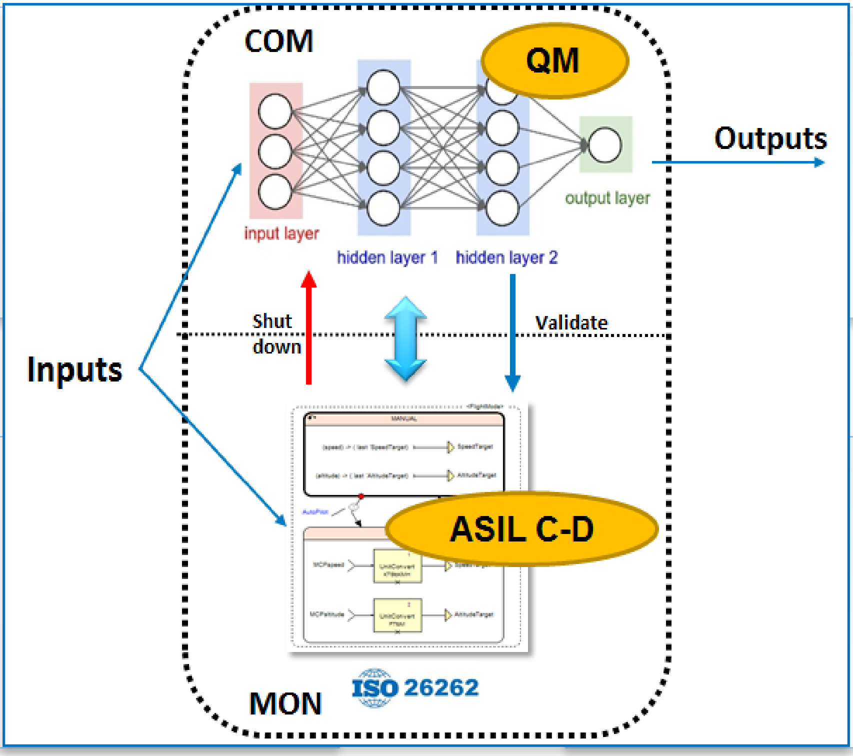
Because it is not possible to trace decisions backward through a neural net and connect them to higher level requirements

A safe software architecture is essential whenever neural nets are used

Safe Software Architecture for Integrating Neural Networks

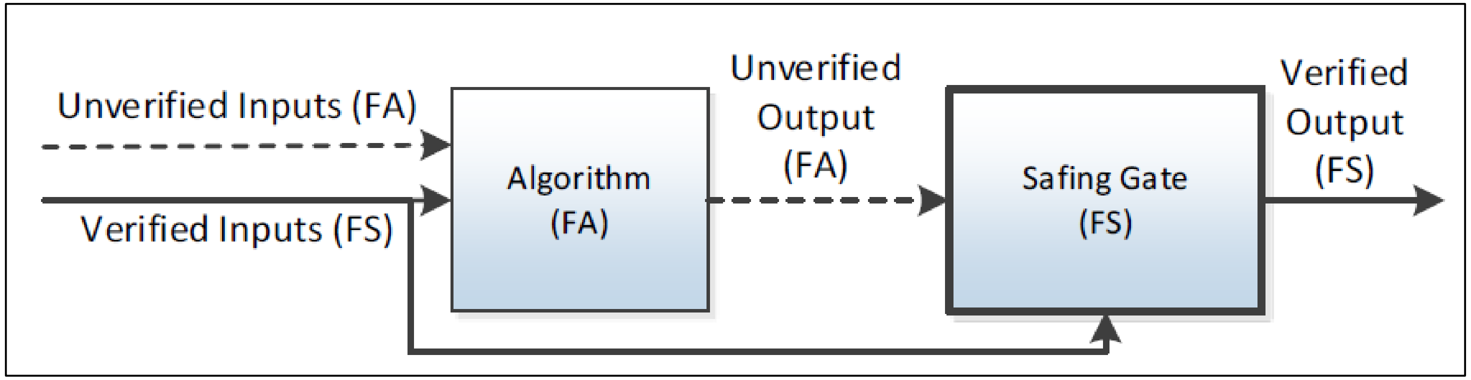
A COM-MON (Command and Monitor) architecture is used when using neural nets

The "DOER" Algorithm can fail arbitrarily (FA) meaning that it can do wrong things in the worst possible way



Safety is allocated to the monitor. The monitor is developed using MBSE, safety analyses, certified code generation

The Safing Gate (the "CHECKER") turns the Algorithm into a fail silent (FS) component, only producing correct data or shutting down



Source: Carnegie Mellon University

Bringing it all together

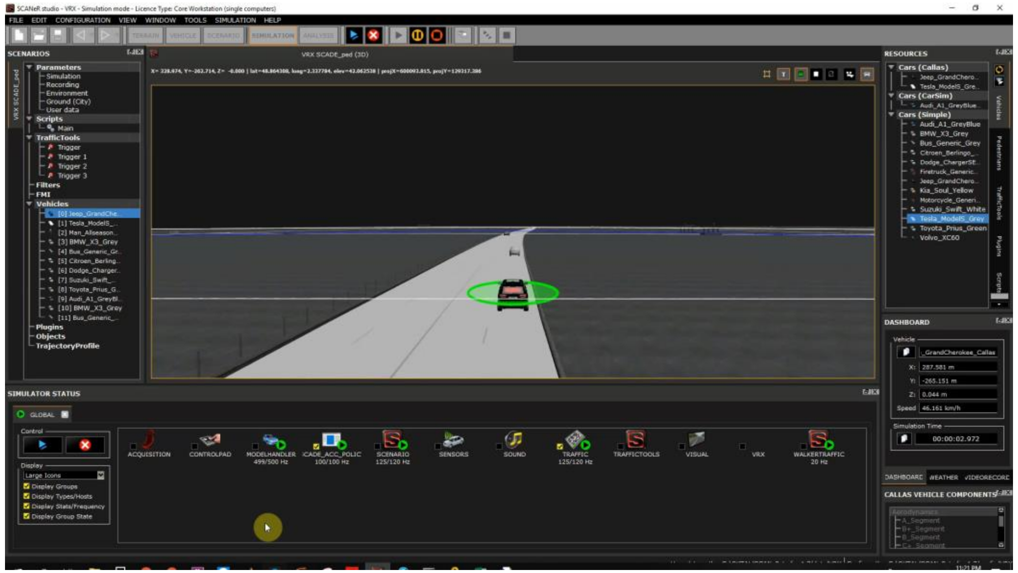
1. Functional Safety & Cybersecurity Analysis
2. Sensors
3. Closed-Loop Simulation
4. Control Software
5. Automated Driving Software
6. Vehicle Platform

Vehicle Dynamics

Key Features

- Complete accurate multi-body vehicle dynamics in VRXPERIENCE Driving Simulator

  Or – connect any custom vehicle model through FMI, C/C++, ANSYS Twin Builder
- Car, Truck and off-road vehicle models
- Consider road friction variation and wind

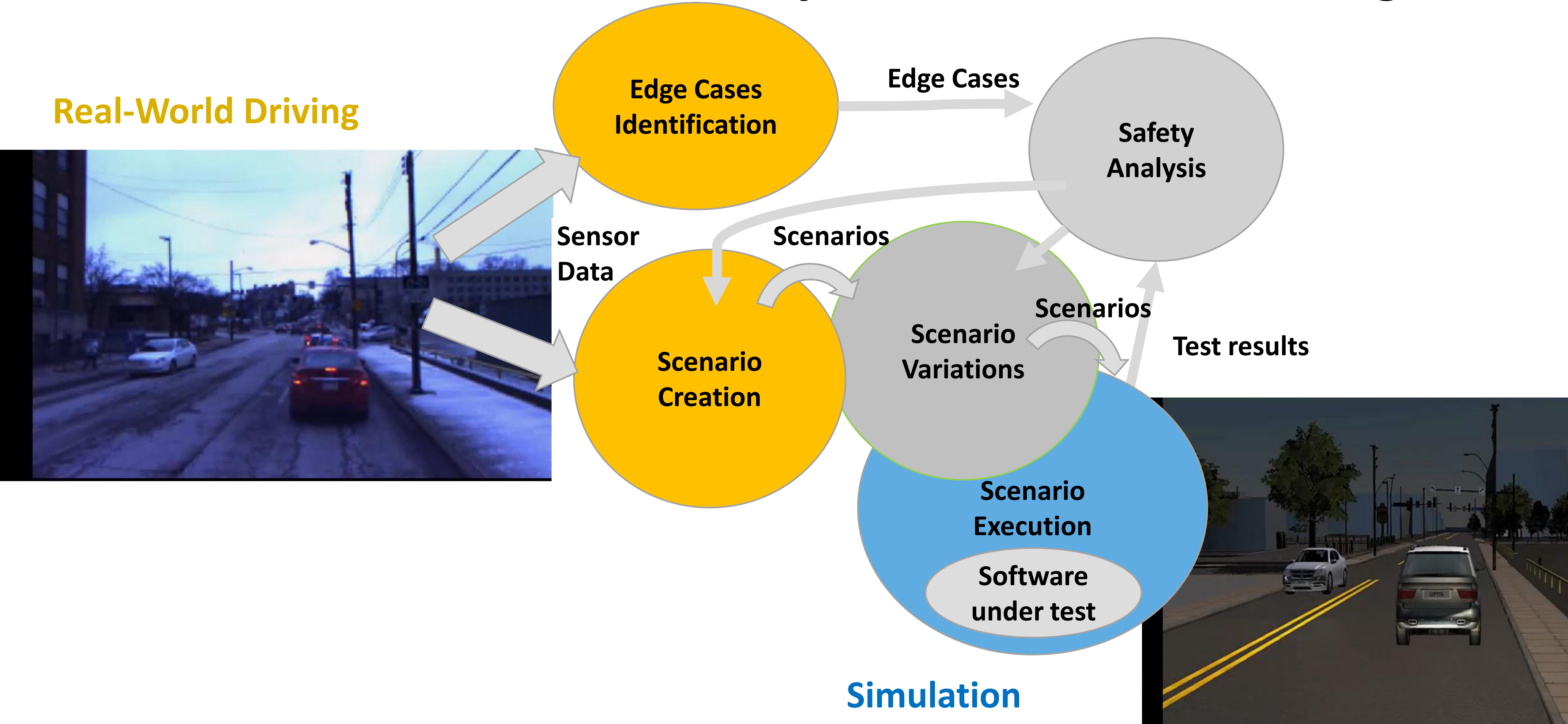


Use case

- Bring accurate vehicle dynamics model in the loop
- Accurate dynamic simulation with driver in the loop

Summary

Summary: Connecting Real-Real World Driving and Simulation to Achieve Safety of Autonomous Driving



Q&A