A Different Perspective - A Bottom Up View

ASAM OpenSCENARIO 2.0 concept discussion group
jupp.tscheak(at)daimler.com, RD/ASE, Sindelfingen, 21.02.2019

Mercedes-Benz
The best or nothing.

Content

# Textual vs. Visual Programming

1D (Column position)

```
double ttc = v1 < v2 ? ds / (v2 - v1) : inf;
```

- A **line of code** can be considered as Turing tape and thus is **1D**

- Visual programming languages are designed to represent information with **geometric forms**. They can be considered as being **2D**.

- The **information entropy** tends to be higher for textual languages than for visual ones.

- Some content is better suited for a **visual** description (**UML**), others can be described better using a **textual** presentation (**Mathematical proof**).

2D (x, y)

AST created from C
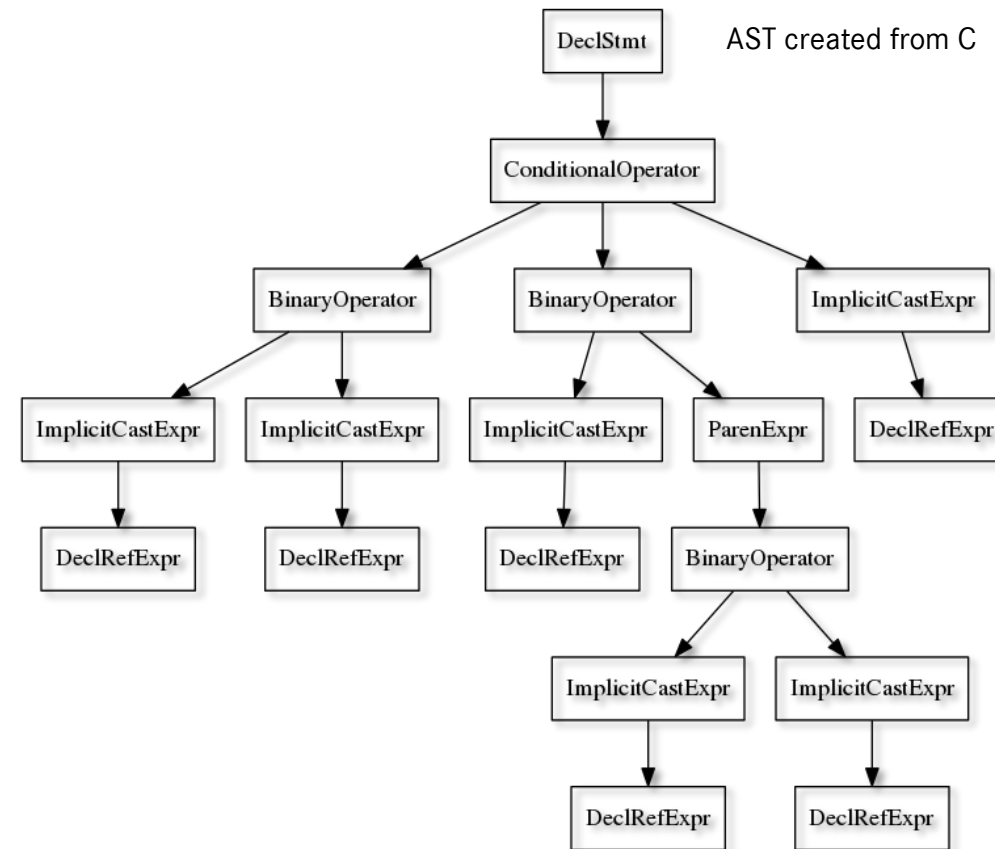
A Visual Approach
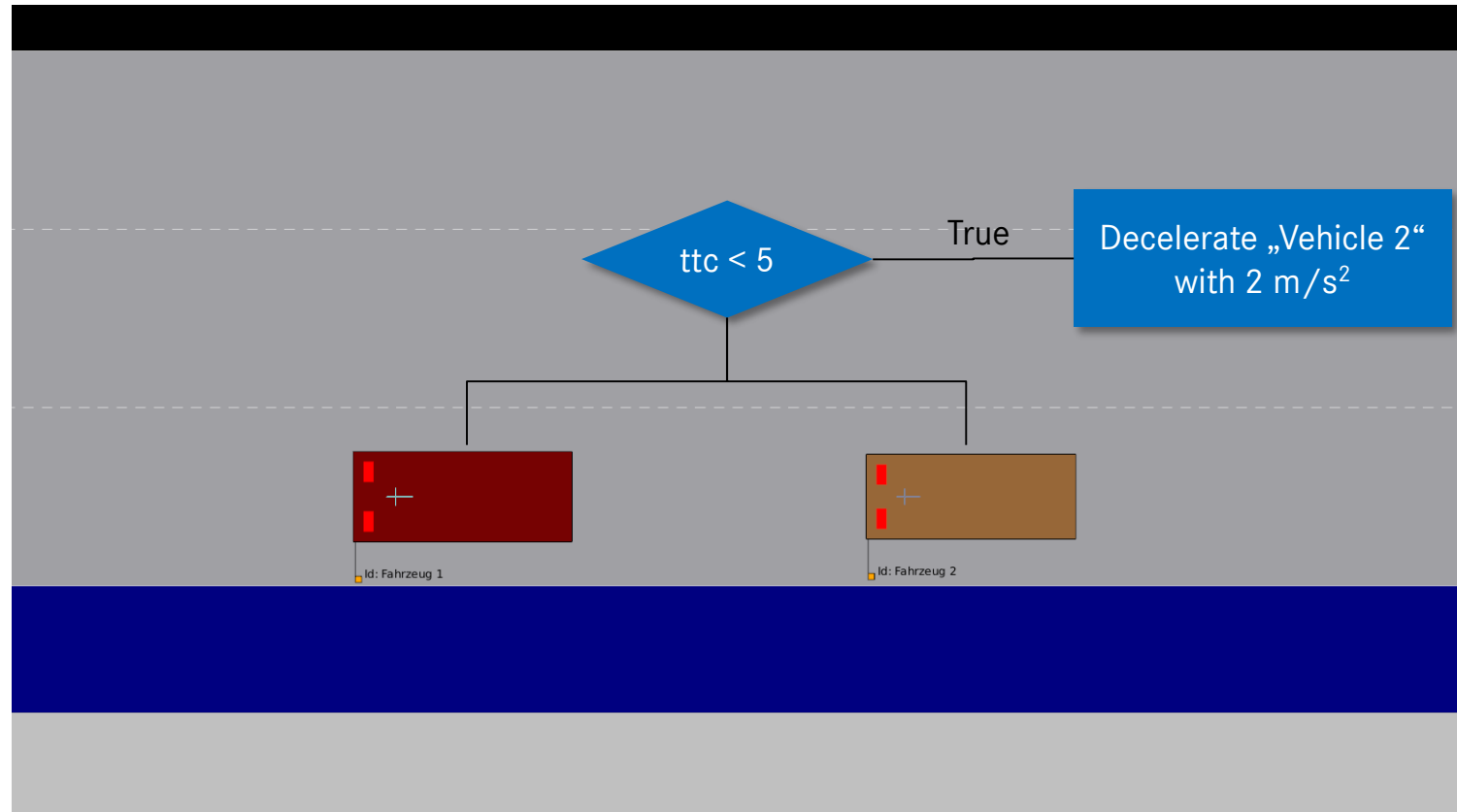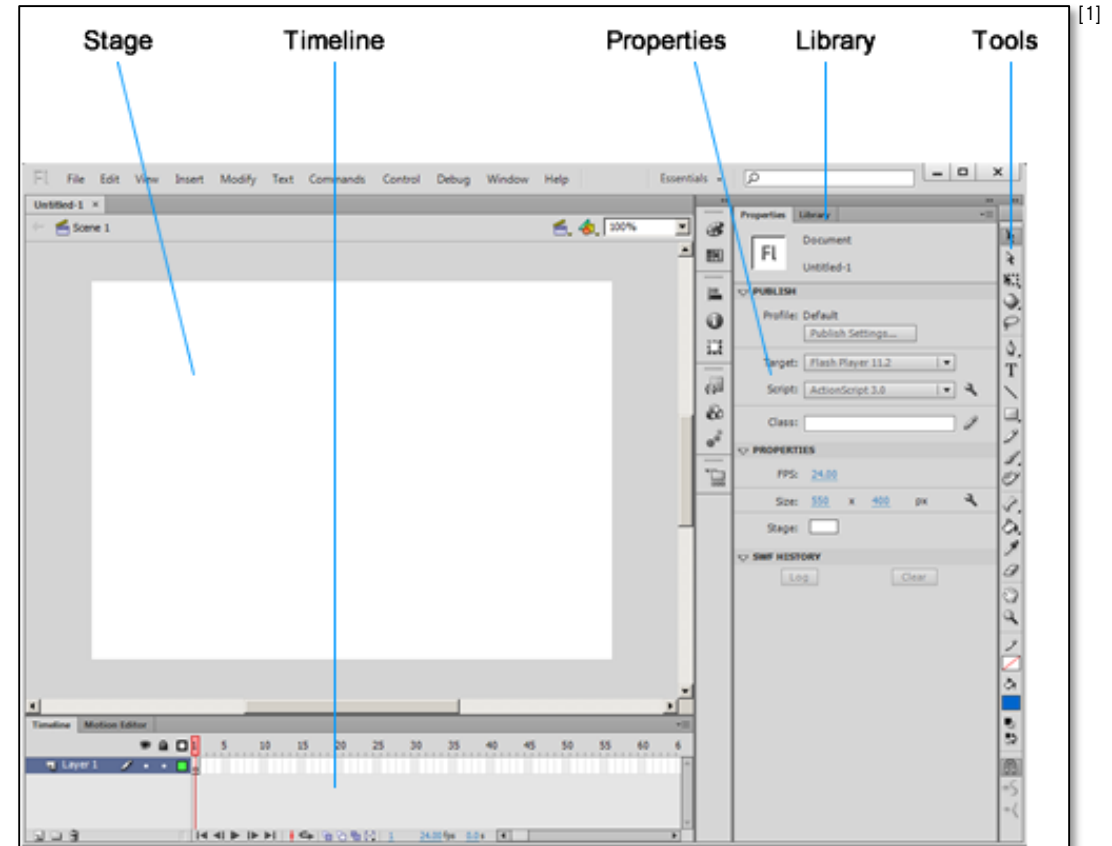
The Timeline

Timeline

Let's get inspired: Adobe Animate CC

Adobe Animate CC is a powerful authoring tool that abstracts the creation of games and films.

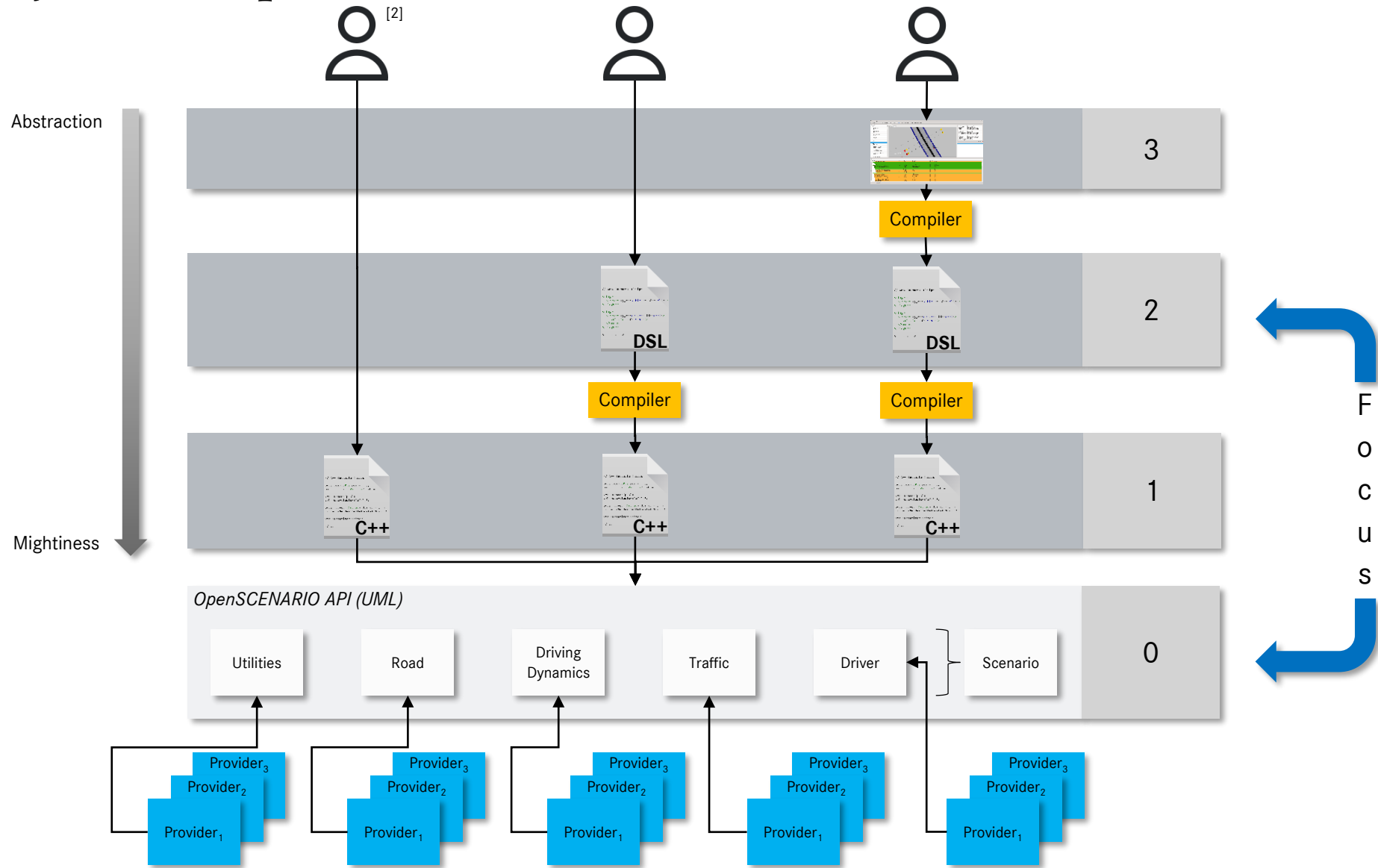Aspects that relates to OpenSCENARIO:

- Timeline based,

- Event system,

- ActionScript 3.0, ECMAScript (ECMA-262),

- Huge API/Library.

Any application created with Adobe Animate CC could have been written **by just using ActionScript**.

[1]

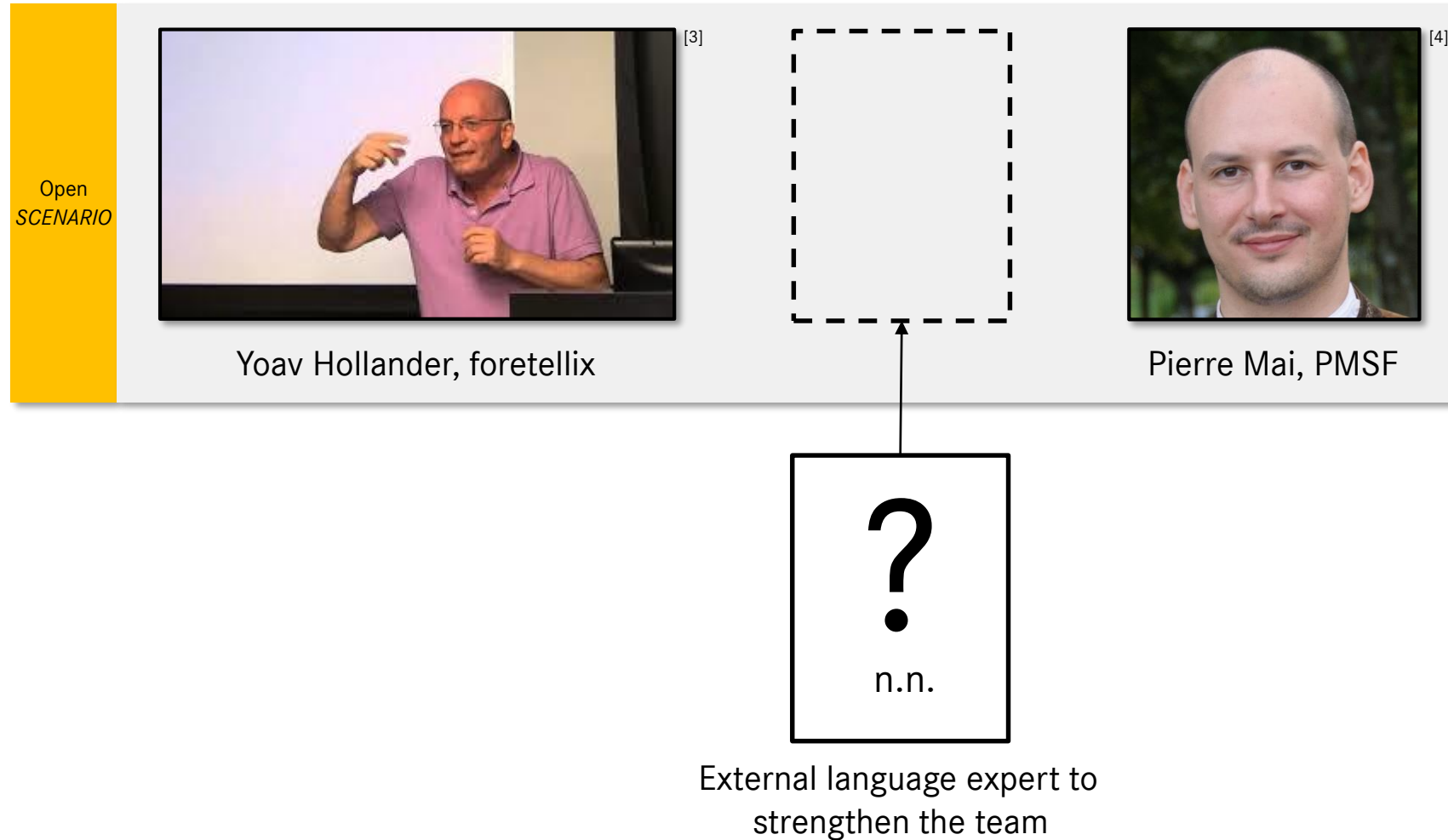Stage   Timeline   Properties   Library   Tools

Different Layers Of OpenSCENARIO

[2] Breeze User Icon https://github.com/KDE/breeze-icons developed by the KDE Community, LGPL Version 3 https://www.gnu.org/licenses/gpl-3.0.html

Mercedes-Benz

DSL Experts

Open
SCENARIO

[3]

[4]

Yoav Hollander, foretellix

Pierre Mai, PMSF

?

n.n.

External language expert to
strengthen the team

Conclusion Part I

We would like to ask the community

*to make the design of a DSL and the underlying type system the goal of the concept project, with focus on **Readability** and **Mightiness**.*

Content

1	Textual versus visual programming

| 2	Use case presentation

Use Case 1 – „Construction Site"

As a scenario designer, I would like to define a construction site that occupies one or more lanes. The construction site should be enclosed by definable traffic signs (e.g. beacons). The outer border should be described by a piecewise defined function. In order not to load the image system, a visibility window should be definable.

Parameters:

- Definition of source T_1 and target T_2 offset with $\Delta T = |T_2 - T_1|$,

- Length Δs_1 and function interpolating source to target offset (**phase 1**)

- Length Δs_2 of constant offset (**phase 2**),

- Length Δs_3 and function interpolating target to source offset (**phase 3**),

- Piecewise defined function describing phase 1, 2 and 3:

$$T: s \to= \begin{cases} T_1 + \Delta T * norm(s), & s \leq \Delta s_1 \\ T_1, & s > \Delta s_1 \\ T_1 + \Delta T * (1 - norm(s)), & s \geq \Delta s_1 + \Delta s_2 \end{cases}$$

- Visible range [rear,front].
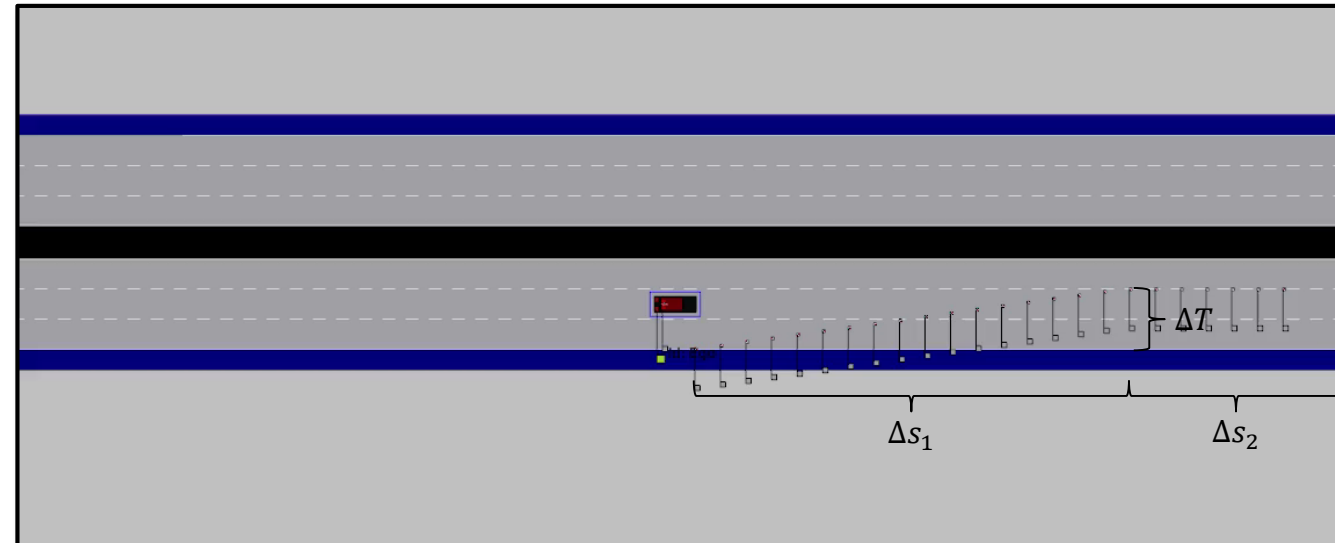
Use Case 2 – „AV Junction Behaviour Test"

As a function developer, I want to test the behavior of an AV in intersection areas. It should not be necessary to create a trajectory for every intersection and every vehicle. Instead, the number of incoming roads should be determined automatically and equipped with vehicles. The turning directions of the ambient vehicles are to be determined stochastically.

Parameters:

- Junction id,

- Attachment distance Δs per incoming road,

- Number of vehicles and turn directions with probability $P = \left(p_{left}, p_{ahead}, p_{right}\right)$ per incoming road.

Mercedes-Benz
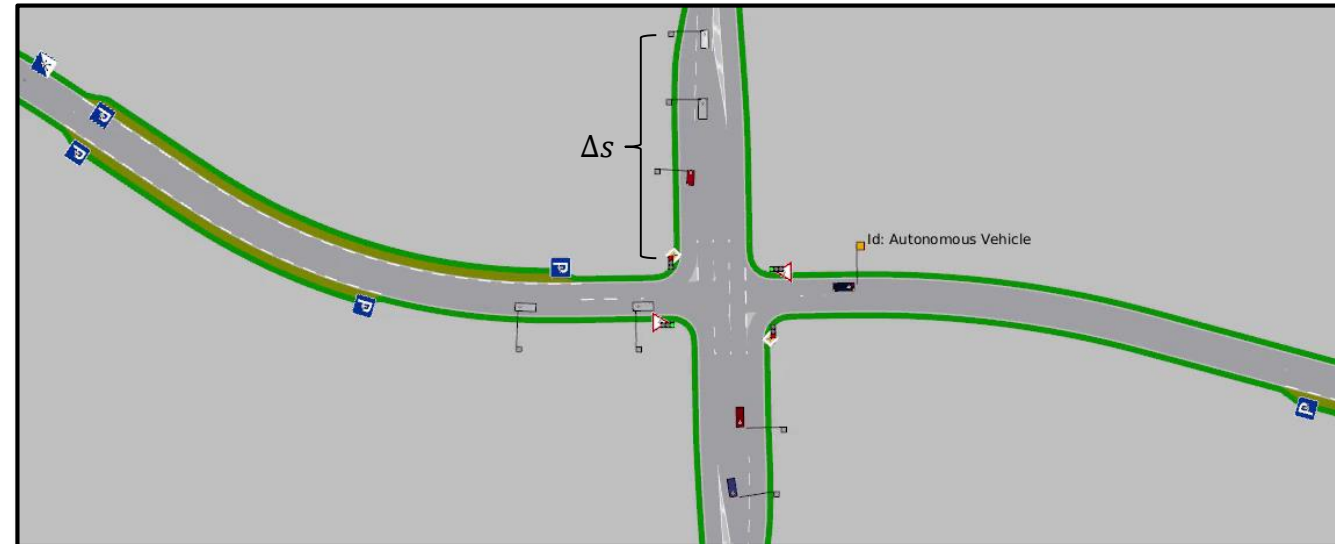
Use Case 3 – „Ambient Highway Traffic"

As a scenario designer, I want to have the possibility to create ambient traffic on highways around the vehicle under test. In order to use rather complex driver models the time spent to calculate the traffic is limited. Therefore the ambient traffic should only be calculated in a certain range around the vehicle under test. It should be possible to validate the generated traffic against real world data as provided by traffic census, showing a realistic fundamental diagram of traffic flow.

Parameters:

- Traffic density,

- Visibility range Δs_{rear} and Δs_{front},

- Probability $P = \left(p_{calm}, p_{active}, p_{sporty}, p_{unsecure}, p_{...}\right)$ of the occurrence of certain driver types

- Speed distribution
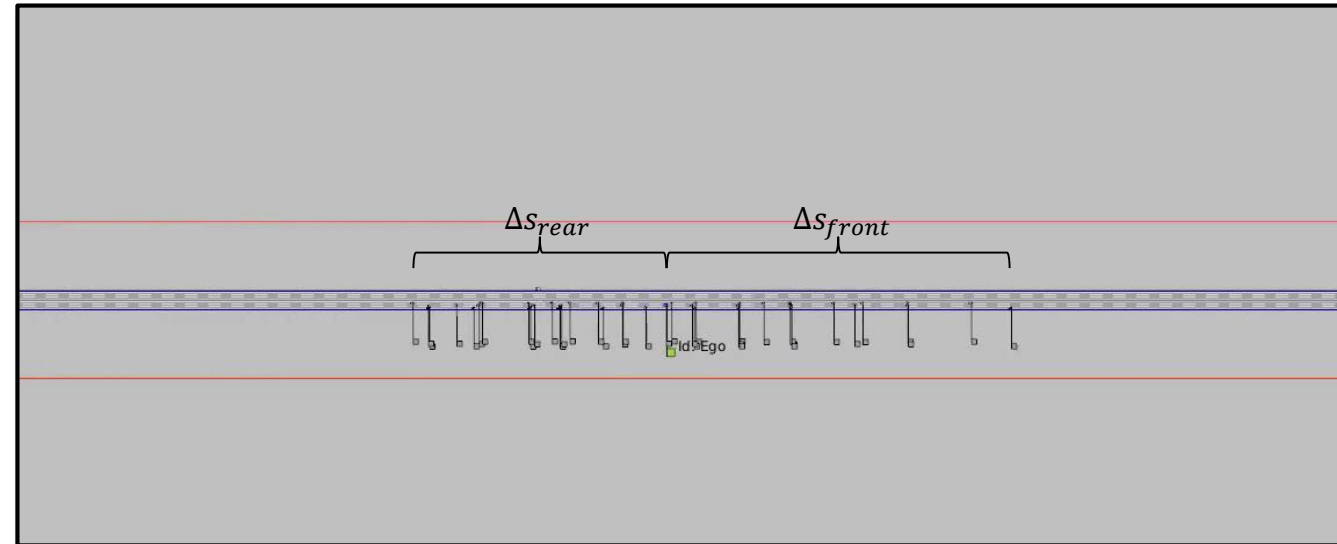
Conclusion Part II

We would like to ask the community

to **provide use cases** that take into account not only current but also **future requirements** of the different roles. We could than use them to

- *Describe those scenarios using well-established languages like OpenSCENARIO 0.9.1, BMW's Motion NextGen, foretellix's DSL, Daimler Driving Simulator's VDScenario etc.,*

- *Learn what is good and what is bad about the existing languages,*

- *Determine the scope of the language and the resulting entities like operators, functions, objects, containers, etc.*

- *Be agil* ☺

The End

Thank you 😊