HORIZON EUROPE PROGRAMME
TOPIC HORIZON-CL5-2022-D5-01-08
Clean and competitive solutions for all transport modes
GA No. 101084046

Zero Emission flexible vehicle platform with modular powertrains serving the long-haul Freight Eco System

ZEFES - Deliverable report

D4.4 – Decision making platforms

Funded by
the European Union

Deliverable No.	ZEFES D4.4	
Related WP	WP4	
Deliverable Title	Decision making platforms	
Deliverable Date	2025-02-24	
Deliverable Type	REPORT	
Dissemination level	Public (PU)	
Author(s)	Florian Krietsch (PTV) Tool 1 – Vojtech Dolejsi, Pavel Masita, Robert Naylor, David Sweet (RIC) Tool 2 & 3 – Florian Krietsch (PTV), Nikhil Muthakana (HAN) Tool 4 – Sajib Chakraborty, Sachin Kumar Bhoi (VUB) Tool 5 – Farraen Mohd. Azmin, Volodymyr Hoikhman, Ezhan Elias (RIC); Richard Kratzing (FHG)	
Checked by	Pascal Revereault / Francois Griffond-Boitier (RIC)	2024-11-08
Reviewed by (if applicable)	Ted Zotos (IRU) / Stefanie van Damme (ALI)	2024-11-13
Approved by	Omar Hegazy (VUB) – Project coordinator	2025-02-02
Status	Final	2025-02-24

Publishable summary

The ZEFES project is dedicated to developing innovative powertrain solutions for heavy trucks, specifically focusing on electric and hydrogen technologies and their real-world applications. This deliverable, titled "Decision Making Platforms: Buying Decisions, Route Planning, Vehicle Assignment, Dynamic Correlation, Predictive Maintenance, and AI Applications," outlines the services and their potential interactions within collaborative IT platform scenarios.

A decision-making platform typically provides the following features:

1. ***Data Integration**: These platforms gather data from various sources, making it easy for users to access relevant information. This integration ensures that decisions are grounded in comprehensive and accurate data.*
2. ***Predictive Analytics**: Leveraging advanced algorithms and machine learning, these platforms can predict future trends and outcomes. Predictive analytics helps organizations anticipate challenges, evaluate risks, and uncover opportunities.*
3. ***Scenario Analysis**: Users can create simulations of different scenarios to assess the potential impacts of various decisions. This feature facilitates the visualization of outcomes and the evaluation of the best course of action under different conditions.*
4. ***Collaboration Tools**: Many decision-making platforms incorporate features that enhance collaboration among team members, fostering shared insights and collective problem-solving, which can lead to more informed decisions.*

The ZEFES "Decision Making Platform" supports electric vehicle (EV) users, such as logistics companies, throughout the entire zero-emission vehicle (ZEV) lifecycle. The tools provided by ZEFES assist customers in making investment decisions and managing fleet operations, whether they operate fully zero-emission fleets or mixed operational scenarios. Additionally, the platform focuses on predictive maintenance and repair through its suite of tools and services.

Contents

List of Figures

Abbreviations & Definitions

Abbreviation	Explanation
API	Application Programming Interfaces
BEV	Battery electric vehicle
CCS	Combined Charging System
CGH2	Compressed Gaseous hydrogen
CPO	Charge Point Operator
CRG	Curved Regular Grid
CSV	Comma-separated values
DC	Direct Current
DT	Digital twin
DL	Deep Learning
EOL	End of Life
EV	Electric Vehicle
FCD	Floating Car Data
FCEV	Fuel cell electric vehicle
GTFS	General Transit Feed Specification
H2	Hydrogen
HD	High density

HoLa	Hochleistungsladen im Lkw-Fernverkehr (high performance charging for long-haul trucking)
HRS	Hydrogen Refuelling Stations
JSON	JavaScript Object Notation
JSON-LD	JavaScript Object Notation for Linked Data
LH2	liquid hydrogen
MCS	Megawatt Charging System
ML	Machine learning
NACS	North American Charging Standard
NeTEx	Network and Timetable Exchange. NeTEx is a CEN/ Technical Standard
NRG	Energy Consumption
OCPP	Open Charge Point Protocol
PB	Physics based
REST	Representational State Transfer
ROM	Reduced Order Model
SoC	State-of-charge
SoH	State-of-Health
TCO	Total costs of ownership
t	Ton
TAB	Tabulator
TEN-T	Trans-European Transport Network
TSP	Traveling salesman problem
URI	Universal Resource Identifier
V	Volt
VSS	Vehicle storage system
ZEFES	Zero Emissions flexible vehicle platforms with modular powertrains serving the long-haul Freight Eco System
ZEV	Zero emission vehicle

1 Introduction

The ZEFES project is geared towards testing of novel powertrain solutions for heavy-duty vehicles, i.e., battery electric vehicles (BEV), hydrogen fuel cell electric vehicles (FCEV), and their application in commercial road transport operations in 15 European countries. To enable and to support this application in the field different software and data processes need to be established and tested.
Central to the ZEFES project is the establishment of a digital platform that will serve as data and tool representation of the real operations and will support and optimise the operation of heavy-duty BEV and FCEV vehicles and related logistics processes. The specific goal for the digital twin development within ZEFES is to design and provide an optimization framework to further improve and facilitate heavy-duty BEV and FCEV powertrain deployment considering the interaction with the charging and refuelling infrastructure and long-haul logistics requirements.

The ZEFES platform is a central IT platform enabling trusted, safe and secure connectivity and a facilitator for dedicated tools and services. It enables workflows and connectivity for ZEFES tools. Building upon a central platform and common digital twin data representation and models, task 4.5 provides a toolbox for decision support for key processes in transport planning and operations. Procedures and algorithms will be developed to optimise key deployment scenarios for ZEV (battery and/or fuel cell) support the use cases:

- Decision making support for fleet purchasing, i.e. what type of vehicle should be purchased to fit best the current mission profile? This case is supported by ZEFES tool 1: Buying decision tool.
- Decision making support for planning a mixed fleet, i.e. which vehicle of an existing or future fleet should be used for a specific mission. This use case is supported by ZEFES tools 2 and 3
- Decision making support for timing the maintenance, i.e. support to identify most beneficial timeslots to optimally go for maintenance.

In this regard, 5 specific tools were created and implemented into the ZEFES platform:

Tool 1: Buying decision - a platform function that helps to find a suitable ZEV fleet replacement for a certain (ICE) fleet operation. Default types of different vehicles and vehicle-trailer configurations will be addressed. In a second stage dynamic fleet data will be used to twin the vehicle models to the real-world vehicles to enhance confidence in the tool outputs and ensure resilient fleet choices.

Tool 2: Mission planning - a platform function that optimises the routing including charging stops, for a certain mission by using an operator's fleet specification (both technical characteristic of the vehicles and the missions to conduct) including procedures and algorithm for this mission planning.

Tool 3 - Right vehicle in right duty - a platform function that selects the most suitable vehicles from the fleet for certain operations and addresses the problem of different weight and dimensions safety restrictions for European Modular System (EMS) deployment.

Tool 4 - Dynamic correlation – a platform service to support synchronization between real world and model data with intensive longitudinal data. The dynamic correlation tool builds upon the technique that can estimate the similarity of two curves for irregularly spaced observations and test population-level inferences.

Tool 5 - Predictive Maintenance – a platform function that can predict vehicle maintenance and/or software updates, for the ZEV specific components and systems, particularly the battery, the fuel cell, the e-dolly and e-trailer, potentially the tyres.

2 Methodology & Timeline

This task focuses on integrating various digital ZEFES tools into the ZEFES platform. Considering this technical focused task, a hybrid agile approach has been followed. In accordance with that, teams from all partners have been working individually and meeting on a regular scheduled basis.

To achieve the overall task objective, a sequence of micro-milestones has been set additionally, these can be grouped into two phases:

Local Development Phase:

- Develop each tool individually using a staged approach.
- Deploy the different versions of each tool locally at the partner level.
- Create workflows for tool-level interaction with the platform.
- Adapt and enhance the ZEFES platform, including workflows and the graphical user interface.

Migration and Platform Integration Phase:

- Design migration pathways from local tool deployments at partners' sites to the ZEFES platform.
- Test individual tool implementations in isolation at the platform level.
- Develop workflow patterns to enable seamless interactions among different tools within workflows.
- Testing of the integrated workflows.
- Deploy and demonstrate an intermediate release at the second ZEFES Symposium.

Throughout the development and adaptation phase, a hybrid approach was adopted, involving regular web meetings at work package level to share updates and identify any necessary bilateral interactions. Additionally, one physical workshop was conducted. The workshop was intended to bridge the gap between the local development phase and the migration and deployment phase.
During the migration and platform integration phase, several bilateral and trilateral technical meetings between the ZEFES platform team of TNO and the involved tool providers took place at scheduled or ad-hoc level.

Project timeline for ZEFES task 4.5 can be found in Figure 1.

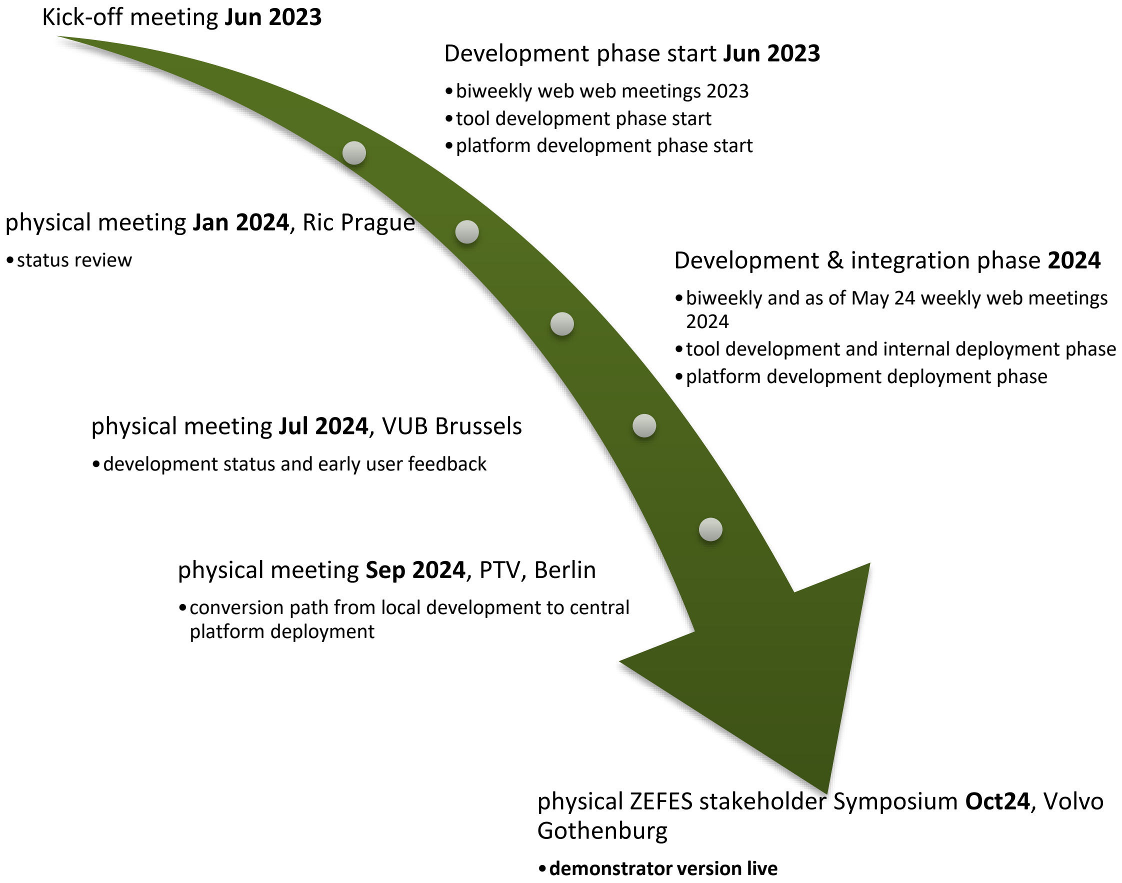


Figure 1 – ZEFES task 4.5 timeline.

3 Decision Making Platform Tools

The ZEFES "Decision Making Platform" (see also D4.1) supports the ZEV users (logistics companies) at several steps of the ZEV lifecycle. This support can be useful to the users for achieving, e.g. CO2 awareness and leveraging CO2 reduction potential. The ZEFES tools cover the investment decision process from a customer's viewpoint as well as the operations of a fleet. The fleet can be a full zero emission fleet, as well as an intermediate, mixed fleet operational scenario. Additionally, predictive maintenance, repair and overhaul is addressed by the tools and services of the decision-making platform.

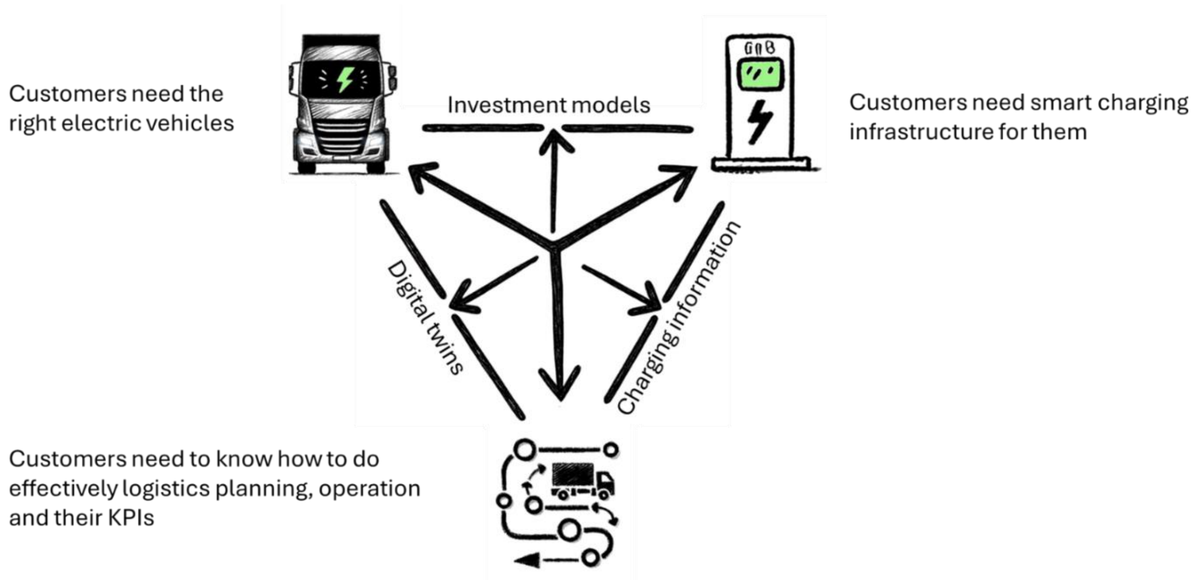


Figure 2 – ZEFES decision making platform concept

This chapter introduces consecutively the 5 different tools of the ZEFES decision making platform.

3.1 Tool 1 - Buying Decision

3.1.1 Tool description

The "Buying Decision Tool" is the first option in the suite of decision-making tools. It predicts energy requirements for zero-emission tractor and trailer combinations using fast running models and historic logged route data. This provides the operational range and total cost of ownership of each suitable vehicle configuration, in order to inform operators on operationally and economically viable options to decarbonise their fleet.

In sustainable commercial transportation, fleet operators are presented with novel challenges when selecting Zero-Emission Vehicle fleets that match their specific operational needs. The need to retain operational efficiency and commercial profitability remain non-negotiable, and therefore their ability to choose ZEV solutions that are suitable for their operations, and in particular the level of certainty and confidence with which those decisions are made, is of paramount importance.

Fleet managers face challenges in estimating a total cost of ownership (TCO) that will be close to reality, gauging the environmental impact of the fleet, optimizing the configuration of the ZEV tractors and trailers. Moreover, the integration of charging infrastructure for the BEVs and refilling stations for the FCEVs presents complexities to mitigate underutilization.

The buying decision tool aims to improve the process of selecting, configuring, and optimizing ZEV fleets, thereby contributing to a more streamlined and effective approach to implementing sustainable transportation. To understand the operators' need, a questionnaire was sent to the ZEFES fleet operators. These conversations provided valuable insights into the limitations of their current process and the expectations they have for a future tool. From these discussions, the functionalities and features that the ZEFES tool should encompass were identified. These are described in the following sections.

3.1.2 Use Case

The diagram below outlines the functionality included in the buying decision tool user interface and the supporting processing workflows needed to generate the models and data for the tool.

- The user interacts with the buying decision tool through the user interface where they can create and save scenarios.
- The scenario represents the usage profile of one or more vehicles that are to be purchased.
- The scenario can be viewed in both map views with various colour axis options (speed, altitude, etc.)
- The user can configure the mission / segment parameters with variables such as load and volume requirements so that only appropriate vehicle models are used in the prediction.
- The user then submits the scenario for processing.
- The results are viewable with the same map and time series plots as described previously – with additional colour axis options available (SOC, H2 level).
- There are also mission and vehicle summary tables to allow the selection of the best vehicles to continue with TCO assessment (using the linked Idiada tool)

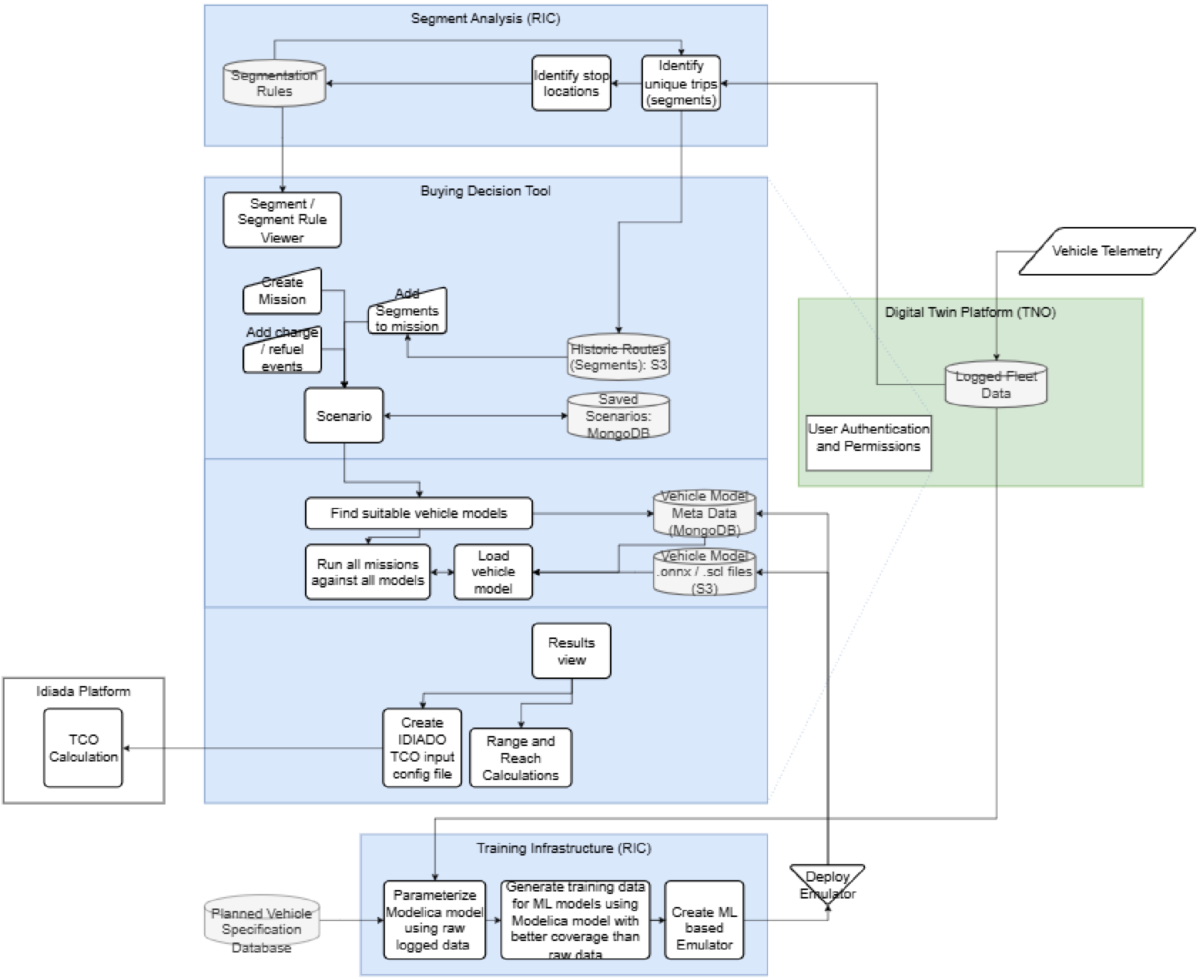


Figure 3 – Tool 1 overall functionality and workflow.

3.1.3 Functions, features & interplay

3.1.3.1 Function and Features

The following are the integral functions and features of the buying decision tool to facilitate informed decision making when selecting a new vehicle for a fleet.

Create realistic operating scenarios:

Build representative mission profiles from route segments isolated from bulk raw GPS data and prepared for simulation / prediction. Using real world GPS data gives the most realistic driving profiles.

Account for driving behaviour:

The segmentation process allows for min / max driving profiles to be identified based on driving 'aggressiveness' allowing the user to see a range of expected consumption depending on driving style.

Add charge / refuel locations:

Add charger and refuel locations to the map at known locations or future potential locations and review the impact on mission feasibility.

Account for temperature on H2 refuelling:
A key concern with FCEV has been identified as the impact of temperature on refuelling. This can be factored into the scenario planning in the buying decision tool such that the mass of hydrogen added when refuelling is temperature dependent.

On demand prediction:
Based on the user input, on demand predictions are made for all truck / trailer configurations that are suitable for the mission using the python server. The modelling techniques used have been selected to ensure that complex scenarios can be predicted with multiple truck options as quicky as possible.

Link to IDIADA TCO calculation Tool:
IDIADA has produced a tool that estimates the Total Cost of Ownership of the vehicle throughout its operational lifespan (documented in ZEFES Deliverable D2.1). The UI facilitates this by allowing the user to download an input configuration file compatible with the IDIADA TCO calculation tool.

3.1.3.2 Interplay
The following interplay is key to the function of the tool.

Raw GPS data from TNO platform:
The route segmentation tool and Modelica model parameterization required access to raw GPS data stored in the TNO platform.

User authentication and permissions from TNO platform:
The TNO platform provides an authentication and permission service that the buying decision tool uses.

3.1.4 Implementation
The buying decision tool consists of a user interface and background processing workflow that prepare the models and analyse the data needed to operate the user interface.

3.1.4.1 Background Processing
There are three main background processing workflows needed for the tool:

- Route segmentation
- Modelica model parameterization
- Reduced Order Model (ROM) tuning

The following sections describe the implementation of each process.

3.1.4.1.1 Route Segmentation:

The overall segmentation process is shown below:

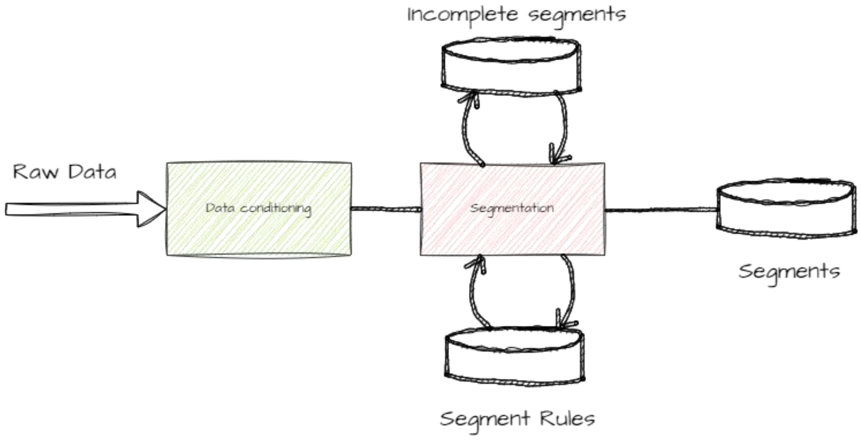


Figure 4 – General route segmentation process.

Conditioning and smoothing the data:

The raw data is conditioned by handling empty rows, clipping and interpolation of the channels and resampling of the dataset to achieve a uniform 1s time step. Some synthetic or calculated channels are created that are required by downstream processing such as total driven distance and 'ignition on' flag. The altitude is smoothed in the distance domain as the models need a 'realistic' gradient without noisy spikes to generate sensible results.

Segmentation rules:

The tool segments the raw GPS data based on:

- Known locations
- Change in payload
- Long stop durations

Known locations: Known locations that have been defined by the user or detected in previous segment processing are stored as location rules. If a vehicle stops inside these defined locations, then the segment is terminated. This rule is always applied.

Change in payload: If the truck is stationary and the sum of the load measured by the truck axle weight sensors changes more than the threshold value the segment is terminated. This rule is always applied.

Long Stop Duration: Segmentation based on long stop durations has the potential to create very short segments so after this segmentation has been completed the segments are re-joined if the total driving time is below an upper threshold with a reduced stopped period.

Short segments: If the total driven distance of a segment created during the segmentation process is less than a distance threshold (3 km by default) the segment is ignored.

Automated location rule creation: When a new location is discovered, it is stored in the segment rules database as a 'known location' rule. Locations dividing two re-joined segments are set as 'inactive'.

The following diagram shows the flow of the segmentation process:

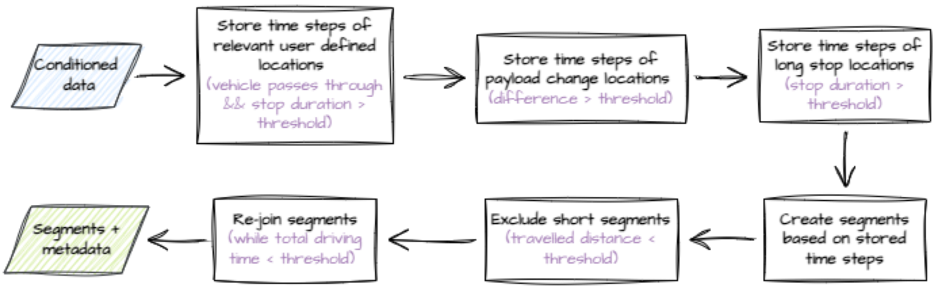


Figure 5 – Detailed route segmentation process.

Incomplete segments:
The last detected segment in the dataset is always checked for completeness. Incomplete segments are stored in the database and matched against future data. If a match is found the stored incomplete segment is prepended to the new data to create a full segment.

Driver behaviour:
To capture driver behaviour, segments are scored for 'aggressiveness' and total segment duration, which is stored with the segment metadata. Full data is stored only for the to-date minimum and maximum 'aggressiveness' segments.

When new segments are found they are matched against the database (start and end locations, distance). If a match is found, then the aggressiveness of the new segment is used to determine if it is a min or max segment in which case the full data is stored, and the segment metadata is updated accordingly.

3.1.4.1.2 Modelica Model Parameterisation:

To create digital twins of each of the use case vehicles, a process for parameterizing Modelica models automatically from processed GPS based route segments was developed.

Model Preparation:
Models were prepared representing the key high-level architecture of the vehicles of interest, along with framework control strategies where required. No special treatment was applied to the models during this stage of preparation, other than implementing a time-based drive cycle reader (rather than the typical distance-based reader used for heavy duty applications).

Specific sub-system models were also built to allow for the tuning of certain parameters in an isolated environment. For example, to accurately tune a battery model, the battery demanded current must be correct. In a full vehicle model, the battery current demand is dependent on vehicle body characteristics, transmission efficiency, e-motor characteristics etc. However, if the battery current has been measured, it is possible to build a standalone battery tuning model by simply connecting the battery to a current source block, and tune battery parameters in an isolated environment, before updating the full vehicle model with these tuned parameters for further system tuning.

The BEV and FCEV vehicle models' main purpose is to accurately predict battery energy or hydrogen consumption over the missions created by the end-user. As such, key features of both the mission profiles and the model vehicle are captured. This includes, but is not limited to:

- Mission vehicle speed / acceleration profile.
- Mission road gradient profile.
- Mission payload.
- Ambient temperature.
- Key vehicle component (battery, e-motor, fuel cell, auxiliary) performance and efficiency data.
- Key vehicle control aspects, e.g., regenerative braking or charge sustaining logic for FCEV vehicles.

Distance or time based: Running the model on a distance basis allows the model to be run on a cycle for which it is not fully capable of following, e.g. maximum velocity, whilst still completing the required distance (and while still respecting the position and durations of the defined stops within the cycle). This is best used for situations where a target vehicle speed trace is being simulated with an alternative vehicle, or at a higher payload. A time-based cycle is required when trying to compare results of the model against the raw data for mean square error evaluation in order to ensure events are happening at the same time for both test and model (in the case of driver error, model de-rate etc.).

Model Parameterization:

A workflow has been developed to iteratively tune the Modelica model when new data is received.

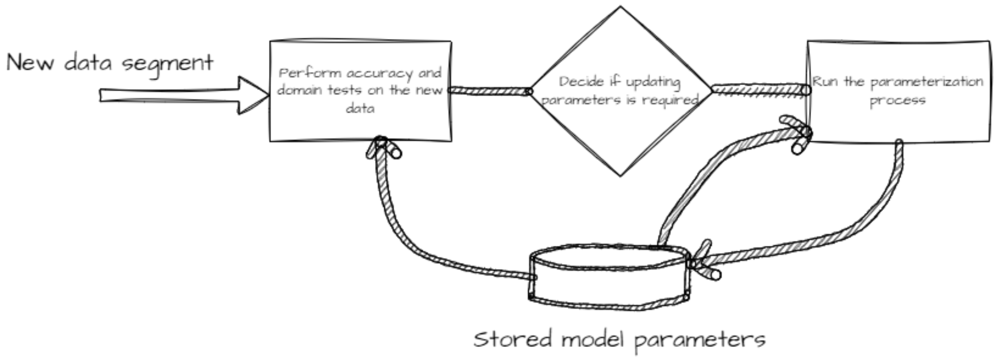


Figure 6 – Model tuning decision logic.

First the incoming data is tested to decide if it is worthwhile tuning the model on the new data using three main criteria.

- **Current accuracy performance of the model on the new data:** there is less needed to train the model using the new data if the model already predicts the output well.
- **Domain of the new data compared with previously trained data:** the domain of the data - whether it has been 'seen before' is accessed using a 2D method across a configurable set of axes. There is less needed to train the model using the new data if it has been seen before.
- **Quantity of data in this domain has been seen:** if lots of similar data has been seen then there is less needed to train the model even if the accuracy of the model is poor.

The decision as to whether to continue with the parameterization is made automatically using preset thresholds and decision logic.

Next - if required - the parameterization process takes the existing parameters and using the new data attempts to improve the fit of the model.

To do this, a list of required dependencies is generated for each vehicle model. For example, a battery electrical model may be tuned first, the results of which may then be fed into a battery thermal model. The combined tuned battery model results (electrical + thermal) may then be input directly to the vehicle model or may be used to tune a battery cooling system model. Once all dependencies are complete (some of which may be done in parallel in separate containers) – the tuned parameters are input to the full vehicle model to tune any further parameters that may be needed.

The parameter tuning process works using optimiser algorithms to minimise the time-based mean square error between a tested variable and a model variable (e.g. battery current/power) which is influenced by the variable. If the parameter has not previously been tuned, a global optimiser (genetic algorithm) based approach is used to generate an initial estimate of the parameter(s) defined in the task. Following the initial estimate, a gradient descent algorithm is used to refine the optimisation.

If a model has previously been tuned, the process is instead initialised using the previously tuned parameters, and only the gradient descent method is used.

Once the parameterization is complete the new parameters are stored in the parameter database relative to this model.

3.1.4.1.3 Reduced Order Model tuning for BEV and FCEV (RIC)

There are several options for creating a Reduced Order Model; mainly based on physics of the modelled system (physics-based models, PB models) or data driven models (machine learning models, ML models). Or various combinations of these. One of these combined approaches is applied in this case as described in the following paragraphs.

Inputs and outputs: The ROMs predict energy consumption (electric energy, hydrogen fuel) which are the outputs based on given drive cycle and conditions (ambient temperature, payload) which are the inputs.

Overall approach description:

The ROMs are constituted of a Physics Based model which broadly represents the vehicle part of the system and a Machine Learning (ML) model which represents the powertrain/energy part of the system as described in the following figure.

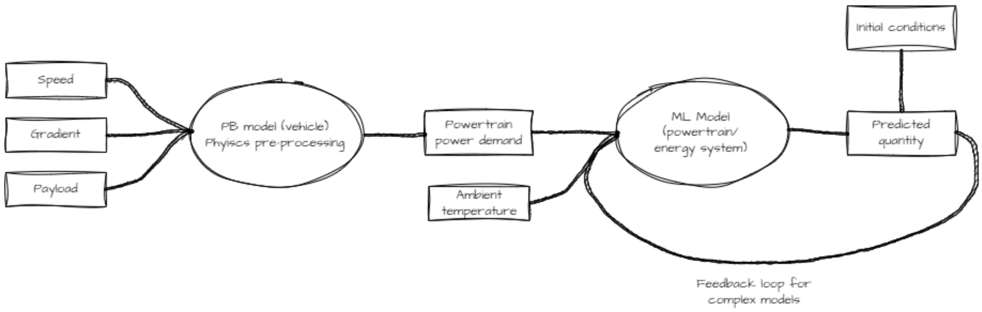


Figure 7 – ROM exploded view.

Sparse Training Data:

Since ML models are data driven, a reasonable amount of data is required to generate such a model however the raw data available is expected to be somewhat sparse and heavily biased to constant speed motorway operation.

As such it was decided to investigate training the ML model using data generated using a Modelica model that was in turn parameterized using the raw data, the idea being that if the Modelica model could capture the characteristics of the real-world truck it could be used to generate data with more complete and uniform coverage of the operating limits than the raw data.

It is expected that the Modelica model - being based on physics equations - will generalize better than a ML model trained using the same data.

Physics Based (PB) vehicle model:

After investigating various options, a two-stage process was implemented with a python physics-based model to represent the vehicle in terms of transformation from drive cycle and payload to powertrain power demand that is then fed to the ML model. The PB model is parameterised using genetic algorithm to find appropriate values based on the training data. The PB model runs open loop and is very quick to evaluate at 1s resolution.

Machine Learning (ML) model:

The machine learning models were artificial Neural Networks (NN) designed and trained using the TensorFlow and Keras packages before being converted to the general ONNX format. These models are stored in S3 and are loaded by the UI back-end server for inference.

The NN models take the powertrain power demand from the PB models and ambient temperature as inputs and produce relative values of energy consumption as outputs. Where possible the models are run open loop as this is fastest but in the case that an output is required to be used as an input (e.g.

SOC in the FCEV) then it is run in a loop which is accelerated by using a 10s sample time for these models.

Stop modelling:

It was found that the neural network model behaviour when the vehicle stood still was unreliable - most notably with the FCEV. It seems that creating a model that works equally well when the vehicle is moving as when it is stationary is not easy. As such it was decided that the neural network model would be used when the vehicle is moving, and a simple physics-based model would be used for stationary sections.

3.1.4.1.4 Reduced Order Model (ROM) tuning for BEV (VUB)

The ROM for the BEV in this tool is designed to predict three crucial parameters: (a) the State of Charge (SoC), (b) Energy Consumption (NRG), and (c) Charging Event (η) based on a given mission profile or speed profile. Traditionally, these predictions can be obtained using a detailed analytical physics-based vehicle and battery model. However, a major drawback of this approach is the significant computational time requirement. Since dynamic models rely on differential equations to generate predictions, they can take a considerable amount of time (typically hundreds of seconds) to process a standard speed profile. This is not ideal for web-based tools, such as a Buying Decision Tool, where quick response times are critical for usability.

To address this, a ROM has been developed to provide fast predictions of SoC, Energy Consumption, and η from speed profiles, typically completing an 1800-second standard WLTP speed profile in just a few hundred milliseconds. This ROM leverages deep learning (DL) and neural networks to simplify a complex dynamic vehicle model. Figure 8 illustrates the block diagram of the ROM for BEV.

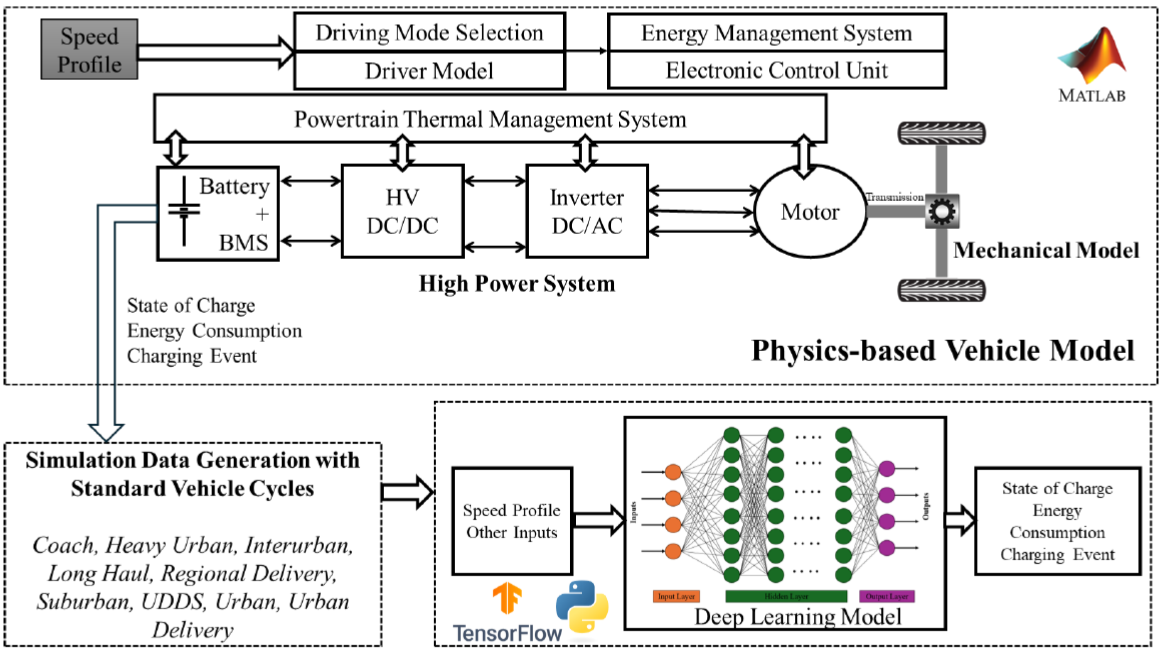


Figure 8 – Overview of development of ROM for BEV.

A comprehensive dataset is generated from the physics-based vehicle model by running simulations across VECTO standard profiles. This dataset serves as the foundation for training the deep learning model. The deep learning model is trained in Python within the TensorFlow environment, where optimization is conducted to determine the ideal model configuration for achieving the highest accuracy in the ROM.

3.1.4.2 User Interface

There are two main components of the user interface:

3.1.4.2.1 Web based user interface (UI):

The user interface is responsible for handling user inputs, getting required data from databases and sending requests to the server to make scenario predictions. It also receives and displays results from the server in map, timeseries plot and tabular format. The user can save and recover scenarios using the UI.

The user interface is written using a modern web application framework (NextJS) to ensure performance and security. Authentication will be handled using the TNO ORY authentication platform which allows the same authentication and permission management as is used on the TNO platform components.

3.1.4.2.2 Python scenario prediction server:

The python server is responsible for loading required models and data from databases and making scenario predictions on request from the UI.

The server supports a streaming response request type which has two main benefits:

- Results are returned with a progress status allowing the UI to update and show results as they are available and keep the user informed of progress.
- It allows inference sessions that last longer than a typical HTTP POST request would allow that would time out.

Ricardo hosts the application components on Amazon Web Services using a Kubernetes cluster in which Helm is used to deploy Docker containers containing the application code. Ricardo is using Argo to orchestrate the data processing workflows described above.

User Interface

Home screen

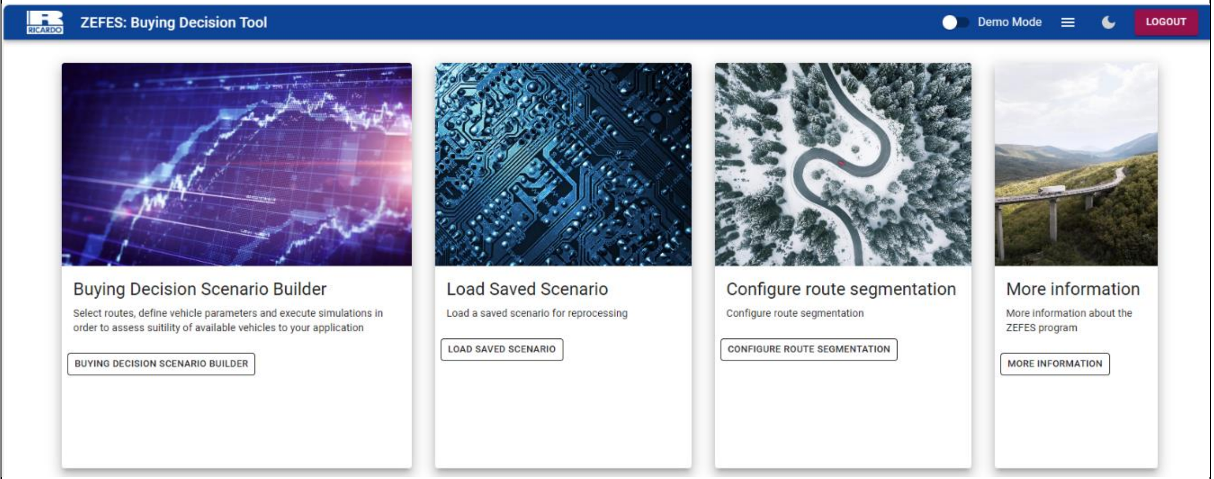
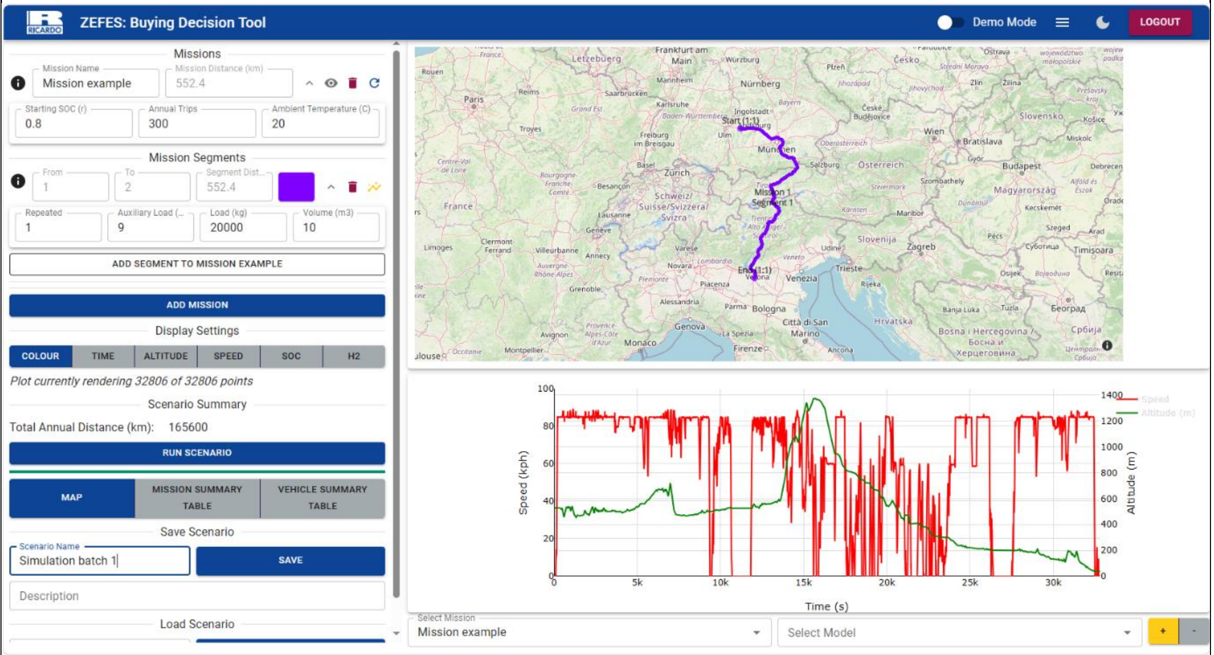


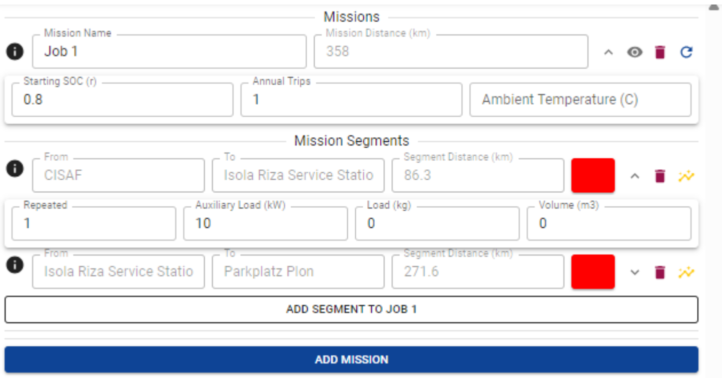
Figure 9 – Buying Decision Tool user interface home screen.

User Interface / Buying decision tool overview



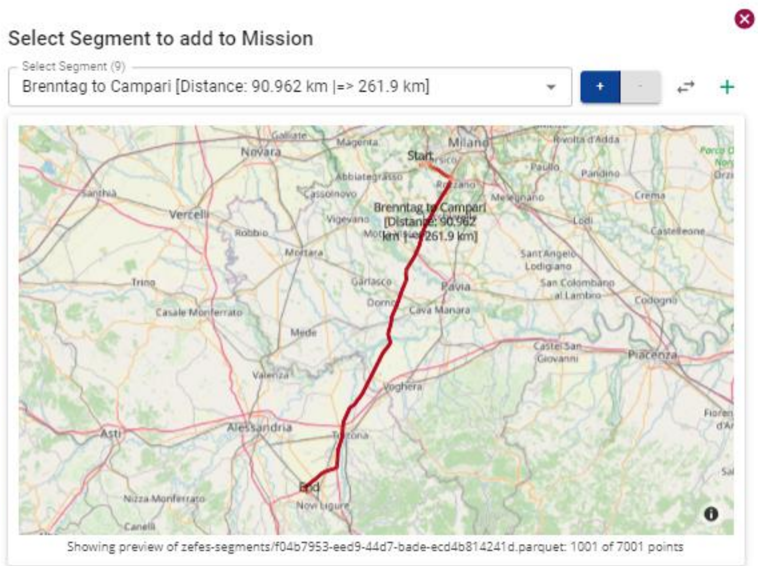
1: Add missions and segments

Parameterize missions and segments

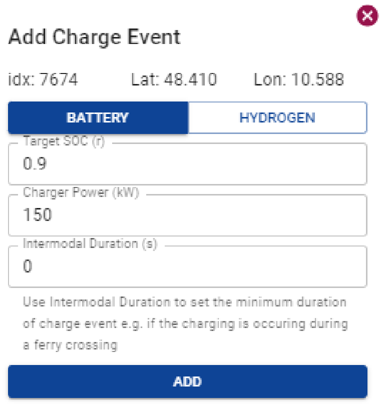


2: Add segments to mission using segment select modal

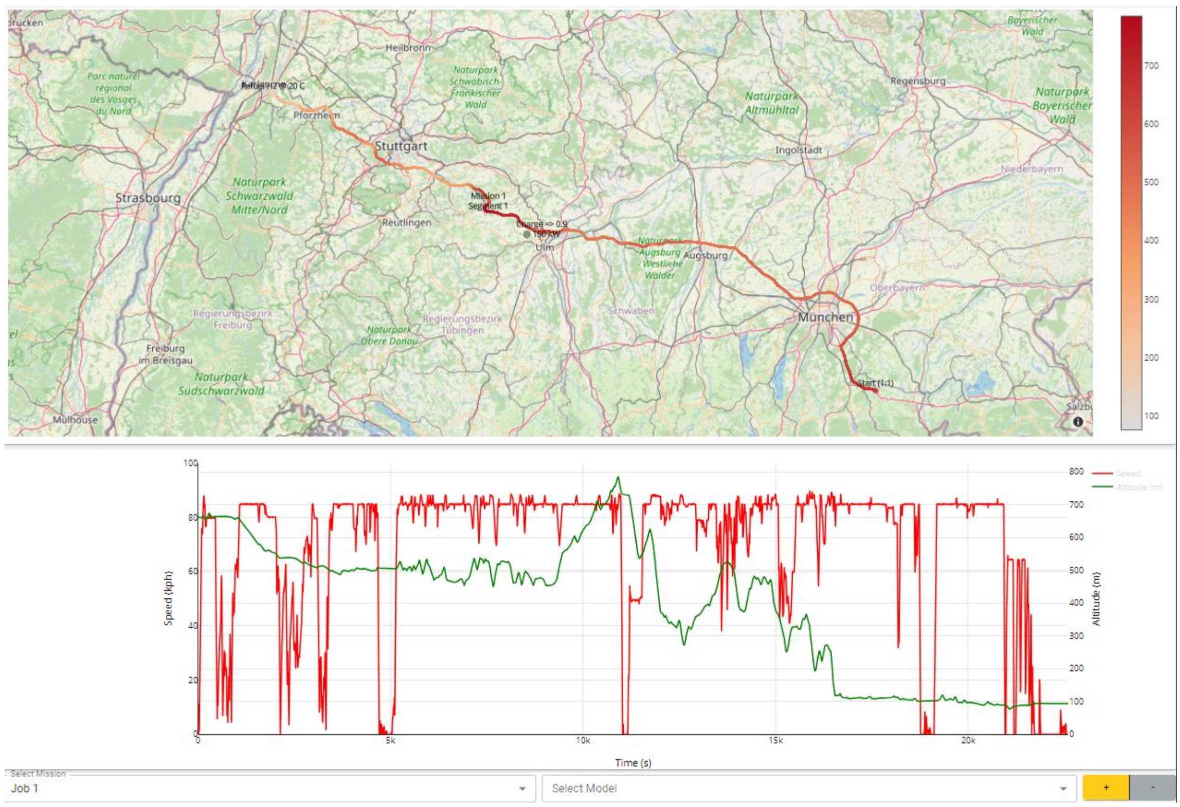
When adding subsequent segments, options are ordered by distance from the end of the previous segment to the start of the next segment



3: Add refuel location for either battery recharge or hydrogen refuel



4: Simulate and view scenario including results on the map and in the timeseries plot

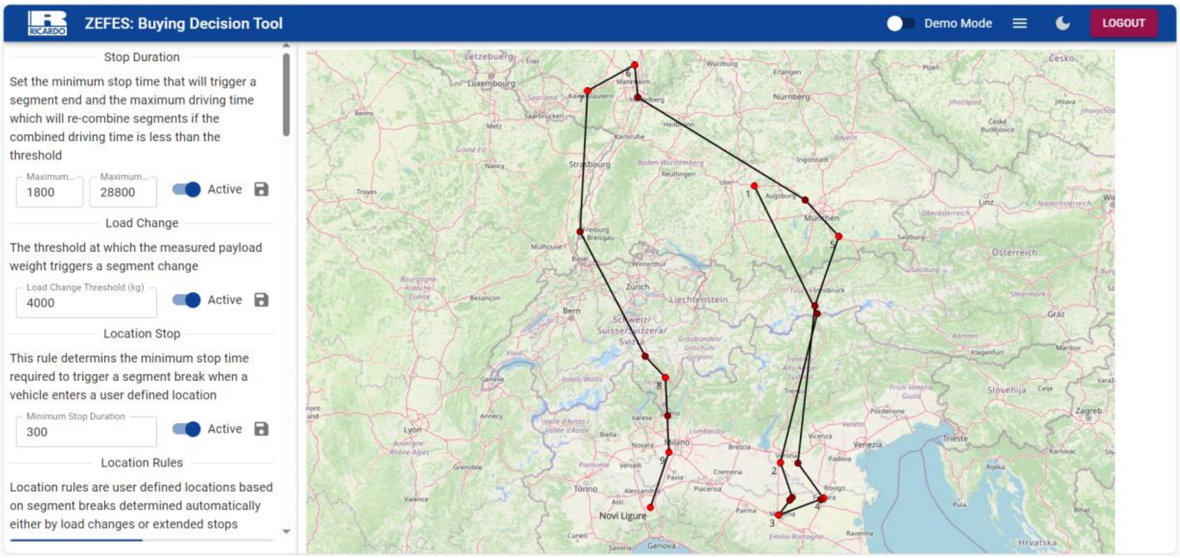


5: View summary results in tabular format

Model Name	Complete Missions	Total Distance [km]	Battery Size [kWh]	Electric Consumption [kWh/100km]	H2 Consumption [kg/100km]	Average Load [kg]	Ending SOC [%]	TCO
FORD1_FCEV	1/1	666	196		6.5	0		
SCA1_BEV_e_trailer	1/1	666	921	105.5		0	57	
REN1_BEV	1/1	666	728	108.9		0	48	

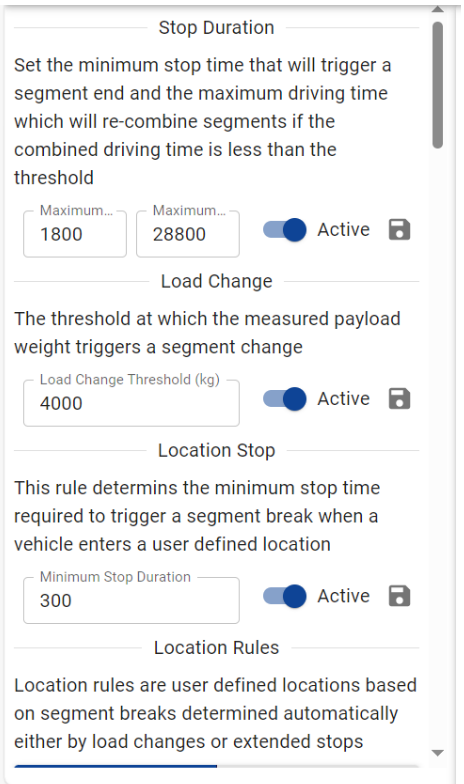
Figure 10 – Features of buying decision tool

User Interface / Route segmentation tool overview

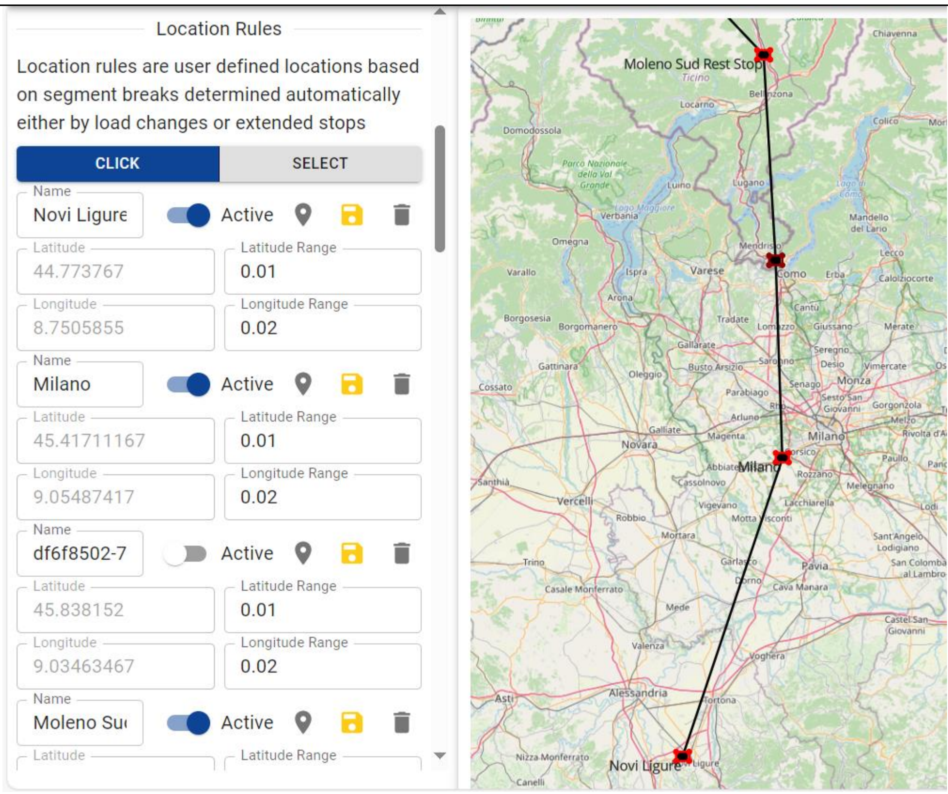


1: Segmentation Rules Configuration

This window allows the user to enable or disable the three main segmentation rules. Thresholds for each segmentation rule can be adjusted here.



2:



Location Rules

All locations detected during the segmentation process (both active and inactive) are stored as location rules including coordinate ranges. Ranges and location names can be changed by the user.

3: Details about stop locations

Details about stop locations can be viewed on the map. Location coordinates and name are displayed along with the reason why the location was identified as a segment end. Locations are surrounded by red or brown rectangles indicating location boundaries defined by latitude and longitude ranges. Red represents an active location, and brown represents an inactive location.

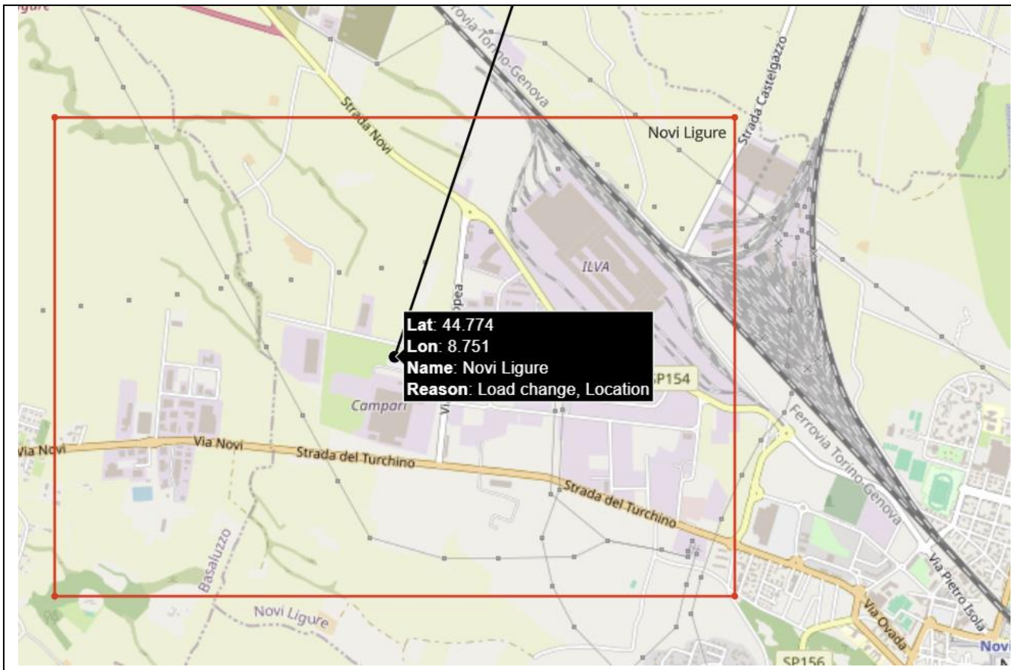


Figure 11 – Features of route segmentation tool

3.2 Tool 2 – Mission Planning

3.2.1 Tool description

With the fast-growing importance of alternative electric and hybrid vehicles in logistics it is vital to understand how these vehicles behave and to be able to use them in a standard routing use case. The operating range of electric vehicles is not only affected by the vehicle's own properties such as its battery capacity, weight, rolling resistance etc. but also by external factors such as temperature, weather conditions, driving style and road gradient, all of which can significantly impact energy consumption. Charging takes much longer than refuelling a petrol or diesel vehicle, so when and where to charge and for how long become crucial factors in a routing calculation. Furthermore, unique to EVs is the possibility of recuperation, and so elevation variance takes on particular significance when routing for EVs (a comprehensive overview of included factors is presented in figure 12).

The tool 2 - mission planning is designed specifically for logistics companies that are interested in electric fleets. It provides a comprehensive catalogue of electric vehicles in Europe, including the vehicles used within ZEFES demonstrations and a powerful tool to calculate & plan routes for electric trucks.
The tool 2 helps logistics companies to discover electric commercial vehicles, and to plan realistic scenarios (1 route or multiple routes) for them, to see, if these vehicles can fulfil the route and transport constraints. This can be performed for a fictive fleet or for an existing mixed fleet. The migration of fleets to electric vehicles helps, reduce their carbon footprint, and contribute to a more sustainable future. The BEV Truck Route Planner provides a comprehensive catalogue of commercial electric vehicles in Europe, including information on their range, charging possibilities and other key specifications. This catalogue is regularly updated and enhanced, to ensure that logistics companies have access to the latest information on electric vehicles. The tool is customizable to fit the needs regarding vehicles or individual functionality.

The route is based on PTV's state-of-the-art routing algorithms and know-how. It calculates the optimal route, considering all relevant vehicle-specific restrictions as well as current and historical traffic conditions. For realistic vehicle consumption on the route, we calculate payloads at each stop and consider influences such as driving behaviour, elevation, temperature and wind effects, vehicle age and battery usage.

3.2.2 Use Case

The use case of tool 2 is split into 2 core functions:

Exploration of EV vehicles

The Comprehensive catalogue of Truck & Van EVs (Electric Vehicles) in Europe provides a catalogue of currently available vehicles. For each vehicle, a realistic consumption calculation is available.

Dedicated route planning of EVs.

deliverer optimized routes for Electronic Vehicle (EV) by factoring in vehicle restrictions, time windows, service times, loadings, traffic, and mandatory rest periods. The EV routing considers relevant vehicle-specific restrictions, driving behaviour, elevations, temperature, and wind influences.

3.2.3 Functions, features & interplay

Elevating Electric Fleet Efficiency

The core functionality of the ZEFES tool 2 lies in its ability to calculate energy consumption for specific electric trucks or vans. By utilizing digital twins and models of commercial electric vehicles, the tool considers critical vehicle attributes such as battery capacity, size, weight, wind resistance, and energy efficiency. This data, combined with realistic simulations of driver behaviour, weather information and environmental conditions, provides precise predictions for range and consumption on each route. This level of accuracy is crucial for optimizing the performance and efficiency of electric fleets.

Handling Complex Logistics Requirements

Planning routes for electric vehicles involves addressing more than just straightforward navigation. The ZEFES tool 2 supports intricate logistical needs, including multiple stops, loading times, time slots, service durations, vehicle restrictions, driving and rest periods, toll calculations, and emissions assessments. By integrating historical and real-time traffic data, the tool API ensures that routes are optimized to meet daily logistical challenges, balancing time and cost efficiency. This comprehensive approach is essential for precise consumption calculations and effective fleet management.

Incorporating Weather, Elevation, and Charging Optimization

Weather conditions and elevation changes can significantly influence the performance and range of electric vehicles. The ZEFES tool 2 takes these factors into account, allowing fleet managers to plan routes that consider potential adverse conditions, see figure below. Additionally, the API enhances charging strategies by determining optimal charging times and locations along the route. This minimizes downtime and maximizes operational efficiency, especially for long-distance trips where effective charge management is vital.

EV – Consumption Influencing Factors

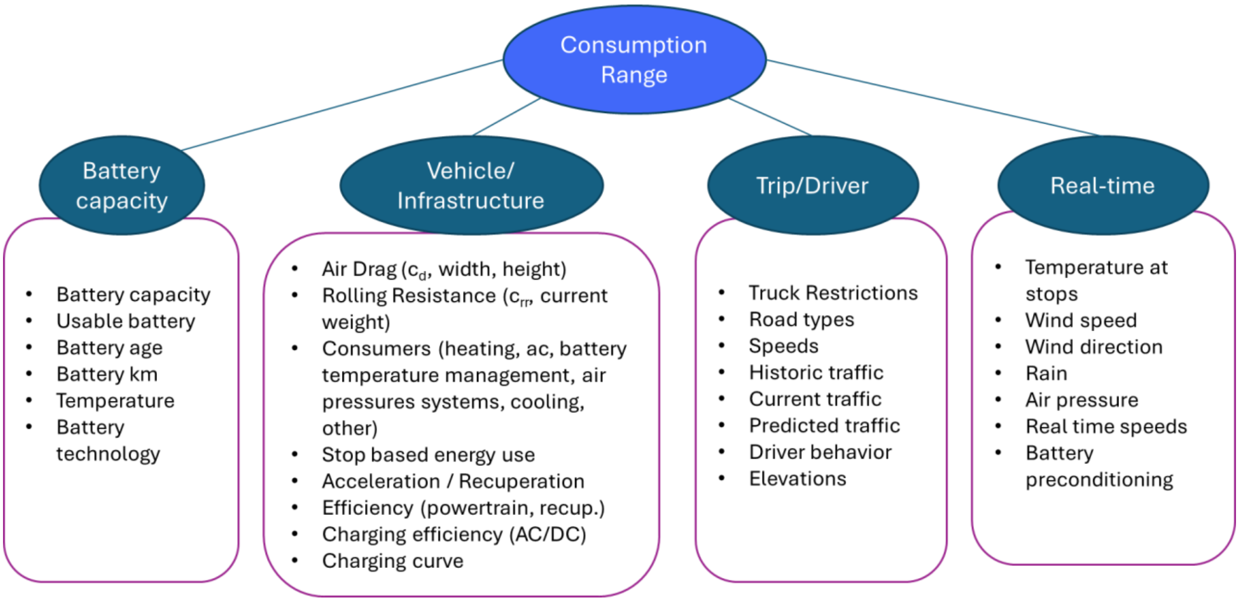


Figure 12 – ZEFES tool 2 and tool 3 consumption influencing factors considered by PTV to estimate the energy need.

Vehicle Database

Electric vehicle models are provided in a separate database and can be retrieved by the *getVehicleModel*s method in the Data API. Although there are a few general reference vehicles provided the idea is to have a collection of vehicle models based on existing vehicles.

Each vehicle contains data like:

- Manufacturer and model.
- Vehicle type and variant.
- Battery capacity and official range.
- Maximum power consumption.
- Charging capabilities and speed.
- Diesel and other fuels powered vehicles (for comparison).

Consumption Calculation

There are many factors which affect energy consumption for electric vehicles such as:

- Acceleration and deceleration.
- Weight and rolling resistance.
- Air drag.
- Wind speed, temperature and weather conditions.
- Grade resistance.
- Power consumption at waypoints (e.g. tail lift, crane, cooling).

Elevation profile

Elevation changes, or the road gradient along the route, can have a substantial effect on energy consumption, not only increasing consumption but also contributing to energy recuperation. The effect is further enhanced by the (load) weight of the vehicle.

(Load) weight

The combined weight of the vehicle and its load can affect rolling resistance, grade resistance, and acceleration factors. For instance, acceleration will impact the energy consumption of an LCV less than it will for a fully loaded HGV, thus different road types can become desirable depending on the overall weight when calculating a route.

Weather

Weather has an impact on energy consumption in various ways, both positive and negative. Temperature affects the battery performance. Wind speed and wind direction affect air drag, especially for vehicles with a big silhouette. The amount of precipitation affects the rolling resistance. The API provides input parameters to set current weather data at each waypoint. If no weather is specified and the start and arrival time are no more than 14 days in the future, weather will be used according to the forecast provided by the weather forecast service Open-Meteo (https://open-meteo.com/). Open-Meteo partners with national weather services to bring you open data with high resolution, ranging from 1 to 11 kilometers. If no live weather is used for the calculation, user individual data is used. As default temperature the tool is configured with a temperature of 23°C and a wind speed of 0 km/h is for the least impact on the energy consumption.

Power consumption at waypoints

The electricity consumption of electrical appliances used during service at each waypoint can be individually defined and accounted for. The energy consumed at a waypoint is removed from the battery at arrival, always before considering a possible charging stop at the same waypoint. This means that the state of charge prior to departure from one waypoint must be sufficient to cover both the subsequent leg's power requirements and any additional consumption at the next waypoint.

Charging

The consumption calculation supports charging at defined waypoints. If waypoints are provided with charging capability, then charging will be incorporated in the route calculation automatically where necessary. Factors taken into consideration include:

- The maximum charging speed at the available charging station(s).
- The predicted state of charge when reaching a charging station (which will impact the charging speed).
- The minimum energy required to avoid dropping below the user-defined *minimumStateOfCharge.*
- Whether it is possible to use service time, i.e. the time the vehicle stops to e.g. load/unload, for charging, or the driver is doing his driver rest time.

The charging algorithm minimizes the additional time needed to charge the battery. Charging during the service time does not need additional time so charging stations where charging is possible during service time may be preferred over those where this is not possible. Similarly, charging with a higher charging power allows to charge the same amount in less time and therefore may be preferred. In the current version, neither a dynamic cost or slot booking nor a station specific energy cost is implemented. This could be part of a future update of tool 2.

The charging time is a proposal, currently for information only. It is not included in the travel time of the route and the start time of subsequent events is not offset by it. The execution of the proposed charge events may lead to inaccurate route results, that need to be considered when planning charging. The delayed departure at a waypoint with a charging station can lead to different issues, including inaccurate traffic conditions, differing toll costs due to time-dependent costs or inaccurate break and rest times.

The state of charge may fall below the defined minimum state of charge or even below zero, if the electricity consumption exceeds the available energy in the battery and charging is not possible before falling below zero. In this case, a warning will be added to the response, informing that the route may not be feasible. The reported state of charge is not limited to zero but may continue below zero; to provide some insight, how much energy is missing to complete the route or how many charging stops would be needed along the route. Charging cannot be planned unless the state of charge remains above 0%. Below this threshold, the battery's charging curve is undefined, making it impossible to create feasible planned routes.

Implementation

Vehicle Models are a more detailed representation of vehicles that extend the predefined profiles. Each vehicle model is based on one predefined profile, which is used for route calculation. The Vehicle Model inherits its properties from the predefined profile but may overwrite them. Furthermore, it extends the predefined profile with additional properties essential for energy consumption and charging calculation. When the EV response fields are requested, the consumption relevant factors are applied on the route and the according response fields are generated. This data then enables cost comparisons with current vehicles and routes.

In addition to EV profiles, the implementation enables the user to check and compare results against combustion engine vehicles.

Fuel Cell and H2 vehicles are to be included during ZEFES WP7 implementation as part of piloting once vehicles and infrastructure are available for modelling. Related external APIs are planned to be linked to the ZEFES platform.

Overwriting Parameters

The existing overwriting hierarchy is extended by one layer - the vehicle model:

- Request parameters.
- Vehicle model.
- Predefined profile.

Base is the predefined profile. Properties in the predefined profile are overwritten by values defined in the vehicle model. This can again be overwritten by requesting parameters in the vehicle object. However, when using a vehicle model some request parameters in the vehicle object are not supported. Those are specifically engineType, fuelType, electricityType, averageFuelConsumption, averageElectricityConsumption, bioFuelRatio, hybridRatio, dualFuelRatio, cylinderCapacity, emissionStandard and co2EmissionClass.

Integration

To use the model-based energy consumption calculation you need to use the POST variant of the calculateRoute operation. Set up your route request with waypoints and other options as usual and follow these steps to get the energy consumption and charge results:

- Set a vehicle model id in the profile parameter.
- Set relevant request parameters at waypoints like load weight, charging possibilities and weather.
- Add the EV_* results you want to see in the response to the results parameter.
- Code Samples: code sample.

With this the Routing API can provide:

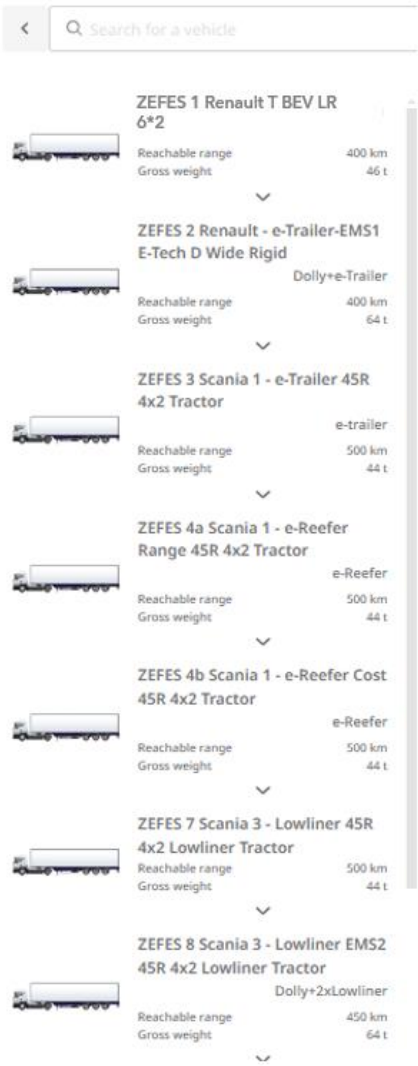
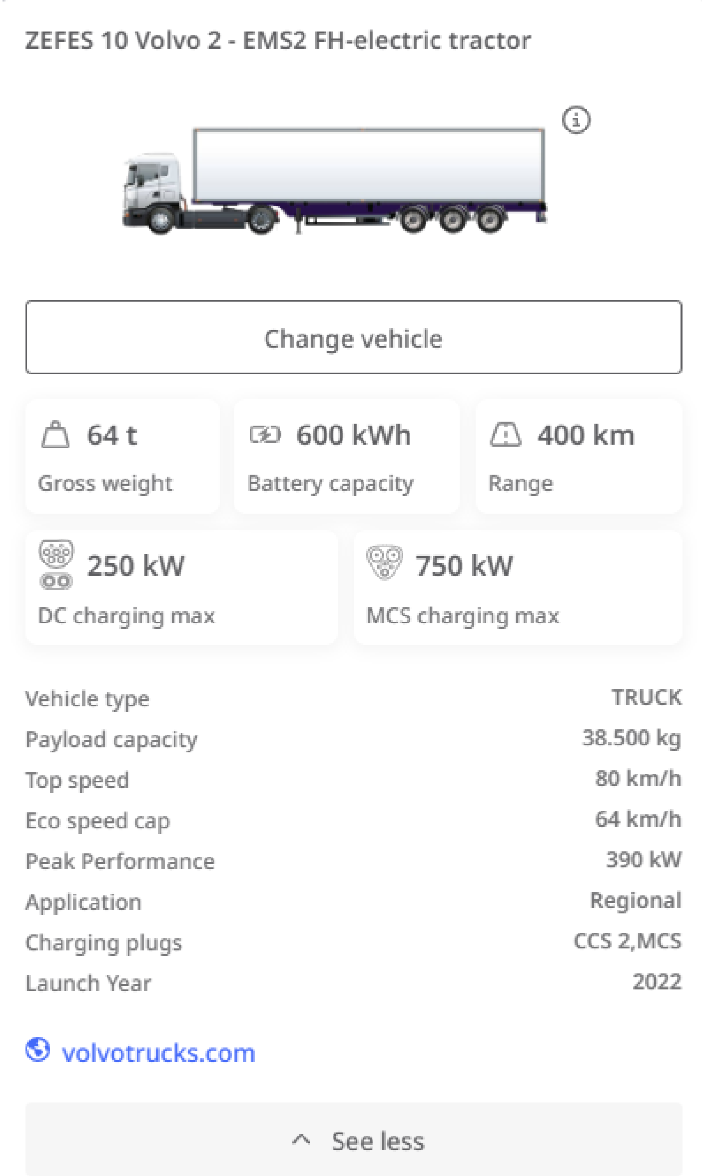
- The state of charge of the battery and energy consumption at the end of the route and at the end of each leg.
- A plan, at which waypoints charging is necessary to reach the destination in the form of charge events.
- Regular update events on the state of charge and energy consumption, optionally with the polyline of the respective route section.

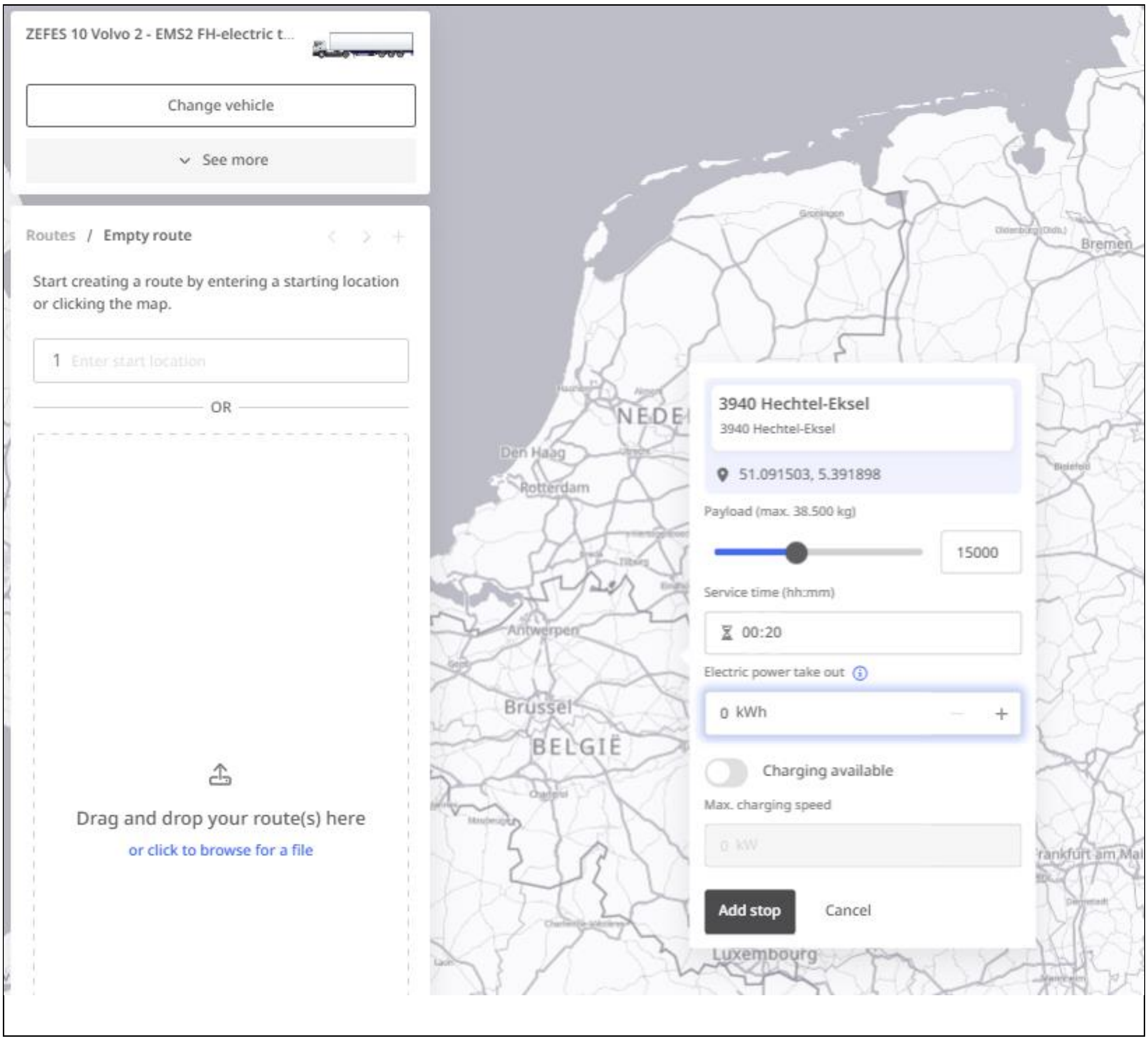
Emissions and Vehicle Models

Emissions can be calculated with vehicle models the same way as with predefined profile. However, some vehicle properties cannot be changed when using a vehicle model, so some emission results are not applicable to vehicle models. Emission relevant properties are automatically used as defined in the vehicle model.

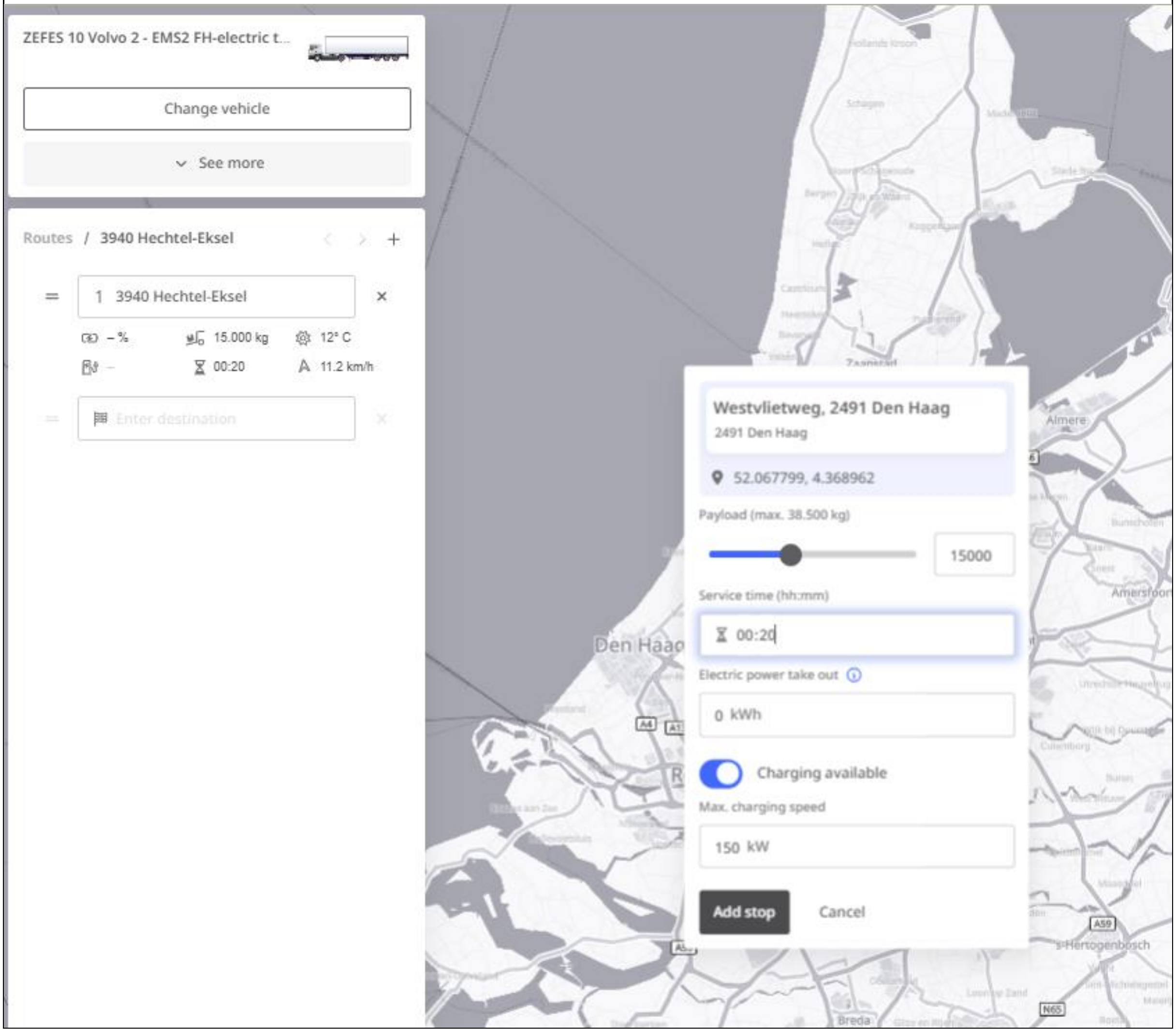
The emission result EMISSIONS_ISO14083_2023 requires vehicle [electricityConsumption] to be provided with predefined profiles. ISO 14083:2023 is applicable to electric vehicles. This standard provides a methodology for quantifying and reporting greenhouse gas (GHG) emissions from transport chains, including those involving electric vehicles. In the case of vehicle models, this value is automatically calculated from the electricity consumption calculated by the Routing API, so it does not have to be provided with the request.

3.2.4 Frontend implementation

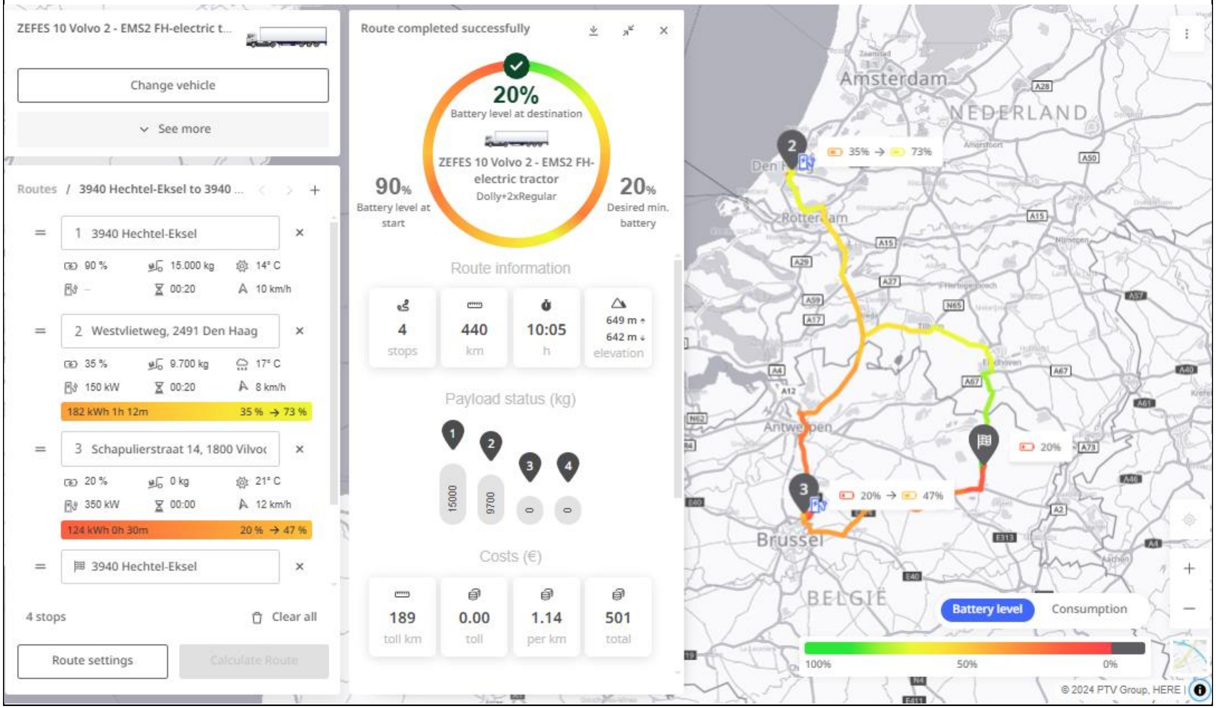
<table>
<tr>
<td>Search and select a vehicle profile (out of 15 stored tractor-trailer combinations)

</td>
<td>Review the vehicle KPI`s per profile.

</td>
</tr>
<tr>
<td colspan="2">Interact with the map to create stop locations</td>
</tr>
</table>

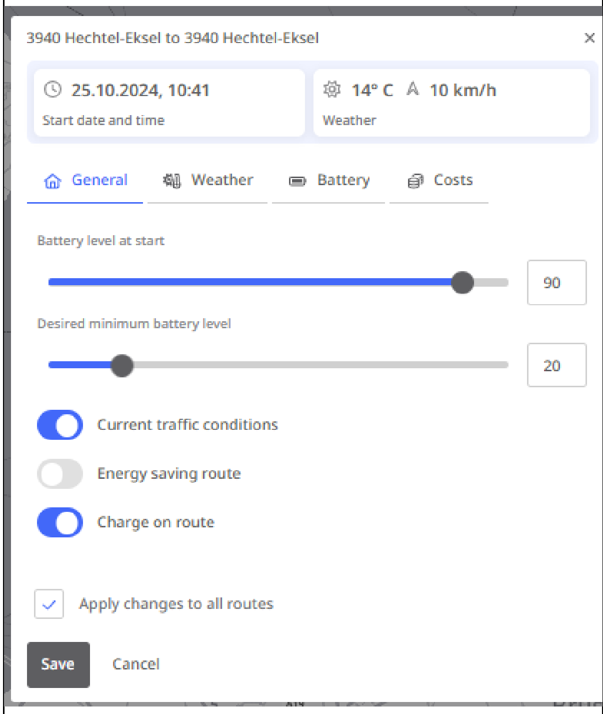
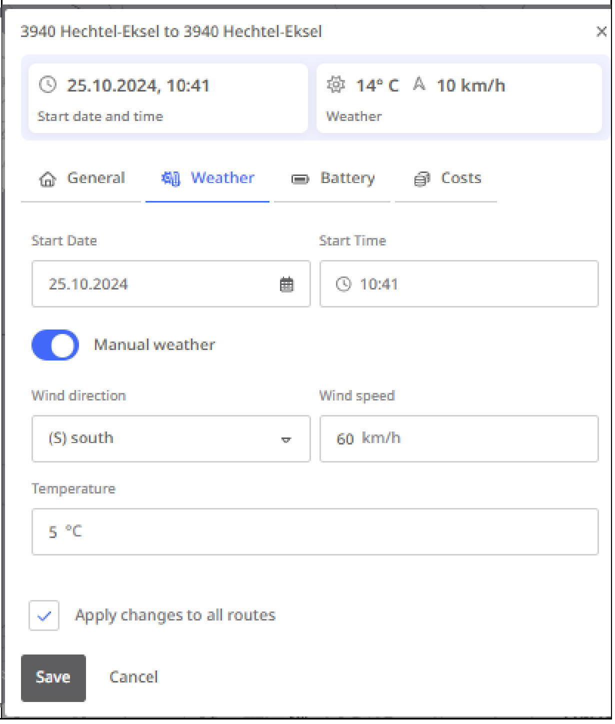

ZEFES 10 Volvo 2 - EMS2 FH-electric t...
Change vehicle
See more
Routes / Empty route
Start creating a route by entering a starting location or clicking the map.
1 Enter start location
OR
Drag and drop your route(s) here
or click to browse for a file
3940 Hechtel-Eksel
3940 Hechtel-Eksel
51.091503, 5.391898
Payload (max. 38.500 kg)
15000
Service time (hh:mm)
00:20
Electric power take out
0 kWh
Charging available
Max. charging speed
0 kW
Add stop
Cancel
Den Haag
Rotterdam
Antwerpen
Brussel
BELGIË
Luxembourg
Bremen

Setup a trip sequence and by adding additional stop points including advanced parameters, e.g. charging availability and maximum charging speed.



Calculate the trip for the specified vehicle profile. Receive information on feasibility, payload per segment, battery state-of-charge level, recharging instructions per stop and trip KPI`s.



Modification of general route settings	live weather or manually set weather
	

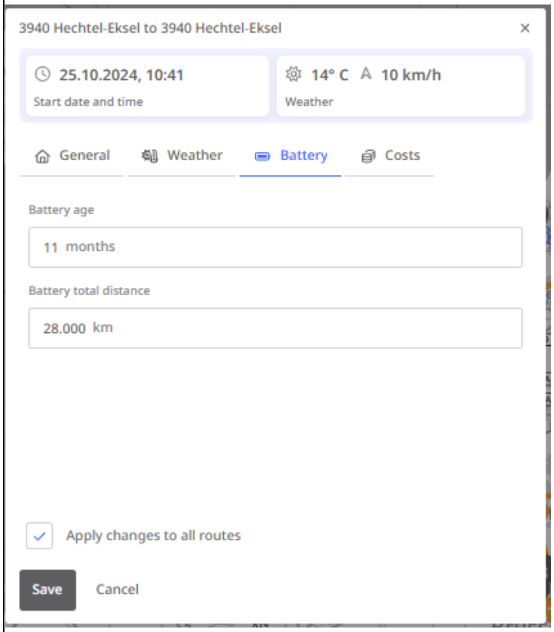
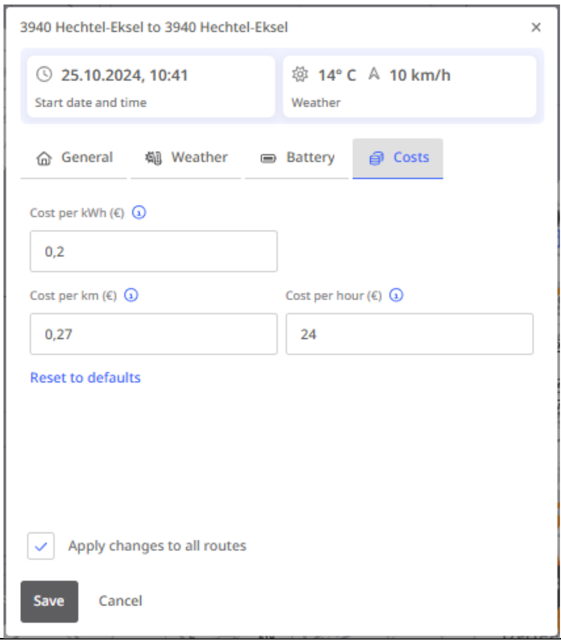
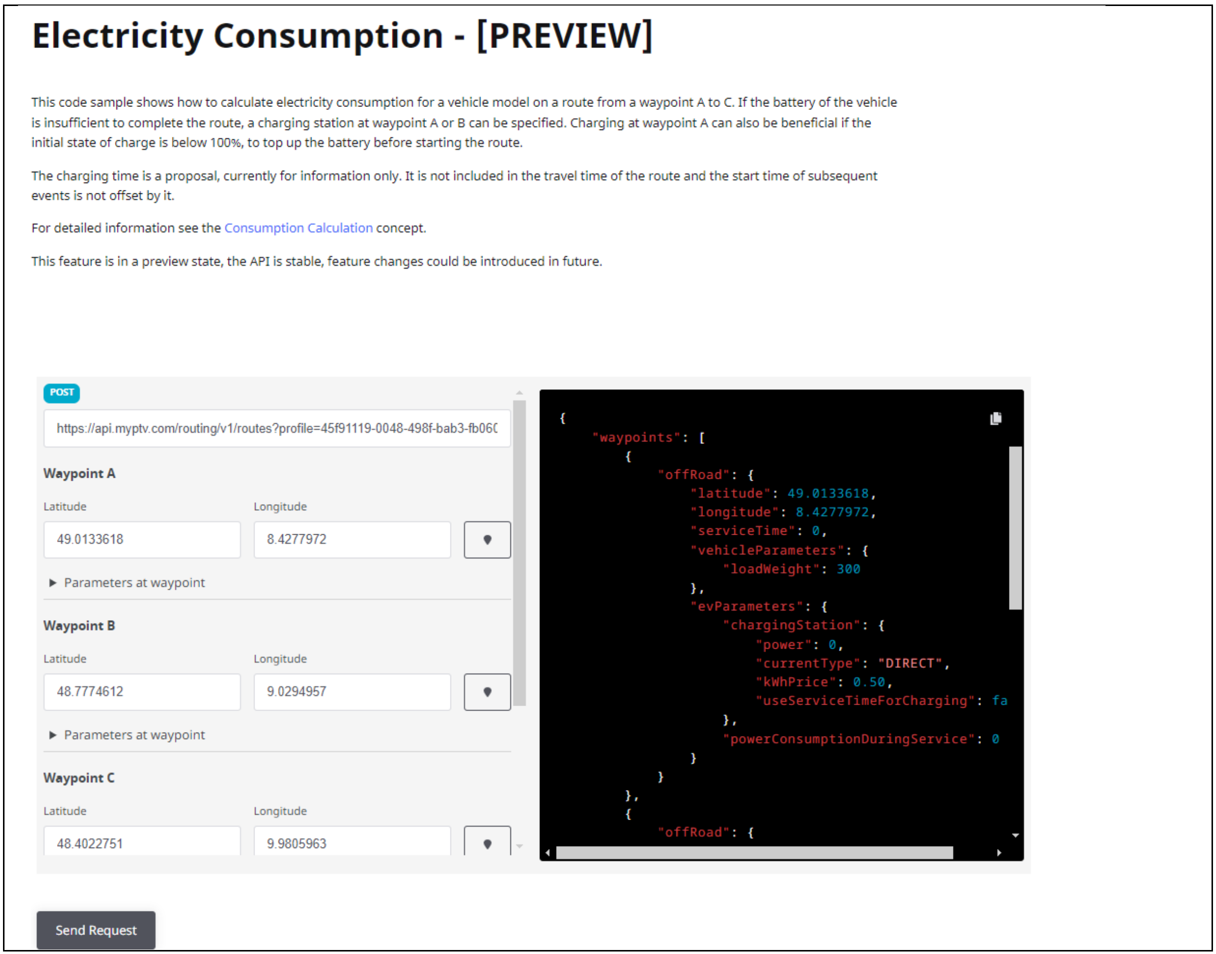
Setting battery age and total distance driven	Modification of calculation cost values
	

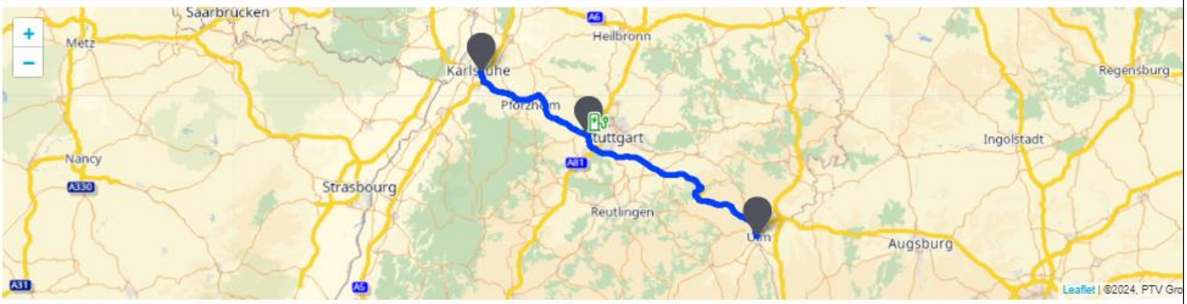
Figure 13 – ZEFES tool 2 implementation screenshots frontend.

3.2.5 API endpoint implementation



Response


```
{
  "distance": 157084,
  "travelTime": 7130,
  "trafficDelay": 50,
  "violated": false,
  "legs": [
    {
      "distance": 59937,
      "travelTime": 2642,
      "trafficDelay": 5,
      "violated": false,
      "evReport": {
        "electricityConsumption": 27.84,
        "batteryStateOfCharge": 44.09,
        "weatherAtStart": {
          "temperature": 14,
          "windSpeed": 8,
          "windDirection": 250
        },
```

Request Sample

C# | JavaScript | Python | Java | PowerShell

```
var client = new RestClient("https://api.myptv.com/routing/v1/routes?profile=45f91119-0048-498f-bab3-fb060341b9a
client.Timeout = -1;
var request = new RestRequest(Method.POST);
request.AddHeader("apiKey", "YOUR_API_KEY");
```

Figure 14 – ZEFES tool 2 implementation screenshots backend

3.3 Tool 3 - Right Vehicle in Right Duty

Route optimization is the process of calculating the most efficient journey for a vehicle, based on numerous different factors. These factors include everything from the vehicle's capacity to road conditions or regulations.
The vehicle routing problem is an optimization challenge with the goal of finding optimal routes for different vehicles visiting different locations. This is known as the "traveling salesman problem" (TSP), or in the transportation industry as the vehicle routing problem.

Whilst the vehicle routing problem is an interesting example, it doesn't even include the complexity of the real-world challenges. There are additional factors and restrictions that limit the number of possible combinations and add a lot of complexity to the delivery route planning process. These limitations might include:

- Fixed delivery dates and assigned time slots at ramps.
- Customer wishes or restrictions at delivery location.
- Vehicle- and driver-specific equipment and abilities (e.g. operation range of the vehicle, temperature-controlled, hazardous goods, truck-mounted forklift, specific driver's license of skill).
- Driving and rest times or working time regulations.
- Combination of pick-up & delivery.
- Several depots.

Existing, state-of-the-art route planning systems help to increase the efficiency, reliability and competitiveness of companies in the logistics industry. They make it possible to adapt to the diverse challenges of modern logistics and improve operational performance at the same time. However, theses existing solutions are, today, not capable of covering EV fleets or mixed fleets. Therefore, the ZEFES tool 3 "Right Vehicle in Right Duty" expands the scope of existing trip planning solutions. The approach foresees to (re-)use existing planning solution results and to provide via ZEFES tool 3 an addon solution to cover the EV aspect in an appropriate fashion. Existing trip structure can be analysed for different trip and vehicle setups. Ultimately the trips can be adopted or even redesigned towards EV executability.

3.3.1 Tool description

Tool 3 builds upon the framework of ZEFES tool 2, while it expands the functionality towards trip planning and analytics at trip level.
The adoption of electric vehicles is accelerating, driven by sustainability goals, regulatory changes, and cost incentives favouring clean technologies. Tool 3 supports this transition by offering a robust toolkit for managing electric fleets.
With detailed analyses of consumption and range, the API aids in the strategic deployment and efficient operation of electric vehicles.
The PTV Developer Routing API is not only a route planning tool—it is a strategic asset for companies managing future logistics fleets. Its advanced features, such as consumption calculation, range

estimation, and charge planning, make it indispensable for optimizing various electric vehicle operations including the mixed fleet aspect.
Ready to drive the green transformation of the logistics industry, the API supports looking to enhance their logistics operations with electric fleets. Routing API provides advanced features, making it a vital component in the sustainable evolution of logistics. By the integration of electric vehicles to an existing fleet in mixed fleet but also at full electric fleet level, the tool supports optimization of trip planning at all levels. Considering to current state of play, EVs often introduce an additional set of constraints to the planning process. The optimization covers these constraints and provides, based on the modelled planning task and vehicle settings, an optimized solution.

3.3.2 Use Case

The ZEFES project offers a robust solution for optimizing vehicle assignments within a fleet, ensuring the right vehicle is selected for each specific trip. The use case workflow is as follows:

1. Requirements Assessment:
    - Planners input trip requirements such as loading capacity, travel distance, operational costs, necessary equipment, and vehicle availability.
2. Bulk Planning:
    - Users can import multiple trips at once, allowing for efficient planning and optimization across various scenarios.
3. Mixed Fleet Optimisation:
    - The system supports planning with a diverse fleet, enabling comparisons between diesel and electric vehicles, which is crucial for transitioning to greener options.
4. Scenario Analysis:
    - Planners can evaluate different scenarios to identify the most efficient vehicle combinations, considering factors like cost-effectiveness and environmental impact.
5. User-Friendly Interface:
    - An intuitive interface makes it easy for logistics planners to enter data manually or through file imports, streamlining the planning process.
6. Technical feasibility analysis of the vehicle, especially for EMS vehicles:
    - To adopt EMS combinations in the existing infrastructure, the manoeuvrability of the vehicles in sharp turn like roundabouts needs to be assessed. The tool performs a feasibility check for the given route, to provide planners with the information needed to choose the right vehicle combination.

The central benefits of this use case are:

- Cost Efficiency: By selecting the most suitable vehicle, operational costs can be minimized.
- Flexibility: The ability to plan for multiple scenarios aids in adapting to changing logistics needs.
- Sustainability: Supports the transition to electric vehicles, helping organizations meet sustainability goals.

- Overall, the ZEFES project empowers logistics planners to make informed decisions, enhancing operational efficiency and sustainability in fleet management.

3.3.3 Functions, features & interplay

Logistics planners create trip plans for given transport orders (including constraints). They aim to assess shifting options towards electric vehicles and to identify the most beneficial vehicles for each individual mission.

Therefore, planned trips can be

- planned in the tool or imported to ZEFES tool 3,
- planned trips can be analysed towards electrification feasibility and cost KPIs,
- different combustion truck profiles and electric vehicle profiles can be compared and analysed to support optimal decision making towards the “right vehicle for right duty”.

The key functions tailored for electric vehicles (EVs) to enhance route planning and logistics efficiency of ZEFES tool 3 are:

1. **Charging Station Locator**:
    - Provides information on the location, availability, and type of charging stations along the route. Booking a parking and charging slot is not included in the current version, but a planned feature for a future version.
2. **Range Calculation**:
    - Calculates the optimal route based on the vehicle’s battery range, factoring in terrain and traffic conditions.
3. **Route Optimization**:
    - Optimizes routes to minimize charging stops while ensuring that travel time and distance are efficiently managed.
4. **Charging Time Estimation**:
    - Estimates the time required for charging based on battery level and charging station specifications, helping to plan stops effectively.
5. **Battery Consumption Modelling**:
    - Models’ energy consumption based on various factors, such as vehicle type, load, and driving behaviour, allowing for accurate trip planning.
6. **Multi-Stop Planning**:
    - Supports planning for multiple stops, incorporating charging needs at each point to ensure that the vehicle remains within its range.
7. **Dynamic Updates**:
    - Offers real-time updates on traffic conditions and charging station availability, ensuring the route remains efficient and practical.
8. **Vehicle feasibility**
    - Offers a post calculation method with focus to EMS vehicles to analyse the feasibility of a vehicle to execute the planned trip safely.

3.3.4 Implementation

To ensure realistic route calculations that account for various factors like road restrictions and traffic conditions, the tool utilizes vehicle profiles. Each profile includes relevant properties for route calculation, such as motorization, dimensions, toll subscriptions, and the environmental zones for which the vehicle is authorized. Additionally, a vehicle profile can be associated with a cost profile, allowing for a more precise cost estimation for the route based on the combined settings of both profiles.

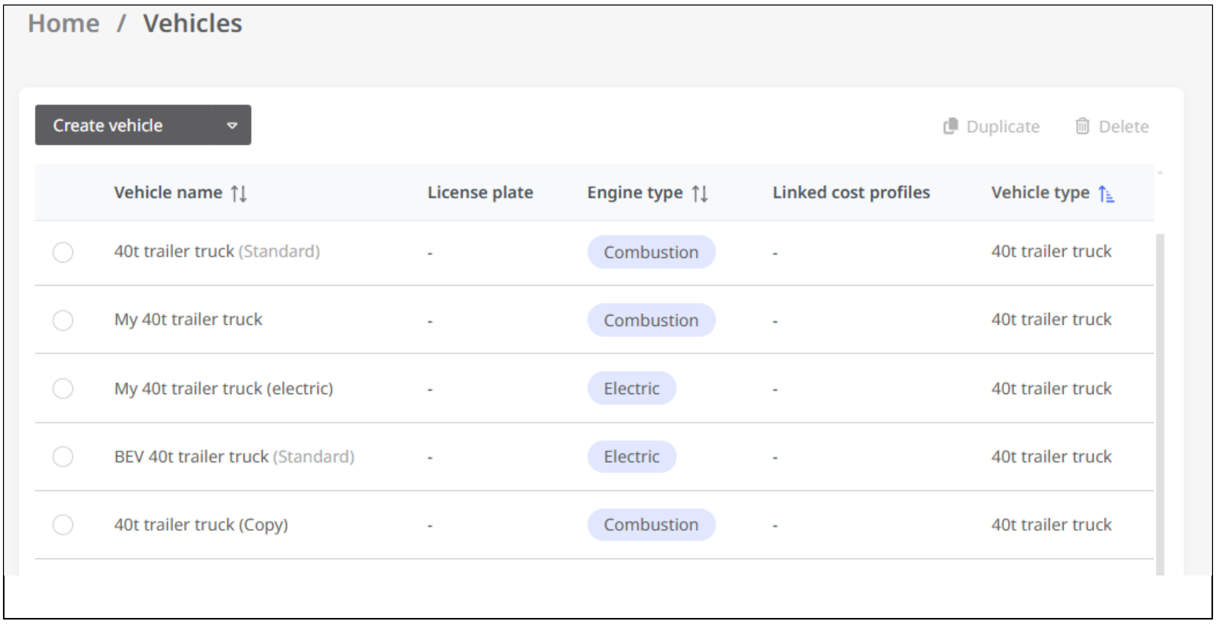


Figure 15 – Selection and management of vehicle profiles

To achieve the most accurate calculation of costs for a planned route, the tool employs cost profiles. Each profile includes settings that determine how route costs are calculated. For instance, costs per kilometre can be based on the vehicle's fuel consumption or set as a fixed amount. Additionally, costs can be derived from time factors, or users can input fixed route costs. Each cost profile can be connected to one or more vehicle profiles.
Default profiles can be used, or individual profiles can be created.

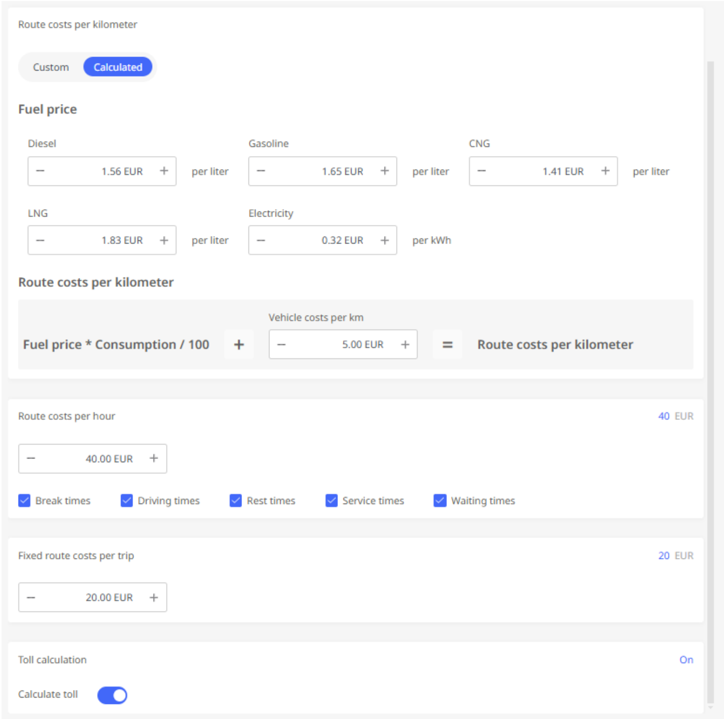


Figure 16 – Cost calculation parameters

Multiple features allow to setup the calculation attributes.

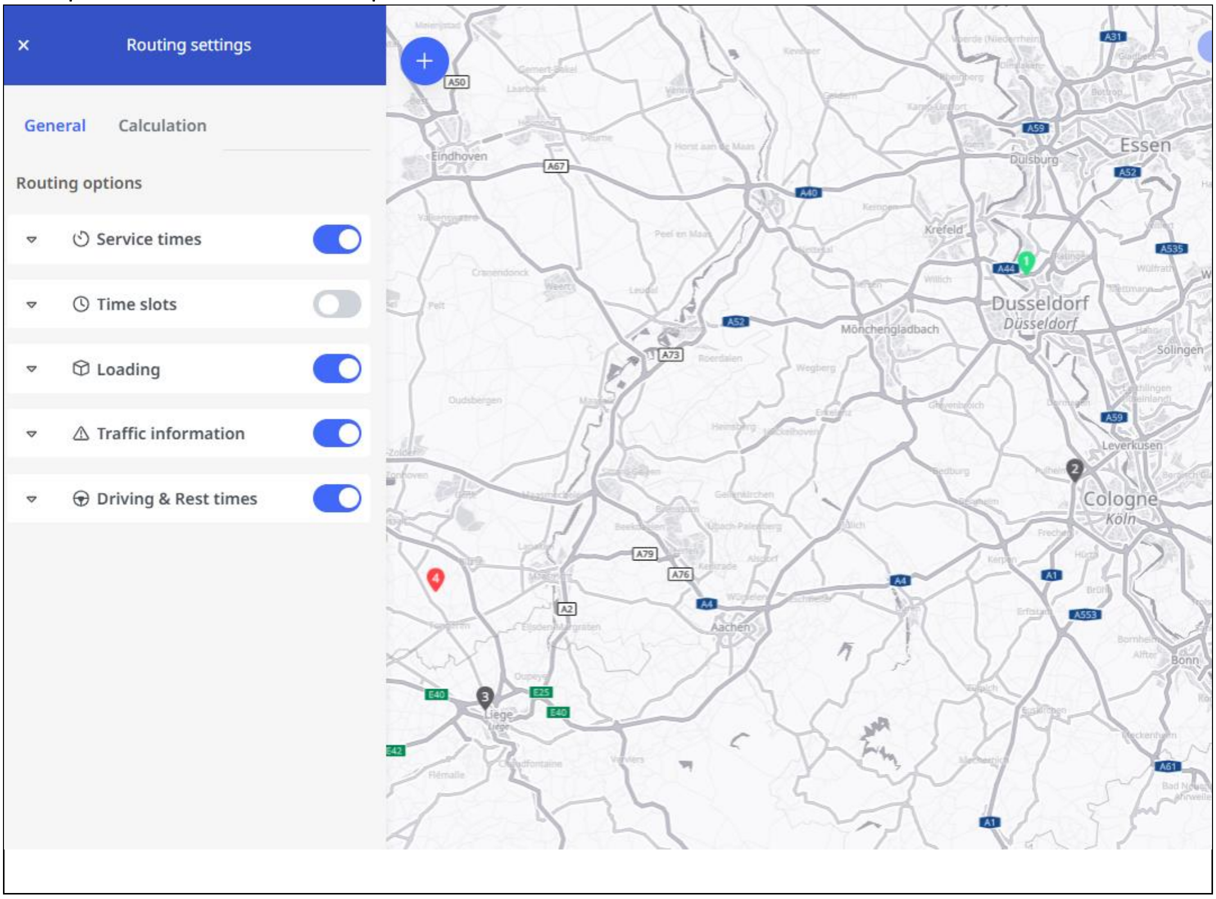


Figure 17 – Configuration of planning parameters

The tool provides a state-of-the art graphical user interface, enabling to include different information layers, e.g. low emission zones, truck restrictions, live traffic, planned construction sites.

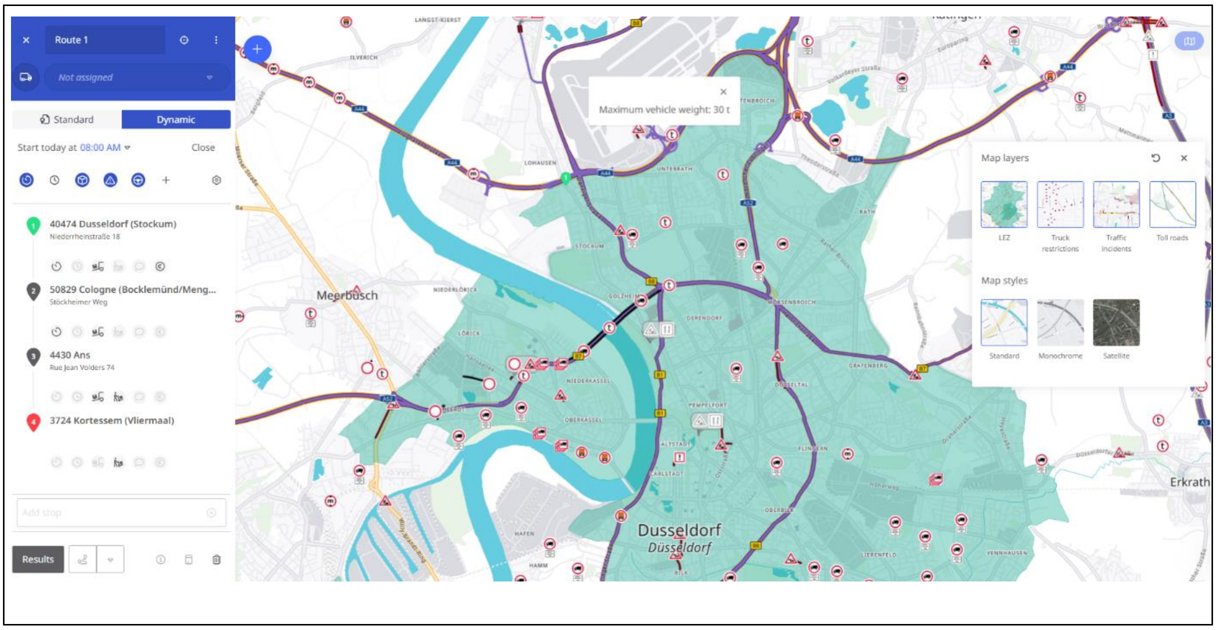


Figure 18 – ZEFES tool 3 - different information layers of the GUI

EV truck trips are presented in a coloured polyline, visualizing the battery state of charge at each part of the trip.

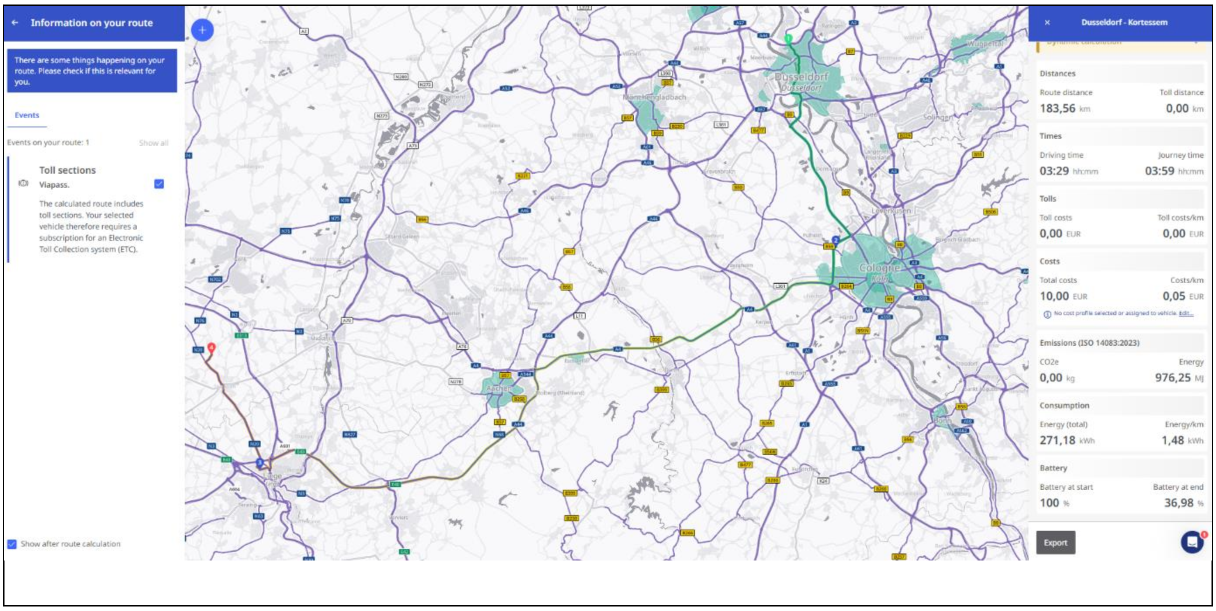


Figure 19 – ZEFES tool 3 GUI

For a planned trip, results like stops, time & distances, emissions and consumptions can be exported. Finally, a GLEC report can be created for the planned trip.

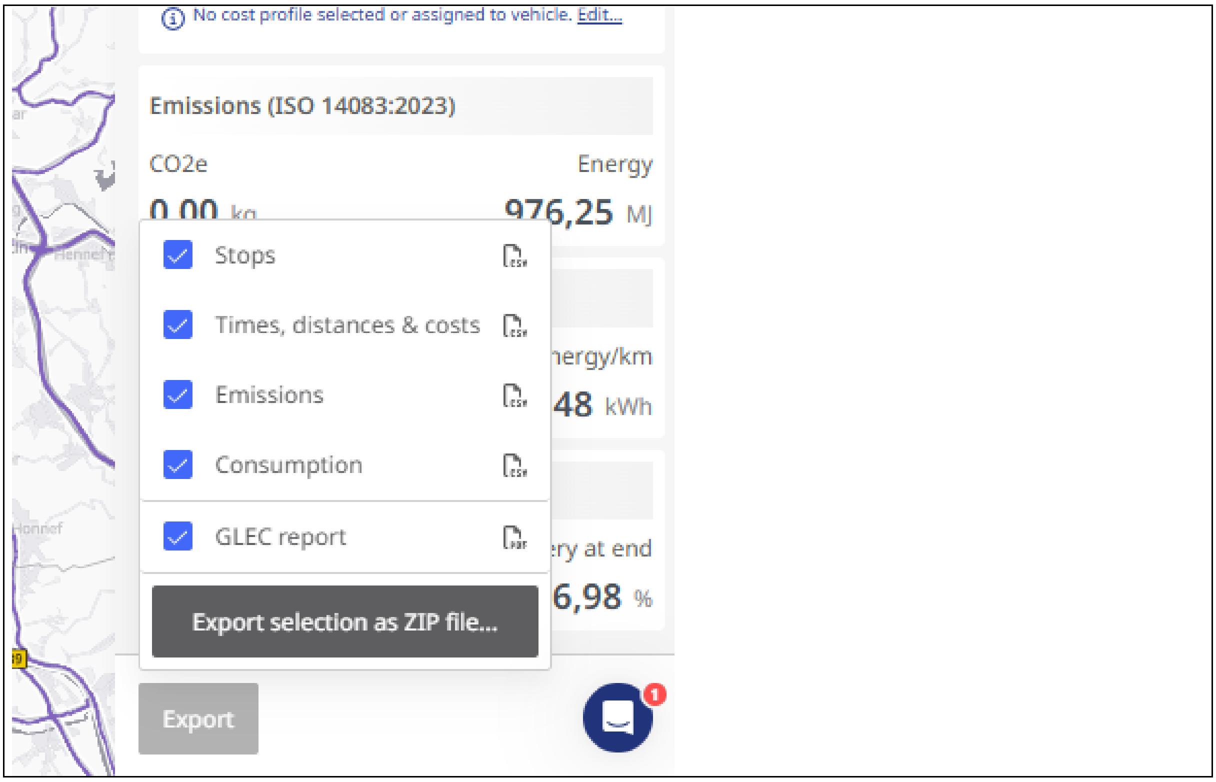


Figure 20 – Mission planning export.

The support the decision making for identification of the right vehicle for the right duty, the tool features the functionality to compare routes, using different setting, especially different vehicle profiles.

Comparing routes

With the right vehicle for right duty tool the user can create two routes or two different variants of a route and compare them with each other. This can help the user to optimize the order of the stops on a route and also to optimize the selection and usage of different vehicles. When comparing these two routes, all results (distance, travel time, costs, etc.) are compared.

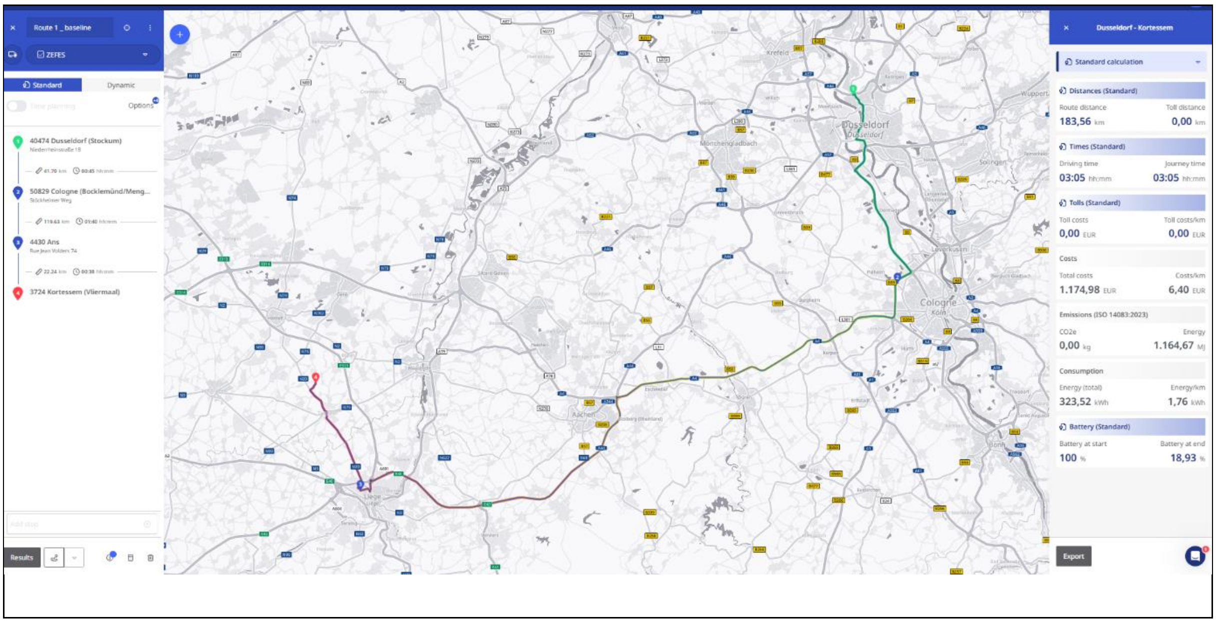


Figure 21 – Mission planning for reference truck.

For better comparison, the two routes are shown in two different colours in the list view, the map view and the details comparison view.

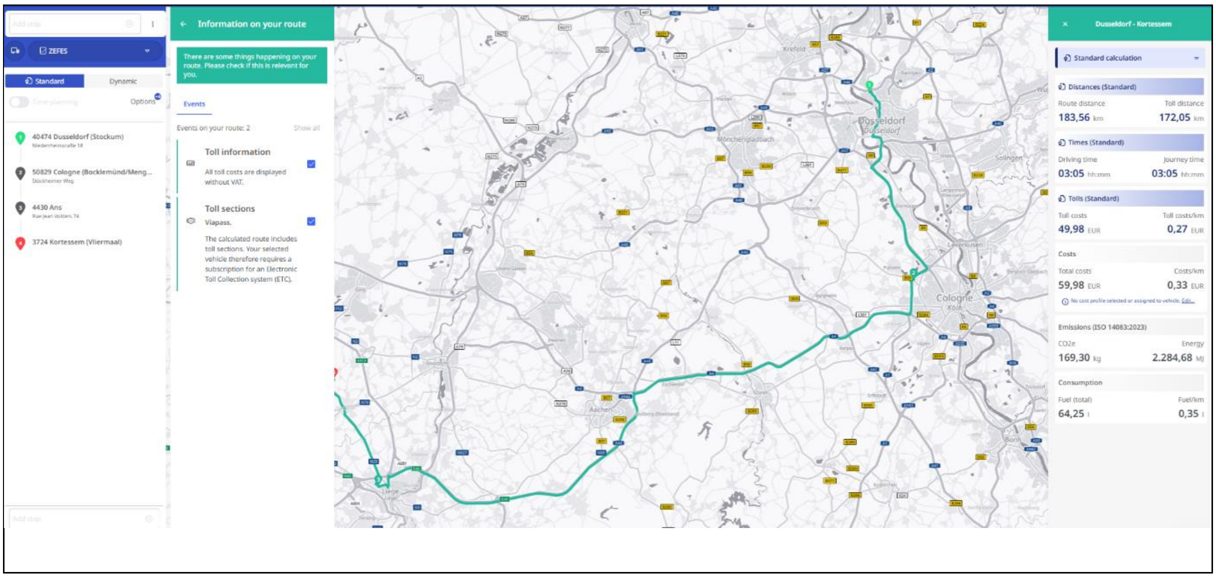


Figure 22 – Mission planning for EV truck and Diesel truck.

The following picture shows the comparison between a Diesel and an EV truck for the same route:

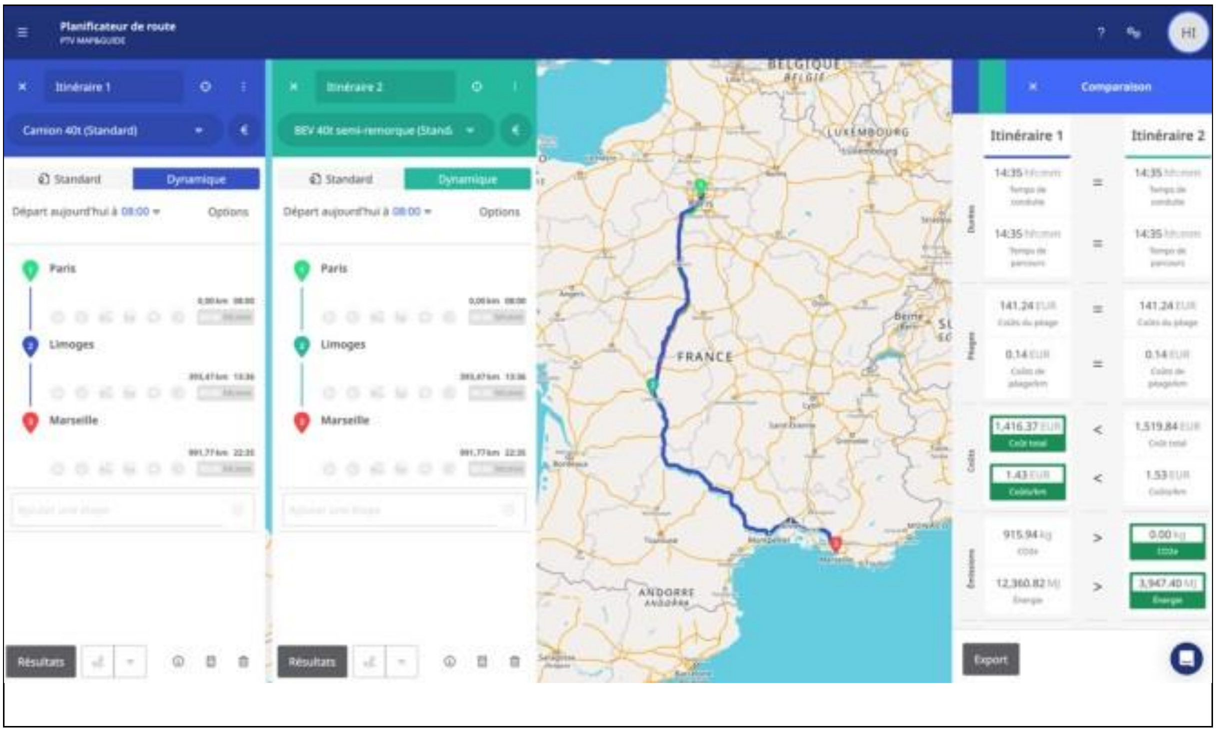


Figure 23 – Comparison of different vehicles for the mission.

Finally, a post calculation to assess the spatial compliance of the infrastructure is done. This is enabled by a sub-tool made by HAN University of Applies Sciences and a workflow of the ZEFES platform by TNO that enables the user to perform a post calculation for the planned trip. The sub-tool simulates the chosen vehicle combination, including EMS vehicles piloted in ZEFES, at critical sections in the route (e.g. roundabouts, sharp turns and highway ramps) to assess its compliance with the available space. Even though legal limitations still exist for the deployment of vehicle combinations exceeding the length and weight limits prescribed by Directive 96/53/EC, such as EMS vehicles, broader adoption of these combinations in Europe is anticipated soon. This is mainly due to their non-negligible positive contribution to reducing the carbon footprint from road freight.

The result of this tool is thus a status flag of the trip – feasible, conditional feasible, not feasible. The flags are a result of a comprehensive post calculation simulation. The polyline (GPS waypoints) of the planned route is handed over to the workflow handler and forwarded to the HAN post calculation service. The polyline is injected to a simulation which deep analyses all road elements of the envisaged driving route. The tool contains vehicle models of all ZEFES vehicles.

For critical sections, e.g. roundabout, computer vision techniques extract the road width available from satellite images to detect construction attributes, like the inner circle of a roundabout. The roundabout boundaries are then modelled, and the vehicle is simulated on it.

An example of the identified roundabout boundary and simulation reference path is shown in Figure 24 - Tire trajectories of an EMS2 combination. The reference path is made considering the dimensions of the vehicle and driving constraints like maximum steering angle, possible rate of steering application, etc.

Figure 24 shows the result of a simulation where an EMS2 vehicle drives a first exit roundabout, where the second trailer's tires go over the boundary, which will lead to a red (not feasible) flag.

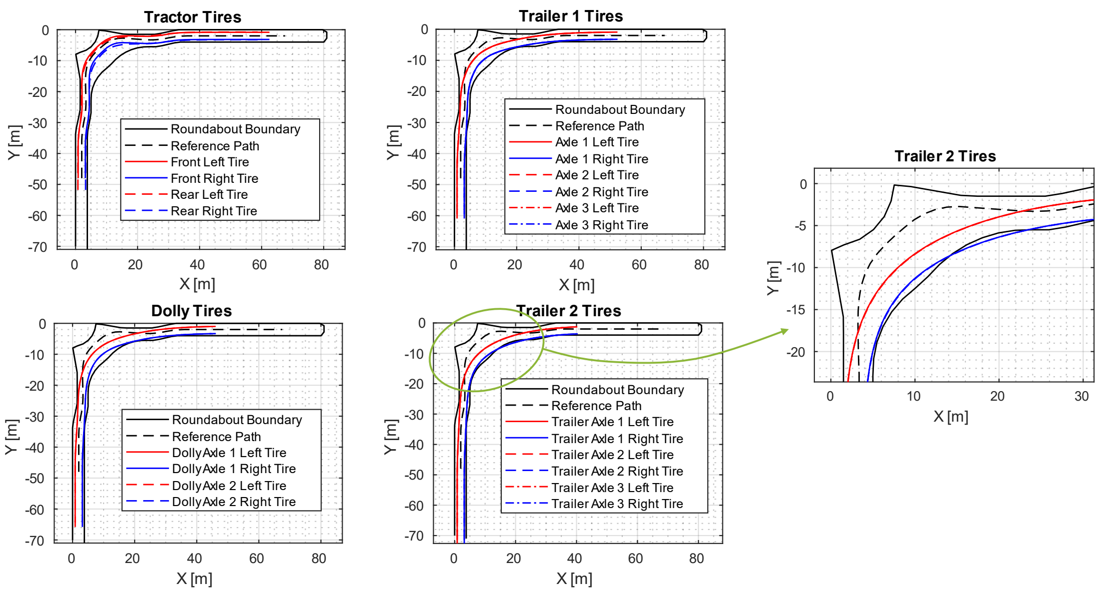


Figure 24 - Tire trajectories of an EMS2 combination

The vehicle simulation post calculation tool has the following features:

- Automated checking of all critical sections in the route, no manual intervention required.
- The ability to save road width extraction results so that computer vision algorithms do not need to be executed for the same roundabout.
- The same workflow can be considered roundabouts from any country, due to the modelling of country specific infrastructure specifications.
- Flags are provided for sharp turns and roundabouts as Red, Amber or Green to indicate feasibility. Flags are also provided for highway ramps where a recommended speed is calculated based on the rollover threshold of the vehicle combination.

The Figure 25 shows the display of the flags to the user, where the red point on the map depicts a roundabout that is not feasible for an EMS1 combination, which is chosen in the route planner.

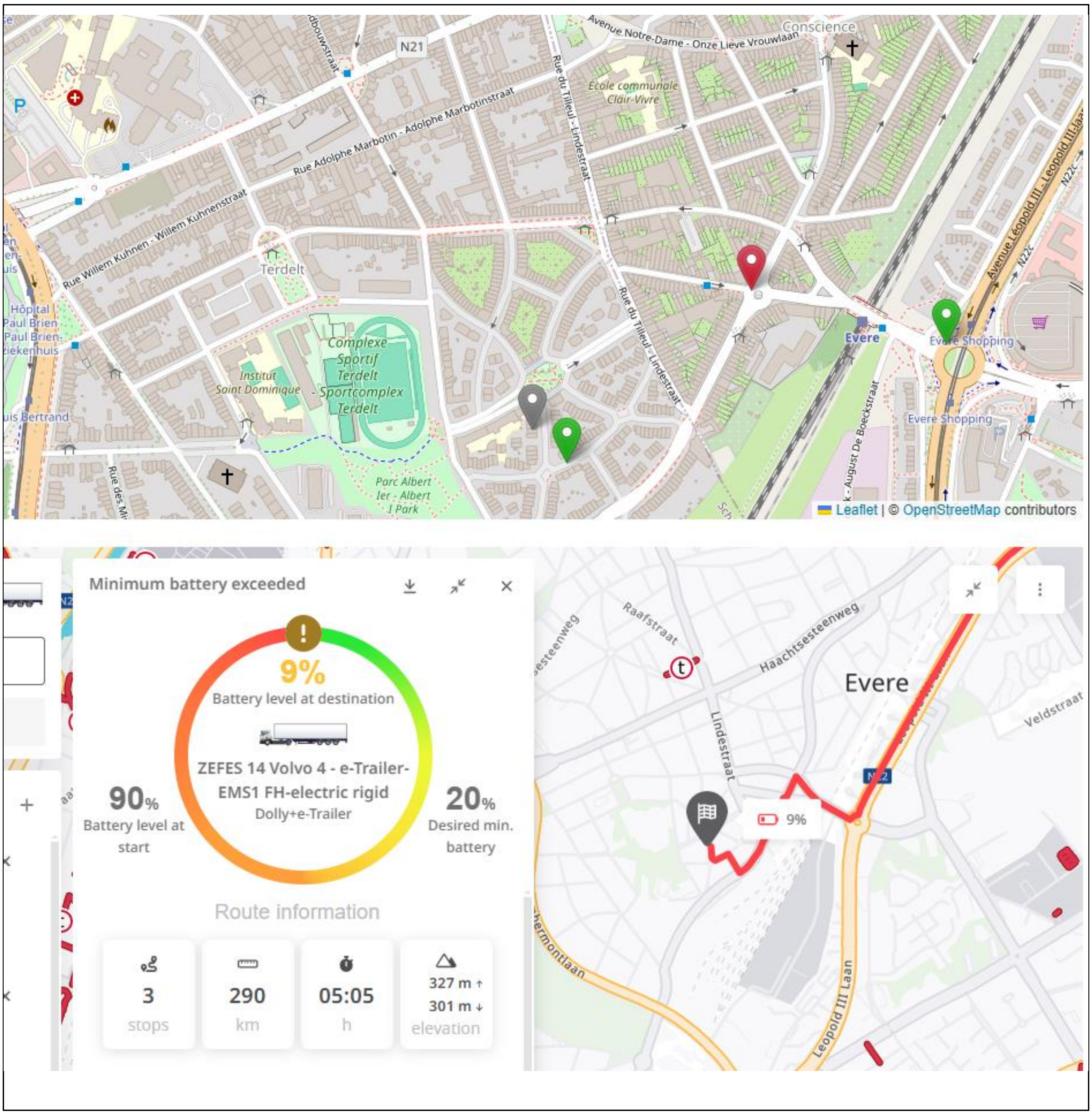


Figure 25 – Post calculation of technical trip feasibility for certain vehicle profile.

3.4 Tool 4 - Dynamic Correlation

3.4.1 Tool description

To implement Digital Twin (DT) technology for vehicles, it is essential to continuously assess the similarity between vehicle telematics data and the physical or data-driven DT models. This ongoing comparison ensures confidence in the accuracy and reliability of the DT model over time. The ***"Dynamic Correlation tool (DCT)"*** offers DT model developers an all-in-one solution to estimate and analyse the similarities between two key data curves: vehicle telematics and digital twin data.

The aim is that this tool will enable faster accuracy checks of the digital twin model by model developer without effort for data processing. Hence, this tool will not have access to any internal variables or parameters of the digital twin model, rather the model developers will access this tool via an API and receive the accuracy performance of their DT models.

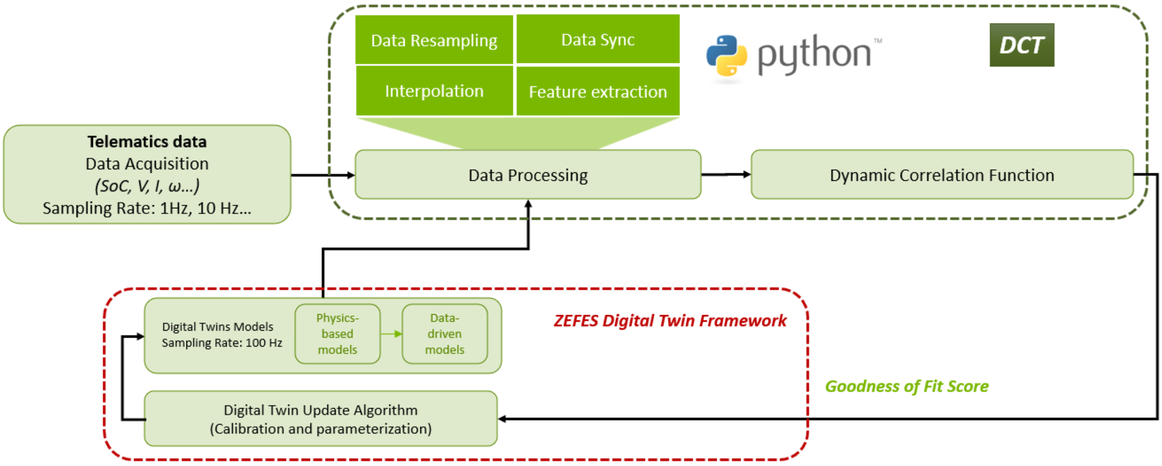


Figure 26 – Dynamic Correlation Tool Overview using Python 3.9.2.

Vehicle telematics data is real-world data that inherently includes uncertainties. The primary uncertainties are as follows:

- Presence of noise and outlier data within the vehicle telematics information.
- Irregular time-domain spacing of data, as recorded by vehicle data acquisition systems.

Additionally, data generated in the cloud by DT models will likely differ from vehicle data due to:

- The sampling frequency at which data is captured. DT models may produce data at a higher frequency, reflecting the fidelity of the model used.
- Differences in units between the two data sources, depending on sensor specifications and modelling approaches.
- A lack of synchronization in time, preventing direct comparison.

The Dynamic Correlation Tool (DCT) is designed to provide an effective solution for accurately assessing similarity between these two data sources.

3.4.2 Use Case

Error! Reference source not found.26 provides an overview of how datasets from vehicle telematics and ZEFES digital twin models (i.e. Physics-based and Data-driven models) are integrated within the DCT to support model developers. Both data sources—vehicle telematics and DT models—are inputted into the DCT, where they undergo a multi-step process of data processing and filtering. Following this, the data is sent to the correlation function, and the resulting output is delivered back to the DT model developer. The DT model developer can utilize the DCT in a continuous loop to optimize and calibrate the DT model's parameters.

3.4.3 Functions, features & interplay

3.4.3.1 Functions and features

The following functions and features are integrated into the DCT.

- **Data Resampling**: The data from both vehicle telematics and DT models are resampled to match the highest sampling frequency available between the two datasets. This ensures that both datasets are compatible and comparable, allowing for accurate analysis.
- **Interpolation**: Since data resampling may create gaps, interpolation is used to fill these gaps and maintain continuity in the data. Two interpolation methods are available:
    - **Linear Interpolation**
    - **Cubic Spline Interpolation**: Uses piecewise cubic polynomials to create a smooth curve between data points, which can result in a more precise estimation than linear interpolation.
- **Data Normalization**: This step removes any bias due to differences in units between the two data sources, enabling a direct comparison. Normalization scales the data to a common range, making it easier to evaluate the relationship between the datasets.
- **Correlation Metrics Generation**: The DCT tool offers multiple correlation metrics to quantify the relationship and similarity between the datasets. These metrics include:
    - **Pearson Correlation**: Measures the linear relationship between two variables, ranging from -1 (perfect negative correlation) to +1 (perfect positive correlation).
    - **Spearman Correlation**: A non-parametric metric that assesses the monotonic relationship between variables by ranking them, making it less sensitive to outliers.
    - **Kendall's Tau**: Evaluates the strength of association between two variables by considering the concordance and discordance of data pairs, useful for ranked data.
- **Performance Metrics Generation**: The DCT tool offers multiple performance metrics to quantify the accuracy between the datasets. These metrics include:
    - **Mean Absolute Error (MAE)**: Calculates the average absolute difference between the observed and predicted values, giving a straightforward measure of error.
    - **Mean Square Error (MSE)**: Finds the average squared difference between observed and predicted values, highlighting larger errors due to squaring.
    - **Root Mean Square Error (RMSE)**: The square root of MSE, providing an error measure on the same scale as the data, which is more interpretable than MSE alone.
    - **R-Square (R^2)**: Represents the proportion of variance in the observed data explained by the model, indicating the model's goodness of fit.
- **Visualization and Output**: The output of the analysis is visualised in the webpage; however, the API only returns analysis results in JSON format.

The challenge of synchronizing both datasets in time is currently under investigation. Various algorithms and feature engineering techniques are being explored for potential implementation.

3.4.3.2 Interplay

The following interplays are crucial for the tool's functionality:

- Vehicle data and DT model data must be provided in CSV files, with an exact column-to-column alignment of variable names.
- Interaction with the API must follow a specific protocol. The key for selecting features or functions for output analysis must be explicitly provided; otherwise, the tool will return an error.

3.4.4 Implementation

This section describes the implemented tool, available both as a webpage interface and an API for automated scripting. Two options are available to choose in both interface:

- The kind of interpolation that will be used in analysis.
- The correlation metrices that is needed to be given as output.

3.4.4.1 Web Interface

Error! Reference source not found. displays the input interface for the DCT, where users can select options and initiate analysis. After analysis, the results are presented along with a visual plot comparing the modified DT data and vehicle telematics data. Two CSV files containing data are uploaded to the interface. Figure 28 shows the calculated correlation metrics and the corresponding graphical plot for one variable.

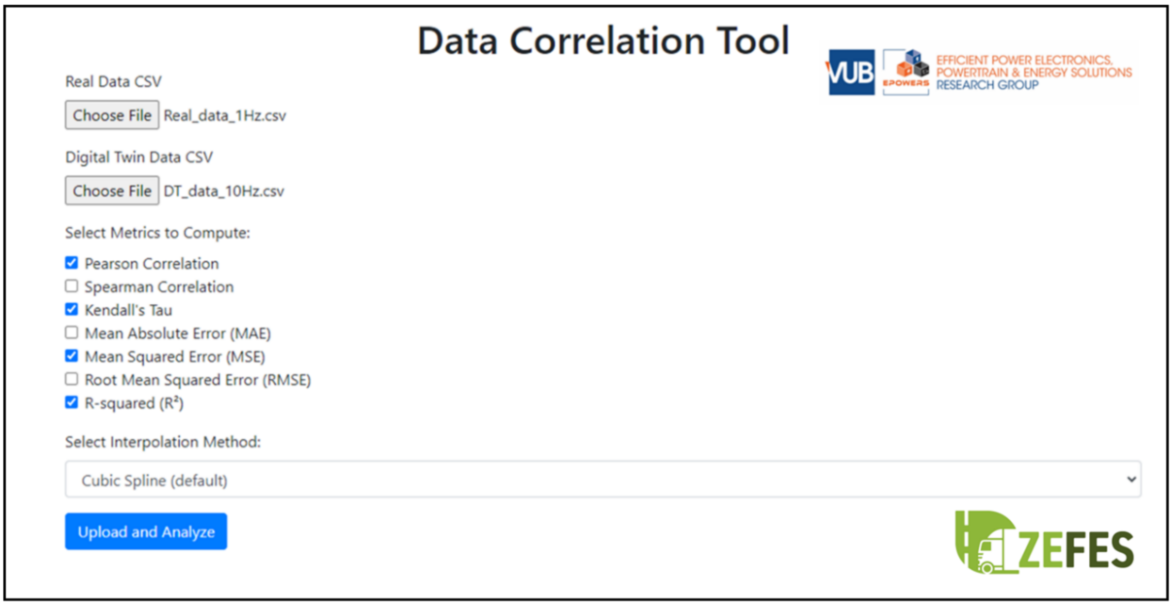


Figure 27 – Dynamic Correlation Tool Web interface

Analysis Results

Metrics Table

Data Column	Pearson's Correlation	Kendall's Tau	P-Value	MSE	R^2 (%)
x1	0.987	0.85	0.0	5.266	97.313
x2	0.986	0.847	0.0	4.157	97.261
x3	0.971	0.816	0.0	8.919	94.239
x4	0.977	0.803	0.0	5.578	95.412

Plots

x1

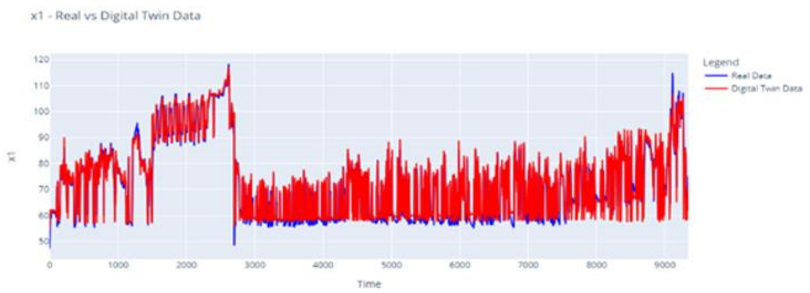


Figure 28 – Result of Correlation analysis.

3.4.4.2 Web API

The code for web API on a local host is depicted in Figure 29. Using the below script a POST request can be sent to the API URL to receive a JSON data providing the analysis result. It is always desirable to use web API any development work, as it can be used iteratively.

```python
# Define the URL for the API endpoint
api_url = 'http://127.0.0.1:5000/api/upload'

# Define the paths to your CSV files
real_data_path = 'Data_Analysis_Tool/Real_data_1Hz.csv'
digital_twin_data_path = 'Data_Analysis_Tool/DT_data_10Hz.csv'

# Define the metrics and interpolation method
metrics = ['pearson', 'spearman', 'kendall','mea','mse','rmse']  # Choose the metrics you want to compute
interpolation_method = 'cubic'  # Or 'linear'

# Open the CSV files
files = {
    'real_data': open(real_data_path, 'rb'),
    'digital_twin_data': open(digital_twin_data_path, 'rb')
}

# Define the data to be sent with the request
data = {
    'metrics': metrics,
    'interpolation': interpolation_method
}

# Send the POST request to the API
response = requests.post(api_url, files=files, data=data)

# Check the response status
if response.status_code == 200:
    # Print the JSON response from the server
    result = response.json()
    print("API Response:", result)
else:
    print(f"Request failed with status code {response.status_code}")
    print("Response:", response.text)

# Close the files
files['real_data'].close()
files['digital_twin_data'].close()
```

Figure 29 – Web API python script.

3.5 Tool 5 - Predictive Maintenance and AI applications

3.5.1 Tool description

The key vehicle technologies deployed within the context of ZEFES include battery systems, fuel cell systems, and novel tyre technologies aimed at electric traction systems. Available information on how these systems age during real world operation is still relatively limited, and one of the ZEFES project aims is to develop toolsets that will enable monitoring of the system's health, and the prediction of future degradation or wear.

The predictive maintenance tool will enable OEM's, fleet owners or fleet operators to monitor system ageing, detect anomalies and predict degradation or failures in advance. By leveraging robust machine learning algorithms, this tool analyses patterns from the vehicle's logged data to predict degradation and identify failures. As a result, fleet operators can make better-informed decisions regarding vehicle maintenance and usage profiles.

3.5.2 Use Case

The predictive maintenance tool relies on 2 main analytical processes:

1. Diagnostic analytics
    - Use past/present data to tell you the past state
    - Diagnostics refer to condition-based monitoring that monitors for failures that have started to happen
    - It monitors the vehicle raw logged data at almost real-time to detect anomalies or find patterns of failure severity
    - This helps the operator to take necessary actions to prevent more severe failures once they have been detected
2. Predictive analytics

- Use past/present data to tell you the future state
- This type of analytics relies on prior vehicle usage or usage projection ahead of time to predict future degradation or wear-rate leading up to point of failure or EOL (End-of-life)
- This type of analytics is much more complex and will involve a lot of preprocessing and continuous learning

Both Diagnostics and Prognostics functions may be supported using predictive models. Within the context of the ZEFES Predictive Maintenance Tool, three models have been provided by partners:

- Tyre wear black-box model by Michelin
- Fuel cell degradation black-box model by AVL (to be delivered)
- Battery degradation black-box model by FHG

The predictive maintenance tool is therefore articulated around three key data sources: data acquisition from the vehicle whilst in operation, simulation models, and machine learning algorithms. The full process is described in the following sections.

3.5.3 Functions, features & interplay

The diagram below outlines the system layout for the predictive maintenance tool. The platform consists of three main components:

1. Telemetry Data Processing:
   - In this initial stage, telemetry data is processed. The system collects and evaluates data from various sources.
   - This data is used to provide regular updates on system health (Diagnostics), and to forecast future usage patterns and future state of health (Prognostics)
2. Diagnostic Analysis:
   - The diagnostic analysis component monitors telematics data for anomalies.
   - Anomalies represent points that deviate from the expected behaviour of normal data points.
   - Two types of anomalies can be identified using this platform:
     - Local Anomalies: These are point anomalies—individual data points that fall outside the expected pattern, range, or norm of nearby points.
     - Global Anomalies: These points lie outside the overall distribution of data. Global anomalies can be either collective (affecting multiple data points) or contextual (based on the data's expected pattern).
   - On top of anomaly detection, diagnostic analysis derives system wear or state of health from the collected vehicle data. This is supported using ageing simulation models.
3. Prognostic Analysis
   - Predictive maintenance aims to forecast future failures or wear rates.
   - By detecting the initial failure pattern in the data, fleet operators have ample time to take preventive action.
   - Notably, the platform offers a prediction horizon that spans from one week to several months. This information empowers fleet operators to make informed decisions based on anticipated component wear.

The general principles of the predictive maintenance logic for all three vehicle systems are depicted in Figure 30. A description of the individual workflow for each system follows.

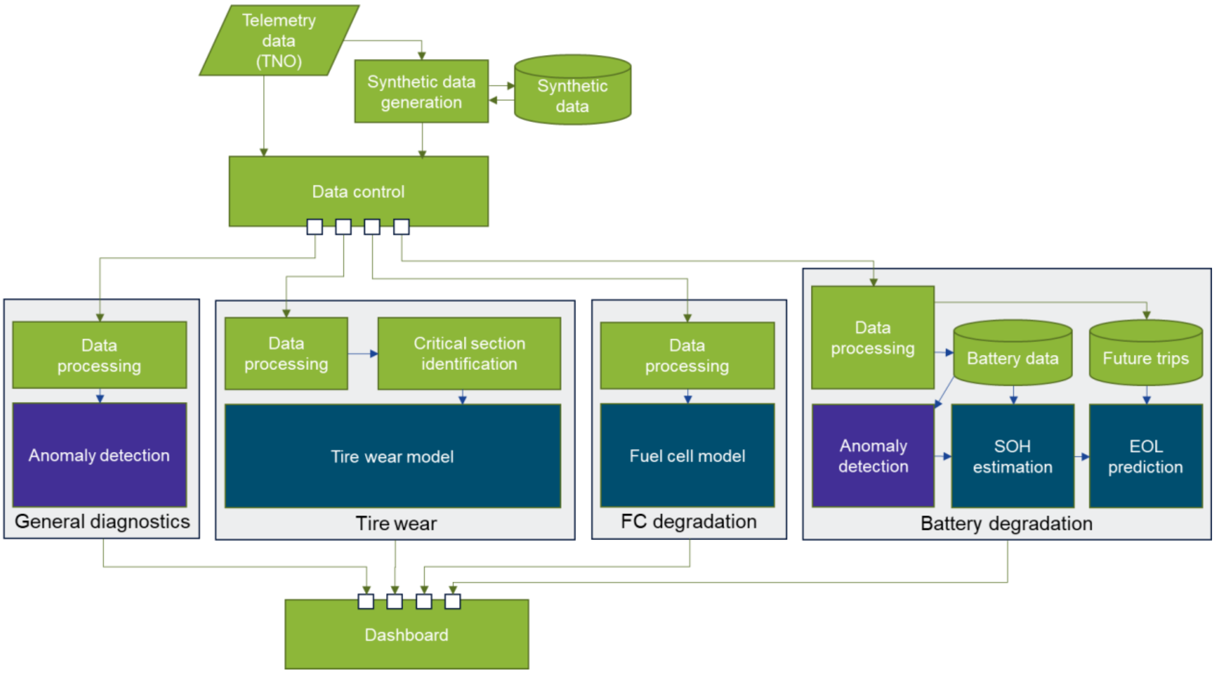


Figure 30 – Predictive maintenance tool overall process.

Predictive Maintenance of Battery:

Figure 31 illustrates the battery predictive maintenance workflow. The inputs listed below are based on Ricardo's experience, but these may be updated once more information becomes available.

The input to the model consists of current and forecasted telematics data, which feeds into the battery model. The model predicts State of Health (SOH) and anomalies for current and future truck usage (i.e., 1 month, 3 months, or 6 months ahead of time). Within the battery model itself are snapshots of historical driving patterns. These snapshots facilitate faster computation of End-of-Life (EOL) predictions for the batteries, thereby reducing the computational overhead.

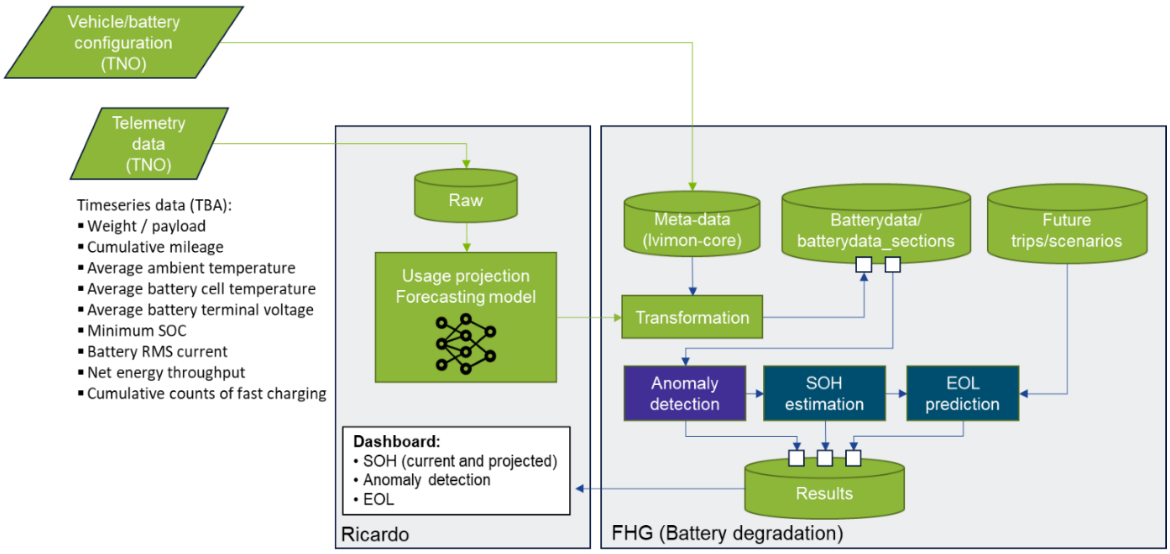


Figure 31 – Battery degradation monitoring workflow.

The workflow within the battery degradation model and its interfaces to the TNO platform is illustrated in the following figure. The overall setup is a local instance of the models and needed data at FHG premises which is communication with the Digital Twin Platform defined interfaces/ APIs.

The model consists of two parts: an SOH estimation module and an end of life (EOL) prediction module. The SOH estimation module needs periodically input data as timeseries of battery signals for Voltage (U), current (I), state of charge (SOC) and temperature (T). These data sets are transformed and enriched with meta data about the battery to a local battery data database as baseline for internal anomaly detection and state of health estimation. The results of this estimation will be stored in a local results database and can be transferred with periodical queries via API by the Predictive Maintenance Tool. The estimation is carried out on a weekly basis as ageing is a relatively slow process.

The EOL module uses the current SOH estimation as starting point for the EOL prediction, which is carried out by simulation with a provided set of reference driving cycles or scenarios. The idea is to forecast over a similar use-case how long it takes to achieve a residual capacity of 80 % and/ or a doubled internal resistance as end-of-life definition. The date, where these values will be achieved, are provided as EOL output.

Predictive Maintenance of Tyres:

This part of predictive maintenance predicts the tire wear using the Michelin tire model. The goal is to predict the tire tread depth. The prediction is not just for current data but also for projected vehicle usage based on current driving patterns.

There will be two models working alongside each other. The main model is the high-fidelity black box model provided by Michelin. The model requires the tire forces to be computed for all critical segments as shown in the figure below. The forces are computed beforehand and are stored in a database; this is because the computational overhead is high and makes it impossible to run real-time processing.

Another model on the platform is the machine learning tire model. This model learns from the high-fidelity model, and it executes much faster than the actual tire model. This allows instantaneous wear prediction.

Both model types complement each other, and ML model is updated every time there is a new set of training data available. The update frequency is estimated to be on a weekly basis.

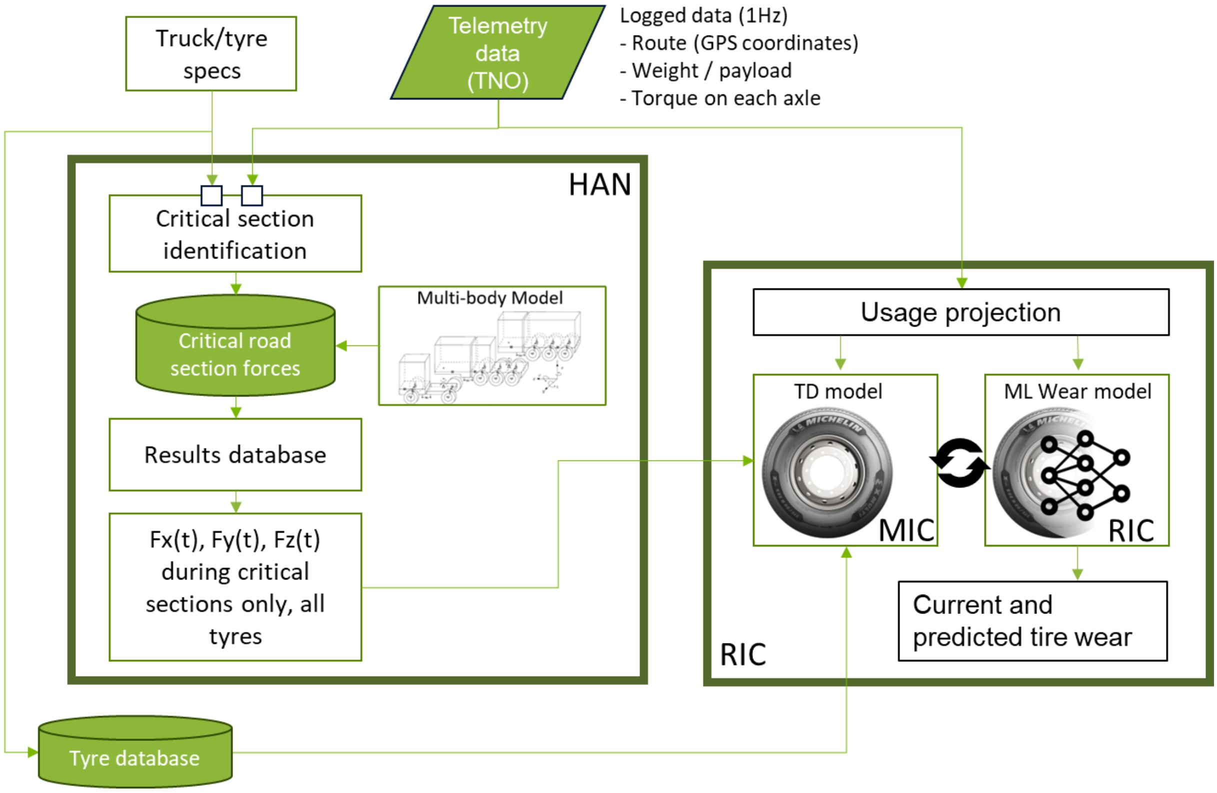


Figure 32 – Tyre degradation monitoring workflow.

Predictive Maintenance of Fuel Cell:

The diagnostics and predictive maintenance tool for fuel cell systems will be built around the fuel cell ageing model developed by POW and AVL in Work Package 2, Task 2.7. Availability of this model is forecasted for the latter part of the project, and therefore development of diagnostics and predictive maintenance strategies for those systems will be executed on reception of this model. It is expected the workflow will be like that deployed for the other systems and described in previous sections.

3.5.4 Implementation

The Tool 5 user interface (UI) is implemented using a React Web App. The main objective of the UI is to inform fleet operators of issues that may exist in their fleet of vehicles and provide fleet information. These issues may manifest as anomalies, fault codes, component degradation, and wear. Additionally, users can predict ahead of time and view predicted wear rates and degradation for components such as the battery, tires, and fuel cells.

The homepage of the Tool 5 web app is shown below:

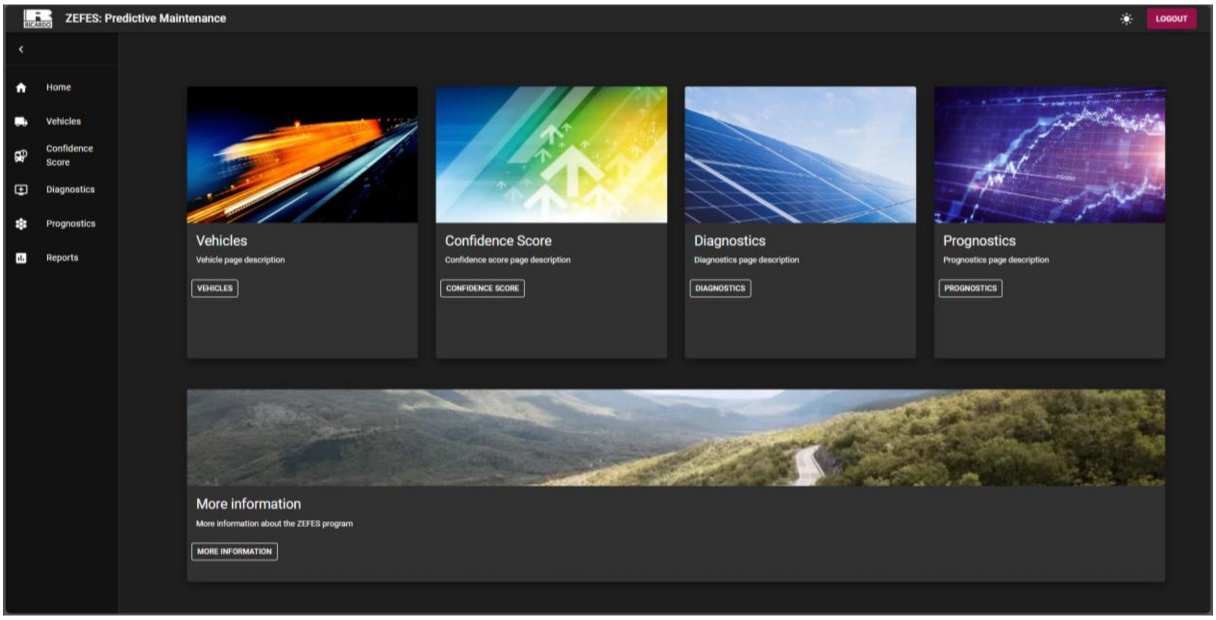


Figure 33 – Tool 5 homepage

The Tool 5 UI main components are as follows:

1. Vehicles page
2. Confidence score page
3. Diagnostics page
4. Prognostics page

Vehicles page

The vehicles page provides information on the historical telematics of a vehicle and its overall health status.

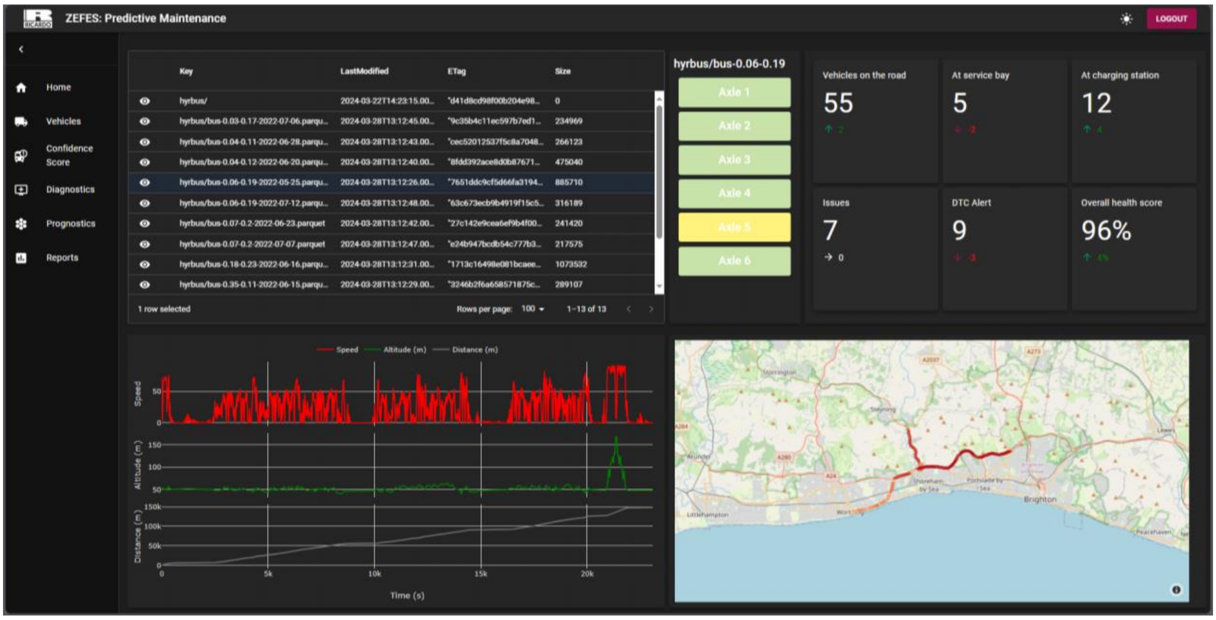


Figure 34 – Vehicles page

Confidence score page

The confidence score page is used to help determine if specific datapoints have been trained on for the anomaly detection model. Moving forwards, this page will not be part of the Tool 5 UI.

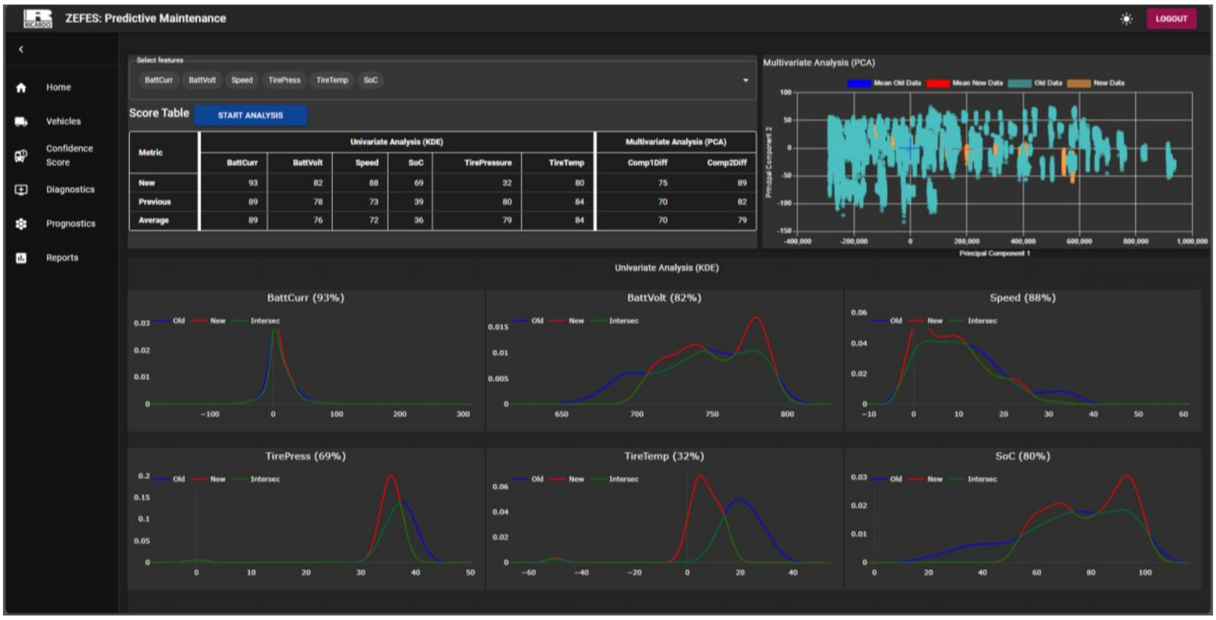


Figure 35 – Confidence score page

Diagnostics page

The diagnostics page lists the anomalies detected within the vehicle fleet and displays the locations on a map.

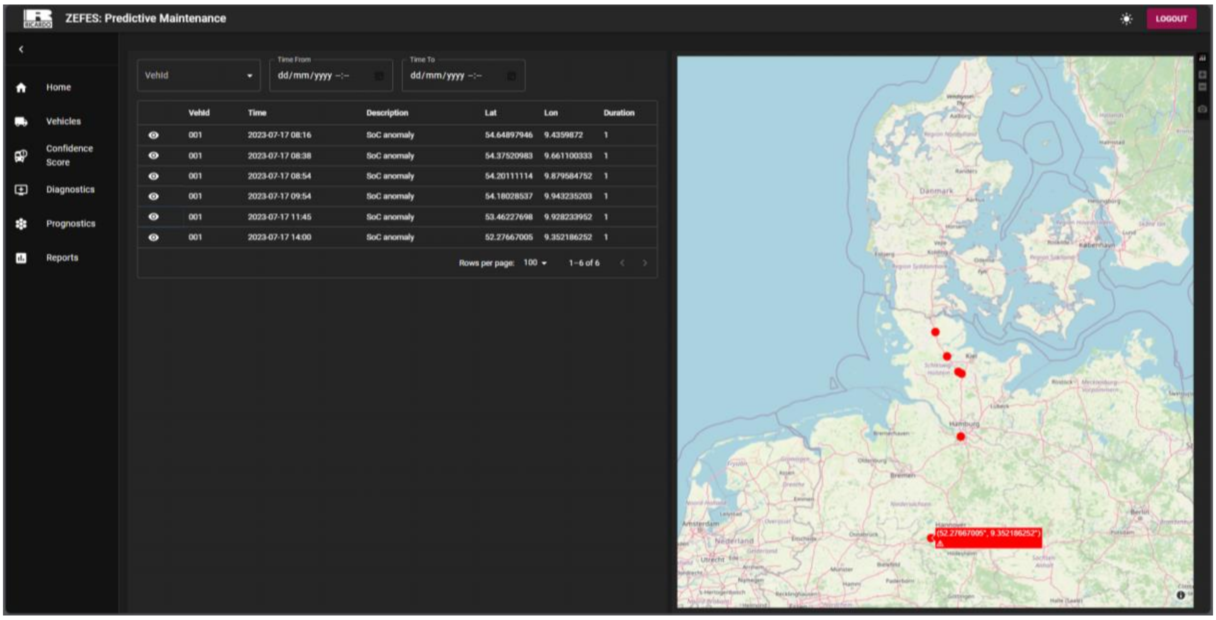


Figure 36 – Diagnostics page

Prognostics page

The prognostics page displays the predictions of critical system degradation such as batteries, fuel cells and tires.

Figure 37 – Prognostics page

4 Data integration and interplay with the platform

The gathered data and information should be provided to the Digital Twin Platform (Task 4.3) for keeping or requesting it in secure manner to fulfil the goals of the tools which should be elaborated within ZEFES project. This chapter is dedicated to the providing infrastructure data to the platform and the ways to request it in secure, authorized manner.

4.1 Infrastructure data injection to platform

There are different ways data could be injected to the platform. The specific way depends on the choice to provide the dataset to the platform or to allow to request data by demand.
Every data owner has its own authorized account on the Digital Twin platform. Only by using such an account ZEFES participant can access the platform for any action on tools, models and data.

Dataset uploading to platform

If the dataset is provided to keep at the platform, it could be uploaded to the platform by the data owner. Every authorized data owner receives the possibility to upload needed dataset to the platform through web form with existing button for uploading. For the upload, a special temporary secure session is open to guarantee that data is being uploaded in secure manner. The dataset under upload is being kept in the isolated data folder of the data owner. By default, the uploaded dataset is visible only to the data owner. After uploading, the data owner can choose the users account from the platform with which data owner wants to share data. Only those accounts chosen by data owner can access data afterwards.

Request for data through owner API

Data owner can provide the API to access the requested dataset in the case of need. In this case the dataset is being kept on the data owner premises. There should be an API, always available to listen to the upcoming requests for data. From the platform side a special Docker image will be constructed, which is responsible to request dataset, establish communication under that request, download required part of dataset and provide it directly to the processing without keeping it on platform side. When an instance of such Docker container starts, it takes the data request parameters from the user, establishes the communication with the premises of data owner, sends request for data, waits for the response with the dataset, and forms the received dataset for the immediate use in the processing.
In this case data owner still needs an account on the Digital Twin platform. The dataset should be registered in Metadata registry. For each dataset data owner provides the list of ZEFES accounts which are allowed to access this dataset. Before starting the communication, Docker instance checks whether the given account can access the requested dataset. If the account is not in the list, data request does not proceed, and user receives the message that the request cannot be fulfilled because they have no right to access the dataset. Users can communicate with data owners to request to include the account of the user to the accounts list for data access.

4.2 Exposing collaborative data via platform

Every ZEFES participant has their own user account. The rights to use specific dataset is associated with such user account.

Every dataset is exposed only to the user accounts chosen by data owner to access that specific dataset. The list of such accounts is being managed by data owner. Only authorized user accounts which are in the list of data owner can access data from that owner.
Every data owner should register their datasets in Metadata registry by providing the description of the dataset and the contact point information for managing that dataset. The users of ZEFES platform in case of need can communicate with the data owner to include him/her into the list of the accounts which are able to access that specific dataset. After being included in the list of the authorized user accounts by data owner, such dataset becomes available for that specific user account. Data could be used as an input to the ZEFES tools.

4.3 Valorising collaborative data via platform

In the ZEFES project, 5 tools are being made for the stakeholders of the project. Every tool in the end is represented as a workflow to run to reach the required results. Such workflow could be run only by an authorized account of a Digital Twin platform user. The access to the allowed workflows is provided by the rights associated with the user account. If the user account requires to start specific workflow, the right to access the datasets associated with this workflow and specifically to access the input dataset is checked before running the workflow. If the account does not have the right to access datasets, needed to run that specific workflow, the workflow will not be started, and user receives the message that they do not have rights to access that dataset. Users can check Metadata registry on who is the owner of the dataset and request the right to access the specific dataset from data owner. After the data owner puts the user account into the list of the accounts which are allowed to use that specific dataset, the user can run workflow with the necessary infrastructure data.

5 Results & Discussion

This Chapter summarizes the main results of the task 4.4 of M1-M9. The results will be taken up as basis for application and integration course of T4.5.

5.1 Results

Main objective of the task 4.5 is to provide and to implement the service functionalities for decision and planning support within the ZEFES platform.
Based upon the ZEFES platform architecture, different tools and workflows have been implemented accordingly.
In this regard 3 sub results can announced:

Result 1: development and provision of 5 project specific tools.

Result 2: implementation of the project specific tools to the platform infrastructure and interaction with the platform. Furthermore, set up of workflows to interact between different services and tools.

Result 3: deployment of demonstrator versions of the platform and the tools in a comprehensive setup.

5.2 Contribution to project (linked) Objectives.

Overall, this deliverable contributes to project objective 3 “provide digital and fleet management tools specifically for HD ZEVs, fleet integration with remote operational optimisation of vehicle performance “and objective 4 “demonstrate missions on cross-border, TEN-T corridors, fulfilling the requirements for range and payload, and comparing the deployability of BEVs and FCEVs for different mission profiles.”

Objective 3 foresees ZEFES to design an open platform to represent European logistics missions, enabling an assessment of the impact on environment and society of using HD ZEV. The platform will include different modules, such as “life-cycle assessment”, “assignment and routing”, “vehicle performance”, “vehicle condition”, “logistics performance” etc.
Part of this objective is to develop and validate truck Digital Twins (DTs) and fleet management tools. A DT here is a virtual representation of an object or system, possibly spanning its lifecycle, that is updated using real-world, possibly real-time data. Such a DT uses simulation, machine learning and reasoning to help decision-making[7].
The DT in ZEFES has different layers: component, vehicle, infrastructure and fleet operation. A platform will be created to make DTs of individual ZEV HDV operations with common data representations. The DTs of ZEFES will be applicable in the following domains: design and build, testing and validation, logistics operations, and charging infrastructure and management.
This deliverable as part of task 4.4 **contributes to the digital twin and digital platform services for infrastructure related data, model, and services.**

Objective 4 aims to demonstrate missions on cross-border, TEN-T corridors, fulfilling the requirements for range and payload, and comparing the deployability of BEVs and FCEVs for different mission profiles.

To perform the tests in real world environment, different digital services regarding the given infrastructure need to be facilitated in DTs of ZEFES.
This deliverable **contributes to the digital twin and digital platform services for infrastructure related data, model, and services for testing in course of objective 4 and related test scenarios.**

5.3 Contribution to major project exploitable result

The ZEFES project has 5 main objectives:

1. Execute real-world demonstrations, multimodal and cross border, of long-haul BEVs and FCEVs across Europe to take zero-emission long-haul goods transport in Europe to the next level.
2. Create a pathway for long-haul BEVs and FCEVs to become more affordable and reliable, more energy efficient, with a longer range per single charge and reduced charging times able to meet the user's needs.
3. Develop technologies which can deliver promised benefits (easy handling, similar driving hours & charging/hydrogen refuelling stations, high speeds, and ability to operate in complex transport supply chains).
4. Demonstrate an interoperable Megawatt Charging System (MCS) and the location deployment strategy for hydrogen refuelling stations (HRS). Make the mapping of flexible and abundant charging/refuelling points and novel charging concepts.
5. Create novel tools for fleet management to seamlessly integrate the rising number of long-haul BEVs and FCEVs vehicles in the logistics supply chains, in the form of a Digital Twin.

The deliverable provides a major contribution to the ZEFES offering and testing of software, process is and solutions for zero emission vehicles. Thereby, infrastructure data is key to enable core scenarios of ZEFES. With infrastructure data for services and applications, it supports objective 1, 2, 3, and 5.

6 Conclusion and Recommendation

6.1 Technical Achievements

Conclusion 1:

The development of the tools was preceded by consultations with Logistics Operators, to understand what information they need to allow them to assess the potential, and challenges, of transitioning towards and operating ZEV's in their fleet. Reviews of progress were organized during the tools development to gather feedback and suggestions. Some of these have been incorporated, and it is expected that the vehicle demonstration phase and the associated data logging will offer further opportunities for tools' updates.

Conclusion 2:

The development of the various tools has been conducted individually, and this approach has proven beneficial. Each partner responsible for a tool could concentrate on its development, while coordination on interfaces with other stakeholders in the work package was managed as needed. This focused approach has resulted in high-quality solutions for each individual tool.

Conclusion 3:

The local deployment of each tool provided opportunities for demonstration at various events. Additionally, it served as a fallback option in case of unexpected issues during development and integration, thereby reducing project-level risks for this specific work package.

6.2 Tasks Management and Partners' Collaboration

Conclusion 4:

Regular weekly updates are essential to keep everyone informed about the progress of the different tool developments. Each partner could monitor the development paths of the tools, and necessary interactions were coordinated as needed.

Conclusion 5:

A physical meeting was crucial to bridge the different phases and to kick off the migration to platform integration. This technical meeting was highly valued by all partners, facilitating vital discussions that resulted in prioritizing open points and determining concrete next steps. This kick-off enhanced integration efficiency and shortened the timeline.

Conclusion 6:

The physical event, in this case, the ZEFES symposium, created a sense of urgency for tool and platform providers to deliver high-quality results. The intermediate version of the platform, along with the associated workflows and integrated tools, established an excellent foundation for future developments and testing within the project.

Considering these six conclusions, we believe this approach is effective and recommend it for future projects. However, it's important to note that technical developments tied to strict deadlines can create pressure, necessitating careful planning and robust project management at all stakeholder levels. To mitigate this pressure, good communication among partners at an appropriate frequency, as highlighted within conclusion 4, is highly recommended.

7 Risks and interconnections

7.1 Risks/problems encountered

Risk No.	What is the risk	Probability of risk occurrence[1]	Effect of risk[1]	Solutions to overcome the risk
WP4.1	Delay in provisioning of data and services	2	2	• Conduct regular status meetings in order to track progress and strengthen the interaction with WP partners/ stakeholders. • Develop appropriate mitigation strategies at operational level in case these are needed.
WP4.2	Data quality of artefacts not as intended	3	2	• Contributing partners of data and services artefacts to check quality, to indicate quality and to perform quality improvements if needed
WP4.3	Legal issues preventing data processing and data valorisation	2	1	• Conduct regular status meetings in order to track progress and strengthen the interaction with WP partners/ stakeholders. • Develop appropriate mitigation strategies at operational level in case these are needed. • Communicate openly and as early as possible the use case and the condition of the data & services to raise awareness at contributor level.
WP4.4	Contributing partner leaving the consortium	2	1	• In case of a partner leaving the project, effected tasks have to be covered by replacement. Escalation to project board level.

[1)] Probability risk will occur: 1 = high, 2 = medium, 3 = Low

8 Linked deliverables

The objective of the task 4.4 which is linked to this deliverable it set the foundation for integrating and putting together a platform for major data objectives including external data sets (road, rail, filling stations, traffic, and environmental data etc.) that can interact with surrogate vehicle models of BEV and FCEV. Considering the requirements from WP1 as well as the specifications from WP2, 3, 5,6 and WP7 a common data representation will be provided including static and dynamic data.

The graphic below shows the project work package and the interdependencies of the workstreams.

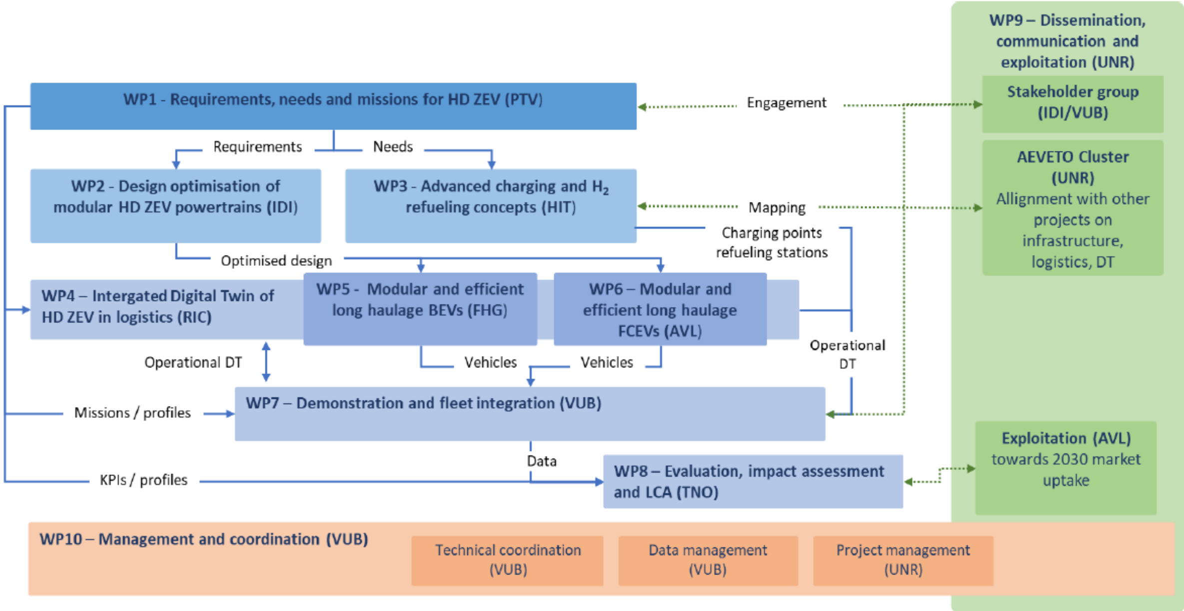


Figure 38 - ZEFES work package structure

At a technical detail level, this deliverable is strongly connected with WP1 deliverables

- D1.2 - **Defined Use Cases, Target metrics and needs** - KPIs per use case on energy savings and mission efficiency (T1.2)
- D1.3 - **ZEFES ecosystem specification** - use case KPI needs, stakeholder business needs, consolidation towards consistent system, TCO (T1.3, 1.4, 1.5)
- D1.4 - **Supply chain mapping** - mapping of ZEFES use cases at a supply chain level (T1.3)

Towards WP4, this deliverable is strongly connected to all WP4 deliverables:

- D4.1 - **Digital twin and platform specification & architecture** - selection of tools, scaling of models, model fidelity, run time and usage assessment (T4.1 and T4.2).
- D4.3 - **Interfaces standard and tool** – tool based on current commercial products and the interfaces handle confidential data from various partners. (T4.3)
- D4.4 - **Decision making platforms** - buying decision, route planning, vehicle assignment, dynamic correlation, predictive maintenance and AI applications (T4.5)
- D4.5 - **Process and outcomes of model validation** - LCA and EIA interfaces and outcomes from WP4. A validated digital twin platform with the functionality of decision-making platforms (T4.6, T4.7)

9 References

https://cordis.europa.eu/project/id/874997

10 Acknowledgement

The author(s) would like to thank the partners in the project for their valuable comments on previous drafts and for performing the review.

Project partners:

#	Partner short name	Partner Full Name
1	VUB	VRIJE UNIVERSITEIT BRUSSEL
2	FRD	FORD OTOMOTIV SANAYI ANONIM SIRKETI
4	KAE	KASSBOHRER FAHRZEUGWERKE GMBH
5	REN	RENAULT TRUCKS SAS
6	SCA	SCANIA CV AB
7	VET	VAN ECK TRAILERS BV
8	VOL	VOLVO TECHNOLOGY AB
8.1	CPA	CPAC SYSTEMS AB
9	ABB	ABB E-MOBILITY BV
9.1	ABP	ABB E-MOBILITY SPOLKA Z OGRANICZONAODPOWIEDZIALNOSCIA
9.2	ABG	ABB E-MOBILITY GMBH
10	AVL	AVL LIST GMBH
11	CM	SOCIEDAD ESPANOLA DE CARBUROS METALICOS SA
11.1	APG	AIR PRODUCTS GMBH
12	HEPL	HITACHI ENERGY POLAND SPOLKA Z OGRANICZONA ODPOWIEDZIALNOSCIA
13	MIC	MANUFACTURE FRANCAISE DES PNEUMATIQUES MICHELIN
14	POW	PLASTIC OMNIUM NEW ENERGIES WELS GMBH
15	RIC-CZ	RICARDO PRAGUE S.R.O.
15.1	RIC-DE	RICARDO GMBH
16	UNR	UNIRESEARCH BV
17	ZF	ZF CV SYSTEMS HANNOVER GMBH
18	ALI	ALLIANCE FOR LOGISTICS INNOVATION THROUGH COLLABORATION IN EUROPE
19	DPD	DPD (NEDERLAND) B.V.
20	COL	ETABLISSEMENTEN FRANZ COLRUYT NV
21	GRU	GRUBER LOGISTICS S.P.A.
22	GBW	GEBRUEDER WEISS GESELLSCHAFT M.B.H.
23	PG	PROCTER & GAMBLE SERVICES COMPANY NV
23.1	PGP	PROCTER AND GAMBLE POLSKA SPOLKA Z OGRANICZONA ODPOWIEDZIALNOSCIA
23.2	PGA	PROCTER & GAMBLE AMIENS
23.3	PGG	PROCTER&GAMBLE SERVICE GMBH
24	PRI	PRIMAFRIO SL

25	PTV	PTV PLANUNG TRANSPORT VERKEHR GmbH
26	Fraunhofer	FRAUNHOFER GESELLSCHAFT ZUR FORDERUNG DER ANGEWANDTEN FORSCHUNG EV
27	HAN	STICHTING HOGESCHOOL VAN ARNHEM ENNIJMEGEN HAN
28	IDI	IDIADA AUTOMOTIVE TECHNOLOGY SA
29	TNO	NEDERLANDSE ORGANISATIE VOOR TOEGEPAST NATUURWETENSCHAPPELIJK ONDERZOEK TNO
30	UIC	UNION INTERNATIONALE DES CHEMINS DE FER
31	CFL	CFL MULTIMODAL S.A.
32	GSS	Grupo Logistico Sese
33	HIT	Hitachi ABB Power Grids Ltd.
34	IRU	UNION INTERNATIONALE DES TRANSPORTS ROUTIERS (IRU)
35	RIC-UK	RICARDO CONSULTING ENGINEERS LIMITED

Disclaimer/ Acknowledgment

Co-funded by the European Union. The 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 Commission. Neither the European Union nor the granting authority can be held responsible for them.

.

11 Appendix A – Review of tools generated during previous EU funded projects

11.1.1 Road network information

- Road network information
  - Attributes obtained from HERE databases are:
    - · linkID
    - · Speed limit
    - · Traffic flow average speed
    - · Free flow speed
    - · Link length
    - · functional class (road type)
    - · List of longitude/latitude/elevation
    - · Termination of the link (traffic light, stop road signs, etc.)
    - · Road roughness
- Weather information
  - forecast weather description
  - · Temperature
  - · Atmospheric pressure
  - · Solar irradiance.
- Traffic information
- Traffic events
- Charging point locations