BERTHA

Deliverable 4.1: Drivable scenes for simulation

López, Antonio M. (CVC-CERCA)
Levy, Alexandre F. (CVC-CERCA)
Abad, Rubén (CVC-CERCA)

Version 1.1
May 2nd , 2025

Funded by the European Union

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Climate, Infrastructure and Environment Executive Agency (CINEA). Neither the European Union nor the granting authority can be held responsible for them.

TABLE OF CONTENTS

IMPORTANT

This document serves as a template for deliverables and follows a proposal structure. The mandatory sections include: Executive Summary, Introduction and Objectives, and Conclusions. The remaining sections are customizable.

Funded by the European Union

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Climate, Infrastructure and Environment Executive Agency (CINEA). Neither the European Union nor the granting authority can be held responsible for them.

EXECUTIVE SUMMARY

BERTHA's details

Project name	BEhavioural ReplicaTion of Human drivers for CCAM
Project acronym	BERTHA
Grant Agreement number	101076360
Duration and dates	36 months (1 November 2023 – 31 October 2026)
Call and topic	HORIZON-CL5-2022-D6-01-03: Safe, Resilient Transport and Smart Mobility services for passengers and goods
Granting authority	European Climate, Infrastructure and Environment Executive Agency (CINEA), under the powers delegated by the European Commission
Official project website	berthaproject.eu

The BERTHA consortium

N°	NAME	ROLE	COUNTRY
1	INSTITUTO DE BIOMECANICA DE VALENCIA (IBV)	Coordinator	Spain
2	INSTITUT VEDECOM (VED)	Beneficiary	France
3	UNIVERSITE GUSTAVE EIFFEL (UGE)	Beneficiary	France
4	DEUTSCHES FORSCHUNGSZENTRUM FUR KUNSTLICHE INTELLIGENZ GMBH (DFKI)	Beneficiary	Germany
5	CENTRE DE VISIO PER COMPUTADOR (CVC-CERCA)	Beneficiary	Spain
6	CAPGEMINI ENGINEERING DEUTSCHLAND SAS & CO KG	Beneficiary	Germany
6.1	VORTEX - ASSOCIACAO PARA O LABORATORIO COLABORATIVO EM SISTEMAS CIBER-FISICOS E CIBERSEGURANCA (VOR)	Affiliated entity	Portugal
7	CONTINENTAL AUTOMOTIVE FRANCE SAS (CON)	Beneficiary	France
8	FUNDACION CIDAUT (CIDAUT)	Beneficiary	Spain
9	AIT AUSTRIAN INSTITUTE OF TECHNOLOGY GMBH (AIT)	Beneficiary	Austria
10	UNIVERSITAT DE VALENCIA (UVEG)	Beneficiary	Spain
11	EUROPCAR INTERNATIONAL	Beneficiary	France
12	F. INICIATIVAS, CONSULTADORIA E GESTAO, UNIPESSOAL, LDA (FI)	Beneficiary	Portugal
12.1	F. INICIATIVAS ESPANA I MAS D MAS I SLU (FI_ES)	Affiliated entity	Spain
15	SMART EYE AKTIEBOLAG	Beneficiary	Sweden

Funded by the European Union

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Climate, Infrastructure and Environment Executive Agency (CINEA). Neither the European Union nor the granting authority can be held responsible for them.

Project's summary

The main objective of BERTHA is to develop a scalable and probabilistic Driver Behavioural Model based mostly on Bayesian Belief Networks (BBN). The DBM will be implemented on an open-source HUB (repository) to validate the technological and practical feasibility of the solution with industry, and provide a distinctive approach for the model worldwide scalability. The resulting DBM will be translated into a simulating platform, CARLA, using various demonstrations which will allow the construction of new driving models in the platform.

BERTHA will also include a methodology which, using the HUB, will allow to share the model with the scientific community, in order to facilitate its growth.

The project includes a set of interrelated demonstrators to show that the DBM can be used as a reference to design human-like, easily predictable and acceptable behaviours of automated driving functions in mixed traffic scenarios.

BERTHA is expected to go from TRL 2 to TRL 4. The requested EU contribution is €7,981,801. The consortium, formed by several entities from different countries, deems this Project as vitally relevant to the CCAM industry due to its impact for safer and more human-like CAVs and its market and societal adoption.

Document details

Deliverable type	Code (SCENIC scripts) & descriptive report.
Deliverable n°	D4.1
Deliverable title	Drivable scenes for simulation
Lead beneficiary	CVC-CERCA
Work package and task	WP4 – T4.1
Document version	1.1
Contractual delivery date	April 2025
Actual delivery date	May 2nd , 2025
Dissemination Level	Public
Purpose	SCENIC scripts (.scenic) allow to have an executable version of the UCs of BERTHA project (D1.1) in CARLA simulator.

Funded by the European Union

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Climate, Infrastructure and Environment Executive Agency (CINEA). Neither the European Union nor the granting authority can be held responsible for them.

Document's abstract

WP4 involves training AIs to learn from or cooperate with a DBM in the context of autonomous driving and driving assistance. This allows us to evaluate the DBM's utility. The corresponding experimentation will be conducted in the open-source CARLA simulator. Regardless of the driving experiences considered in WP4 (including random driving), it is necessary to cover the use cases (UCs) defined in WP1. These consist of five UCs, with one more added from WP4.

It has been established that the SCENIC language will be used to implement these UCs. Consequently, six *".scenic"* files have been created to encode the corresponding UCs. These serve as the basis for generating specific scenarios. Therefore, in WP4 terminology, the established scenes have been successfully codified.

Document's revision history

The following table describes the main changes done in the document since it was created.

REVISION	DATE	DESCRIPTION	AUTHOR (PARTNER)
V1.0	April 30, 2025	1st complete version	Antonio M. López, Alex F. Levy, Rubén Abad (CVC)
V1.1	May 2nd, 2025	Final version after revision	Andrés Soler, Helios de Rosario (IBV)

Terminology and acronyms

TERM/ACRONYM	EXPLANATION
DBM	Driver Behavioral Model

Disclaimer

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Climate, Infrastructure and Environment Executive Agency (CINEA). Neither the European Union nor the granting authority can be held responsible for them.

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Climate, Infrastructure and Environment Executive Agency (CINEA). Neither the European Union nor the granting authority can be held responsible for them.

1. INTRODUCTION AND OBJECTIVES

This deliverable (D4.1) is a direct result of task T4.1 of WP4.

Within the context of WP4, proof-of-concept demonstrations are developed to showcase the potential uses of DBMs. These demonstrations are organized as tasks. Except for task T4.8, all other demonstrations are carried out using the CARLA simulator [1].

Essentially, in a demonstration, a system is subjected to a driving experience. This system can range from an AI for autonomous driving to an AI that supports human driving. The idea is to see how DBMs can help the system achieve better results in some aspects of interest.

In any case, this leads us to the need to design and implement the aforementioned driving experiences in the CARLA simulator. For this, there are two fundamental aspects:

- Maintain close collaboration with WP1, responsible for DBM development, with the aim of establishing relevant driving experiences.
- Determine the best way to implement these driving experiences in the CARLA simulator.

More specifically, each driving experience can be divided into two logical components: one static, the other dynamic. The static component defines a traffic event; for example, a simple case would be stating that “a vehicle has to make a left turn at a traffic-light-controlled intersection.” The dynamic component is established by the details; in the previous example, this would involve setting the speed at which the vehicle approaches the traffic light, the type of vehicle, the sensors on board, the state of the traffic light, etc. In the context of WP4, we term ***scene*** to an example of the static component, while we term ***scenario*** to an example of the dynamic component. In short, the scenarios can be seen as instances of the scenes.

Deliverable D4.1 focuses on scenes, and it is complemented by deliverable D4.2 [4] that focuses on scenarios. These deliverables actually consist of executable code, but we complement the code with documents providing the necessary context for a good understanding. Accordingly, **this document focuses on scenes**.

2. METHODS

A priori, from the WP4 perspective, scenes and scenarios could arise randomly while driving through the different towns offered by the CARLA simulator. However, considering the work carried out in WP1, it must be taken into account that, in the context of the BERTHA project, although one can speak of a DBM architecture, the components used, and their parameterization will be tailored to specific use cases [3].

In particular, in the context of the BERTHA project, five use cases (UCs) have been established:

- *UC1: “Collision risk avoidance”* on highway.
- *UC2: “Insertion on highway”*.
- *UC3: “Pedestrian crossing”* in urban area.
- *UC4: “Left turn at urban intersection”* with traffic lights.
- *UC5: “Pull back in”* on urban highway.

Funded by the European Union

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Climate, Infrastructure and Environment Executive Agency (CINEA). Neither the European Union nor the granting authority can be held responsible for them.

Once WP4 started, we found also useful to add a variant of UC2, that we could term:

- *UC2b: "Insertion on new highway lane due to current lane termination"*.

Therefore, we have six UCs in WP1 terminology and so six scenes in WP4 terminology.

From the point of view of what we call scenes and scenarios in WP4, this implies:

- Automating specific driving experiences in CARLA.
- Deploying the UCs in the towns available in CARLA.
- Applying best practices from the field of machine learning when defining the development conditions of the DBMs (WP1) and the conditions for their validation (WP4).

2.1. Automating specific driving experiences in CARLA

Two approaches to force programmatically driving experiences in CARLA have been evaluated. One approach is based on CARLA's tool known as ScenarioRunner [5]. The other approach is based on a third-party open-source tool which is gaining popularity, known as SCENIC [6]. Both can be adapted to have AI driven vehicles, rule-based vehicle behaviors, or human drivers in the loop. As a main difference, we have observed that the ScenarioRunner approach allows long drives on a CARLA's town, along which different driving events appear, while SCENIC focuses on short drives, i.e., on performing short-time/small-area driving maneuvers. Both approaches have been determined to be complementary and usable in WP4. However, to implement the UCs, we found SCENIC to be a better match. Moreover, it is possible to translate SCENIC programming to OpenScenario programming with tools developed by BERTHA partners.

2.2. Deploying the UCs in CARLA

The 15 towns available in version 0.9.15 of CARLA, which is the version established as a reference in the BERTHA project, were considered. The locations where each of the UCs could be deployed in these towns were identified. Therefore, the six respective scripts ("*.scenic*") were programmed in the SCENIC language, i.e., one for each UC. Essentially, each script establishes:

- The CARLA's town to be used.
- Road characteristics (number of lanes, their direction, etc.) so that the SCENIC system can deploy the UC in different locations within the specified town.
- Objects dynamically participating in the UC: cars, pedestrians, traffic lights.
- Behavior of these objects.
- Parameters that define the UC, each with an associated range of valid values (or a constant value). Some parameters are related to the objects (location, direction, and speed of the vehicles, location and direction of the pedestrians, state of the traffic lights, etc.), others are related to environmental conditions (time of day, cloudiness, etc.).

Funded by the European Union

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Climate, Infrastructure and Environment Executive Agency (CINEA). Neither the European Union nor the granting authority can be held responsible for them.

Regarding the behavior of the vehicles, we must take into account that, at this time of BERTHA project, there are no DBMs available that can be executed, something that is planned to happen after the delivery of this document and its associated "*.scenic*" files. Therefore, for now, to validate the operation of the "*.scenic*" files, we have programmed what could be summarized as rule-based reasonable behaviors. However, these behaviors can be easily changed. Thus, for each UC, an executable DBM will be assigned to one or another participating vehicle once they are available in CARLA.

2.3. Machine learning best practices

As we have mentioned, in BERTHA a DBM architecture will be developed and tailored for each of the considered UCs. These UCs will be used in different WP4 tasks to demonstrate DBM usefulness. Some of these tasks (e.g. T4.4 and T4.5) include the use of AI models that can be trained or fine-tuned based on DBMs or can collaborate with DBMs. Therefore, from a machine learning perspective, it is advisable to work with data segregated for "development" and "validation":

- Development: tuning of the DBMs themselves (WP1) and training of AIs (WP4).
- Validation: testing of the AIs (WP4).

Taking into account the tasks in WP1 and WP4, it has been established that the clearest way to set this separation is as follows:

- Assign different town locations for development and validation of each UC provided it is possible, otherwise run the UCs with different parameters (scenarios).
- Regarding the predefined "weathers" in CARLA, assign some for development, others for validation, and others for both as is common practice [2]. This mainly affects the appearance of the images captured on board the vehicles.

3. RESULTS

3.1. Scenic files

The main assets of task T4.1 are therefore six "*.scenic*" files which are named as follows:

- *uc1_collision_risk_avoidance.scenic*
- *uc2_insertion_on_highway.scenic*
- *uc2b_insertion_on_new_highway_lane.scenic*
- *uc3_pedestrian_crossing.scenic*
- *uc4_left_turn_at_urban_intersection.scenic*
- *uc5_pull_back_in.scenic*

Funded by the European Union

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Climate, Infrastructure and Environment Executive Agency (CINEA). Neither the European Union nor the granting authority can be held responsible for them.

As seen in [4], SCENIC also uses the extension "*.scene*". However, in WP4 terminology, it is not to refer to scenes but to scenarios. In theory, the nomenclature could lead to confusion, but in practice it is not the case since the "*.scene*" are the files that can be actually executed since these are instances of the "*.scenic*". **These six "*.scenic*" files are distributed together with the scenarios ("*.scene*") in THIS LINK, as described in [4]**.


```
"""
The ego vehicle drives on a straight road and another car suddenly incorporates into its lane abruptly.
"""

#### --- CONFIGURATION --- ####

param town = "Town04"
param map = localPath(f'../maps/{globalParameters.town}.xodr')
param weather = 'ClearNoon'

model scenic.simulators.carla.model

# Parameter selectors
param adv_speed = 0
param adv_switching = 2
param adv_spawning = 0
param dynamic = True
param end_scenario = True
param ego_behavior = True

# Value ranges
RANGE_ADV_SWITCHING = [[0, 2], [2, 7], [7, 20]]
RANGE_ADV_SPAWNING = [[20, 40], [40, 60], [60, 80]]
RANGE_ADV_SPEED = [[70, 80], [80, 90], [90, 100]]

# Apply selected parameter ranges
ADV_SPEED = Range(RANGE_ADV_SPEED[globalParameters.adv_speed][0], RANGE_ADV_SPEED[globalParameters.adv_speed][1])
ADV_SWITCHING = Range(RANGE_ADV_SWITCHING[globalParameters.adv_switching][0], RANGE_ADV_SWITCHING[globalParameters.adv_switching][1])
ADV_SPAWNING = Range(RANGE_ADV_SPAWNING[globalParameters.adv_spawning][0], RANGE_ADV_SPAWNING[globalParameters.adv_spawning][1])
SWITCH_LEFT_LANE_OFFSET = -3

EGO_MODEL = "vehicle.lincoln.mkz_2017"

EGO_SPEED = 20

#### --- BEHAVIORS --- ####

# Adversary behavior: fast car moves forward, then changes lanes if ego is close
behavior FastCarBehavior():
    if globalParameters.dynamic == True:
        while ((car_spec can see ego) or ((distance from self to ego) < ADV_SWITCHING)):
            do AutopilotBehavior(speed=ADV_SPEED)

        do DisableAutopilotBehavior()
        while (ego.laneSection != car_spec.laneSection):
            do AutopilotBehavior(speed=ADV_SPEED, lane_change="Right")

behavior EgoBehavior():
    if globalParameters.ego_behavior == True:
        do AutopilotBehavior(speed=10)

#### --- GEOMETRY & LANE SELECTION --- ####

# Choose lanes with a left lane available (for potential lane changing)
laneSecsWithLeftLane = []
for lane in network.lanes:
    for laneSec in lane.sections:
        if laneSec._laneToLeft is not None and laneSec._laneToRight is None and laneSec._laneToLeft.isForward == laneSec.isForward:
            laneSecsWithLeftLane.append(laneSec)

assert len(laneSecsWithLeftLane) > 0, \
    'No lane sections with adjacent left lane in network.'

initLaneSec = Uniform(*laneSecsWithLeftLane)

#### --- OBJECT PLACEMENT --- ####

spawnPt = new OrientedPoint on initLaneSec.centerline

# Ego vehicle on the rightmost lane
ego = new Car at spawnPt,
    with blueprint EGO_MODEL,
    with rolename "hero",
    with behavior EgoBehavior()


spawnPtLeft = new OrientedPoint at spawnPt offset by (SWITCH_LEFT_LANE_OFFSET, 0)

# Fast car placed ahead in left lane
car_spec = new Car following roadDirection from spawnPtLeft for -ADV_SPAWNING,
    with regionContainedIn None,
    with rolename "adv_car",
    with behavior FastCarBehavior()

#### --- REQUIREMENTS --- ####

DIST_THRESHOLD = 3*ADV_SPAWNING
require (distance from ego to intersection) > DIST_THRESHOLD

#### --- TERMINATION CONDITIONS --- ####
if globalParameters.end_scenario == True:
    terminate when (((ego can see car_spec) and ((distance from ego to car_spec) > 60)) or ((distance from ego to car_spec) > 100))
```

Figure 3.1.1. Example of ".scenic" for the UC5, i.e., the "uc5_pull_back_in.scenic" file.

Funded by the European Union

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Climate, Infrastructure and Environment Executive Agency (CINEA). Neither the European Union nor the granting authority can be held responsible for them.

As illustrative example, Figure 3.1.1 shows the "*.scenic*" file for the UC5, which is illustrated in Figure 3.1.2. In this file, the relevant parameters have a default value within a stablished range. However, the file can be seen as a template to be later instantiated by setting particular values to generate specific scenarios (see [4]). Figures 3.1.3 to Figures 3.1.7 summarize the purpose of different parts of this SCENIC code.

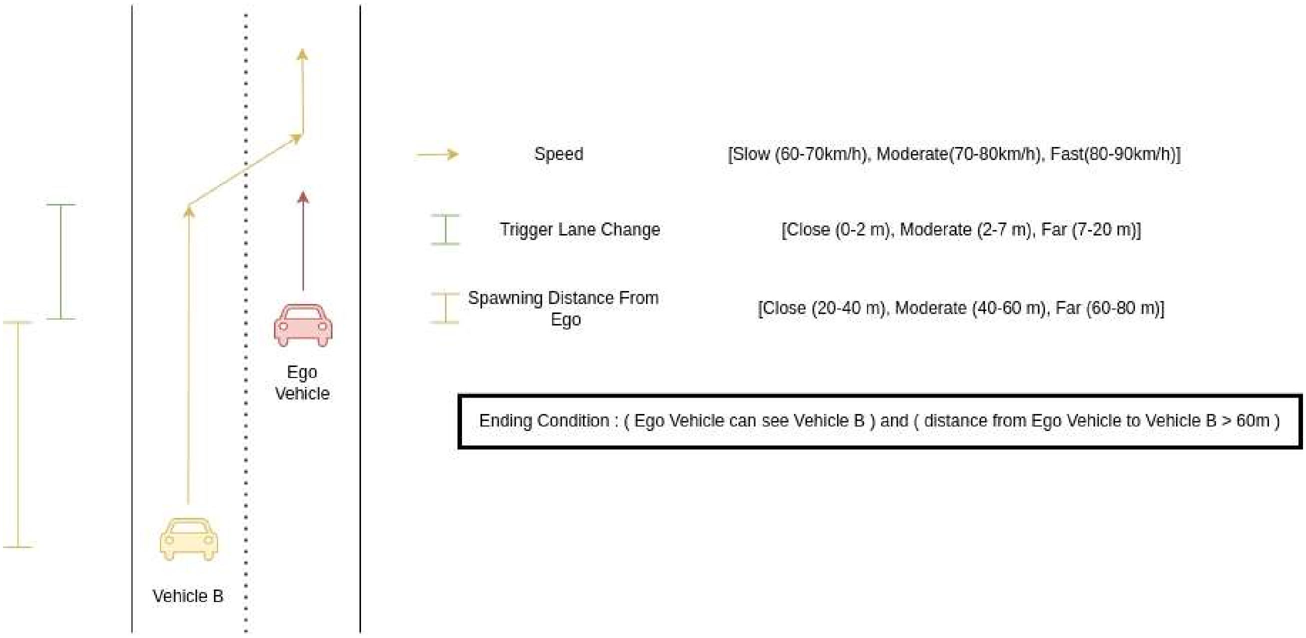
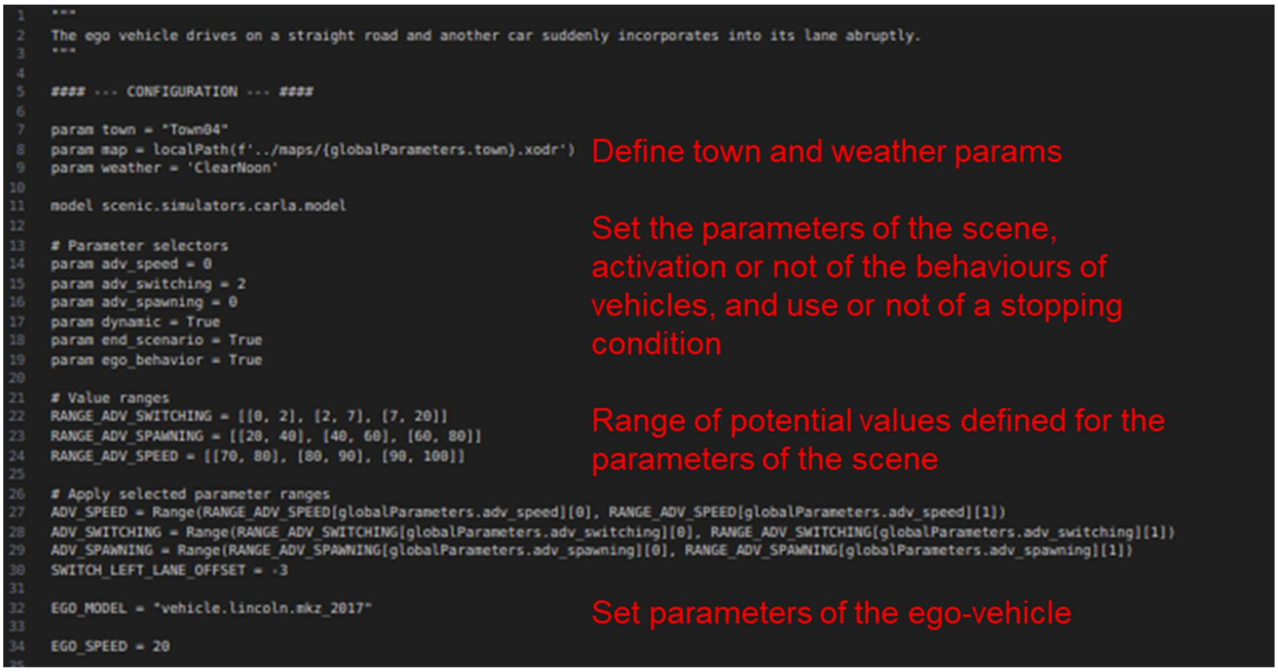


Figure 3.1.2. Visual description of UC5 ("Pull back in" on urban highway).



```
"""
The ego vehicle drives on a straight road and another car suddenly incorporates into its lane abruptly.
"""

#### --- CONFIGURATION --- ####

param town = "Town04"
param map = localPath(f'../maps/{globalParameters.town}.xodr')
param weather = 'ClearNoon'

model scenic.simulators.carla.model

# Parameter selectors
param adv_speed = 0
param adv_switching = 2
param adv_spawning = 0
param dynamic = True
param end_scenario = True
param ego_behavior = True

# Value ranges
RANGE_ADV_SWITCHING = [[0, 2], [2, 7], [7, 20]]
RANGE_ADV_SPAWNING = [[20, 40], [40, 60], [60, 80]]
RANGE_ADV_SPEED = [[70, 80], [80, 90], [90, 100]]

# Apply selected parameter ranges
ADV_SPEED = Range(RANGE_ADV_SPEED[globalParameters.adv_speed][0], RANGE_ADV_SPEED[globalParameters.adv_speed][1])
ADV_SWITCHING = Range(RANGE_ADV_SWITCHING[globalParameters.adv_switching][0], RANGE_ADV_SWITCHING[globalParameters.adv_switching][1])
ADV_SPAWNING = Range(RANGE_ADV_SPAWNING[globalParameters.adv_spawning][0], RANGE_ADV_SPAWNING[globalParameters.adv_spawning][1])
SWITCH_LEFT_LANE_OFFSET = -3

EGO_MODEL = "vehicle.lincoln.mkz_2017"

EGO_SPEED = 20
```


Figure 3.1.3. From Figure 3.1.1, initialization section.

Funded by the European Union

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Climate, Infrastructure and Environment Executive Agency (CINEA). Neither the European Union nor the granting authority can be held responsible for them.

```
#### --- BEHAVIORS --- ####

# Adversary behavior: fast car moves forward, then changes lanes if ego is close
behavior FastCarBehavior():
    if globalParameters.dynamic == True:
        while ((car_spec can see ego) or ((distance from self to ego) < ADV_SWITCHING)):
            do AutopilotBehavior(speed=ADV_SPEED)

        do DisableAutopilotBehavior()
        while (ego.laneSection != car_spec.laneSection):
            do AutopilotBehavior(speed=ADV_SPEED, lane_change="Right")

behavior EgoBehavior():
    if globalParameters.ego_behavior == True:
        do AutopilotBehavior(speed=EGO_SPEED)
```

Set changing lane when it is over the specified distance to the ego-vehicle

Set a simple autopilot behaviour for the ego vehicle

Figure 3.1.4. From Figure 3.1.1, establishing dynamic behaviours.

```
#### --- GEOMETRY & LANE SELECTION --- ####

# Choose lanes with a left lane available (for potential lane changing)
laneSecsWithLeftLane = []
for lane in network.lanes:
    for laneSec in lane.sections:
        if laneSec._laneToLeft is not None and laneSec._laneToRight is None and laneSec._laneToLeft.isForward == laneSec.isForward:
            laneSecsWithLeftLane.append(laneSec)

assert len(laneSecsWithLeftLane) > 0, \
    'No lane sections with adjacent left lane in network.'

initLaneSec = Uniform(*laneSecsWithLeftLane)
```

Choice of a lane with no right lane and a left lane going in the same direction

Figure 3.1.5. From Figure 3.1.1, establishing road information to execute the driving experience.

```
#### --- OBJECT PLACEMENT --- ####

spawnPt = new OrientedPoint on initLaneSec.centerline

# Ego vehicle on the rightmost lane
ego = new Car at spawnPt,
    with blueprint EGO_MODEL,
    with rolename "hero",
    with behavior EgoBehavior()


spawnPtLeft = new OrientedPoint at spawnPt offset by (SWITCH_LEFT_LANE_OFFSET, 0)

# Fast car placed ahead in left lane
car_spec = new Car following roadDirection from spawnPtLeft for -ADV_SPAWNING,
    with regionContainedIn None,
    with rolename "adv_car",
    with behavior FastCarBehavior()
```

Spawning of ego-vehicle with its characteristics

Spawning of Vehicle B with its characteristics

Figure 3.1.6. From Figure 3.1.1, spawning the vehicles participating in the UC.

```
#### --- REQUIREMENTS --- ####

DIST_THRESHOLD = 3*ADV_SPAWNING
require (distance from ego to intersection) > DIST_THRESHOLD

#### --- TERMINATION CONDITIONS --- ####
if globalParameters.end_scenario == True:
    terminate when (((ego can see car_spec) and ((distance from ego to car_spec) > 60)) or ((distance from ego to car_spec) > 100))
```

Making the scenario happening far from an intersection

Set to termination when the Vehicle B is in front of the ego-vehicle at a distance >60 meters, or if the Vehicle B is not in the range seen by ego-vehicle

Figure 3.1.7. From Figure 3.1.1, setting additional requirements and stopping conditions.

Funded by the European Union

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Climate, Infrastructure and Environment Executive Agency (CINEA). Neither the European Union nor the granting authority can be held responsible for them.

3.2. Location of UCs at CARLA's towns

Table 1 summarizes where these UCs are deployed at CARLA towns, considering development and validation phases. For the variable "weather", we have set the development/validation choices shown in Table 2, from those available in CARLA simulator. Note that, although some locations may be similar for the development and validation phases, the parameters instantiating the ".scenes" to generate actual scenarios [4] are different.

Table 1. In what town are located the scenes corresponding to UCs. Yellow means that the scenarios associated to the scenes are planned to be used for validation purposes, blue for development purposes, green for both, and white means no scenes allocated. The column "Figures 3.2.N" refers to figures "N" in section 3.2 showing the corresponding town locations.

	T03	T04	T05	T06	T10	T15	Figures 3.2.N
UC1							2, 3, 4
UC2							2
UC2b							2, 4
UC3							6
UC4							2
UC5							1, 2, 3, 4, 5

Table 2. Split of weathers for development and validation. Colour codes are the same than in Table 1.

ClearNoon	
ClearSunset	
CloudyNoon	
CloudySunset	
HardRainNoon	
HardRainSunset	
MidRainyNoon	
MidRainSunset	
SoftRainNoon	
SoftRainSunset	
WetCloudyNoon	
WetCloudySunset	
WetNoon	
WetSunset	

Funded by the European Union

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Climate, Infrastructure and Environment Executive Agency (CINEA). Neither the European Union nor the granting authority can be held responsible for them.

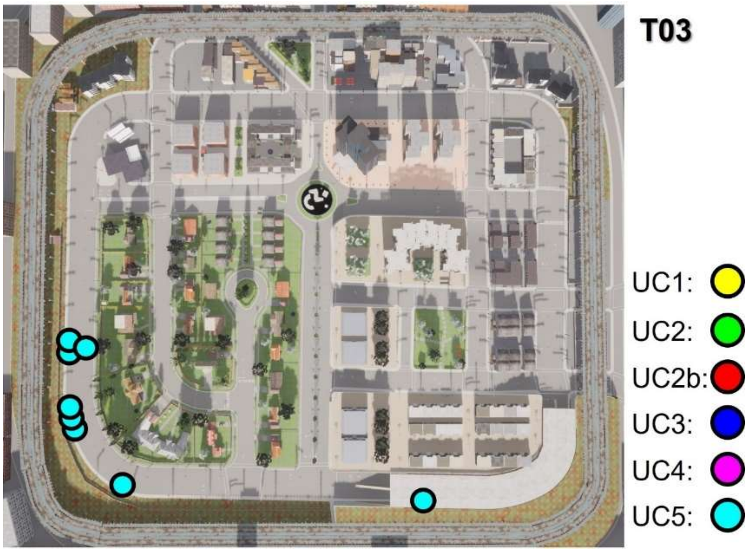


Figure 3.2.1. Top view of CARLA's town 3 (T03), indicating the UCs considered in it. This town is only used in the development phase.

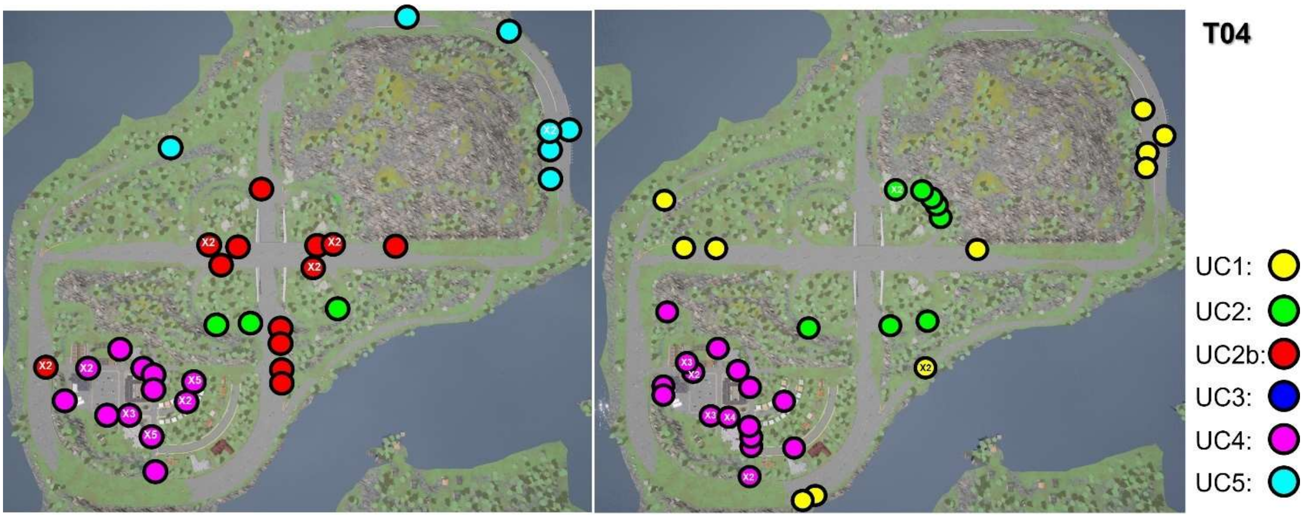


Figure 3.2.2. Top view of CARLA's town 4 (T04), indicating the UCs considered in it. This town is used in the development (left) and validation phases (right). "xN" stands for "N cases" around the indicated location (which would be difficult to show here due to the resolution of the figure).

Funded by the European Union

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Climate, Infrastructure and Environment Executive Agency (CINEA). Neither the European Union nor the granting authority can be held responsible for them.

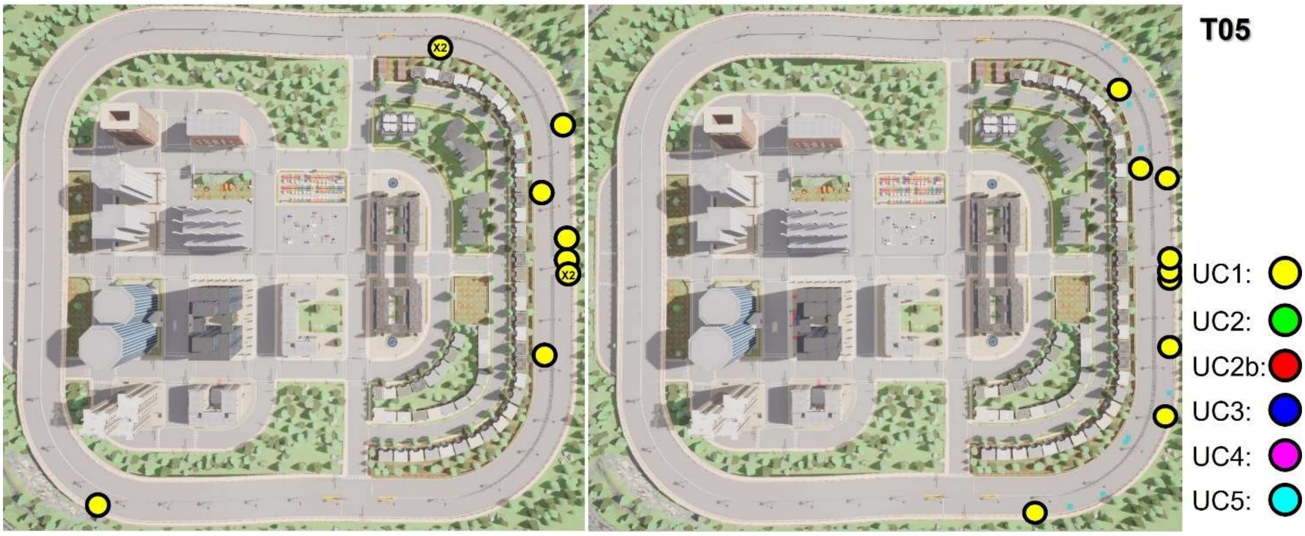


Figure 3.2.3. Top view of CARLA's town 5 (T05), indicating the UCs considered in it. This town is used in the development (left) and validation phases (right). "xN" stands for "N cases" around the indicated location (which would be difficult to show here due to the resolution of the figure).

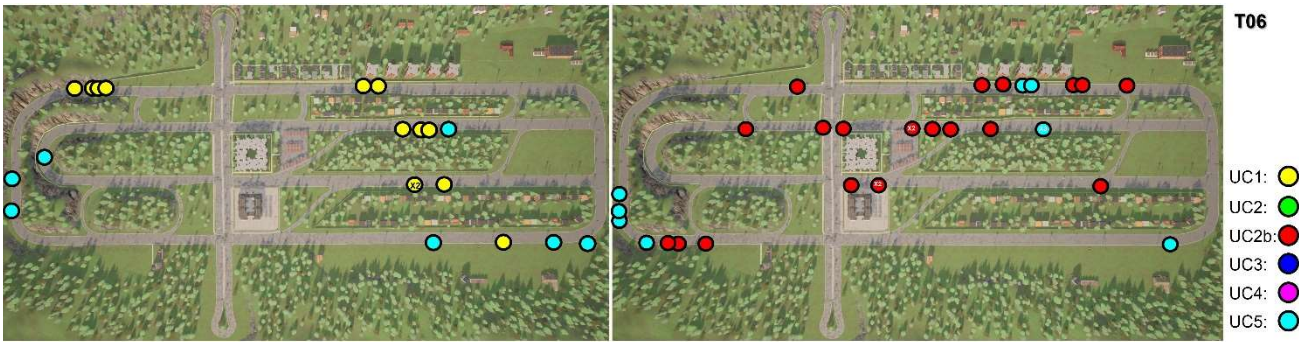


Figure 3.2.4. Top view of CARLA's town 6 (T06), indicating the UCs considered in it. This town is used in the development (left) and validation phases (right). "xN" stands for "N cases" around the indicated location (which would be difficult to show here due to the resolution of the figure).

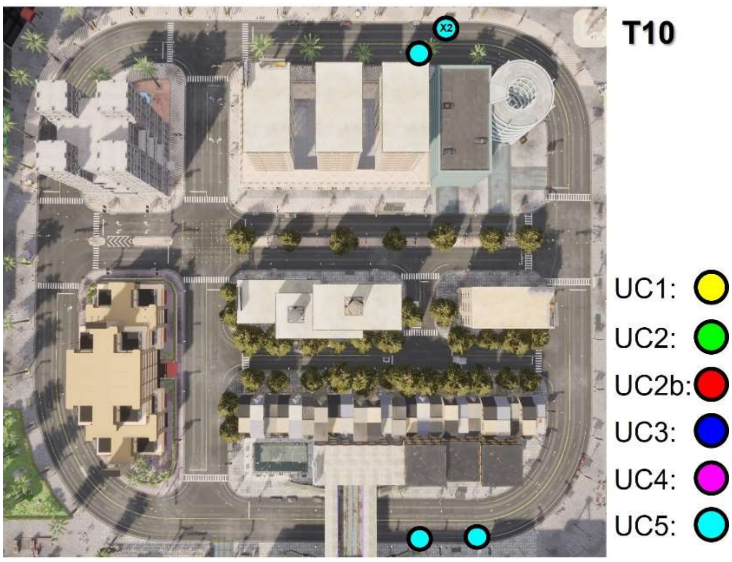


Figure 3.2.5. Top view of CARLA's town 10 (T010), indicating the UCs considered in it. This town is only used in the validation phase. "xN" stands for "N cases" around the indicated location (which would be difficult to show here due to the resolution of the figure).

Funded by the European Union

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Climate, Infrastructure and Environment Executive Agency (CINEA). Neither the European Union nor the granting authority can be held responsible for them.

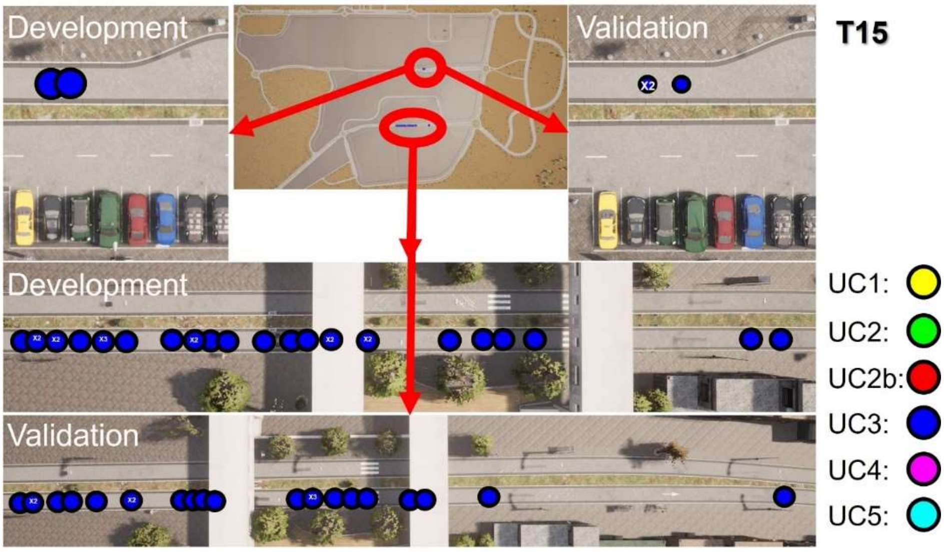


Figure 3.2.6. Top view of CARLA's town 15 (T015), indicating the UCs considered in it. This town is used in the development and validation phases as indicated in the figure. "xN" stands for "N cases" around the indicated location (which would be difficult to show here due to the resolution of the figure).

4. CONCLUSIONS

Various tasks in WP4 involve training an AI to learn from or cooperate with a DBM in the context of autonomous driving and driving assistance. This allows us to evaluate the DBM's utility. The corresponding experimentation will be conducted in the open-source CARLA simulator. Additionally, regardless of the driving experiences considered in WP4 (including random driving), it is necessary to cover the use cases (UCs) defined in WP1. These consist of five UCs, with one more added from WP4.

It has been established that SCENIC will be used to implement these UCs. Consequently, six "*.scenic*" files have been created to encode the UCs. These serve as the basis for generating specific scenarios [D4.2. Currently, since no DBM has been developed yet, hand-crafted behaviors have been set for vehicles, but these can be modified throughout the project.

In conclusion, in WP4 terminology, the established scenes have been successfully codified.

5. REFERENCES

[1] CARLA. https://carla.org

[2] Felipe Codevilla, Eder Santana, Antonio M. López, Adrien Gaidon, "Exploring the Limitations of Behavior Cloning for Autonomous Driving". In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV),* 2019.

[3] BERTHA project: *"Deliverable 1.1. Use cases for the identification of the model".*

[4] BERTHA project: *"Deliverable 4.2. Runnable scenario scripts for simulation".*

[5] ScenarioRunner. https://scenario-runner.readthedocs.io/en/latest/#quick-start

[6] SCENIC. https://github.com/BerkeleyLearnVerify/Scenic

Funded by the European Union

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Climate, Infrastructure and Environment Executive Agency (CINEA). Neither the European Union nor the granting authority can be held responsible for them.