Defining and Structuring ODD and Usage Scenarios for SOTIF

Ideation Workshop ASAM OpenODD

2020-04-23

Bernhard Kaiser

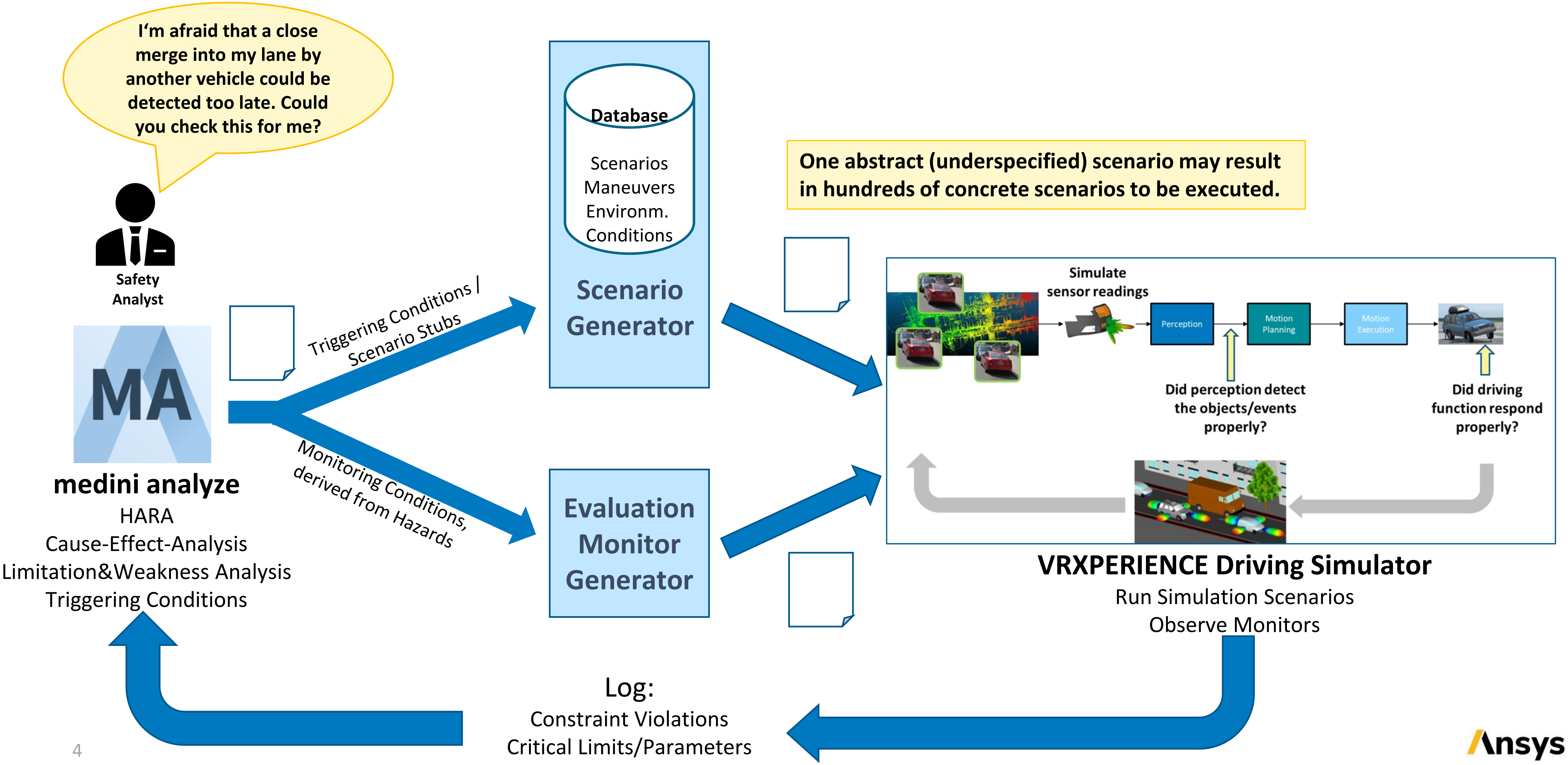
Agenda

- **Who we are and what use cases we have in mind**
- **Why a structured definition of ODD and usage scenarios is important**
- **How this was done in the past**
- **What are the issues in the AD Safety / SOTIF domain**
- **Ideas for research, discussion and standardization**

Who we are and what use cases we have in mind

- ANSYS - In the first place a simulation company
  - Basically anything that can be simulated: Mechanical Structures, Fluids&Aerodynamics, Electromagnetic Fields etc.
  - Increasing focus on automated driving: Traffic Situations, Vehicle Dynamics, Physically Accurate Camera, Lidar, Radar Simulation
- Acquired medini analyze in 2016
  - Basically a safety analysis, safety concept (requirements + architecture), safety mangement tool
  - Distinguishing feature: everything is based on a SysML model (consistency and traceabiltiy ensured)
  - All sorts of Functional Safety Analyses (Hazard Analysis, Fault Tree Analysis, FMEA, HAZOP etc., acc. ISO 26262 and other safety standards
  - Increasing focus on SOTIF, CyberSecurity and integral safety of automated vehicles
- Safety (SOTIF) and Automated Vehicles Simulation will have to work together closely in the future!
  - Designing an AV system and then trying to validate it by testing, testing, testing is not efficient → do systematic analysis, and do it early!
  - But behavior of AD (involving AI!) is so unpredictable that even systematic analysis does not prevent bad surprises later on the road → validation and simulation becomes part of the safety loop

Who we are and what use cases we have in mind



I'm afraid that a close merge into my lane by another vehicle could be detected too late. Could you check this for me?

Safety Analyst

medini analyze

HARA
Cause-Effect-Analysis
Limitation&Weakness Analysis
Triggering Conditions

Triggering Conditions / Scenario Stubs

Monitoring Conditions, derived from Hazards

Database

Scenarios
Maneuvers
Environm.
Conditions

Scenario Generator

Evaluation Monitor Generator

One abstract (underspecified) scenario may result in hundreds of concrete scenarios to be executed.

Simulate sensor readings

Did perception detect the objects/events properly?

Did driving function respond properly?

VRXPERIENCE Driving Simulator

Run Simulation Scenarios
Observe Monitors

Log:
Constraint Violations
Critical Limits/Parameters

Why a structured definition of ODD is important

- Aspect 1: Know what is in your universe (i.e. everything you have to master)
  - An automated vehicle must be designed and validated to safely operate **under any conditions within ODD**
  - The **remaining risk** of unforeseen conditions that are in the real ODD but not in the specficiation or test cases must be estimated („edge cases“)
  - The ablity to accurately describe **all aspects inside the ODD** (all road types, objects, environmental conditions, expectable maneuvers etc.) is key to an accurate and complete requirements specification, safety analysis and test case set for the AD function

- Aspect 2: Know where your universe ends (i.e. continuing outside is unsafe)
  - The capability of an automated vehicle to **determine where it is safe to operate and where not**
  - The challenge is that the automated vehicle must detect the ODD edges **onboard** (i.e. onboard sensors, HD maps and self-localization, V2X) and **in real time** to base safety-critical decisions on it
    - E.g. when to forbid activation of a function, when to call for driver-takeover or stop the vehicle, when to modify behavior, e.g. by disabling subfunctions or limiting the speed
  - The ability to accurately describe the conditions that determine **the limits of the ODD** is key to design safe ODD sensing and decision making algorithms, and corresponding test cases to validate these

How to describe / structure the ODD

- Aspect 1: Know what is in your universe (i.e. everything you have to master)
  - **Enumerative approaches** (list all road types, environmental conditions, speed ranges, surrounding vehicles including their present behavior etc, then form the combinatory of it all)
  - **Declarative approaches** (e.g. v < 60 km/h, number of lanes >= 2, pedestrians do not exist)

  Difference is rather theoretic, because enumerative approaches also need equivalence classes delimited by parameters, and declarative approaches also need categories that can be enumerated

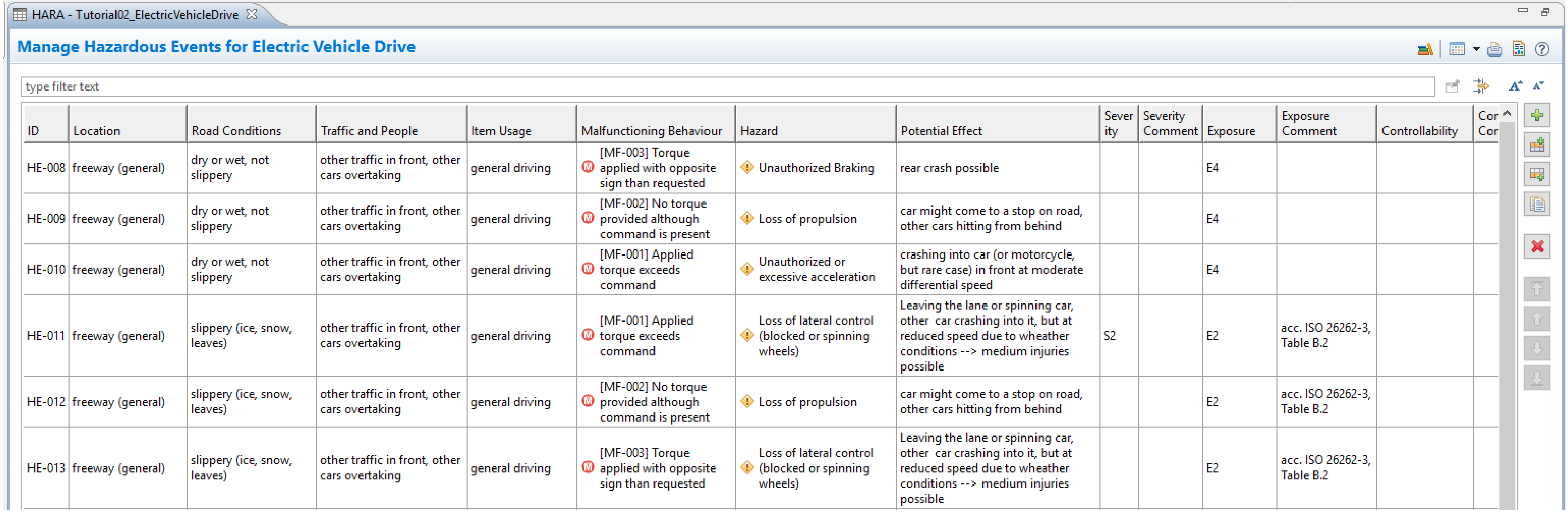
- Aspect 2: Know where your universe ends (i.e. continuing outside is unsafe)
  - Hard to enumerate the outside (you cannot list everything your system is *not* specified for)
    - So fallback to Aspect 1: Rather say what is inside the ODD, not what is outside
  - Possible to enumerate features (presence or absence, optional parameters), e.g.
    - At least 2 lanes per direction, lanes for opposing directions separated by construction
    - Lanes at least 3m wide
    - No traffic lights, crossroads, cyclists or pedestrians
  - Declarative approach desirable for on-board decision algorithm – but how to formalize it and how to know what aspects are relevant?

How was this done in the past (manually driven cars)?

- Aspect 1: Know what is in your universe
  - Not an explicit need! The human driver is supposed to be trained, intelligent and vigilant.
    - Human driver has the responsibility to know all the rules and cope with all possible situations.
    - …even the ones he has never encountered before!
  - When there was no defect on the car (→ Product Liability, FuSa), the human (ego car driver or other traffic participant) is usually blamed for any accidents
  - **Still some need to structure ODD during vehicle development**, e.g.
    - Requirements and Design Parameters for Developmente (e.g. for dimensioning of chassis, brakes, powertrain)
    - Situations for Hazard Analysis (ISO 26262)
    - Situations for which ABS, ESC, AEB and the like are designed and tested
    - Test case catalog for closed-track and road testing of the vehicle

- Aspect 2: Know where your universe ends
  - Basically not relevant at all! A car can go (almost) **anywhere** on solid ground
  - And if not, the responsibility to know the limits and cope with them was with the **human driver**, e.g.
    - Don‘t drive into the water
    - Don‘t get stuck in the mud
    - Reduce your speed when there is snow on the road or foggy weather

How was this done in the past – Example: Hazard Analysis



HARA - Tutorial02_ElectricVehicleDrive

Manage Hazardous Events for Electric Vehicle Drive

type filter text

ID	Location	Road Conditions	Traffic and People	Item Usage	Malfunctioning Behaviour	Hazard	Potential Effect	Severity	Severity Comment	Exposure	Exposure Comment	Controllability	Cor Cor
HE-008	freeway (general)	dry or wet, not slippery	other traffic in front, other cars overtaking	general driving	Ⓜ [MF-003] Torque applied with opposite sign than requested	◈ Unauthorized Braking	rear crash possible			E4			
HE-009	freeway (general)	dry or wet, not slippery	other traffic in front, other cars overtaking	general driving	Ⓜ [MF-002] No torque provided although command is present	◈ Loss of propulsion	car might come to a stop on road, other cars hitting from behind			E4			
HE-010	freeway (general)	dry or wet, not slippery	other traffic in front, other cars overtaking	general driving	Ⓜ [MF-001] Applied torque exceeds command	◈ Unauthorized or excessive acceleration	crashing into car (or motorcycle, but rare case) in front at moderate differential speed			E4			
HE-011	freeway (general)	slippery (ice, snow, leaves)	other traffic in front, other cars overtaking	general driving	Ⓜ [MF-001] Applied torque exceeds command	◈ Loss of lateral control (blocked or spinning wheels)	Leaving the lane or spinning car, other car crashing into it, but at reduced speed due to wheather conditions --> medium injuries possible	S2		E2	acc. ISO 26262-3, Table B.2		
HE-012	freeway (general)	slippery (ice, snow, leaves)	other traffic in front, other cars overtaking	general driving	Ⓜ [MF-002] No torque provided although command is present	◈ Loss of propulsion	car might come to a stop on road, other cars hitting from behind			E2	acc. ISO 26262-3, Table B.2		
HE-013	freeway (general)	slippery (ice, snow, leaves)	other traffic in front, other cars overtaking	general driving	Ⓜ [MF-003] Torque applied with opposite sign than requested	◈ Loss of lateral control (blocked or spinning wheels)	Leaving the lane or spinning car, other car crashing into it, but at reduced speed due to wheather conditions --> medium injuries possible			E2	acc. ISO 26262-3, Table B.2		

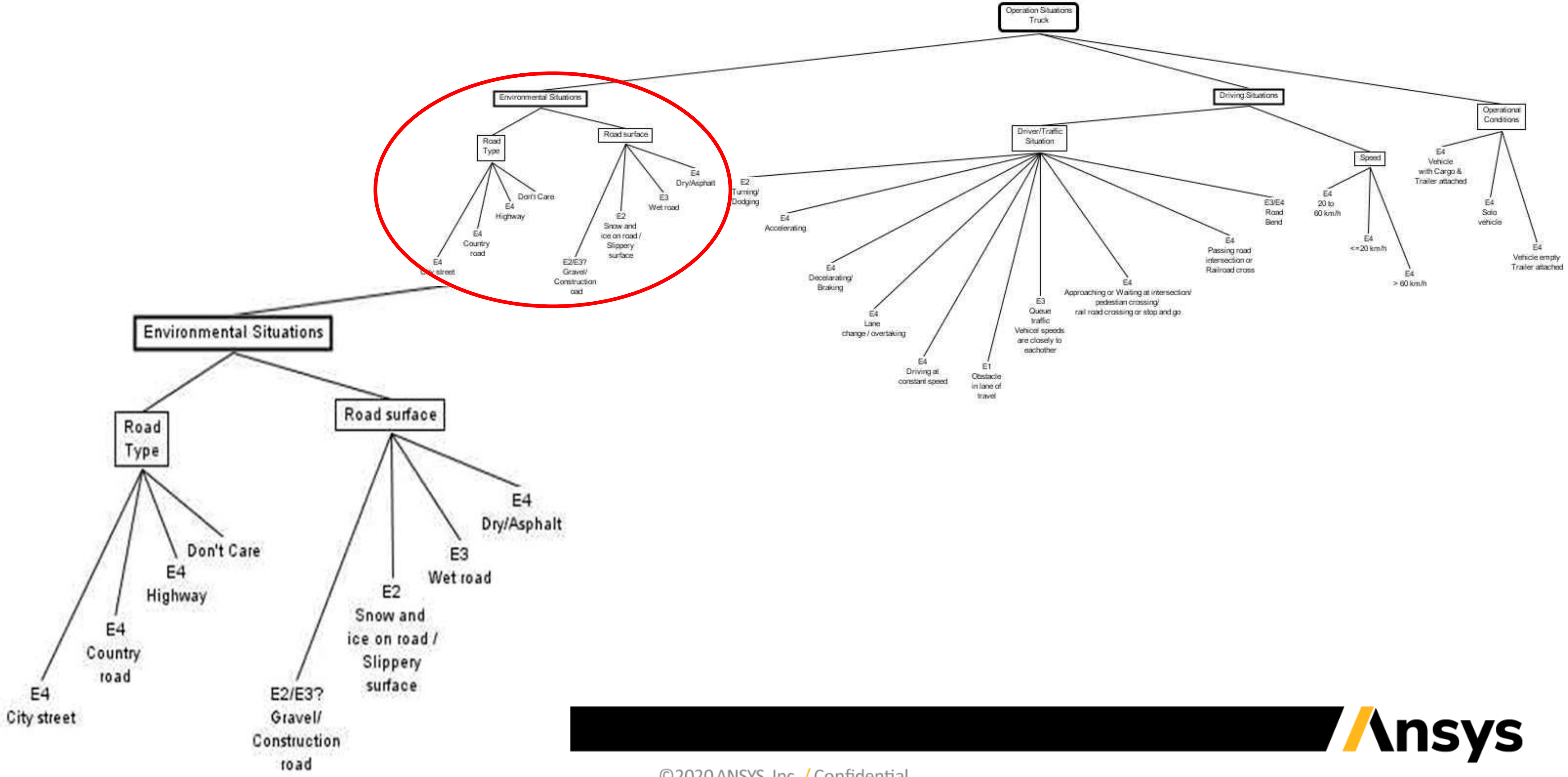
Some companies tried to be very systematic (ending up at millions of individual situations), some tried to be pragmatic, using a (more or less) justified selection of 50…300 individual situations.

Lessons Learned from HARA Situation Catalog Building

1. Define classifciation (e.g. road type, driving situation, load condition, vehicle configuration etc)
2. Define equivalence classes (e.g. road type := {highway, country road, city street, offroad}
3. Specify parameters (e.g. medium speed := 60 km/h < v < 100 km/h)
4. Crosscheck with situations from ISO26262, test case catalogs, past experience, educated guessing etc.
5. Sort out special cases (not to blow up combinatory) ,e.g. parking, backing up, car wash, being towed etc.
6. Build combinatory (road types x driving situations x load conditions ...) – can still be > 10,000 situations
7. Reduce number of cases and justify your decision
   - Irrelevant classifications (no influence on considered hazards)
   - Equivalence classes (variants that have same influcence on hazards)
   - Subsumptions (worst case of all variants)
   - Unrealistic cases (e.g. city traffic & speed > 100km/h)
   - Focus on combinatinons with special relevance (e.g. side wind & on bridge), treat others on coarser level
   - Function specific elimination (e.g. brake function→ day or night irrelevant vs. light function → dayor night is relevant)
8. Assign exposure values (E parameter) according to ISO26262

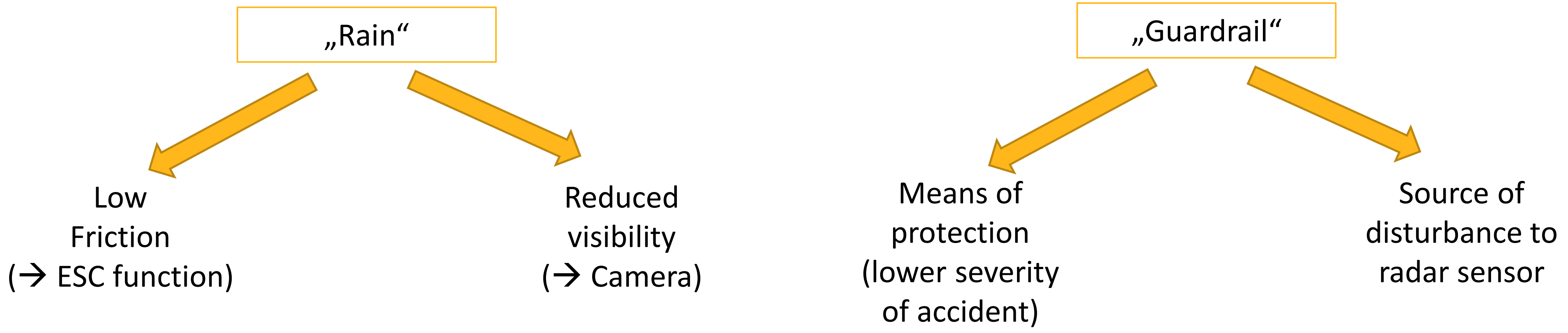
Being systematic and complete while constraining the analysis to a manageable number of situations is a big challenge!

Lessons Learned from HARA Situation Catalog Building


Operation Situations Truck
Environmental Situations
Road Type
E4 City street
E4 Country road
E4 Highway
Don't Care
Road surface
E2/E3? Gravel/ Construction oad
E2 Snow and ice on road / Slippery surface
E3 Wet road
E4 Dry/Asphalt
Driving Situations
Driver/Traffic Situation
E2 Turning/ Dodging
E4 Accelerating
E4 Decelarating/ Braking
E4 Lane change / overtaking
E4 Driving at constant speed
E1 Obstacle in lane of travel
E3 Queue traffic Vehicel speeds are closely to eachother
E4 Approaching or Waiting at intersection/ pedestian crossing/ rail road crossing or stop and go
E4 Passing road intersection or Railroad cross
E3/E4 Road Bend
Speed
E4 20 to 60 km/h
E4 <=20 km/h
E4 > 60 km/h
Operational Conditions
E4 Vehicle with Cargo & Trailer attached
E4 Solo vehicle
E4 Vehicle empty Trailer attached
Environmental Situations
Road Type
Road surface
E4 City street
E4 Country road
E4 Highway
Don't Care
E2/E3? Gravel/ Construction road
E2 Snow and ice on road / Slippery surface
E3 Wet road
E4 Dry/Asphalt

Lessons Learned from HARA Situation Catalog Building

- Not only ask what is there – but also why (and for which aspect) it is of relevance!



„Rain“

Low
Friction
(→ ESC function)

Reduced
visibility
(→ Camera)

„Guardrail“

Means of
protection
(lower severity
of accident)

Source of
disturbance to
radar sensor

Unforeseen Object („black swan“) appearing in the ODD

- Important to know for city speed (i.e 50 km/h) robot taxi
  - At higher speeds we must anticipate, so we must predict objects‘ trajectories, so we must classify objects correctly and have an appropriate model for their movement
- Irrelelvant for low speed (i.e. max 15 km/h) valet parking function
  - Don‘t care what it is – just brake for any object (enough reaction time even without prediction, no fear of provoking a rear crash)

What are the challenges in the AD Safety / SOTIF domain

- The well known ones from the past…
  - Find a notation that is formal enough (need for computerized onboard decision, need for repeatable simulation!), yet close to reality and domain engineers‘ language
  - How to be systematic and complete, but cope with the the combinatorial explosion problem
    - Focussing on the aspects that are really relevant (but without forgetting something important)
    - Forming equivalence classes on the right granularity level
  - How to handle the exots that cannot be described in a declarative manner (equivalence classes and parameter ranges), but have to be enumerated individually

- … plus some new ones
  - How to capture all relevant aspects that may be clear to humans, but incredibly hard to explain to a computer
  - How to cope with the unknown unknowns (you cannot enumerate or parametrize what you don‘t even know!)
  - How to adapt from the objective outside world to a subjective perception of the world seen through the lens of a sensor system or ML algorithm

Challenges – Some Examples

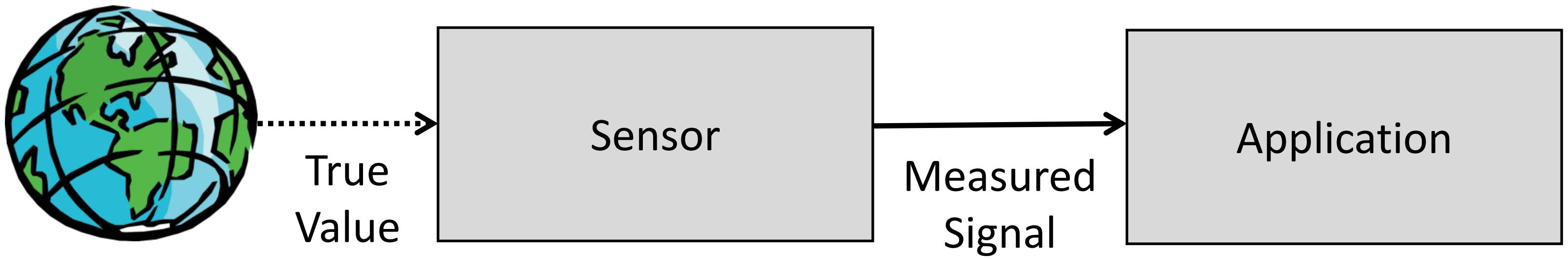
- Highway Pilot
  - ODD: „Only on highways, **but not** construction sites, toll boths and some more“
  - How to be sure about highway end (the road layout can still look the same, but have traffic lights and cyclists)
    - HD-Map? Traffic Sign Recognition? See the world through the sensors‘ eyes! We can only design and validate what is specified!
  - How to know (early enough) that a construction site or tool booth is ahead?
  - What other (unspecified) exceptions may occur in the ODD (animal, breakdown-car, lost cargo,…)?

- Automated City Driving
  - A lot of fuss is being made about “black swans”
  - Can we / do we have to enumerate all of these in the ODD definition?
  - Are they relevant at all for the vehicle’s decisions? (or just brake for anything?)
  - How does this relate to the AD function specification (e.g. low speed only vs. all speed ranges?)
  - How to express that the AD function shall cope with just anything that can occur in a city street?
  - Would it be acceptable to simply exclude certain things from ODD?
    - “Driver, will you take over? I don’t know what this is!”

What are the challenges in the AD Safety / SOTIF domain

- See the world through the sensor‘s eyes!

  - If you can describe what characterizes the ODD limits in terms of objects and parameters that exist in the real world, you‘ve achieved a lot – but still not enough to specify an **onboard** algorithm to make a decision
    - E.g. hand back to the driver, reduce the speed or stop the car when leaving the ODD
  - The onboard computer is not an omniscient observer! It only sees an image of the world through the sensors
    - ODD violation decision algorithm must be specified in terms of sensor outputs, not real world objects and variables → don‘t specify in terms of facts that the onboard computer cannot know!
    - **This may require 2 versions of the ODD definition: Objective and subjective (depending on car‘s perception capabilities)**
    - Consider inaccuracies, range of view, performance limitations, disturbances, fitness for purpose of sensor (SOTIF)
    - Consider failures of sensor (FuSa)



e.g. true distance to another car

e.g. CAN message with distance value from sensor

Ideas for research, discussion and standardization

- Work towards a standard catalog of categories (e.g. road type, weather condition) and classes for these categories (e.g. sunshine, rain, snow, fog for weather), including parameters (e.g. number of lanes)
- Suggest understandable tabular or CNL (constrained natural language) notations to express ODDs and ODD limits
- Suggest structuring approaches (e.g. classification trees) and methods to adjust the level of granularity and to filter out irrelevant details to avoid the combinatorial explosion problem of enumerative approaches from past
- Work towards a formal ontology / meta model and a formal language to express presence/absence of features an objects, parameters of objects, relations between objects and to a certain extent behavior of actors
- Work towards solutions for formal completeness and consistency checking of ODD specifications
- Work towards a formalization of the transformation between objective representation and subjective representation (as seen through the sensor set of the car) of the ODD, to support the creation of verifyable onboard decision algorithms
- Work towards formal integration with OpenScenario / OpenDrive, e.g .enabling a formal check whether a modeled scenario is part of the ODD or not

Foundations and Terminology

Foundations and Terminology

- J3016 defines an Operational Design Domain (ODD) as

> "operating conditions under which a given driving automation system or feature thereof is specifically designed to function, including, but not limited to, environmental, geographical, and time-of-day restrictions, and/or the requisite presence or absence of certain traffic or roadway characteristics."

- An ODD may put limitations on

1. the road environment
2. the behavior of the ADS-equipped subject vehicle; and
3. state of the vehicle.

- The ODD may reflect the requirements of a particular driving automation feature

Source: Krzysztof Czarnecki: Operational Design Domain for Automated Driving Systems - Taxonomy of Basic Terms. Technical Report. 2018. Available from: https://www.researchgate.net/publication/326543176_Operational_Design_Domain_for_Automated_Driving_Systems_-_Taxonomy_of_Basic_Terms [accessed Apr 22 2020].

Foundations and Terminology

- Closely related to ODD is the Operational Road Environment Model (OREM)

> “An operational road environment model (OREM) is a representation of the relevant assumptions about the road environment in which an ADS will operate the ADSequipped vehicle.”

- Environment models capture the properties of the environment that are relevant to the ADS operation, while abstracting irrelevant details.
    - OREMs can represent generic environments, such as two-lane rural road, or actual roads in a specific geographic area.
    - OREMs can take different forms, including specification documents and executable models.
- OREMs provide context for specifying driving tasks of an ADS and for verifying and validating the ADS.
- An ODD of an ADS implies a set of operational environments in which the ADS can operate the ADS-equipped vehicle. These environments can be specified using a set of OREMs.

Source: Krzysztof Czarnecki: Operational Design Domain for Automated Driving Systems - Taxonomy of Basic Terms. Technical Report. 2018. Available from: https://www.researchgate.net/publication/326543176_Operational_Design_Domain_for_Automated_Driving_Systems_-_Taxonomy_of_Basic_Terms

Foundations and Terminology

- Operational World Model / Operational World Model Ontology

> “Operational world model ontology is a conceptualization of the elements that occur in an operational world model. The conceptualization includes element types, element attributes and relationships and, if applicable, behaviors..”

- An OWM ontology contains concepts in five categories:

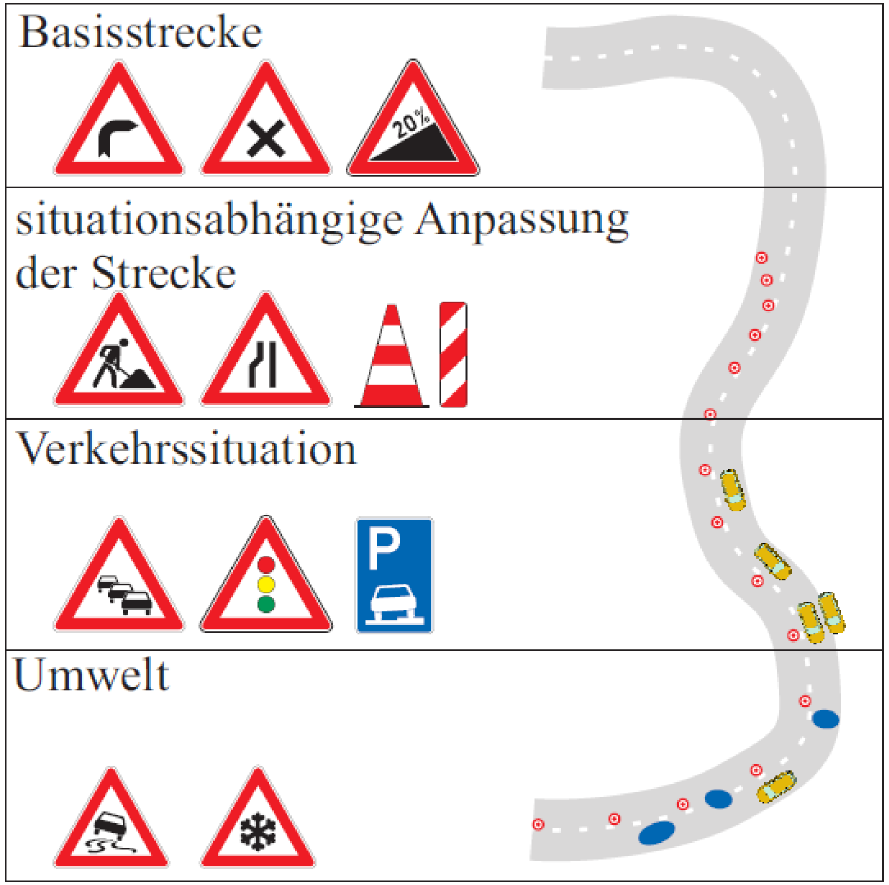
1. **Road structure:** This category includes road geometry, lane configuration, roadside structure, traffic control devices, junctions, and temporary structure.
2. **Road users:** This category includes vehicles and pedestrians and their behavior.
3. **Animals:** This category covers wild and domestic animals of different sizes.
4. **Other obstacles:** This category covers all other obstacles that might be found on a roadway, such as lost cargo, tree branches, or debris.
5. **Environmental conditions:** This category consists of atmospheric, lighting, and road surface conditions.

Source: Krzysztof Czarnecki: Operational Design Domain for Automated Driving Systems - Taxonomy of Basic Terms. Technical Report. 2018. Available from: https://www.researchgate.net/publication/326543176_Operational_Design_Domain_for_Automated_Driving_Systems_-_Taxonomy_of_Basic_Terms [accessed Apr 22 2020].

Foundations and Terminology

Schuldt et al: 4-Layer Model for the construction of AD Scenarios (not actually meant for ODD spec)

1. **Static Road Layout** including tunnels, bridges, road markings, signs, traffic lights…
2. **Temporary modifications** e.g. road construction
3. **Traffic Situation** other traffic participants like vehicles, pedestrians, etc.
4. **Environmental conditions** like weather, light and sight



Source: Schuldt, Saust, Lichte, Maurer: Effiziente systematische Testgenerierung für Fahrerassistenzsysteme in virtuellen Umgebungen, 2014