This Project is co-funded by the European Union under the 7th Framework Programme

HIGH TOOL

Deliverable D5.2

Pre-Final Version of the HIGH-TOOL Model: Documentation

Project Information

Project number	321624
Project acronym	HIGH-TOOL
Project full title	Strategic high-level transport model
Start date of project	15 April 2013
Duration	34 months
Partners	KIT, FŐMTERV, MCRIT, MKmetric, Panteia, Significance, TNO, TML

Document Information

Work package	WP5: Development of the HIGH-TOOL Model
Deliverable Title	D5.2: Pre-Final Version of the HIGH-TOOL Model: Documentation
Version	2.0
Date of submission	16 December 2015
Main Editor(s)	Oriol Biosca (MCRIT)
Contributor(s)	Andreu Ulied, Cristian Lorenzo, Efrain Larrea (MCRIT)
Reviewer(s)	KIT, Significance
This document should be referenced as	Biosca O. (2015): Pre-Final Version of the HIGH-TOOL Model: Documentation, HIGH-TOOL Deliverable D5.2, project co-funded by the European Commission under the 7th Framework Programme, Barcelona.

Classification – This report is:									
Draft		Final	X	Confidential		Restricted		Public	X

History			
Version	**Issue Date**	**Status**	**Distribution**
0.1	17 July 2015	Draft	Consortium
0.2	3 September 2015	Draft	Consortium
0.3	9 October 2015	Draft	Consortium, Peer Review
0.4	16 October 2015	Draft Final	Consortium, Peer Review
1.0	20 October 2015	Final	Consortium, Project Officer
2.0	16 December 2015	Revised Final	Consortium, Project Officer

Table of Contents

Index of Figures

Index of Tables

Glossary

Angular.js	A client framework to build web applications in JavaScript
Apache	Web server technology
API	Application Programming Interface: A set of routines, protocols, and tools for building software applications
Bootstrap	Mobile first responsive web framework
Bower	A package management system for client-side programming
C	General-purpose imperative computer programming language
C++	General-purpose programming language. It has imperative, object-oriented and generic programming features.
C#	Multi-paradigm programming language. Developed by Microsoft within its .NET initiative and later approved as a standard by ECMA (ECMA-334) and ISO (ISO/IEC 23270:2006).
CO_2	Carbon Dioxide
CSV	Comma-Separated Values
DEM	HIGH-TOOL Demography module
D3.js	JavaScript library used to manipulate documents based on data. It produces HTML and SVG outputs from raw data.
DG MOVE	Directorate-General for Mobility and Transport
DOM	Document Object Model. Cross-platform and language-independent convention for representing and interacting with objects in XML documents
EC	European Commission
ECMA	European Computer Manufacturers Association
ECMAScript	Scripting language standardized by ECMA International in the ECMA-262 specification and ISO/IEC 16262
ECR	HIGH-TOOL Economy & Resources module
ENV	HIGH-TOOL Environment module
EU	European Union
EU28	28 Member States of the European Union
Eurostat	Statistical Office of the European Union
FRD	HIGH-TOOL Freight Demand module
GDP	Gross Domestic Product
GHG	Greenhouse Gas
GIS	Geographic information system
Grunt	JavaScript task runner
GTK	Cross-platform widget toolkit for creating graphical user interfaces
HTML	Hyper Text Markup Language. The standard markup language used to create web pages.
HTML5	Fifth version of the HTML Language
HTTP	Hyper Text Transport Protocol. An application protocol for distributed, collaborative, hypermedia information system.
HTTPS	Hyper Text Transport Protocol Secure. A communications protocol for secure communication over a computer network.
IO	Input-Output
IOT	Input-Output Table

IWW	Inland Water Ways
Java	Object oriented programming
JavaScript	A dialect of ECMAScript programming language
JSON	JavaScript Object Notation. Open standard format that uses human-readable text to transmit data objects consisting of attribute–value pairs. It is used primarily to transmit data between a server and web application, as an alternative to XML.
JVM	Java Virtual Machine. An abstract computer used to translate Java programs into actual machine code and execute them.
Linux	The kernel of the GNU/Linux operating system. It is usual to refer GNU/Linux just as Linux.
Node.js	A platform built on Chrome's JavaScript runtime for building network applications
NO_x	Nitrogen Oxides
NPM	A JavaScript package manager used in Node.js
NUI	Natural User Interface. A User interface design paradigm.
NUTS	Nomenclature of Territorial Units for Statistics. A number assigned to it defines the level of granularity, such as 0 for the country level.
OECD	Organisation for Economic Co-operation and Development
OGC	Open Geospatial Consortium
ORM	Object-relational mapping. In computer science is a programming technique for converting data between incompatible type systems in object-oriented programming languages, creating a "virtual object database" that can be used from within the programming language.
PAD	HIGH-TOOL Passenger Demand module
Pagination bar	Pagination is the process of dividing (content) into discrete pages, either electronic pages or printed pages. A pagination bar is used in User Interface designs as horizontal bar allowing to select different option pages.
Pagination tab	Pagination is the process of dividing (content) into discrete pages, either electronic pages or printed pages. Pagination tabs are used in User Interface designs as a set of horizontal tabs allowing to select different option pages.
pkm	Passenger-kilometre
Porting	In software engineering, porting is the process of adapting software so that an executable program can be created for a computing environment that is different from the one for which it was originally designed (e.g. different CPU, operating system, or third party library). The term is also used when software/hardware is changed to make them usable in different environments.
PostGIS	An open source software programme that adds support for geographic objects to the PostgreSQL object-relational database.
PostgreSQL	A object-relational database
Python	General-purpose, high-level programming language
Qt	Cross-platform application framework
RES	Renewable Energy Sources
RESTFul API	Representational State Transfer (REST) API. A software architecture style consisting of guidelines and best practices for creating scalable web services.
RSS	Rich Site Summary. Family of standard web feed formats to publish frequently updated information: blog entries, news headlines, audio and video.
SAB	Scientific Advisory Board
Sails.js	A server framework to build Node.js applications
SAV	HIGH-TOOL Safety module
Schema	Each of the databases corresponding to a single HIGH-TOOL model run

SDK	Software Development Kit, a set of software development tools
SQL	Structured Query Language
SVG	Scalable Vector Graphics. A markup language used to create two-dimensional graphics with support for interactivity and animation.
Swing	Graphical user interfaces widget toolkit for Java
tkm	Tonne-kilometre
TRANSTOOLS	Network-based European Transport Model
TPM	Transport Policy Measure
VES	HIGH-TOOL Vehicle Stock module
VOC	Volatile organic compound
vkm	Vehicle-kilometre
WebSocket	Protocol providing full-duplex communication channels over a single connection
WP	Work Package

Executive Summary

This Deliverable presents the work developed in the HIGH-TOOL Work Packages WP1 to WP6, that lead to the production of the pre-final version of HIGH-TOOL model.

The HIGH-TOOL model pre-final version is a decentralised modelling system composed of independent but highly interconnected modules such as projection models and databases, as well as system's management and administration components that are linked to a web User Interface. In-depth testing and validation will be carried out in WP7 and WP8, after the submission of the pre-final version.

The HIGH-TOOL pre-final version is an advanced beta version of the HIGH-TOOL model. It will be evaluated by DG MOVE in order to support its further customisation. Furthermore, it will be checked by modellers to enhance the capabilities and accuracy of their models, and by software developers to optimise the whole system. The HIGH-TOOL model allows the user two types of policy analyses:

- Running a single Transport Policy Measure (TPM): for a number of pre-defined policy measures, HIGH-TOOL identifies variables where policy can potentially impact, and users stress the intensity of these impacts.
- Running a User-Defined Policy Scenario: will allow the possibility to define tailor-made policy scenarios.

This document is split into seven chapters:

Chapter 1: Introduction. Presents the scope, current status and future upgrade strategies for the HIGH-TOOL Model.

Chapter 2: Meeting the user needs. Synthesises the user expectations on HIGH-TOOL that motivated the model characteristics, as stated during its development phase.

Chapter 3: Operating HIGH-TOOL. Presents the main features and rationales behind the HIGH-TOOL model, based on a complete overview of the User Interface capabilities.

Chapter 4: Model structure and performance. Synthesises the key organisation of the model from a computation point of view. It presents each of the forecast modules involved in the model, its data management strategy and structure and concludes with a number of indicators that present current model performances.

Chapter 5: System Specifications. Describes the software solution adopted for HIGH-TOOL and the hardware requirements of the model.

Chapter 6: User Handbook. Illustrates how to perform common tasks in the HIGH-TOOL model.

Since testing has begun, validation and adjustment continues. The information provided by this document is not final and will likely change.

1 Introduction

1.1 Approach

Work Package 5 "Development of the HIGH-TOOL model" is about developing the core modelling capabilities of the strategic High-Level tool and its basic structure. The tasks of this work package are centred on the development of each individual module (i.e. demography, economy, passenger and freight demand, vehicle stock, as well as environmental and safety impacts), and the later integration of these individual modules in a logical sequence, so that all together they provide a HIGH-TOOL model capable of analysing the potential interest of European transport policies at a strategic level.

The integrated model developed in Work Package 5 is supported by the data stock and data exchange platform developed in Work Package 3, which provides all data related functionalities (read and write of inputs, outputs, parameters and intermediate variables).

The development of the model in Work Package 5 goes hand in hand with the development of its User Interface in Work Package 6. The User Interface provides all visualisation and reporting capabilities of the tool, including all functionalities related to the entry of inputs (design of transport policy packages, definition of single transport policy measures, definition of socioeconomic framework conditions), their transformation into quantitative values for model variables, the management model runs, and all functionalities related to data retrieval, including raw data exports and generation of processed Policy Assessment Reports.

The HIGH-TOOL model is based on the set of mathematical formulations (equations and elasticities) identified in Work Package 4. The overall software architecture was defined in Work Package 2. The development will be accompanied by activities in Work Package 7 and Work Package 8 concerning validation and robustness of the model.

From the implementation dimension, the HIGH-TOOL approach follows a three-stage tool development process: a **prototype** version, a **pre-final** version, and **final** version. In this sequence, the pre-final version is the second out of three model versions to be delivered by the HIGH-TOOL team. The aim of each of the stages is described below, along with main milestone dates:

- **Prototype**: First fully operational aggregated model at EU level developed in Microsoft EXCEL, useful in order to determine user-needs and system's requirements for the actual model to be developed (D5.1, delivered in May 2014).
- **Pre-final**: Second fully operational disaggregated model at NUTS-2 level developed in Java (D5.2, delivered in October 2015).

- **Final**: Final fully operational disaggregated model at NUTS-2 level with improvements and further customisation after in-depth testing and validation of the pre-final version (D5.3, to be delivered in May 2016).

This document will evolve into Deliverable D5.3 during further development of the project, and will correspond to the final version of the HIGH-TOOL model.

1.2 Aim of HIGH-TOOL

The HIGH-TOOL model is developed within an interactive and user-friendly approach designed to achieve the following objectives:

- Strategically assess different policy measures and trends in terms of economic, environmental and social impacts, in order to define and study alternative scenarios, and to obtain preliminary results, which can be assessed then in depth by advanced and more detailed instruments (such as TRANSTOOLS, TREMOVE or other models).
- Prepare policy-oriented communicative reports of results, with tables, graphics, and maps, to be used in high-level discussion groups and negotiations.

1.3 Structure of the Model

The core of the HIGH-TOOL model consists of seven modules, which are summarised in Table 1.

Table 1: HIGH-TOOL modules

Module	Content
Demography (DEM)	Development of the population and its structural changes.
Economy & Resources (ECR)	Quantifies the economic impacts of transport policy, such as transport related employment and GDP growth.
Vehicle Stock (VES)	Considers the development of the vehicle fleet and its structural changes.
Passenger Demand (FRD)	Reflects passenger transport demand whereby it is sensitive to policy measures effecting the generation, distribution and modal split.
Freight Demand (PAD)	Describes freight transport demand whereby it is sensitive to policy measures effecting the generation, distribution and modal split.
Environment (ENV)	Deals with emissions caused by transport activities and its changes.
Safety (SAF)	For road transport, the module deals with fatalities, serious and slight injuries, and total accident costs. For the other modes, fatalities and total accident costs are predicted.

Furthermore, the system consists of the **Data Stock** which ensures the compilation of data for the different modules (including data parameters definition, sources and units) and the definition of the data exchange mechanism, which will be used to download and upload data (see Kiel et al, 2014).

System's management and administration includes utilities for the system's manager to define user's profiles (e.g. in terms of access restrictions, database space available to store outputs...) and control the links, runs and synchronism of the different modules.

The **User Interface** includes three main components:

- **Pre-processing**: A scenario-building component allowing the derivation of input parameters for framework conditions and policy variables which can be used by the different modules.
- **Processing**: A model run component allowing different running options.
- **Post-processing**: A policy assessment component responsible for generating tables and graphs as policy-meaningful outputs.

The overall conceptual scheme of the HIGH-TOOL modelling system is shown in Figure 1.

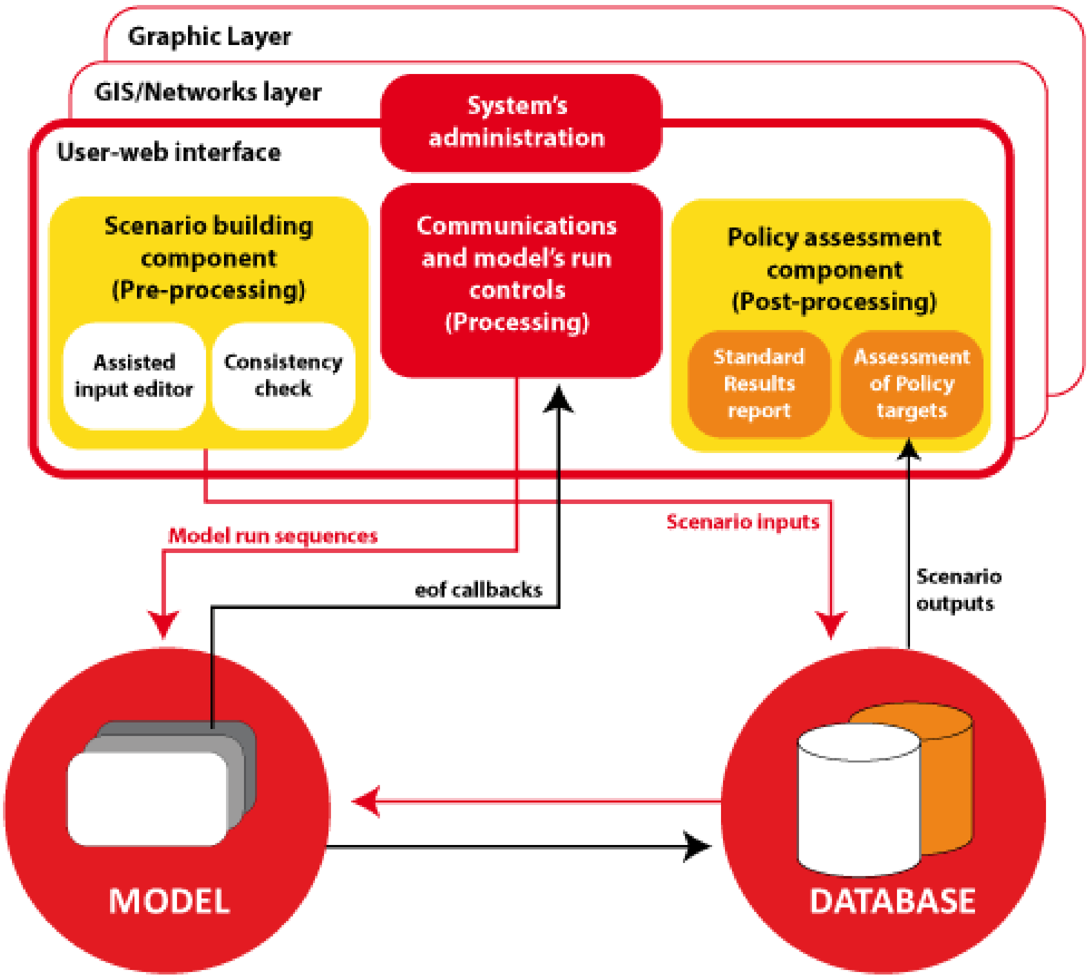


Figure 1: Conceptual scheme of the HIGH-TOOL modelling system

1.4 Restrictions Regarding the Use of Modelling Results

According to the objectives of the pre-final version, it has not undergone any extensive validation and testing activities. Therefore, the results cannot be used for any assessment of transport policies.

1.5 Upgrade Strategy

Given that work will continue with HIGH-TOOL beyond the pre-final version delivery in mid-October 2015, the consortium will keep the online version of HIGH-TOOL (http://81.47.175.201:82) updated with the latest available version of the model at all times. Contact with the system administrator is needed to obtain a user name and an access password. This strategy aims to provide the EC with the latest available releases, increasing robustness, completeness and performance of the model. This strategy will be maintained all along the validation process of HIGH-TOOL.

The planned upgrades for the coming weeks are as follows:

- Upgrade of **Economy and Resources** module. The calibration of this module is on-going (trade figures); an update will be released in November 2015.
- Upgrade of **Freight Demand** module. With new trade figures, the calibration of this module will continue. An update will be released in November 2015.
- Upgrade of **Passenger Demand** module. With a new version of the Economy & Resources module, the calibration of this module will be fine-tuned. An update will be released in November 2015.
- Upgrade of **Vehicle Stock** module. An update will be released in November 2015 fine-tuning the calibration of the module.
- Upgrade of **Environment** module. An update will be released in November 2015 based on the updates of all previous modules.
- Upgrade of **Safety** module. An update will be released in November 2015 based on the updates of all previous modules.
- **Policy Assessment Reports**. The template of the report and its figures will be updated in the view of all previous developments.
- **Input translator**. The finalisation of this component of HIGH-TOOL will allow users to run TPM combinations (Policy Packages with Combined TPMs), and tailor-made policy scenarios (User-Defined Policy Scenarios).

2 Meeting the User Needs

2.1 User Requirements in Relation to Policy Definition

The main policy aims of the Transport White Paper 2011 were confronted with the specific user needs in a number of activities held by HIGH-TOOL with DG MOVE (see Vanherle et al, 2014), while keeping in mind the scope of the model. In Table 2, the strategic role of each of the policy topics is assessed in the rightmost columns in relation to the opinions expressed by participants of HIGH-TOOL user workshops and user surveys.

Table 2: Key policies considered in HIGH-TOOL, and assessment of their relevance in user workshops

Internal market	**crucial**	**important**	**optional**
Internal market for road	x		
Internal market for rail	x		
Internal market for inland waterway transport	x		
Internal market for maritime	x		
Internal market for air	x		
Transport security for cargo			x
Transport security for passenger			x
Transport security for land transport			x
Transport security for "end-to-end"			x
Multimodal Transport	x		
European TEN-T core network		x	
Cross border missing links		x	
Key bottlenecks (freight and passenger)		x	
Multimodal freight corridor structures		x	
EU transport infrastructure in view of energy efficiency needs and climate change challenges		x	
Planning procedure (timing, communication framework, environmental issues)			x
Capacity and quality of transport systems		x	

Taxation and pricing	**crucial**	**important**	**optional**
Infrastructure charging / Access management schemes	x		
Internalisation of external costs (or selected external costs categories and individual modes)	x		
Public funding of transport		x	
Other / new financing instruments			x
Fuel taxation		x	
Transport taxation (vehicle taxation, company car taxation, transport service taxation)		x	

Research and innovation	**crucial**	**important**	**optional**
Technology for vehicles	x		
Technology for transport infrastructure / system		x	
Technology for transport information systems, management & services	x		
Framework for transport safety	x		
Framework for promotion & incentives			x
Framework for technology and infrastructure			x
Intelligent Transport System (ITS)		x	

Efficiency standards and flanking measures	**crucial**	**important**	**optional**
Standards for transport safety	x		
Standards for passenger rights		x	
Standards for environment		x	
Flanking measures for promotion, information, dialogue			x
Flanking measures for regulation			x

2.2 User Requirements in Relation to Output Indicators

The following list presents the most relevant impact indicators as expressed by participants in HIGH-TOOL user workshops and surveys:

- Transport impact indicators;
- GHG emissions;
- Economic growth;
- Employment;
- Cost savings;
- Safety;
- Transport sector employment.

By default, assessment results need to be compared to the EU Reference Scenario in order to identify the potential impacts of policies being modelled.

2.3 Usability Requirements

Full documentation related to the User Interface can be found in HIGH-TOOL Deliverable D6.1 "Design Criteria for the User Interface & Policy Assessment Reports" (Biosca et al, 2015). The web-interface allows the HIGH-TOOL model to be run in an easy and intuitive way and provides detailed results for different indicators.

Following a 'user-centric' approach, the design of the system has begun by defining user needs not only in relation to the model but also in relation to its usability. Activities with future users have been organised all along the project life to obtain permanent user feedback, which allows to consider user needs at the core of all design tasks in HIGH-TOOL.

During the first user workshop in June 2013, participants were asked to express their agreement on a number of statements determining the main pathways of development of HIGH-TOOL. The main conclusions were (Vanherle et al, 2014):

Opinions diverged whether HIGH-TOOL should focus on certain policy domains in more detail or whether it should be defined as a broad, strategic model covering all domains.

A certain degree of agreement existed that HIGH-TOOL should focus on a selected number of impact indicators that can be best assessed with the model.

Participants emphasised the desire for a flexible, transparent model. A medium-detail level of policy inputs should be adopted which is a compromise between user requirements and ability to efficiently model.

HIGH-TOOL should allow changing assumptions to the baseline and other reference scenarios to prevent it from becoming outdated shortly after its finalisation.

EC participants would be willing to take responsibility for the consistency of the policy inputs. However, this would not mean that no efforts would have to be made to provide some guidance to the user, e.g. providing sensible ranges for parameters.

Data and hypotheses in HIGH-TOOL could be introduced in the model both as prefab and as custom input. By 'prefab' it is meant that input data and scenarios are (almost) fully pre-prepared during development leaving the users some possibilities to make changes, whereas 'custom' meant that the user is mostly in charge for the input which involves more flexibility but also more complexity.

Other requirements discussed during the user workshops have been included in HIGH-TOOL as much as possible:

HIGH-TOOL should focus on spatial aspects rather than only providing aggregate results at a national level (at NUTS-2 level). Only indicators for which the regional level does not provide added value, such as GHG emissions, aggregate results at national level would suffice.

Runtime of the model would depend on the level of detail for policies, but the expectations on computation time were not overly restrictive, in the order from 30 minutes to several hours.

Relatively balanced opinions were received with regard to the question whether the HIGH-TOOL model should be designed to run locally or online on a remote server. It was mentioned that sharing runs and results with colleagues would be easier with an online version.

3 Operating HIGH-TOOL: Model Capabilities

3.1 General Approach

Figure 2 shows the logical processes needed to operate the HIGH-TOOL model. These processes are performed through the User Interface, and are complemented by the Data Stock for mechanisms regarding data storage and retrieval.

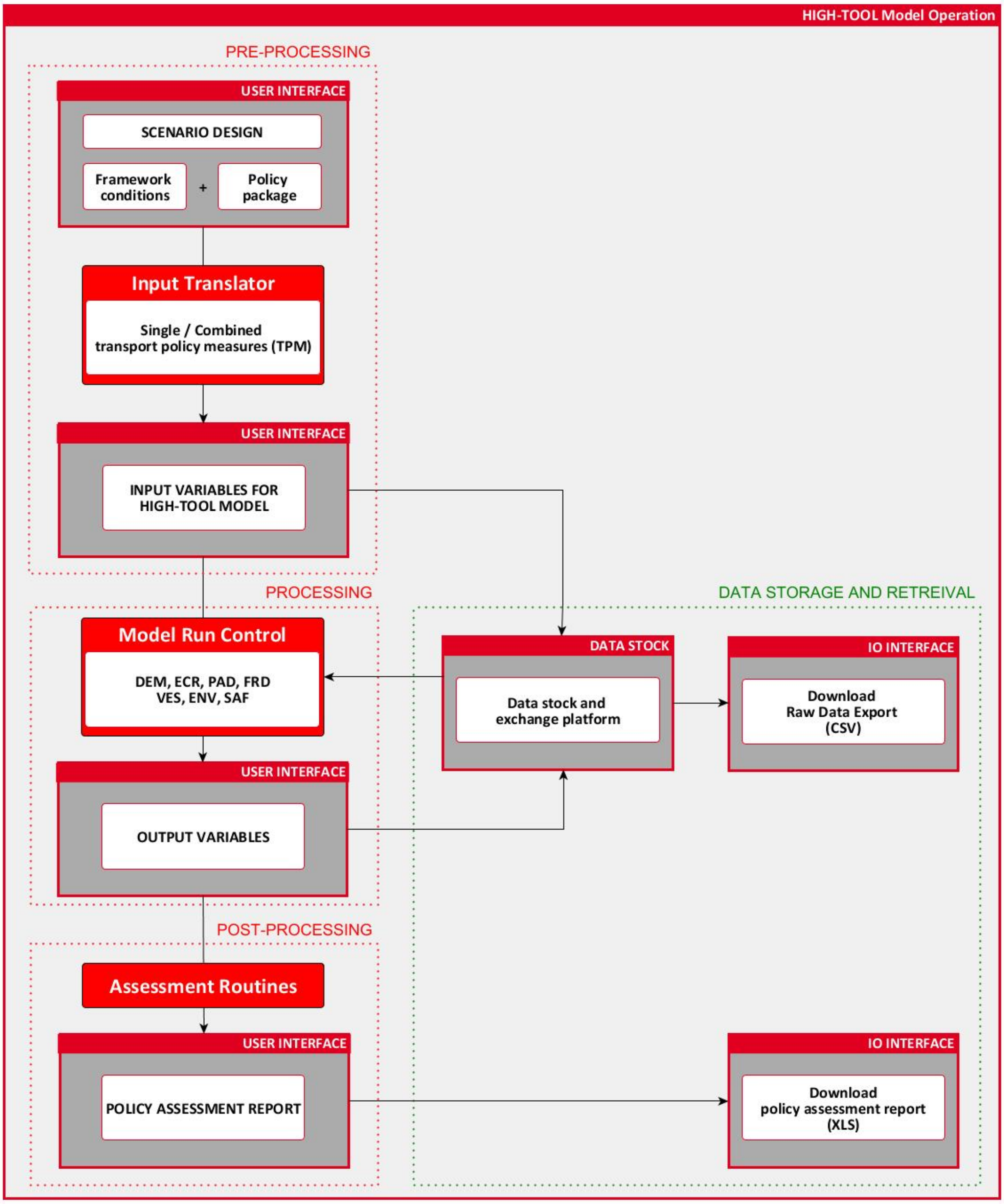


Figure 2: General approach to operation of the HIGH-TOOL model

In brief, the User Interface is integrated by the next components:

- **Pre-processing**: A scenario-building component allowing a derivation of input parameters for framework conditions and policy variables which can be used by different modules.
- **Processing**: A model run component allowing different running options.
- **Post-processing**: A policy assessment component responsible for generating tables and graphs as policy-meaningful outputs.

All User Interface components are reported in detail in the next sections.

3.2 Design Criteria of the User Interface

The final goal of the User Interface is facilitating transport policy experts without a deep modelling background to use the modelling system in a productive way. On one hand the User Interface has to be productive, clear and friendly enough to reduce the user's tasks with low added-value as much as possible (e.g. filling large lists with input variables and parameters for model running, exchanging data between software applications). On the other hand, the User Interface has to be intelligent enough to support tasks with higher-value such as defining consistent sets of input variables and analysing results in a policy-meaningful way. All considered, the web-interface will help users in the pre-processing (e.g. defining scenarios), processing (e.g. running scenarios) and post-processing tasks (e.g. powerful graphic and cartographic visualisation options).

The User Interface is therefore conceived as a 'user-centric' application instead of a 'model-centric' application. It is not just the model's interface, but the user's interface for models and databases; thus it is designed as a translator between the user's thoughts and the model's needs for specific input variables. Only in this way users can really feel the interface intuitively.

The design principles adopted for the HIGH-TOOL User Interface are based on the criteria outlined by Tognazzini (2013). They include **Anticipation**, **Visible Navigation**, **Efficiency**, **Consistency**, **Explorable Interfaces**, **Learnability**, **Readability**, and **Intuitiveness** and **Self-Explaining**.

The criteria have been addressed by the HIGH-TOOL User Interface as follows:

Anticipation: The HIGH-TOOL User Interface is designed to anticipate user needs. Users are not expected to search for information or evoke necessary tools, but the information and tools needed for each step of the process are provided to the users in a natural way. Thus default values provided by HIGH-TOOL are 'intelligent'.

Visible Navigation: Navigation is reduced to a minimum. The HIGH-TOOL User Interface is designed such that users can access configuration, hypotheses, calculations and result screens by a minimum number of clicks and actions. Navigation is clear and natural, allowing a user to quickly transition from novice to expert.

Efficiency: The HIGH-TOOL User Interface facilitates user's productivity, not computer's productivity. The application is designed such that task flows are comfortable, clear and optimal for the user, rather than organised according to programming needs.

Consistency with user expectations:

- Logical procedures stay coherent along the application.
- Buttons, input fields and navigational controls are always located in the same position of the interface.
- Harmonious 'overall look' of the different screens.

Explorable Interfaces: Users of the HIGH-TOOL User Interface are offered a line of least resistance allowing them to do just what they want, getting the job done in the quickest way possible, while still supporting those who want to explore further. This means stable visual elements to allow fast navigation and making actions reversible.

Learnability: The HIGH-TOOL User Interface allows for a quick learning curve that supports the performance of tasks within the minimum time possible (Natural-User Interface, NUI).

Readability: Texts displayed in the HIGH-TOOL User Interface are incorporated in such a way that they can be read properly without effort, favouring dark text on pale backgrounds, avoiding grey backgrounds and using font sizes that are large enough to be readable on standard monitors.

Intuitive and Self Explaining: The layout of HIGH-TOOL and the logic of processes in the User Interface are conceived in a way that makes it possible for users to run the tool on an intuitive basis. Contextual information provided by the User Interface, information made available through icons and tool-tip displays support the self-explaining property of HIGH-TOOL.

All design criteria together empower users to work efficiently with HIGH-TOOL while enjoying all features.

3.3 Pre-Processing: Definition of Policies

The main component of the User Interface is the scenario-building interface, which helps users to translate policy questions and framework conditions into the specific values of input variables. It is foreseen for the final version of the HIGH-TOOL model different ways of assessing transport policies, ranging from quick and simple policy definitions (but with lower degree of possible user customisation) to more holistic policy scenarios (but requiring more time and effort from the user side to ensure coherence and consistency of hypotheses).

In brief, the following options are foreseen for policy analysis and assessment:

- a single Transport Policy Measure (TPM);
- a Policy Package with Combined TPMs (not available in the pre-final version);
- a User-Defined Policy Scenario analysis module.

3.3.1 Single Transport Policy Measure (TPM)

The first module provides a set of pre-defined TPMs to be analysed on a one per one basis; TPMs are always analysed against the EU reference socioeconomic baseline 2010–2050 (official population and economy projections).

This is the quickest way to evaluate transport policies. For each pre-defined TPM, HIGH-TOOL identifies which of the model variables are impacted (only a limited number of variables for each TPM), and what are the reasonable impact ranges. Users define the policy intensity for each of these variables.

Users work directly upon model variables. Users select the evolution of model variables involved in a TPM between 2010 and 2050. Possible variable ranges (maximum and minimum allowed values) are provided as a reference. Such ranges are defined in HIGH-TOOL based on extensive literature review. Details can be found in Deliverable D4.2 "Elasticities and Equations" (van Grol et al, 2015).

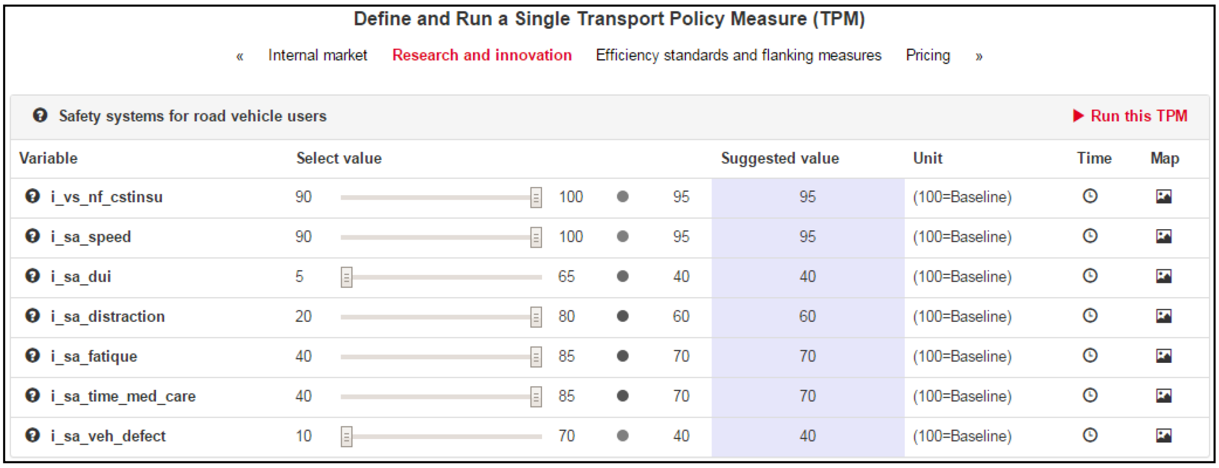


Figure 3: Sample of the interface to define and run a single Transport Policy Measure (TPM)

The 39 TPMs currently considered are presented below in Table 3.

Table 3: Single Transport Policy Measures considered in HIGH-TOOL model

Internal Market measure	Explanation
Acceleration of TEN-T implementation	Policy set to simulate an accelerated implementation of TEN-T projects, under the assumptions that the Core TEN Network will be completed by the year 2025 and the Comprehensive TEN Network by the year 2040.
Access to rail infrastructure	"Ensure effective and non-discriminatory access to rail infrastructure, including rail-related services, in particular through structural separation between infrastructure management and service provision"
Enhance service quality at airports	Clarify and improve conditions to enter and provide quality services, including ground handling: ensure that all actors in an airport system meet minimum quality standards.
Enhance service quality at ports	Review restrictions on provision for port services.
European Rail Traffic Management System	ERTMS (European Rail Traffic Management System)
Freight corridor management	Develop an integrated approach to freight corridor management, including track access charges.
Harmonized handling of dangerous goods	Streamline the rules for the intermodal transport of dangerous goods to ensure interoperability between the different modes.
Harmonisation of rail safety	Progressively achieve a sector-wide approach to safety certification in the rail transport sector, building on existing approaches for infrastructure managers and railways undertakings and evaluating the possibility to rely on a European standard. Enhance the role of ERA in the field of rail safety, in particular its supervision on national safety measures taken by National Safety Authorities and their progressive harmonisation.
Harmonized social rules for truck drivers	EU-wide common job quality and working conditions for truck drivers SEC(2008)2632 Policy set to simulate the regulation of job quality and working conditions for truck drivers, assuming an impact on travel costs and travel time for the truck mode.
Maritime traffic management system	Integrate the use of monitoring tools by all relevant authorities, ensure the full interoperability between ICT systems in the waterborne sectors, guarantee the monitoring of vessels and freight (Blue Belt) and set up appropriate port facilities ("Blue Lanes").
Opening the internal IWW market	Establishes an appropriate framework to optimise the Internal Market for Inland Waterway Transport, and to remove barriers that prevent its increased use. Assesses and defines the necessary tasks and mechanisms for their execution, also with a view to the wider European context.
Opening the internal rail market	Open the domestic rail passengers market to competition, including mandatory award of public service contracts under competitive tendering.
River information system	Policy set to simulate the development of the River Information Services, assuming an impact on IWW transport time thanks to the reduction of administration burdens and the provision of information exchange for freight transport.
Single European road market	Elimination of restrictions on road cabotage.
Single European Sky	Deploy the future air traffic management system (SESAR) in the agreed timeframe.
Single rail vehicle authorisation and certification	Achieve a single vehicle type authorisation and a single railway undertaking safety certification by reinforcing the role of the European Railway Agency.

Pricing measure	Explanation
CO_2 certificate system for road transport	Policy set to simulate the implementation of a CO_2 certificate system. A certificate price in constant (2005) EUR per ton CO_2 can be implemented and the trend for this price over the future time period can be defined. The certificate price is converted in an additional cost by fuel type (depending on their CO_2 specific emission factor), which is applied to all road modes.
CO_2 feebates for road transport	Policy set to simulate the implementation of a feebates to stimulate the diffusion of low emission, fuel-efficient and alternative fuel vehicles. A system offering rebates for purchasers of clean vehicles and in parallel setting fees for vehicles emitting more CO_2 than a certain threshold is called feebates.
Circulation tax for cars	Vehicle taxation (circulation & registration taxes). Policy set to simulate the implementation of common rules for annual car circulation taxes based on CO_2 emissions and the phase out of registration taxes at EU level in order to encourage the use of fuel-efficient cars.
HDV infrastructure charge	Phase in a mandatory infrastructure charge for heavy-duty vehicles. The scheme would introduce a common tariff structure and cost components such as the recovery of wear and tear, noise and local pollution costs to replace the existing user charges.
Internalisation of external costs	Proceed with the internalisation of external costs for all modes of transport applying common principles while taking into account the specificity of each mode.
Urban road charging	Urban Area charging/Cordon pricing.

Efficiency standards and flanking measures	Explanation
CO_2 emissions limits for road vehicles	CO_2 emission limits for HDV, LDV and cars. Policy set to simulate restrictive limits on CO_2 emissions from new vehicles (cars and trucks).
Deployment of efficient vehicles	Support the deployment of new vehicles and vessels and retrofitting.
Diffusion of electro cars	Electromobility road measures to promote increased replacement rate of inefficient and polluting vehicles. The policy is set in order to accelerate the diffusion of vehicles with electric propulsion into the European car vehicle fleet.
Diffusion of H_2 fuel cell cars	H_2 Fuel Cell vehicle measures to promote increased replacement rate of inefficient and polluting vehicles. The policy is set in order to stimulate the diffusion of H_2 fuel cell vehicles into the European car vehicle fleet.
HDV limitation for urban areas	The City Logistics policy includes various measures regarding urban freight distribution (e.g. urban consolidation centres) aimed at reducing the traffic of heavy duty vehicles through cities and metropolitan areas by means of the implementation of technical and planning measures.
Improving local public transport	Policy set to simulate an improvement of frequency and reliability of public transport services (bus or train), in terms of reduction of travel time at local level.
LDV speed limit	Examine approaches to limit the maximum speed of light commercial road vehicles, in order to decrease energy consumption, to enhance road safety and to ensure a level playing field.
Pollutant limits for road vehicles	Standards for controlling air pollution (CO, NO_x, particulate matter). Policy set to simulate restrictive limits on pollutant emissions from new vehicles (cars and trucks).
Replacement of inefficient LDVs and buses	The policy is set to stimulate purchasing clean and energy-efficient commercial vehicles (LDV and buses) at urban level.
Replacement of inefficient car	The policy is set in order to stimulate the replacement of inefficient and polluting vehicles into the European car vehicle fleet.

Research and innovation measure	Explanation
Dynamic traffic management for road	Deployment of next generation of multimodal traffic management and information systems.
Improvement of energy efficiency of vehicles	Technological innovation on vehicle efficiency through new engines, materials and design.
Intelligent road vehicles	Deployment of next generation of multimodal traffic management and information systems.
Intelligent traffic information system for road	Deployment of next generation of multimodal traffic management and information systems.
New fuels and propulsion systems	Technological innovation on new fuels and propulsion systems to achieve cleaner energy use for all modes of passenger and freight transport.
Road vehicle safety technology protecting other transport users	Safer vehicle technologies for vulnerable users such as pedestrians, cyclists and motorcyclists.
Safety systems for road vehicle users	The policy is set to simulate the diffusion of a variety of technical safety systems in road vehicles such as driver assistance systems, seat belt reminder, eCall and vehicle-infrastructure interface.

3.3.2 Policy Package with Combined TPMs

This module will allow the simultaneously merging of different TPMs to simulate more complex Policy Packages. As with single Transport Policy Measures, users work directly upon model variables. The module will be fully developed in the final version of HIGH-TOOL once TPMs become finalised. It has not been implemented yet in the User Interface.

A key issue is how inputs on different TPMs influencing the same model variables can be considered simultaneously. In some cases it may be possible to simply add the impact of two TPMs; but in some other cases there might be impacts in opposite directions, or impacts in the same direction that do not fully add together. Work is advanced on the analysis of possible interdependencies among TPMs. However, this module cannot be fully implemented until single TPMs have been extensively tested.

3.3.3 User-Defined Policy Scenarios

The module will be fully developed towards the final version of HIGH-TOOL, once TPMs and Combined TPMs become final. It has been implemented in the User Interface, but only the interface structure will remain, the content will be adjusted. Highly aggregated variables included in this module for the user to define scenarios will be fine-tuned in the light of performance of transport modules towards the final version.

User-Defined Policy Scenarios can be analysed against the socioeconomic baseline, but also against variations of this baseline defined by users (custom Framework Conditions).

Whereas when working with Single TPMs or Combined TPMs, users will work directly with model input variables (each TPM affects a number of variables of the model), when working with free User-Defined Policy Scenarios, users deal with more aggregated variables that HIGH-TOOL will translate into specific values for model variables.

The Input Translator is the component (currently being developed) which translates values introduced by users to aggregated variables, into specific values for model's input variables. The Policy Translator is built as a number of external tables in MS EXCEL (see Figure 4) through custom formulations. Formulations and multipliers are based on the in-depth analyses performed in Work Package 4 of HIGH-TOOL.

1	interface_variable	formula to translate interface inputs into model	description of the translation rule applied	model variable	dimension_mode	description
12	30	dat_var=dat_var*(1-int_var_30*5)	increase in %gdp spent on new infrastructure becomes a reduction of average distance	i_fd_imp_dist	2	rail
13	31, 33	dat_var=dat_var*(1-(int_var_31+int_var_33)*5)	increase in %gdp spent on new infrastructure becomes a reduction of average distance	i_fd_imp_dist	3	air
15	32, 33	dat_var=dat_var*(1-(int_var_32+int_var_33)*5)	increase in %gdp spent on new infrastructure becomes a reduction of average distance	i_fd_imp_dist	5	sea
16	32, 33	dat_var=dat_var*(1-(int_var_32+int_var_33)*5)	increase in %gdp spent on new infrastructure becomes a reduction of average distance	i_fd_imp_dist	6	short-sea
17	29	dat_var=dat_var*(1-int_var_29*5)	increase in %gdp spent on new infrastructure becomes a reduction of average distance	i_fd_imp_dist	7	coach
18	29	dat_var=dat_var*(1-int_var_29*5)	increase in %gdp spent on new infrastructure becomes a reduction of average distance	i_fd_imp_dist	8	road_car
19	29	dat_var=dat_var*(1-int_var_29*5)	increase in %gdp spent on new infrastructure becomes a reduction of average distance	i_fd_imp_dist	9	road_motorcycl
62	24	dat_var=dat_var*(1-int_var_24*0,1)	decrease of speed improves crash safety standards	i_sa_speed_car		
63	24	dat_var=dat_var*(1-int_var_24*0,95)	decrease of driving drunk improves crash safety standards	i_sa_dui_car		
64	24	dat_var=dat_var*(1-int_var_24*0,8)	decrease of distractions improves crash safety standards	i_sa_distraction_car		
65	24	dat_var=dat_var*(1-int_var_24*0,6)	decrease of fatigue while driving improves crash safety standards	i_sa_fatigue_car		
66	24	dat_var=dat_var*(1-int_var_24*1)	increase of safety belt use improves crash safety standards	i_sa_restraint_car		
67	23	dat_var=dat_var*(1-int_var_23*0,5)	improvement in roads safety equipment and smart infrastructure	i_sa_infra_fault_car		
68	24	dat_var=dat_var*(1-int_var_24*0,9)	decrease of defects in cars improves crash safety standards	i_sa_veh_defect_car		
69	24	dat_var=dat_var*(1-int_var_24*0,6)	decrease of time for medical care to arrive improves crash safety standards	i_sa_time_med_care_car		
70	24	dat_var=dat_var*(1-int_var_24*0,1)	decrease of speed improves crash safety standards	i_sa_speed_p2w		
71	24	dat_var=dat_var*(1-int_var_24*0,95)	decrease of driving drunk improves crash safety standards	i_sa_dui_p2w		
72	24	dat_var=dat_var*(1-int_var_24*0,8)	decrease of distractions improves crash safety standards	i_sa_distraction_p2w		
73	24	dat_var=dat_var*(1-int_var_24*0,6)	decrease of fatigue while driving improves crash safety standards	i_sa_fatigue_p2w		
74	24	dat_var=dat_var*(1-int_var_24*1)	increase of helmet belt use improves crash safety standards	i_sa_helmet_p2w		
75	23	dat_var=dat_var*(1-int_var_23*0,5)	improvement in roads safety equipment and smart infrastructure	i_sa_infra_fault_p2w		
77	24	dat_var=dat_var*(1-int_var_24*0,9)	decrease of defects in cars improves crash safety standards	i_sa_veh_defect_p2w		
78	24	dat_var=dat_var*(1-int_var_24*0,6)	decrease of time for medical care to arrive improves crash safety standards	i_sa_time_med_care_p2w		
79	24	dat_var=dat_var*(1-int_var_24*0,1)	decrease of speed improves crash safety standards	i_sa_speed_pt		
80	24	dat_var=dat_var*(1-int_var_24*0,95)	decrease of driving drunk improves crash safety standards	i_sa_dui_pt		
81	24	dat_var=dat_var*(1-int_var_24*0,8)	decrease of distractions improves crash safety standards	i_sa_distraction_pt		
82	24	dat_var=dat_var*(1-int_var_24*0,6)	decrease of fatigue while driving improves crash safety standards	i_sa_fatigue_pt		
83	23	dat_var=dat_var*(1-int_var_23*0,5)	improvement in roads safety equipment and smart infrastructure	i_sa_infra_fault_pt		
84	24	dat_var=dat_var*(1-int_var_24*0,9)	decrease of defects in cars improves crash safety standards	i_sa_veh_defect_pt		

Hoja1 | _parameters | _em_type | _agegroup | _dimensions | _gender | _agegroup_simple | _db | _area_type | _db_iww | _ht2006_cont | _tables | _cost_type ...

Figure 4: Sample of the User Interface to define policies in User-Defined Policy Scenarios

When dealing with User-Defined Policy Scenarios, the User Interface becomes organised in a policy-meaningful manner. The User Interface groups policy fields by key policy areas aiming to provide a familiar work environment to users of the EC. The User Interface indicates how extreme assumptions are (input values) by using logical colour scales (from light grey to black), indicates reference values for baseline assumptions (for contrast purposes) and provides graphical tools to disaggregate values over time and at regional level.

3.4 Processing: Running Simulations

The User Interface is responsible for calling individual HIGH-TOOL modules in a logical sequence so that a full model run can be developed over time, for the whole simulation period.

When a user launches a run, the User Interface of HIGH-TOOL is responsible for managing this run, calling the different modules in due time, gathering results and presenting them to users. The User Interface allows users to visualise all runs being processed at any time (ongoing runs) and managing the run sequence, e.g. pausing or halting a specific run. HIGH-TOOL is a multitask platform, meaning that it can handle more than one run simultaneously, though available computational resources need to be shared between all active runs. Partial results of an ongoing simulation become available for users before the run has finalised, for instance intermediate results of a module that has finished calculating while the next module is still running can be downloaded using the Export function.

Figure 5 shows the User Interface screen where users can manage ongoing runs. Each line corresponds to a model run. The *Status* field displays whether a run is active, on standby, paused or finalised. The *Step* field indicates which specific step is being undertaken for an unfinished run: e.g. the run ID 6 is processing the Freight Demand module for year 2010. The right-most panel allows users managing active runs. Available options (left to right), are as follows: visualising available results, downloading available data, pause (or reactivating a paused run), and abort/delete a run.

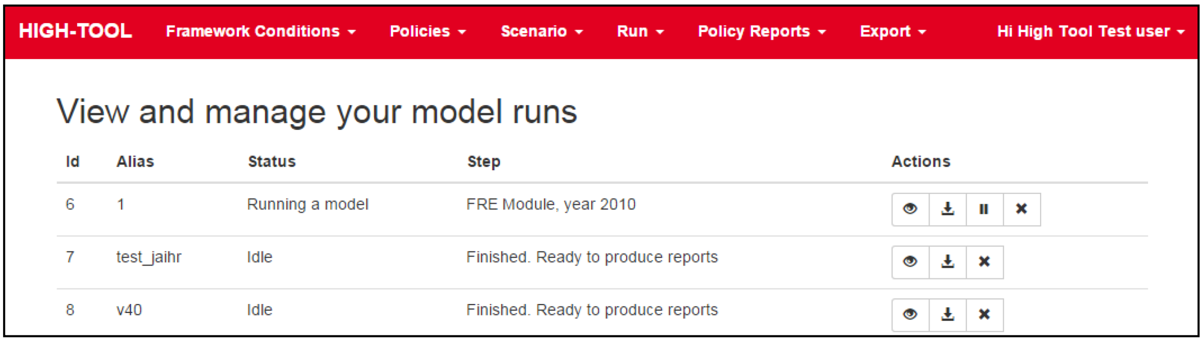


Figure 5: Sample of the User Interface to manage model runs and results

The run sequence of HIGH-TOOL modules is established as illustrated in Figure 6 below.

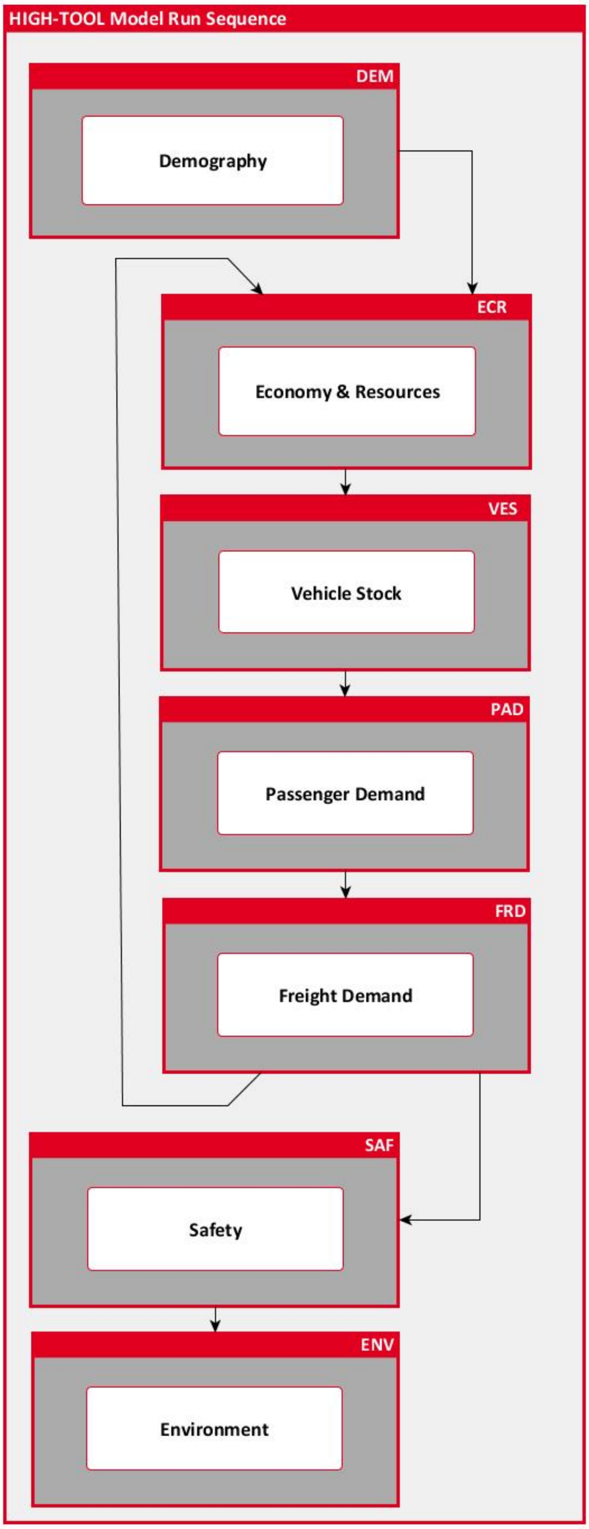


Figure 6: Model run sequence in HIGH-TOOL. The Model run sequence is managed and controlled by the HIGH-TOOL User Interface

The Demography module runs first for the whole time span of a HIGH-TOOL simulation.

A loop is then started where the Economy and Resources, Vehicle Stock, Passenger Demand and Freight Demand modules are run sequentially for each temporal step within the time span of a HIGH-TOOL simulation. Time steps are currently defined as 5 years, but eventually they might be increased to reach an adequate balance between model precision and model performance.

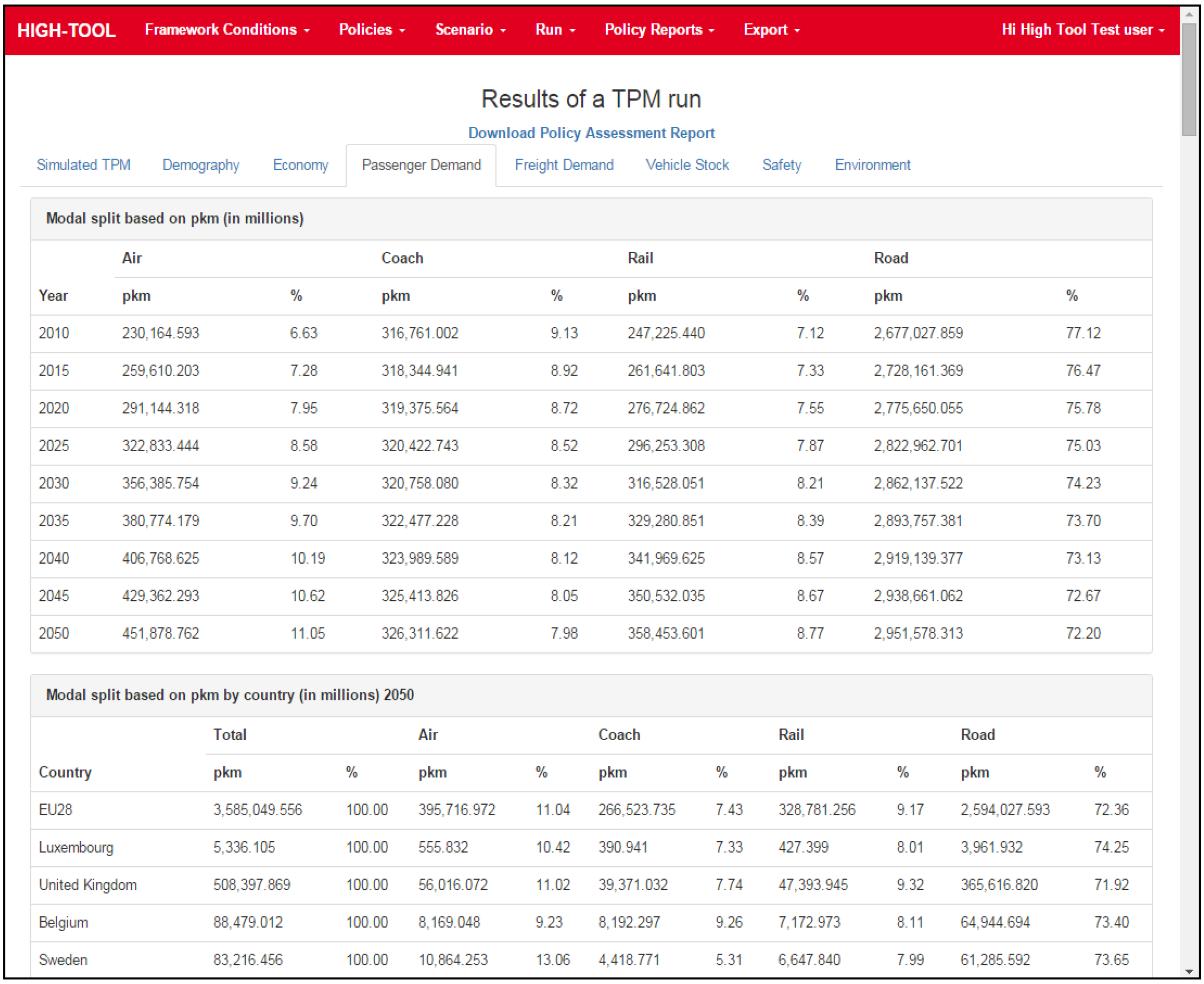
Before the demand can be calculated the user costs have to be determined as input to the demand models. The user costs depend on the degree of motorisation and the vehicle fleet which is determined by the economic situation. In the next time steps the changes in demand are a driver for the Vehicle Stock. The HIGH-TOOL Consortium is planning to conduct a case study if the application of 5-years time steps results to significant differences to the application of 1-year time steps.

The Safety and the Environment modules are called at the end of the previous loop, and run one after the other for the whole time span of the HIGH-TOOL simulation.

3.5 Post-processing: Generation of Policy Assessment Reports

The policy report integrates the outputs of the HIGH-TOOL model into a relatively short number of policy-relevant indicators, and presents these indicators in a meaningful way to the user. Results are provided in a detailed level in the form of tables, figures and maps, for the period 2010–2050 in at least five-year steps.

Indicators included in policy assessment reports are displayed in the User Interface (Figure 7), and are also published in pre-designed MS EXCEL files (Figure 8), so tables and graphics are automatically produced and the users can easily adjust or modify them whenever needed, or use them in other word processor or presentation documents.



Modal split based on pkm (in millions)

	Air		Coach		Rail		Road	
Year	pkm	%	pkm	%	pkm	%	pkm	%
2010	230,164.593	6.63	316,761.002	9.13	247,225.440	7.12	2,677,027.859	77.12
2015	259,610.203	7.28	318,344.941	8.92	261,641.803	7.33	2,728,161.369	76.47
2020	291,144.318	7.95	319,375.564	8.72	276,724.862	7.55	2,775,650.055	75.78
2025	322,833.444	8.58	320,422.743	8.52	296,253.308	7.87	2,822,962.701	75.03
2030	356,385.754	9.24	320,758.080	8.32	316,528.051	8.21	2,862,137.522	74.23
2035	380,774.179	9.70	322,477.228	8.21	329,280.851	8.39	2,893,757.381	73.70
2040	406,768.625	10.19	323,989.589	8.12	341,969.625	8.57	2,919,139.377	73.13
2045	429,362.293	10.62	325,413.826	8.05	350,532.035	8.67	2,938,661.062	72.67
2050	451,878.762	11.05	326,311.622	7.98	358,453.601	8.77	2,951,578.313	72.20

Modal split based on pkm by country (in millions) 2050

	Total		Air		Coach		Rail		Road	
Country	pkm	%	pkm	%	pkm	%	pkm	%	pkm	%
EU28	3,585,049.556	100.00	395,716.972	11.04	266,523.735	7.43	328,781.256	9.17	2,594,027.593	72.36
Luxembourg	5,336.105	100.00	555.832	10.42	390.941	7.33	427.399	8.01	3,961.932	74.25
United Kingdom	508,397.869	100.00	56,016.072	11.02	39,371.032	7.74	47,393.945	9.32	365,616.820	71.92
Belgium	88,479.012	100.00	8,169.048	9.23	8,192.297	9.26	7,172.973	8.11	64,944.694	73.40
Sweden	83,216.456	100.00	10,864.253	13.06	4,418.771	5.31	6,647.840	7.99	61,285.592	73.65

Figure 7: Model results displayed on the User Interface of HIGH-TOOL

A number of pre-defined graphics to illustrate key indicators have been developed in MS EXCEL. In terms of communication, online solutions can be very effective, but in terms of productivity, MS EXCEL is probably the best solution since users are able to get both tables with indicators and graphics pre-designed in a file, and can easily adjust the design, select tables and graphics for their own reports and presentations.

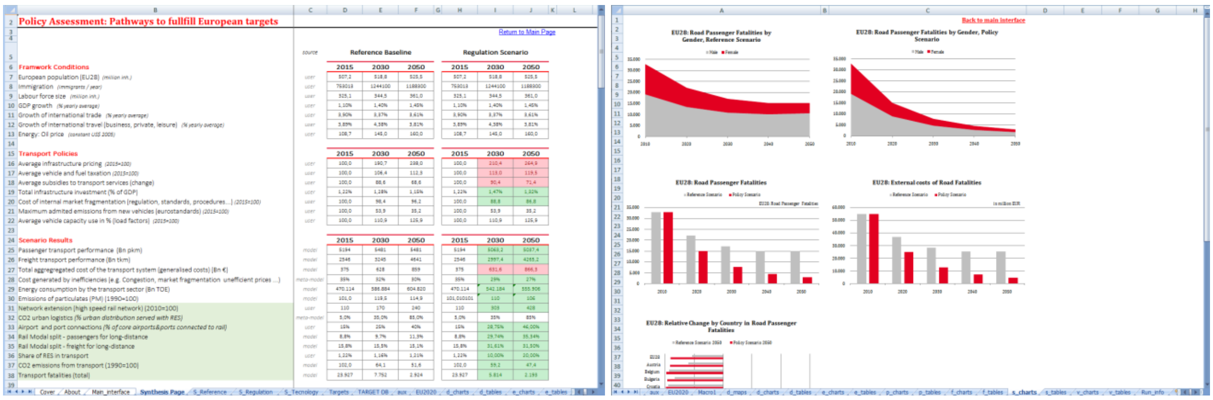
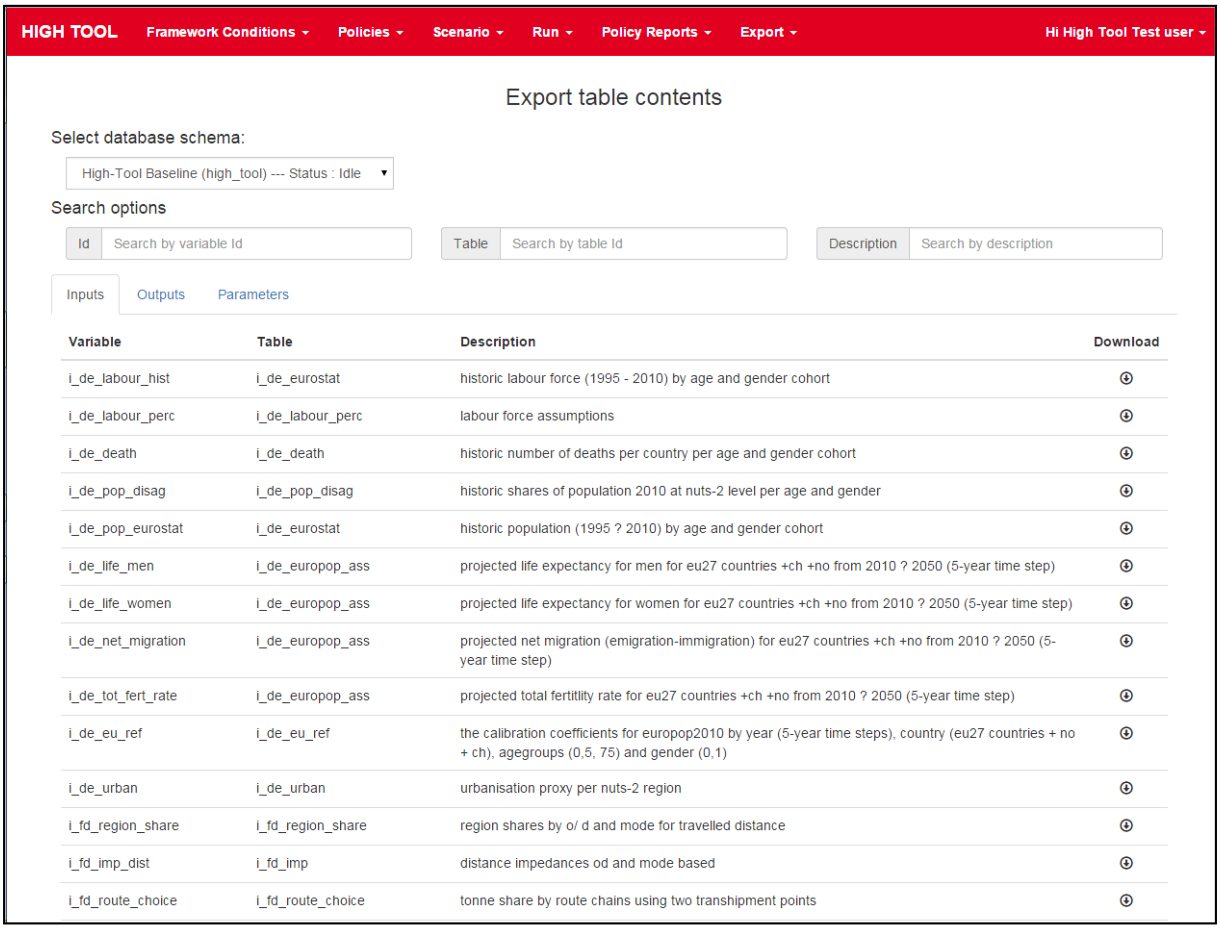


Figure 8: MS EXCEL synthesis of scenario results – Tables and graphics generated by the HIGH-TOOL model

Full results of the HIGH-TOOL model can also be obtained directly in raw formats (CSV) for further user management. The HIGH-TOOL User Interface allows the download of any table produced by a specific HIGH-TOOL module using the Export Table functionality (see Figure 32).



Variable	Table	Description	Download
i_de_labour_hist	i_de_eurostat	historic labour force (1995 - 2010) by age and gender cohort	
i_de_labour_perc	i_de_labour_perc	labour force assumptions	
i_de_death	i_de_death	historic number of deaths per country per age and gender cohort	
i_de_pop_disag	i_de_pop_disag	historic shares of population 2010 at nuts-2 level per age and gender	
i_de_pop_eurostat	i_de_eurostat	historic population (1995 ? 2010) by age and gender cohort	
i_de_life_men	i_de_europop_ass	projected life expectancy for men for eu27 countries +ch +no from 2010 ? 2050 (5-year time step)	
i_de_life_women	i_de_europop_ass	projected life expectancy for women for eu27 countries +ch +no from 2010 ? 2050 (5-year time step)	
i_de_net_migration	i_de_europop_ass	projected net migration (emigration-immigration) for eu27 countries +ch +no from 2010 ? 2050 (5-year time step)	
i_de_tot_fert_rate	i_de_europop_ass	projected total fertitlity rate for eu27 countries +ch +no from 2010 ? 2050 (5-year time step)	
i_de_eu_ref	i_de_eu_ref	the calibration coefficients for europop2010 by year (5-year time steps), country (eu27 countries + no + ch), agegroups (0,5, 75) and gender (0,1)	
i_de_urban	i_de_urban	urbanisation proxy per nuts-2 region	
i_fd_region_share	i_fd_region_share	region shares by o/ d and mode for travelled distance	
i_fd_imp_dist	i_fd_imp	distance impedances od and mode based	
i_fd_route_choice	i_fd_route_choice	tonne share by route chains using two transhipment points	

Figure 9: Export raw data control panel

4 Model Structure and Performance

4.1 General Organisation of the Model

Figure 10 illustrates the structure of the HIGH-TOOL model (Mandel et al, 2015). A full overview and a description of inputs and outputs for each module are provided in Deliverable D2.1 (Mandel et al, 2013).

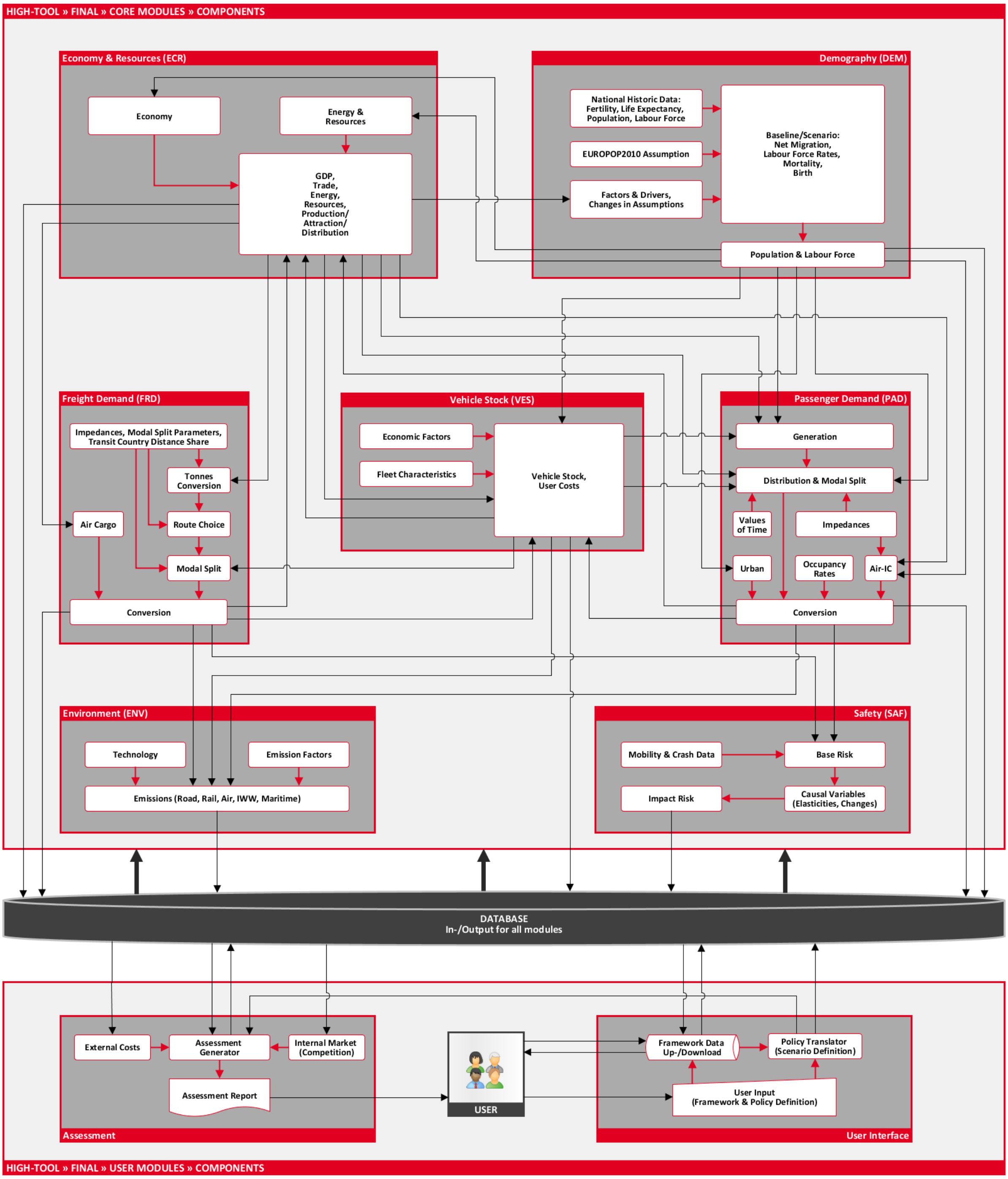


Figure 10: Structure of the pre-final version of the HIGH-TOOL model

4.2 Description of HIGH-TOOL Modules

Full documentation related to models, equations and elasticities can be found in HIGH-TOOL Deliverable D4.2 "Elasticities and Equations" (Van Grol et al, 2015).

4.2.1 Demography Module

The Demography module reflects the demographic development of the regions considered within HIGH-TOOL. Calculations are performed at the level of countries *ci* (NUTS-0 level) and thereafter disaggregated to zones *i* at NUTS-2 level. The development of the population is simulated by the Demography module with a cohort component model that takes the effects of demographic drivers and migration into account. The module is sensitive to changes affecting the exogenous demographics (e.g. fertility rates per specific age group) and endogenous socio-economic drivers (e.g. GDP). For the development of the pre-final HIGH-TOOL model, only exogenous drivers are considered, i.e. the fertility, mortality, and migration rates per gender *g* and age group *a*. The final output of DEM is the estimation of population for the EU28 including Switzerland and Norway at NUTS-2 level by gender *g* and age group *a*. The Demography module also estimates the labour force up to 2050 by gender *g* and age group *a*. The module does not use at this stage any inputs from other HIGH-TOOL modules. This output of the module is shown in Table 4.

Table 4: Interaction of the Demography module with other HIGH-TOOL modules

I/O	Variable	Description	Dimensions	Module(s)	Name in database
Out	*pop*tot	Population [persons]	time period *t*, country *ci*, gender *g*, age group *a*	Passenger Demand, Economy & Resources	o_de_pop
Out	*LB*	Labour force [persons]	time period *t*, zone *i*, gender *g*, age group *a*	Passenger Demand, Economy & Resources	o_de_labour

4.2.2 Economy and Resources Module

The Economy and Resources module (ECR) simulates the impact of transport policies on the economy. Its role is to provide baseline economic projections regarding GDP and value added, to provide projections for inter-regional trade flows and to calculate the impacts of changes in freight and passenger costs on the wider economy. Economic sectors considered in the economic module correspond to the commodities that are needed in the Freight Demand module. At the beginning of each time period, ECR provides projections of inter-regional trade flows, the necessary input to the rest of the HIGH-TOOL modules, and at the end it calculates the impacts of transport policy measures determined by other modules on the GDP, employment and resource use.

The calculations of ECR are implemented as a set of sequential algebraic equations. The inputs and output of the module are shown in Table 5.

Table 5: Interaction of the Economy & Resources module with other HIGH-TOOL modules

I/O	Variable	Description	Dimensions	Module(s)	Name in database
In	*pop*tot	Population [persons]	time period *t*, country *ci*, gender *g*, age group *a*	Demography	o_de_pop
In	*LB*	Labour force [persons]	time period *t*, zone *i*, gender *g*, age group *a*	Demography	o_de_labour
In	*pkm*	Passenger transport mobility [pkm]	time period *t*, origin *i*, destination *j*, purpose *p*, mode *m*	Passenger Demand	od_pd_pkm_od o_pd_airic_pkm_od
In	*vkm*freight	Freight transport mobility [vkm]	time period *t*, origin *i*, destination *j*, mode *m,* commodity *c,*	Freight Demand	o_fd_vkm_od
Out	*GDP*	Gross domestic product [million EUR]	time period *t,* country *ci*	Passenger Demand and Freight Demand	o_er_GDP
Out	*jobs*	Number of working places [jobs]	time period *t,* country *ci*	Passenger Demand	o_er_emp
Out	*income*	Income [EUR]	time period *t,* country *ci*	Passenger Demand	o_er_income
Out	*T*econ	Trade flow [EUR]	time period *t*, origin *i*, destination *j*, mode *m*, commodity *c*	Freight Demand	o_er_trade

4.2.3 Vehicle Stock Module

The main task of the Vehicle Stock module (VES) is to convert passenger and freight demand into the vehicle fleet size. This fleet size is disaggregated by vehicle type *vt* and vehicle age cohort *ac*, which is important for emission and energy use calculations. VES covers 61 road mode vehicle types and 12 non-road mode vehicle types. The module uses the survival rate and depreciation coefficient to determine the number of vehicles that remain from the previous time period *t-1*, and estimates the number of vehicles being sold in time period *t* based on the difference between the predicted transport demand from the passenger and freight demand modules. The choice between different vehicle types *vt* to enter the market is thereafter estimated by a logit model that basically represents user's choices when purchasing a vehicle. The results of the fleet stock calculations are provided at NUTS-2 level for each time period *t*. VES also delivers forecasts of average fixed and variable generalized costs for each transport vehicle type *vt* as well as total tax revenues per country *ci* (NUTS-0 level).The inputs and output of the module are shown in Table 6.

Table 6: Interaction of the Vehicle Stock module with other HIGH-TOOL modules

I/O	Variable	Description	Dimensions	Module(s)	Name in Database
In	*vkm*pas	Passenger transport mobility [vkm]	time period *t*, origin *i*, destination *j*, purpose *p*, mode *m*	Passenger Demand	o_pd_vkm_od
In	*vkm*freight	Freight transport mobility [vkm]	time period *t*, origin *i*, destination *j*, mode *m*, commodity *c*	Vehicle stock, Environment	o_fd_vkm_od
Out	*cost*fix	Fixed vehicle costs [EUR/vkm]	time period *t*, mode *m,* country *ci*	Passenger Demand	o_vs_cstavggen_fix_vkm o_vs_cstavggen_fix_pkm
Out	*cost*var	Variable vehicle costs [EUR/vkm]	time period *t*, mode *m,* country *ci*	Passenger Demand	o_vs_cstavggen_var_vkm o_vs_cstavggen_var_pkm
Out	*stock*	Total number of vehicles	time period t, mode *m,* zone *i*, vehicle type *vt*, age cohort *ac*	Environment	i_vs_veh_stock

4.2.4 Passenger Demand Module

The Passenger Demand module (PAD) largely follows the classical “four-step approach” of transport demand modelling without the network assignment step, given that HIGH-TOOL does not incorporate physical networks. PAD computes trip flows at the level of origin-destination relations at NUTS-2 level, with Europe and neighbouring countries. It distinguishes four trip purposes (business, commuting, private and vacation) and four transport modes (rail, road, air, coach). The Passenger Demand module is not adequate at producing extensive forecasts on intra zonal trips and urban transport respectively in parallel to long haul trips to intercontinental destinations, so it is complemented by two additional sub-models: the urban passenger demand module and the intercontinental air passenger module. The main outputs of PAD are the origin-destination trip matrices by mode *m* and purpose *p*. These trip matrices focus on long distance transport modes and feed an additional sub-model in order to derive other transport indicators like passenger-kilometres and vehicle-kilometres. The inputs and output of the module are shown in Table 7.

Table 7: Interaction of the Passenger Demand module with other HIGH-TOOL modules

I/O	Variable	Description	Dimensions	Module(s)	Name in database
In	*pop*tot	Population [persons]	time period *t*, country *ci*, gender *g*, age group *a*	Demography	o_de_pop
In	*LB*	Labour force [persons]	time period *t*, zone *i*, gender *g*, age group *a*	Demography	o_de_labour
In	*GDP*	Gross domestic product [million EUR]	time period *t*, country *ci*	Economy & Resources	o_er_GDP

I/O	Variable	Description	Dimensions	Module(s)	Name in database
In	*jobs*	Number of working places [jobs]	time period *t*, country *ci*	Economy & Resources	o_er_empl
In	*income*	Income [EUR]	time period *t*, country *ci*	Economy & Resources	o_er_income
In	*cost*fix	Fixed vehicle costs [EUR/vkm]	time period *t*, country ci, mode *m*	Vehicle Stock	o_vs_cstavggen_fix_vkm o_vs_cstavggen_fix_pkm
In	*cost*var	Variable vehicle costs [EUR/vkm]	time period *t*, country ci, mode *m*	Vehicle Stock	o_vs_cstavggen_var_vkm o_vs_cstavggen_var_pkm
Out	*pkm*	Passenger transport mobility [pkm]	time period *t*, destination *j*, origin *i*, purpose *p*, mode *m*	Economy & Resources	o_pd_pkm_od o_pd_airic_pkm_od
Out	*pkm*	Passenger transport mobility by country [pkm]	time period *t*, country *ci*, mode *m*	Safety	od_pd_pkm_transit_safety o_pd_pkm_orig_safety o_pd_urban_pkm_ctry
Out	*vkm*pas	Passenger transport mobility [vkm]	time period *t*, destination *j*, origin *i*, purpose *p*, mode *m*	Vehicle Stock & Environment	o_pd_vkm_od
Out	*vkm*pas	Passenger transport mobility by country [vkm]	time period *t*, country *ci*, mode *m*	Safety	o_pd_vkm_orig, o_pd_urban_vkm_crty
Out	*T*pas	Number of passenger trips [trips]	time period *t*, destination *j*, origin *i*, purpose *p*, mode *m*	Safety	o_pd_airic_trips_od

4.2.5 Freight Demand Module

The Freight Demand module (FRD) consists of four components: generation/attraction, distribution, modal split, and a final conversion component that produces final outputs and derives mobility in terms of tonne-kilometres and vehicle-kilometres. The generation-attraction step is already performed by the ECR module (trade projections by origin-destination pair), provided in EUR (2010, constant values), and are transformed by FRD into tonnes using volume-density assumptions. The trade in origin-destination is distributed using a multinomial logit function distributed across the routes represented by multimodal chains (origin-destination flows routed through transhipment points) collected from the ETISplus database. The modal split component applies a multinomial logit function based on commodity types *c*, as well as cost and time parameters to estimate the modal shares in the legs of the multimodal chains. The cost functions and utilities in the modal split component are based on TRANSTOOLS v2 (Rich et al., 2009). Finally, conversion is applied to obtain transport performance in tonne-km and vehicle-km by O/D and transit. Table 8 shows the interaction of FRD with other HIGH-TOOL modules.

Table 8: Interaction of the Freight Demand module with other HIGH-TOOL modules

I/O	Variable	Description	Dimensions	Module(s)	Database name
In	*GDP*	Gross domestic product [million EUR]	time period *t*, country *ci*	Economy & Resources	o_er_gdp
In	*T*econ	Trade flow [EUR]	time period *t*, origin *i*, mode *m,* commodity *c,* destination *j*	Economy & Resources	o_er_trade
Out	*vkm*freight	Freight transport mobility [vkm]	time period *t*, origin *i*, mode *m,* commodity *c,* destination *j*	Vehicle stock, Environment	o_fd_vkm_od
Out	*vkm*freight	Freight transport mobility in a country [vkm]	time period *t*, country *ci*, mode *m*	Safety	o_fd_vkm_transit
Out	*tkm*	Freight transport performance [tonne-kilometre]	time period *t*, origin *i*, mode *m,* commodity *c,* destination *j*	Economy & Resources	o_fd_tkm_od

4.2.6 Safety Module

The aim of the Safety module (SAF) is to estimate the numbers of fatalities (and injuries) derived from transport, as well as associated social costs. The required input includes historic mobility (from the Database), predicted mobility (from the Passenger and Freight Demand modules), and user hypotheses on safety risk and safety risk causal factors. Risk is defined as the number of 'occurrences' (fatalities, injuries) per unit of mobility (in vehicle-kilometre or trips). The module distinguishes road and non-road modes, which are dealt with in different levels of detail. Road safety is treated most intricately since, besides fatalities, it also predicts the number of serious and slight injuries. The road mode is further split into car, truck, powered-two-wheelers, public transport, bike, and pedestrians. Regarding non-road modes, rail, air, short sea shipping, and inland waterways are considered. SAF presents results per country *ci* (NUTS-0) and time period *t* (in years). Table 9 shows how the module interacts with other HIGH-TOOL modules.

Table 9: Interaction of the Safety module with other HIGH-TOOL modules

I/O	Variable	Description	Dimensions	Module	Name in Database
In	*pkm*	Passenger transport mobility by country [pkm]	time period *t*, mode *m,* country *ci*	Passenger Demand	od_pd_pkm_transit_safety o_pd_pkm_orig_safety o_pd_urban_pkm_ctry
In	*vkm*pas	Passenger transport mobility by country [vkm]	time period *t*, mode *m,* country *ci*	Passenger Demand	o_pd_vkm_orig o_pd_urban_vkm_ctry
In	*T*pas	Number of passenger trips [trips]	time period *t*, origin *i*, destination *j,* mode *m,* purpose *p*	Passenger Demand	o_pd_airic_trips_od
In	*vkm*freight	Freight transport mobility in a country [vkm]	time period *t*, mode *m,* country *ci*	Freight Demand	o_fd_vkm_transit

4.2.7 Environmental Module

The main task of the Environment module (ENV) is to calculate fuel consumption and emissions for each vehicle type *vt*. The module produces estimates for CO_2 emissions as well as five other pollutants, i.e. CO, VOC, NO_x, SO_2 and PM2.5. Fuel consumption and emissions are calculated per origin country *ci* (NUTS-0 level) and thereafter disaggregated to zones *i* at NUTS-2 level. This disaggregation by zones *i* is based on the share of transport demand in each zone within a country *ci*. ENV receives input from PAD and FRD (mobility), as well as from VES (fleet size). The predicted transport demand is disaggregated by vehicle type *vt* and origin country *ci*, then fuel consumption and emissions are derived. Table 10 shows these interactions with other HIGH-TOOL modules.

Table 10: Interaction of the Environment module with other HIGH-TOOL modules

I/O	Variable	Description	Dimensions	Module(s)	Name in Database
In	*vkm*pas	Passenger transport mobility [vkm]	time period *t*, origin *i*, mode *m*, destination *j*, purpose *p*	Passenger Demand	o_pd_vkm_od
In	*vkm*freight	Freight transport mobility [vkm]	time period *t*, origin *i*, mode *m,* destination *j*, commodity *c*	Freight Demand	o_fd_vkm_od
In	*stock*	Total number of vehicles [vehicles]	time period *t*, zone *i*, mode *m*, vehicle type *vt*, age cohort *ac*	Vehicle Stock	i_vs_veh_stock

4.3 HIGH-TOOL Data Stock

Full documentation related to the HIGH-TOOL data stock and data exchange platform can be found in HIGH-TOOL Deliverable D3.1 "Documentation: Input Database for the HIGH-TOOL Model" (Kiel et al, 2014).

The Data Exchange Mechanism prepares an up- and download functionality of the basic datasets used in the HIGH-TOOL model, creating a communication channel with the user for downloading model results as well as feeding in specific user data. The data used (inputs and parameters) and generated (outputs) by the HIGH-TOOL individual modules is also stored in the Data Stock. The Data Stock is a separate module of the HIGH-TOOL model which provides access to the input and output parameters of each module.

The data stock is built with a PostgreSQL + PostGIS database. It hosts the HIGH-TOOL data stock and extra schemata to save user data and to give persistence to the results of model runs. For the version management of the various database files, TortoiseSVN, an open source version management software, is used.

The methodology followed for preparing the Data Stock is the methodology developed in the ETISplus project to build multidimensional models so that they are self-explanatory. This methodology identifies a set of rules for processing multidimensional data.

The database is built from dimensions and parameters:

Dimensions needed were first identified, and for each dimension, e.g. TIME, a table was defined (e.g. _TIME) to store the elements for that dimension (e.g. [2010, 2015, …, 2050]). This methodology allows adaption of the data stock to changes happening along the development of the HIGH-TOOL model (for example, the geographical coverage of the prototype was initially defined at NUTS-0 level, but changed to current NUTS-2 region level for the pre-final version).

Definitions of all the parameters came next. The parameters are sets of numerical factors representing the input and output data of the modules. For example, the economic activity per country and per year is described using GDP. Together all the GDP values per year and country form the GDP PARAMETER. The parameters are complemented by their unit and their source of information. As expected, they are also defined in terms of dimensions. For example, the GDP HIGH-TOOL output parameter was defined by time and country; hence it used the TIME and ZONE dimensions (to be found in the tables _TIME and _ZONE).

To create the database structure required for the HIGH-TOOL Data Stock, definitions of data parameters and dimensions were collected from module developers.

4.4 HIGH-TOOL Model Performance

Table 11 below provides a synthesis of the model performance. Figure 11 displays the relative weight of each module in relation to the total time consumption of a complete simulation in the HIGH-TOOL model.

The Economy and Resources module and the Freight Demand (together they account currently for the 72% of the time of a run) module are the two largest modules, followed by Passenger Demand and Safety modules. Demography, Vehicle Stock and Environment modules together just use 8.6% of total model run.

Figure 12 displays the requirements in terms of minimum RAM memory of each of the modules. Most demanding modules are logically those dealing with largest tables, like the Freight Demand (which models freight transport at the level of different commodities between NUTS-2 regions), the Vehicle Stock (dealing with more than 10 vehicle types and almost 30 fuel technologies), and the Economy and Resources module (including a large number of branches of the European and neighbouring economies).

Table 11: HIGH-TOOL model performance and memory requirement

Module name	ID	Iterations needed for a full 2010–2050 run	Estimated time per iteration (min)	Total time for module run (min)	RAM memory requirement (GB)
Demography	DEM	1	1	1	2
Economy	ECR	9	6,5	58,5	12
Passenger Demand	PAD	9	1,53	13,77	8
Freight Demand	FRD	9	3,25	29,25	14
Vehicle Stock	VES	9	0,5	4,5	12
Environment	ENV	1	5	5	10
Safety	SAF	1	10	10	6
HIGH-TOOL		**39**	**27,78**	**122,02**	**14**

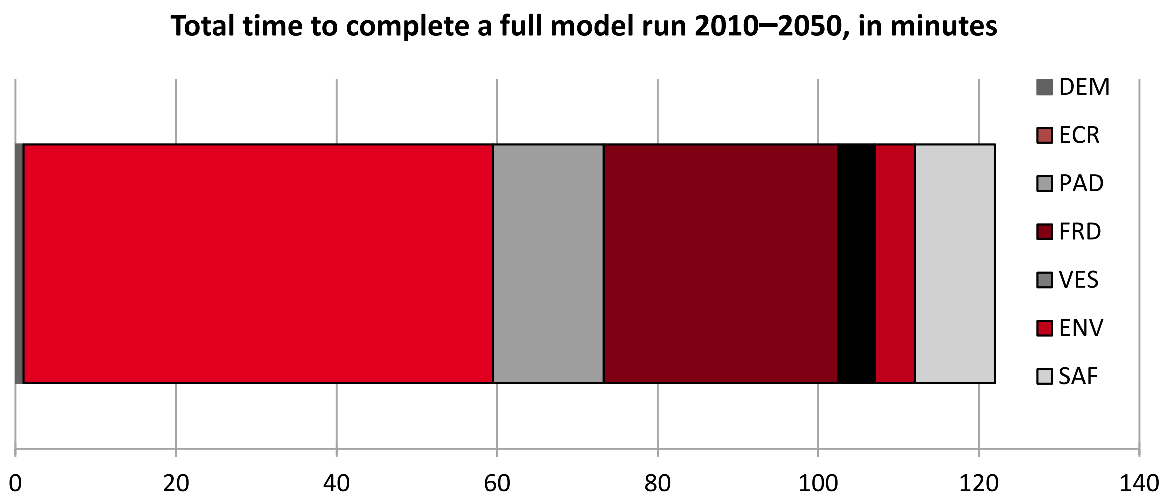


Figure 11: Total time to complete a full model run 2010-2050, in minutes by HIGH-TOOL modules

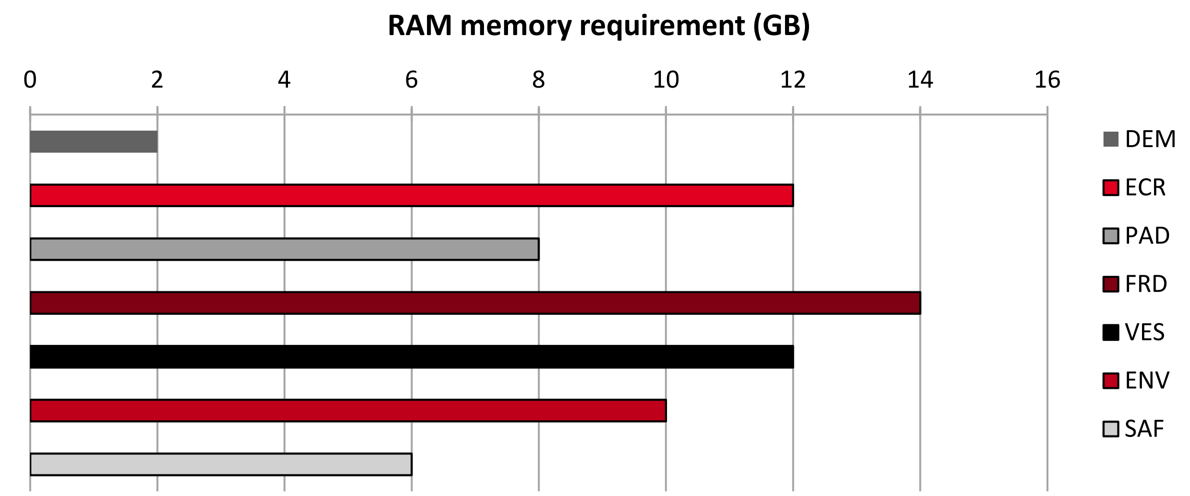


Figure 12: Minimum RAM memory requirements of the HIGH-TOOL modules. 32 GB are recommended for ensuring proper performance of the overall model.

All figures provided above have been determined in test runs performed on a system with the following specifications:

- Ubuntu Server 14.04.3 LTS AMD64 with Linux kernel 3.16.0-46-generic;
- Java version '1.8.0_60' 64 bits, -Xms8G -Xmx12G flags;
- AMD FX-8370 Eight-Core Processor with 16 GB of memory.

5 System Specifications

5.1 Overview

This chapter focuses on the description of technical specifications of the HIGH-TOOL model.

It describes the adopted software solution for programming model components, and for the User Interface, providing the main motivations behind decisions taken along its design process.

It provides the specification for a proper System Administration and other software solutions to increase the overall security of the model application. As a consequence of software choices, the resulting hardware requirements are also analysed in this section.

In brief, the software solution adopted to program a user-friendly web-based system which is compatible with all devices (PCs, tablets, mobile phones) is JavaScript. The User Interface makes use of specialised open source libraries, such as Angular.js (development framework), D3.js (generates data-driven documents), Three.js (webGL, graphic standards), Bootstrap (User Interface and interaction design), among others.

The HIGH-TOOL model is located in a remote server with Linux OS/Apache, interacting with the client's User Interface by RESTFul api. HIGH-TOOL databases are managed by PostgreSQR-PostGis. The server application is also programmed in JavaScript and mostly uses Node.js and Sails.js.

The server computer requires at least an Intel® Xeon® processor E5-1620 (3.60 GHz Turbo, 10 MB), with 32 GB 1600 MHz DDR3 ECC RDIMM memory and SSD hard drive (solid state disk). If the application is installed locally in a desktop computer, this machine will also require equivalent technical specifications in relation to performance.

When working remotely, client computers can be standard (as they will not undertake model calculations), but a reliable internet connection and 'evergreen' web browsers (latest versions of Chrome, Firefox or Internet Explorer/Edge) are a strong requirement

These main features are reported in more detail in the following sections of this chapter.

5.2 Analysis of Alternative Software Solutions

Market trends are generally converging towards the use of web technologies for applications such as HIGH-TOOL. Main reasons for this development are listed below:

- Applications are cross-platform, provided that developers restrict themselves to evergreen browsers and open standards.
- Easiness to update and distribute software.

- They can eventually be transformed into stand-alone applications if needed (using solutions like wrappers, web views).

There are also some pitfalls to overcome:

- Performance standards of web interfaces are usually worse than those developed using other technologies like Java, C/C++, C#… and UI/utility libraries (Qt, GTK, Swift).
- Web browsers are sandboxed due to security concerns (running programs from unverified third parties is restricted). Therefore, web applications can only use web APIs to operate with the browser, which in some cases may not be sufficient for all needed functionalities.
- Browsers are single-threaded. Multi-threading is limited to web workers with a simple messaging system to coordinate threads.

To deal with the second pitfall, most applications use tiers to divide responsibility. The most common solution used by web applications is the three tier architecture, with a presentation tier (that renders the User Interface), an application tier (that handles the business logic), and a storage tier (including data persistence mechanisms and a data access layer that encapsulates persistence mechanisms and exposes the data). This solution is also adopted in HIGH-TOOL.

It is common practice to use frameworks to shorten the development cycle. Frameworks are pieces of software that abstract common features from the low level implementation. The following frameworks have been considered when analysing the best software solution for HIGH-TOOL (see Table 12, Table 13 and Table 14). After evaluating their advantages and disadvantages, it has finally been decided to use AngularJS and SailsJS frameworks.

Table 12: Frameworks considered for the development of the HIGH-TOOL: client applications

Client Framework	Open Source	Programming Language	Main Features	Potential Disadvantages
Web2Py	Yes	Python JavaScript	- Web framework - Easy to run, requires no installation - Commitment to backward compatibility - Multiple database drivers - Speaks multiple protocols - Prevents most common types of vulnerabilities (Cross Site Scripting, Injection Flaws, Malicious File Execution) - Includes a web-based integrated development environment - Allows mapping via standard OGC services (Geoserver, Mapserver, …)	- Requires implementation of Python 2.5, 2.6 or 2.7 (not ported to Python 3) - Less popular than other frameworks (less resources and assistance available) - Monolithic framework

Client Framework	Open Source	Programming Language	Main Features	Potential Disadvantages
Django	Yes	Python JavaScript	- Web framework - Rich featured - Large community using it, increases completeness of resources - Allows mapping via standard OGC services (Geoserver, Mapserver, …)	- Monolithic framework
AngularJS	Yes	JavaScript	- Web framework - Client side web framework - Eases development of HTML5 Web applications - Promotes good practices like modularity, reuse of components and unit testing - Decouples the client side of the application from the server side - Decouples DOM manipulation from application logic - Dependency injection - Allows mapping via standard OGC services (Geoserver, Mapserver, …)	- Cannot be pre-rendered in the server to improve performance
OpenLayers	Yes	JavaScript	- WebGIS framework - Large community using it, increases completeness of resources - Allows mapping via standard OGC services (Geoserver, Mapserver, …)	- Need 3rd party sources to render data or install an OpenStreetMaps tile server - Library, not a full framework

Table 13: Frameworks considered for the development of the HIGH-TOOL User Interface: client/server applications

Server/Client Framework	Open Source	Programming Language	Main Features	Potential Disadvantages
Geomedia Webmap & Smart-client	No	.NET	- GIS platform - Good documented APIs and SDKs	- Subscription-based and proprietary
ArcGIS	No	Core in C++ and OpenGL. APIs and SDK in multiple languages	- GIS platform - Good documented APIs and SDKs	- Subscription-based and proprietary
Joomla	Yes	PHP JavaScript	- Web framework - CMS - Easiness in creating and editing web-based User Interfaces - Serves multiple protocols like HTML, JSON, RSS - Follow a MVC paradigm - Modify the database schema - Allows mapping via standard OGC services (Geoserver, Mapserver, …)	- Most functionality based on contents and user administration - Create modules require knowledge of the platform - Requires ad hoc API for mapping visualisation

Table 14: Frameworks considered for the development of the HIGH-TOOL User Interface: server applications

Server Framework	Open Source	Programming Language	Main Features	Potential Disadvantages
SailsJS	Yes	JavaScript	- Web framework - Client side web framework - 100% JavaScript - Waterline ORM, which abstracts Models to JavaScript objects - Auto-generate REST APIs - WebSocket Support - Reusable security policies - Front-end agnostic - Automation of tasks (using Grunt) - Allows mapping via standard OGC services (Geoserver, Mapserver, …)	- Less popular than Python frameworks

5.3 Architecture of the HIGH-TOOL Model

The HIGH-TOOL model User Interface has been programmed as a stand-alone online application based on AngularJS and SailsJS, both free and open source software components programmed in JavaScript (ECMAScript v6).

The architecture of the HIGH-TOOL User Interface is defined by:

- A web client application developed in AngularJS to run in web browsers, complemented with other libraries used to handle screen responsiveness and GIS visualization.
- A Node.js server application developed in SailsJS. It exposes a REST API, handles authentication and can modify the models either directly or via HIGH-TOOL modules.
- PostgreSQL + PostGIS database. Hosts the HIGH-TOOL data stock and extra schemata to save user data and to give persistence to the results of model runs.
- A Java Virtual Machine (JVM) capable of running Java v8 code. The SailsJS server application calls it to execute HIGH-TOOL modules, which then modifies the data stock. Via callbacks, the SailsJS server application controls execution order and replication of the results.

Figure 13 illustrates the architecture of the HIGH-TOOL model User Interface and its interaction with the HIGH-TOOL components and the data stock.

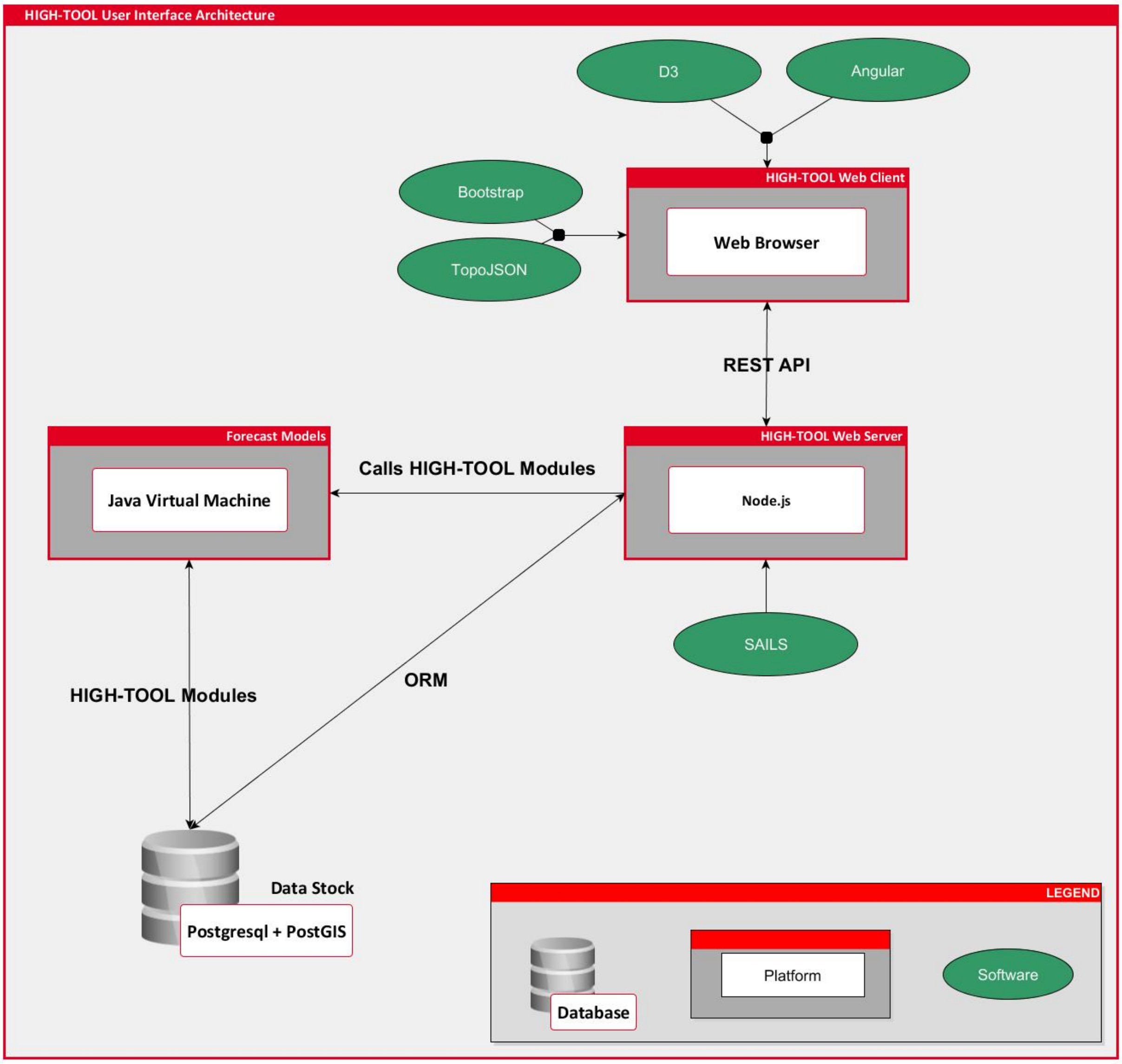


Figure 13: Software Architecture of the HIGH-TOOL User Interface and its interaction with the data stock and HIGH-TOOL components

5.4 Client Software Solution

A web client application developed in AngularJS is required to run in web browsers, complemented with other libraries used to handle screen responsiveness and GIS visualization.

AngularJS has been selected for the client application because it eases the development of HTML5 Web applications and promotes good practices like modularity, reuse of components and unit testing.

The main characteristics of AngularJS are:

- Based on plain 100% JavaScript objects, making code easier to test, maintain and reuse.
- Eliminates DOM manipulation with data-binding protocols, updating the screen view each time the model changes, and updating the model each time the view changes.
- Client-side form validation protocols allow declaring validation rules without having to write code.
- Built-in services are provided on top of XHR using third party libraries, allowing simplification of code by handling asynchronous return of data.
- Uses directives to create reusable components and hiding complex DOM structure, CSS, and behaviour. This allows separate focusing on what the application does or how the application looks.
- Works well embedded in other technologies, whereas many frameworks require full commitment.
- Is designed to be fully testable, encouraging behaviour-view separation. Comes pre-bundled with mocks and takes full advantage of dependency injection. It also comes with an end-to-end scenario runner which eliminates test flakiness.

The AngularJS framework in the HIGH-TOOL model User Interface is complemented with the following specialised libraries:

- Bootstrap, a mobile first responsive framework. Fastens the design of rich User Interfaces that scale from small mobile devices to full desktop screens (allows building of responsive web templates).
- D3js, a library used to manipulate documents based on data. It produces high quality HTML and SVG outputs from raw data (allows the creation of charts based on model data).
- Three.js, a lightweight cross-browser JavaScript library/API used to create and display animated 3D computer graphics on a Web browser (allows implementing 3D functionalities, applied in the HIGH-TOOL GIS application).

5.5 Server Software Solution

A Node.js server application developed in SailsJS has been adopted. It exposes a REST API, handles authentication and can modify the models either directly or via HIGH-TOOL modules.

Using SailsJS for the server application allows development of event driven, non-blocking IO applications. SailsJS is a Node.js Web framework built on Google's Chrome V JavaScript engine. The recommended Node.js version is 0.12.2. Some characteristics of SailsJS are provided next:

- Based on plain 100% JavaScript objects, making code easier to test, maintain and reuse.
- Provides a simple data access layer that works with most databases.
- Built on the familiar relational model, aimed at making data modelling more practical.
- Comes with blueprints that help jumpstart an application server without writing any code thanks to auto-generated REST APIs.
- Easy WebSocket Support translating incoming socket messages compatible with every route.
- Reusable security policies with role-based access control by default.
- Designed to be compatible with any front-end strategy (e.g. Angular, Backbone, iOS/ObjC, Android/Java, Windows Phone).
- Completely customisable asset workflow as it ships with Grunt. Supports already developed Grunt modules.
- Solid foundation thanks to being built on Node.js. Uses Express for handling HTTP requests, and wraps socket.io for managing WebSockets.

Both client and server components use Grunt, NPM and Bower to manage dependencies, environments (development, production, testing) and deployment. PostgreSQL and a JVM capable to run Java v1.8 are required to allow model and data stock interaction.

5.6 Data Stock Software Solution

The data stock is built with a PostgreSQL + PostGIS database. It hosts the HIGH-TOOL data stock and extra schemata to save user data and to give persistence to the results of model runs.

5.7 Multiplatform User Interface

The User Interface is compatible with all devices (desktops, tablets, mobile phones, see Figure 14). It provides responsive web templates which consider the following design features:

- Text reorganises dynamically with varying screen widths. This allows text to always be displayed without the need for horizontal scrolling, regardless of the device used.
- Menu entries reorganise themselves dynamically. When insufficient space is available for them to show horizontally, they fold in a vertical menu that can be visible or folded.
- Control buttons and scroll bars change their appearance based on the device being used, so that all functions remain usable even in small cell phone screens, e.g. sliders in the input editors change to +/- buttons when insufficient space to be displayed is available.

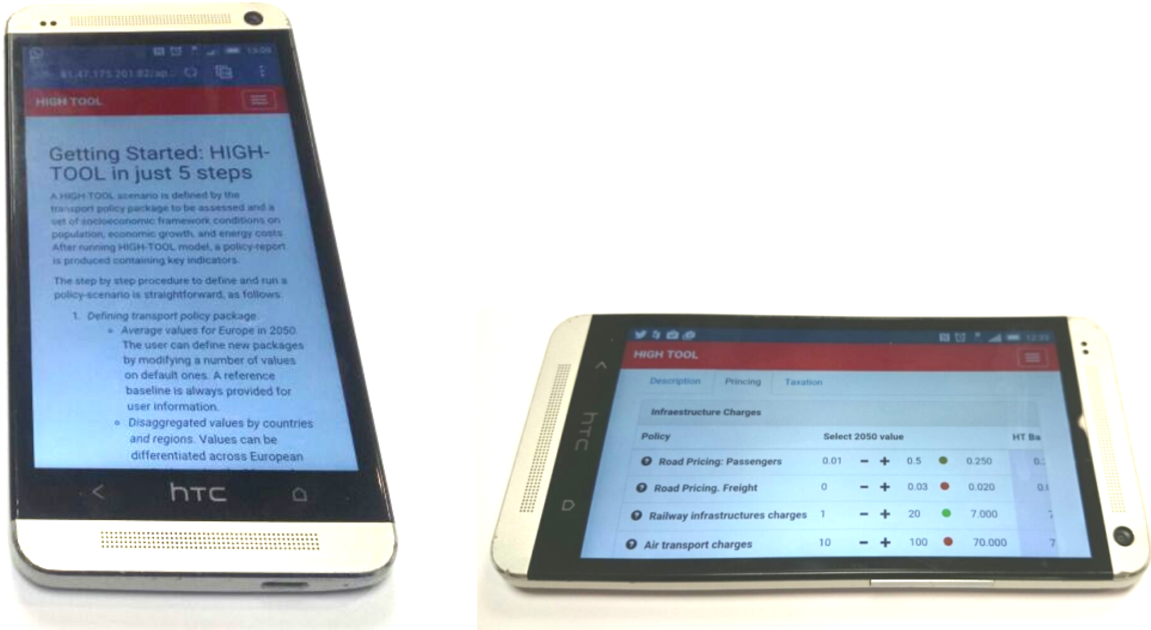


Figure 14: HIGH-TOOL being displayed on a portable device

5.8 Communications between the User Interface and HIGH-TOOL Modules

The SailsJS server application calls to execution the several HIGH-TOOL modules. Individual modules read inputs from and modify the data stock with outputs. Via call backs, the SailsJS server application controls execution order and replication of the results.

5.9 Security Solutions

Given that the HIGH-TOOL model will run on an Internet-connected server, it is important to envisage suitable security solutions to protect the application by preventing and mitigating undesired security threats. The communication between client and server is done via a RESTFul API secured with JSON-Web-Tokens. This strategy replaces session cookies allowing device agnostic communication with the server, Cross Site Request Forgery protection and improved user privacy.

In order for it to be a secure authentication and identification mechanism, JSON-Web-Tokens require HTTPS to prevent man-in-the-middle attacks, e.g. any machine between the client and the server can get the token from the HTTP header and use it to send requests to the server until the token expires.

5.10 System Administration and Management

Different user profiles and authorised roles need to be defined, access controls and passwords, as well as content administration (e.g. modelling results obtained may be confidential or public, or have a restricted access). The HIGH-TOOL User Interface is by default user and password protected. Only authorised users can access HIGH-TOOL.

The first time a user opens the interface, the web page will request an e-mail address and a password (see Figure 15).

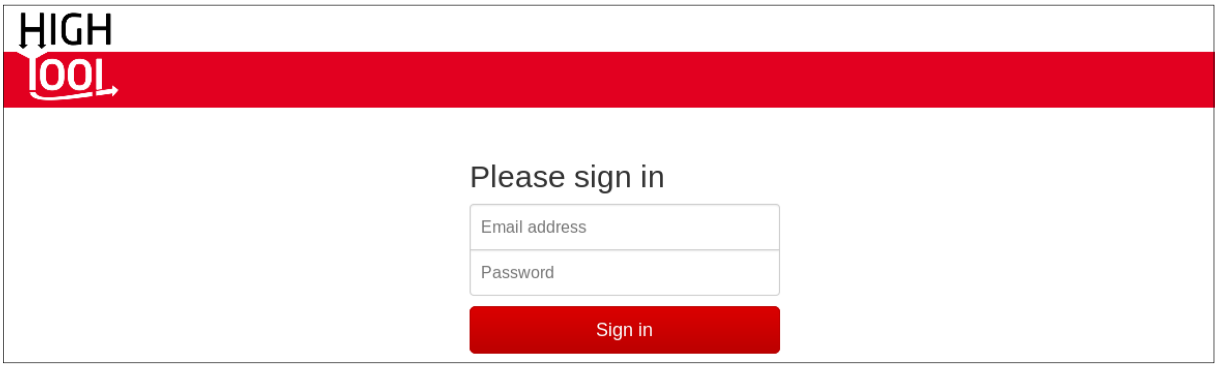


Figure 15: Login screen

User related options of HIGH-TOOL are displayed in the top right corner of the User Interface together with other assistance and help tools. It is envisaged that a Standard User will be able to contact the Administrator via an option button and to change its current profile. A logout option is also provided to the user in this menu.

Two different user profiles are defined in the HIGH-TOOL User Interface:

- **Standard Users**. They can create new packages (specification of Transport Policy Measures, Framework Conditions, and Scenarios) which can be edited, deleted and run. They may declare their packages as public so that any other Standard User can read them. A public package can be opened by other users, but cannot be deleted or modified unless it is saved as a new package (copy).
- **Administrators**. They can edit and delete any packages without taking into account their ownership. They manage the accounts of Standard Users: they can create new users and delete/ban exiting ones.

Contact with the system administrator is needed to obtain a user name and an access password.

5.11 Hardware Requirements

The HIGH-TOOL model is optimised to run smoothly on a Linux or Windows Server. As a reference, it consists of Microsoft Windows Server 2012 R2 64-bit equivalent to 8 x 1 GHz Pentium vCPUs with 32 GB RAM.

Hardware requirements in HIGH-TOOL are given by the operation of the different HIGH-TOOL model components rather than by the User Interface, which is envisaged as a simple resource efficient multiplatform component. Computational modules of the HIGH-TOOL model require that the HIGH-TOOL servers meet at least the following specifications:

- Intel® Xeon® processor E5-1620 (3.60 GHz Turbo, 10 MB);
- 32 GB 1600 MHz DDR3 ECC RDIMM memory;
- SSD (solid state disk) hard drive.

Both memory sets and hard drive technology are critical to ensure proper HIGH-TOOL performance. Given the large amount of data to be processed by HIGH-TOOL modules, ensuring a minimum quantity of RAM memory will allow the system to undertake calculations without the need to physically write in hard drives, while using SSD hard drives will allow performing of read/write operations at the beginning and the end of each module calculation within reasonable timings.

All the software components in the User Interface are open source and portable to the most popular operating systems. This avoids dependence on a specific technology, meaning:

- The client application will work on most environments, including Microsoft Windows, Mac OS, or Linux.
- The server software is currently based on Linux but works in other environments like Mac OS or Microsoft Windows.
- HIGH-TOOL software will also be able to run on stand-alone computers. Guidelines for installation will be provided.

Since HIGH-TOOL is a research project, the software developed will not have security audits and software certification yet. Therefore, it is recommended to install it on a separate computer and/or server environment.

6 User Guidance

This chapter is organised as a user guide, presenting how in practice, users will define assumptions and how a model's output will be presented. The main editors currently available in the User Interface are presented and their organisation is shown. Out of the user perspective the steps necessary to define scenarios respectively to make inputs in the model are listed.

6.1 Installation

Current version of the HIGH-TOOL model operates online with a connection to a remote server (http://81.47.175.201:82). Contact with the system administrator is needed to obtain a user name and an access password.

6.2 Hardware and Software Requirements

When working remotely, client computers can be standard (as they will not undertake model calculations), but a reliable internet connection and 'evergreen' web browsers (latest versions of Chrome, Firefox or Internet Explorer/Edge) are a strong requirement.

6.3 Getting Started

There will be three different ways for the user to define transport analyses in HIGH-TOOL:

Firstly, define and run a ***Single Transport Policy Measure (TPM)*** in the Baseline Framework Conditions (EU reference scenario). A set of pre-defined TPMs can be run individually upon the reference socioeconomic baseline conditions. For each TPM, model input variables representing the TPM and their reasonable impact range are provided. This is the simplest way to analyse the impact of transport policies: the user can go directly to this run option (from the run menu).

Secondly, define and run a ***Policy Package with Combined TPM***s (combining many policies) in the Baseline Framework Conditions (EU reference scenario). This second process is envisaged for the elaboration of more complex policy packages integrating several TPMs at once. The development of this option is ongoing and it is not yet implemented in the User Interface (as it requires previous full performance analysis of single TPMs).

Thirdly, define and run a ***User-Defined Policy Scenario***. It is based on a more general and strategic approach. The variables presented to the user in the User Interface when defining policies and framework conditions are not necessarily the input variables required by the model, but those variables more closely related to policies, and therefore the user's should be more familiar with them. An input translator converts these user variables into the kind of specific input variables HIGH-TOOL modules need to run. Users can also define Framework Conditions. This option is already available at the User Interface, but not yet connected with the modules (as it requires previous full performance analysis of single TPMs).

6.3.1 How to edit and run a Single Transport Policy Measure (TPM)

Single TPMs are always run upon the EU Reference Scenario, allowing users to simply and quickly launch runs of specific policy actions without complex input processes. At the end of the simulation, the HIGH-TOOL model returns a Policy Assessment Report containing all results in a standardised format.

In practice, users are required to undertake the following processes to define and assess TPMs:

- Selection of a TPM to be analysed. From the ***Run menu >> Pre-defined Transport Policy Measure in baseline conditions***. Identify a TPM throughout the interface menus.
- Modify values for TPM variables using the interface. Details on this step are reported in the following chapters 6.5 (variable magnitude), 6.7 (spatial distribution) and 6.8 (temporal dimension).
- Click on ***RUN*** command.

6.3.2 How to edit and run a Policy Package with Combined TPMs

This option is not yet implemented in HIGH-TOOL.

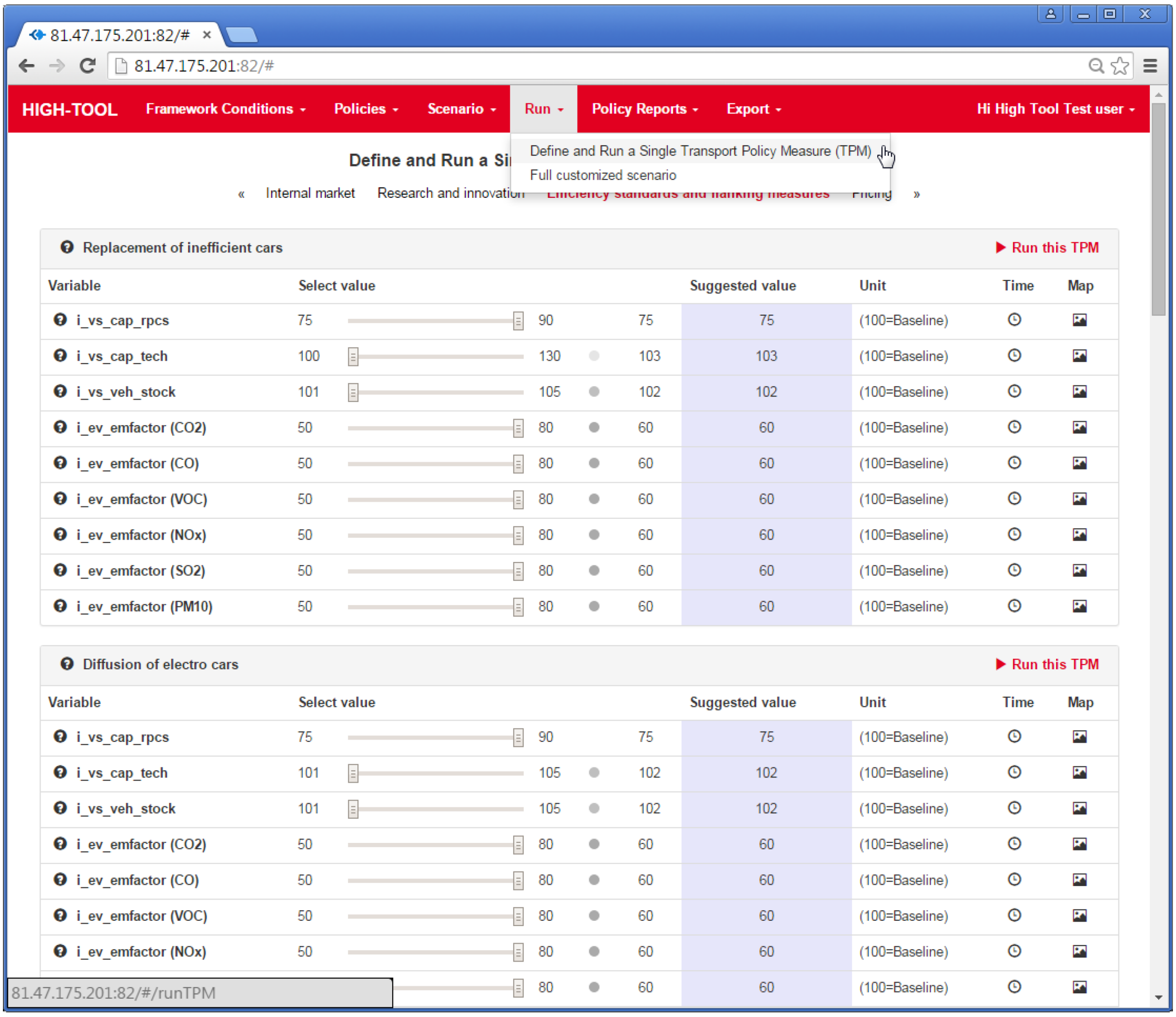


Figure 16: Running a simple TPM in three steps

6.3.3 How to edit and run a User-Defined Policy Scenario

In practice, users are required to undertake the following processes to define and assess transport policies:

- Definition of a **Policy set** (or loading an existing one). It contains specification of the transport policy actions, e.g. related to transport prices and taxation, transport regulation, or further integration of the single market.
- Definition of socioeconomic **Framework Conditions** (or loading an existing one). Under User-Defined Policy Scenarios, policies are always linked to a set of socioeconomic Framework Conditions. These contain assumptions on population, economy and energy costs. The EU reference scenario is therefore just one of many possible socioeconomic frameworks.

- Definition of a **Scenario**, which is by definition a Policy set coupled with a set of Framework Conditions.

At the end of a simulation, the HIGH-TOOL model returns a Policy Assessment Report containing all results in a standardised format. A Policy Assessment Report is composed out of two different components:

- A set of Synthesis Indicators summarising the simulation. The set includes the policy set specifications made by the user, framework conditions selected, and results. This way, results are always linked to the specifications and conditions under which the simulation was executed.
- A complete report in MS EXCEL format containing all results disaggregated by territories and modes. It is generated by the HIGH-TOOL model automatically and can be downloaded from the online application.

6.4 The User Interface Menus

The User Interface consists of the following seven menus:

- **Framework Conditions**: Contains all options related to the specification and management of Framework Conditions. This menu is mainly used under User-Defined Policy Scenarios (see chapter 6.6).
- **Policies**: Contains all options related to the specification and management of policies. This menu is mainly used under User-Defined Policy Scenarios (see chapter 6.5).
- **Scenario**: Defines the Scenario by matching a Policy sets to Framework Conditions. This menu is mainly used under User-Defined Policy Scenarios (see chapter 6.9).
- **Run**: Allows running either a single Transport Policy Measure (TPM) or a User-Defined Policy Scenario (see chapter 6.10).
- **Policy Reports**: Contains all options related to Policy Assessment Reports. Allows showing synthesis indicators of a specific run or downloading a full Policy Assessment Report. (see chapter 6.11).
- **Export:** Allows exporting raw data from tables contained in the HIGH-TOOL database (see chapter 6.12).
- **User Profile**: Allows logging in, logging out, managing users (only for administrator profiles), displaying help materials (getting started, user manual, FAQ, downloading reference reports). This menu is separated from other menus and displayed on the right side of the screen.

General actions in HIGH-TOOL menus are as follows:

- **new**: The user creates a new set of framework conditions, policy sets, or scenarios.
- **load**: The user selects and loads an existing set of framework conditions, policy set, or scenario or policy assessment report.
- **delete**: The user deletes an existing set of framework conditions, policy set, or scenario or policy assessment report.

Besides these main options, some menus contain particular options:

- **see baseline EU Reference Scenario**: This option is located in the Framework Conditions menu, and is intended to allow users to review the key features of the socioeconomic EU Reference Scenario. This reference scenario is used in HIGH-TOOL as a baseline for running single TPMs and Policy Packages as Combined TPMs, and is also provided as an optional framework condition to users for running User-Defined Policy Scenarios.
- **Run >> predetermined Transport Policy Measures in baseline conditions**: Directs users to running a single TPM.
- **Run >> full customised scenario**: Directs users to running a User-Defined Policy Scenario.
- **Export >> tables from databases:** Directs users to retrieving raw data from the HIGH-TOOL databases.

6.5 Defining Policies at Aggregated EU Level

The HIGH-TOOL model allows assessing different policy intensities. Maximum and minimum recommended values for all variables are provided. The structure of policy editors is similar to that of the Framework Conditions editor (see chapter 6.6). The following information and options are provided (see Figure 17):

- Identification of the Transport Policy Measure/policy field.
- A pagination bar groups input fields by different policy dimensions. The following dimensions are currently considered: Internal Market, Research and Innovation, Efficiency standards and flanking measures, Pricing.
- The next screen gives access to the definition of input fields. In the User-Defined Policy Scenarios option, the first tab opens a text editor allowing users to write and explain hypotheses behind their selection of parameters and variables (not available under simple TPM editor).

- Individual parameter and variable values can be specified using the options in the variable editor.

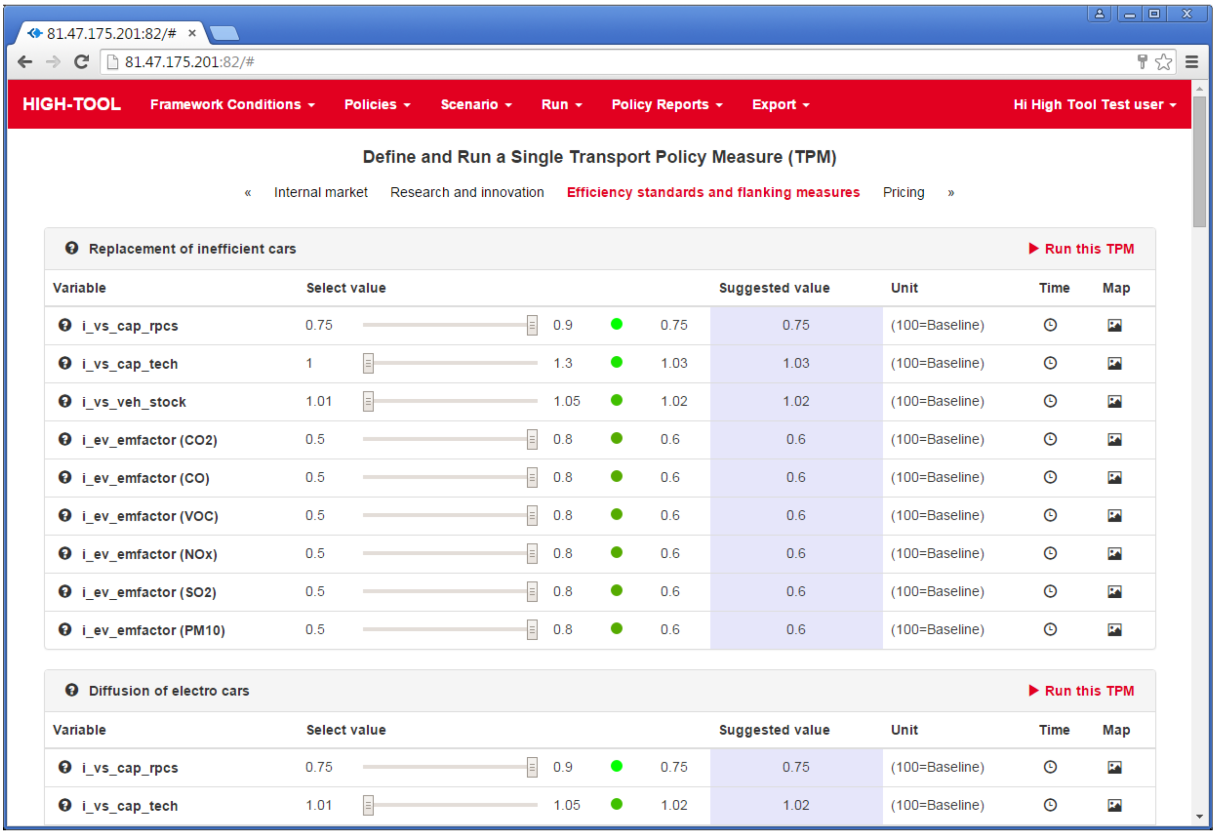


Figure 17: Editing a transport policy measure (user variables to be adjusted for the final version)

The variable editor is organised by rows, where each row corresponds to an input. The available options are, from left to right (see Table 15):

Table 15: Available options in the HIGH-TOOL model variable editor

Option	Interface Display
A help icon and the name of the input parameter or variable	Better use of truck capacity
A slider allowing the user to select a value for the input parameter in 2050. Moving the slider left or right will set out a new 2050 value. Minimum and maximum values are established for each particular input field.	 0.25 1
A colour semaphore. The colour of the semaphore is a linear gradient between the minimum and the maximum values for the input field. Dark tones indicate that the selected value for the input filed is close to the bottom range, whereas lighter tones indicate being close to the top range	

Option	Interface Display
A quantitative reference value for the input field as orientation for the user. This value corresponds to "baseline" policy intensity in line with the EU reference scenario.	Baseline 0.6
The unit of the indicator.	Truck load-factor
The TIME Editor icon allows determining the evolution or trajectory of the input field in the interval 2010–2050.	
The MAP Editor icon allows determining the geographic distribution of the input across Europe.	

6.6 Defining Framework Conditions at Aggregated EU Level

This step is mandatory before simulation of User-Defined Policy Scenarios, but not when editing and running single Transport Policy Measures (TPM).

When simulating a User-Defined Policy Scenario the magnitude of impacts will be different depending on the socioeconomic Framework Conditions considered (framework conditions remain inactive under the running of single TPMs). For instance, road taxation in Eastern European countries will provide different impacts on the transport system depending on the level of economic growth expected for concerned Member States.

Users can define different sets of Framework Conditions to check how policies impact differently under alternative socioeconomic contexts. Framework Conditions include elements of demography and economy and resources.

By default, HIGH-TOOL considers the EU Reference Scenario 2013, mostly based on EUROPOP2010 (Eurostat, 2014), the Ageing Report 2012 (European Commission, 2012) and Energy Trends 2050 (European Commission, 2013). These projections consider the years 2050 (in some cases 2060) as a temporal horizon.

After loading a set of Framework Conditions, the editor screen will show up. The structure of this page is similar to that of the Policies editor (see chapter 6.5). From top to bottom and from left to right the functionalities of the editor are as follows (see Figure 18):

- Identification of the **Framework Condition**.
- A pagination bar groups input fields by main the thematic areas **Demography**, **Economy**, **Energy** and **Trade**.
- A collection of pagination tabs associated with each of the thematic areas display all available options. The first tab is a **text editor** allowing users to write and explain hypotheses behind their parameter and variable selection. All other tabs allow the introduction of **parameter** and **variable values**.

Individual parameter and variable values can be specified using the options of the variable editor. The functioning of the editor is the same as in the editor of policies (see previous chapter).

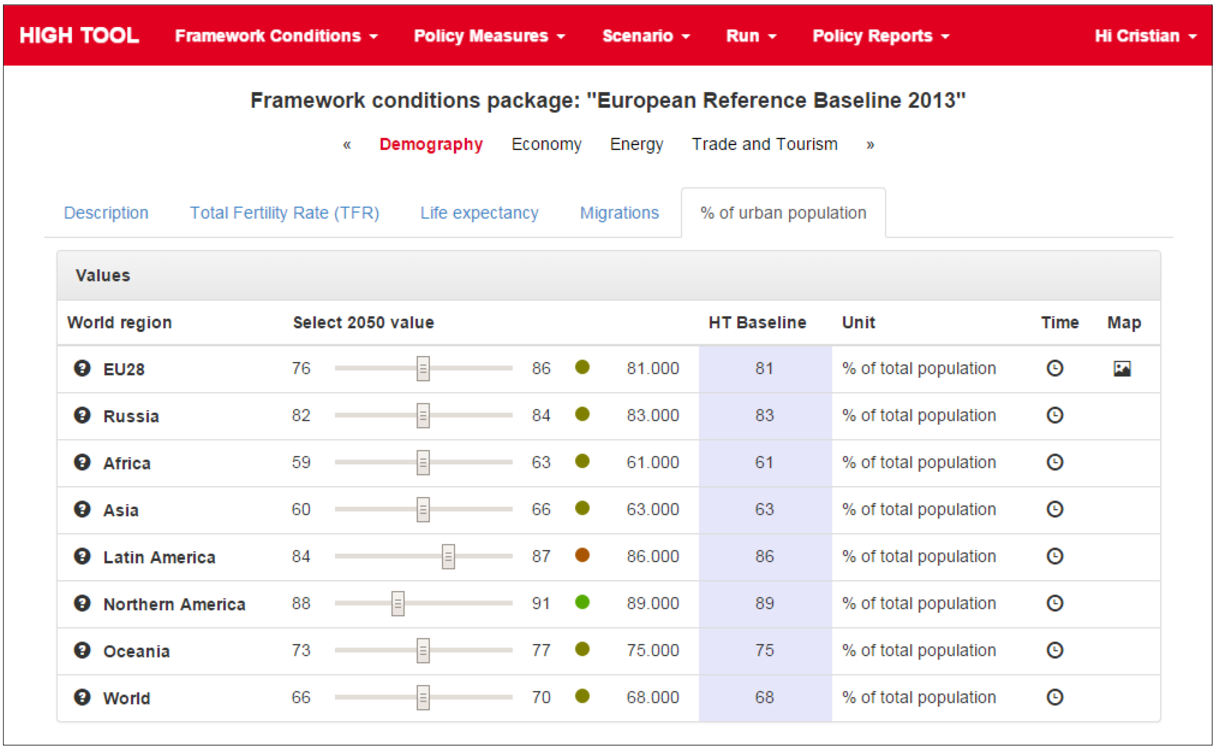


Figure 18: Editing a set of framework conditions

Framework conditions are defined for EU Member States, but also for the rest of the world. World regions are made available to establish hypotheses on e.g. world demographics or economics that drive parameters in HIGH-TOOL like European migrations or trade evolution. In the User Interface, a limited number of regions is provided as a logic input to the more detailed intercontinental bundles used by forecasting modules (see Table 16).

Table 16: World regions by intercontinental bundles used in HIGH-TOOL modules

Region identifier	Region name	ETIS Country_ID	ETIS Zone1_ID	ISO Country_ID and HIGH-TOOL bundle
1170000	Iceland	117	11700	IS
4000000	Africa Nord	400	40000	AFC_NORD
5000000	Africa Central and South	500	50000	AFC_SOUTH
6000000	Africa East	600	60000	AFC_EAST
7000000	Middle East Mediterranean	700	70000	MEA_MEDITERRANEAN

Region identifier	Region name	ETIS Country_ID	ETIS Zone1_ID	ISO Country_ID and HIGH-TOOL bundle
8000000	Middle East East	800	80000	MEA_EAST
10000000	Commonwealth of Independent States	1000	100000	CIS
11000000	Russia, east of Urals	1100	110000	RU_EAST
12000000	Asia/Pacific Indian Subcontinent	1200	120000	ASP_IND
13000000	Asia/Pacific Southern Asia	1300	130000	ASP_SOUTH
14000000	Asia/Pacific Australia/Oceania	1400	140000	ASC_AUS
15000000	Asia/Pacific Far East	1500	150000	ASC_EAST
16000000	America Canada	1600	160000	CDN
17000000	America USA	1700	170000	USA
18000000	America Mexico	1800	180000	MEX
19000000	America Central	1900	190000	AMC_CENTRAL
20000000	America Caribbean	2000	200000	AMC_CARIBBEAN
21000000	America South	2100	210000	AMC_SOUTH
22000000	Antarctica	2200	220000	ANC_ANTARCTICA

6.7 Definition of time evolutions 2010–2050 for input variables

The TIME editor will appear when clicking the icon for any parameter and variable available in the single TPM editor; the User-Defined Policy Scenario editor; and the Framework Conditions editor (see Figure 19).

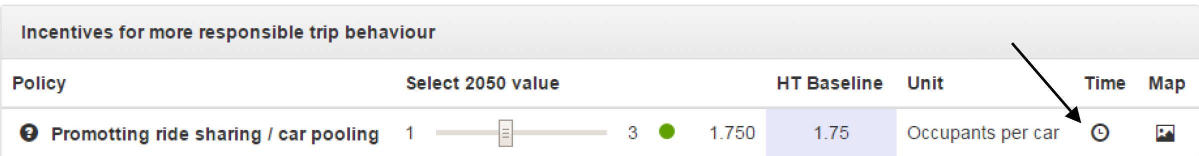


Figure 19: Accessing the TIME editor

The TIME editor allows users to specify values on EU level for a given indicator across the time horizon 2010 to 2050. In principle for each 5-year interval, values can be introduced. This way the policy implementation can be progressive as certain policies may be introduced stepwise. The path from the current situation in 2010 (e.g. current level of road investments) until the final policy level (year 2050) can be reflected (e.g. level of road investments in 2050).

By default, the editor proposes a linear trajectory between 2010 and 2050 but the user can modify this trajectory incorporating intermediate values in the trajectory (see Figure 20). Where available a reference trajectory is provided in grey, representing an average evolution of the policy identified with the EU Reference Scenario.

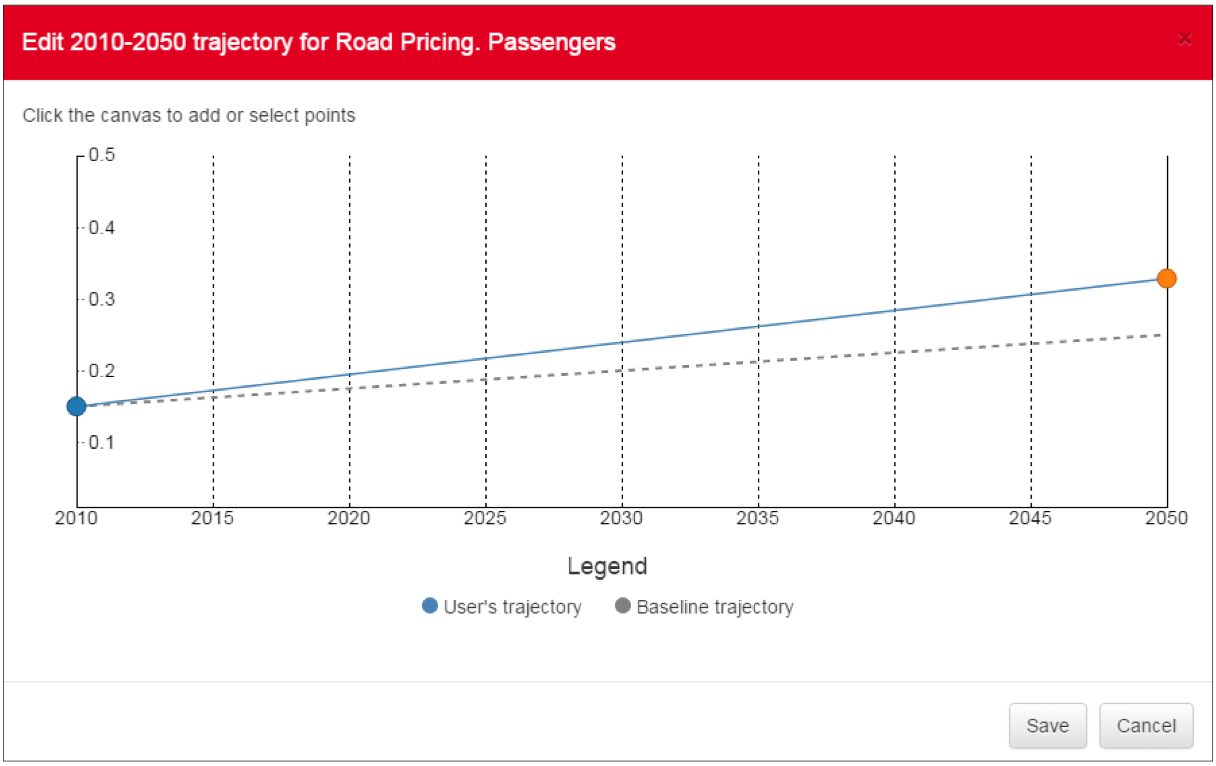


Figure 20: Time evolution of a parameter 2010–2050

The user can add points to the canvas by clicking on it (see Figure 21). After adding a point, the application interpolates using cubic spline values and slopes.

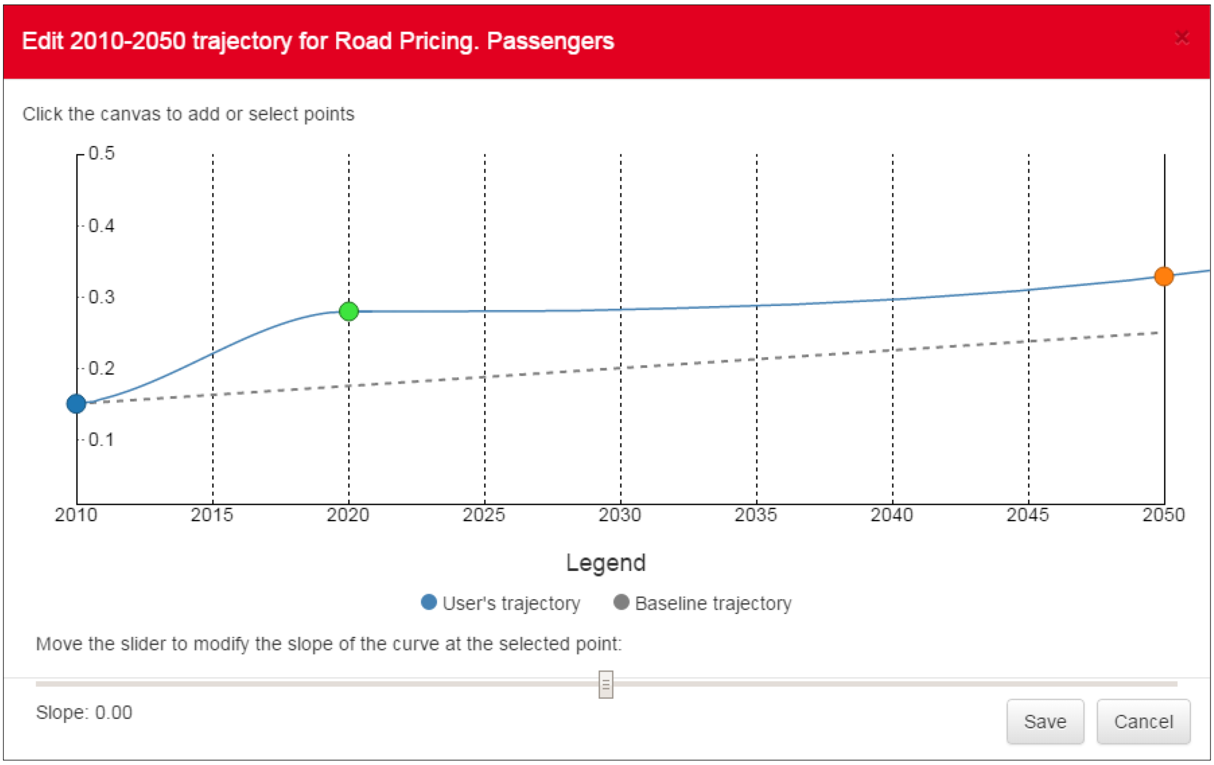


Figure 21: Editing the time evolution of a parameter 2010–2050 by introducing intermediate points

6.8 Definition of Input Variables at Regional NUTS-2 Level

The map editor will appear when clicking the icon for any parameter and variable available in the single TPM editor; the User-Defined Policy Scenario editor; and the Framework Conditions editor (see Figure 22).

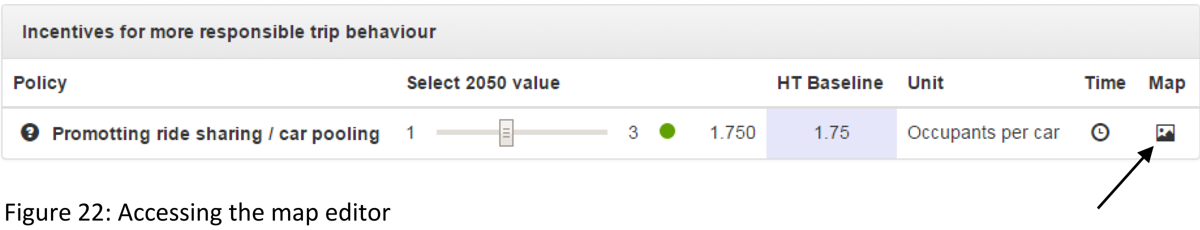


Figure 22: Accessing the map editor

By using the map editor, values can be particularised at a regional level for the different regions in Europe. For example, the level of road investments may be set higher in Eastern European countries in comparison with Western European countries.

After clicking the map editor icon, the application canvas opens showing a map of Europe (see Figure 23). By default all European territory will be displayed equally meaning that the input parameter or variable is being considered uniformly applied across Europe. Therefore the colour is homogeneous, reflecting an equal intensity.

As some policies might be more or less appropriate for some Member States, selectors of the dialog can be used to change the distribution of the policy based on different criteria (see Figure 24):

- Homogenous distribution (default setting);
- Proportional/inversely proportional to population density;
- Proportional/inversely proportional to income per capita.

For example, the intensity of road pricing can be set more intense in wealthy territories while less intense in lagging ones. Another possibility is to set CO_2 taxation more intense in more densely populated areas in Europe and less intense in sparsely populated areas). Territorial distributions of input parameters can be automatically defined based at NUTS-0 or NUTS-2.

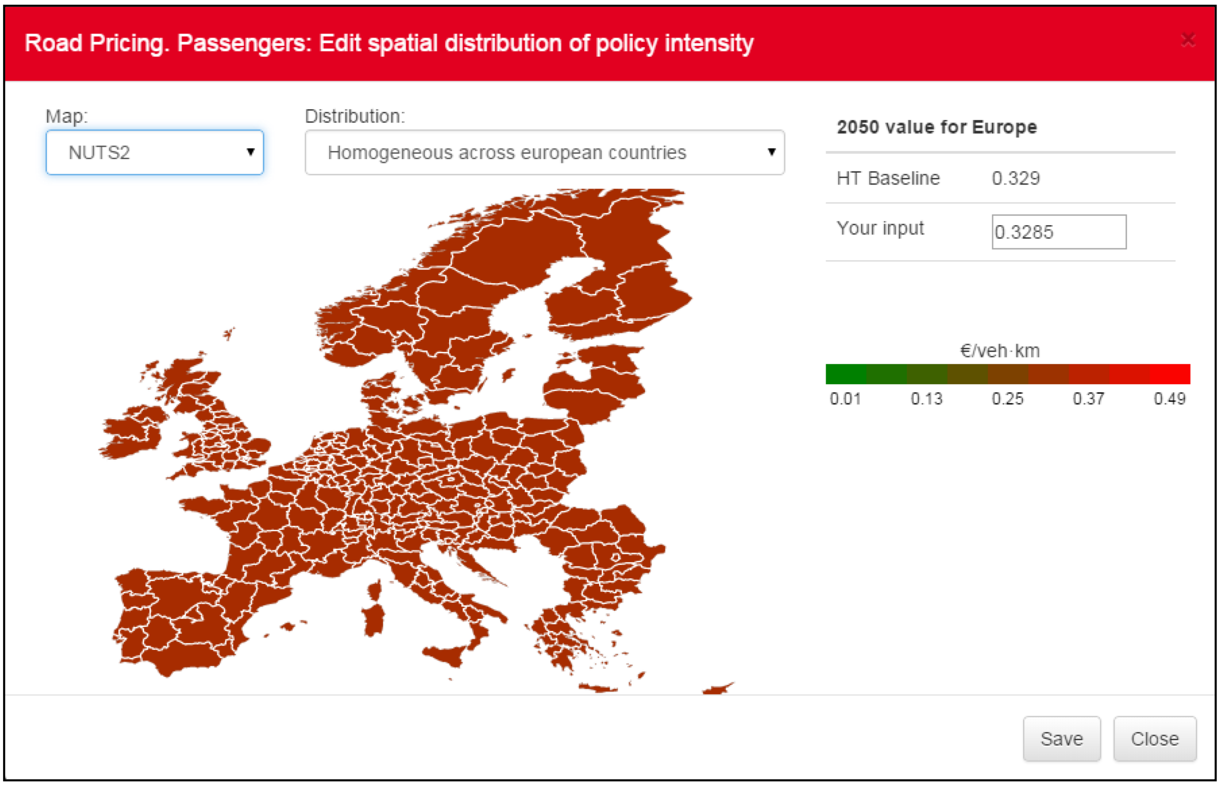


Figure 23: Geographic distribution of a parameter (homogeneous)

Users can also manually change a specific region's value by clicking on the region and changing its input parameter on the right side of the map editor ("Your input" option, see Figure 25). While hovering the mouse over a region, the region name and parameter value in 2050 will be displayed. Double clicking an area of the map allows zooming in for better visualisation of territorial detail.

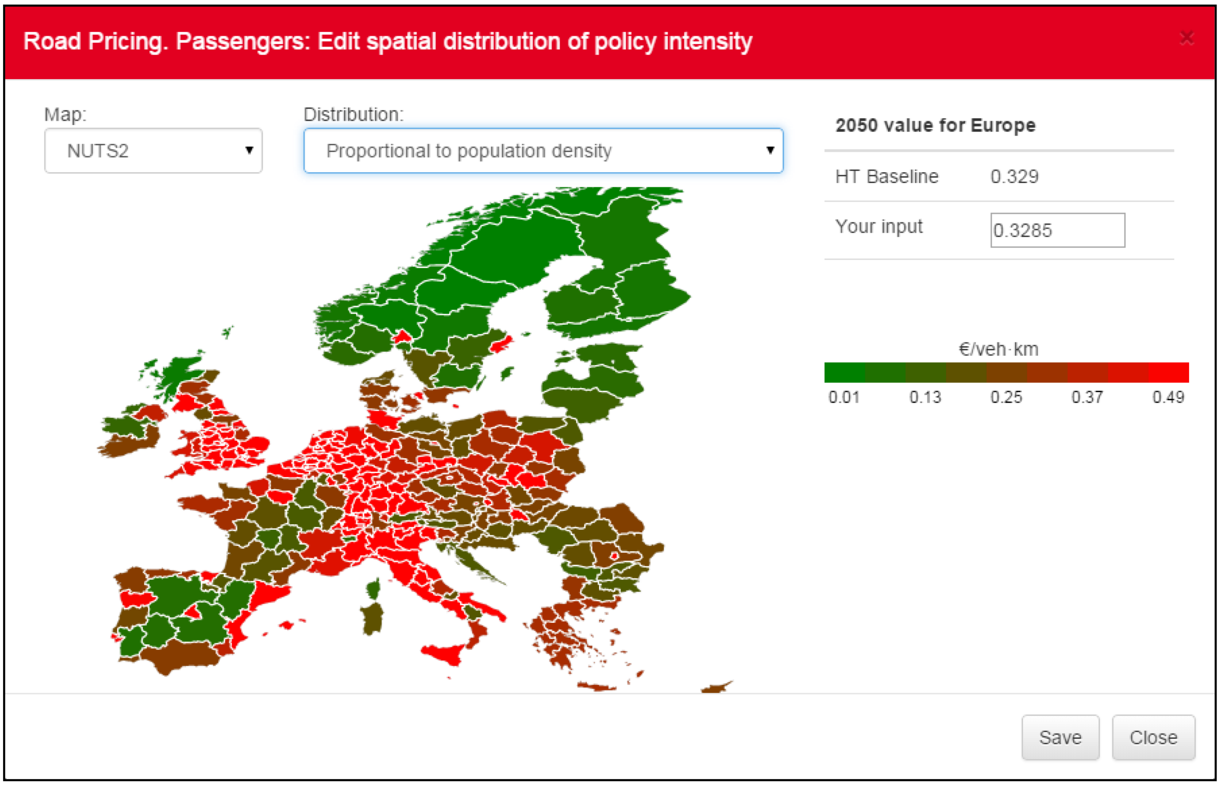


Figure 24: Geographic distribution of a parameter (proportional to population density of NUTS-2 regions)

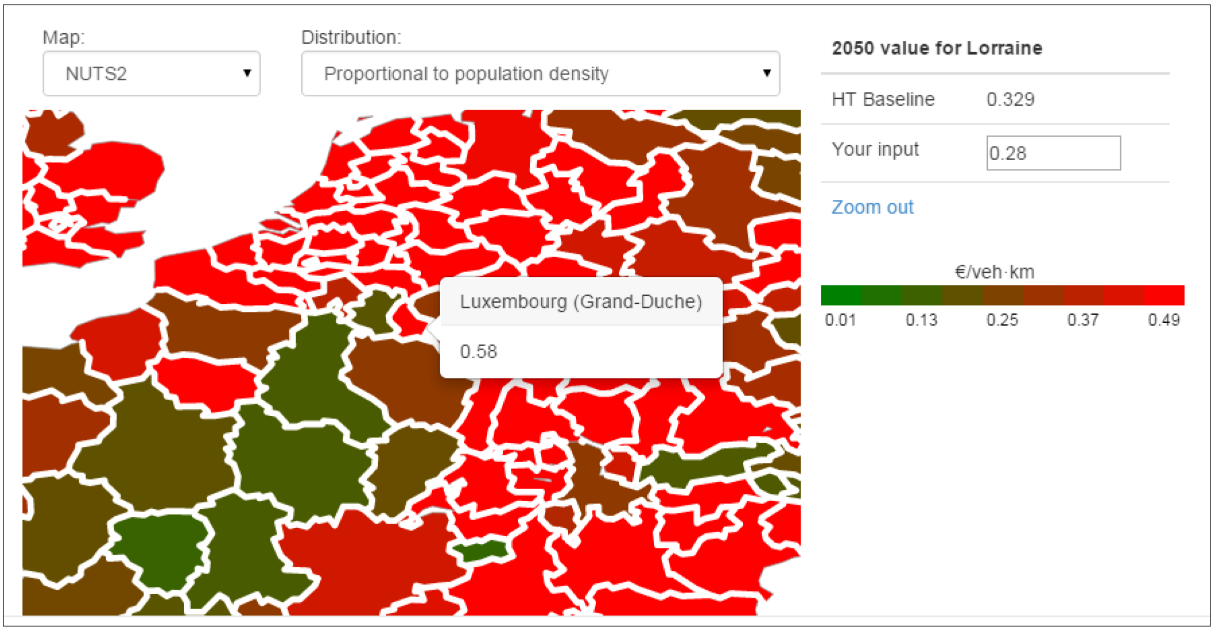


Figure 25: Manual modification of a value for a specific region

6.9 Creating a User-Defined Policy Scenario: Matching Policy Sets with Framework Conditions

This step is mandatory before simulation in User-Defined Policy Scenarios, but not when editing and running single Transport Policy Measures (TPM).

A Scenario consists of a Policy set and a set of accompanying socioeconomic Framework Conditions: The interface for this section is very simple and contains only the minimum amount of options to allow users to define Policy Scenarios:

- A **Framework Conditions selector**: It allows users to select a previously developed package, defining it as the socioeconomic framework under which the simulation will run.
- A **Policy set selector**: It allows users to select the transport policy measure to be assessed from all those defined in previous steps.
- A **text editor**: Users can describe the aim of the simulation.

Figure 26 illustrates how a User-Defined Policy Scenario is built up. In the example, the user has selected reference framework conditions (EU Reference Scenario) with a transport policy exploring pricing and taxation measures. To document the run, the user has included the description of the package and the impacts that he expects from it in the initial box.

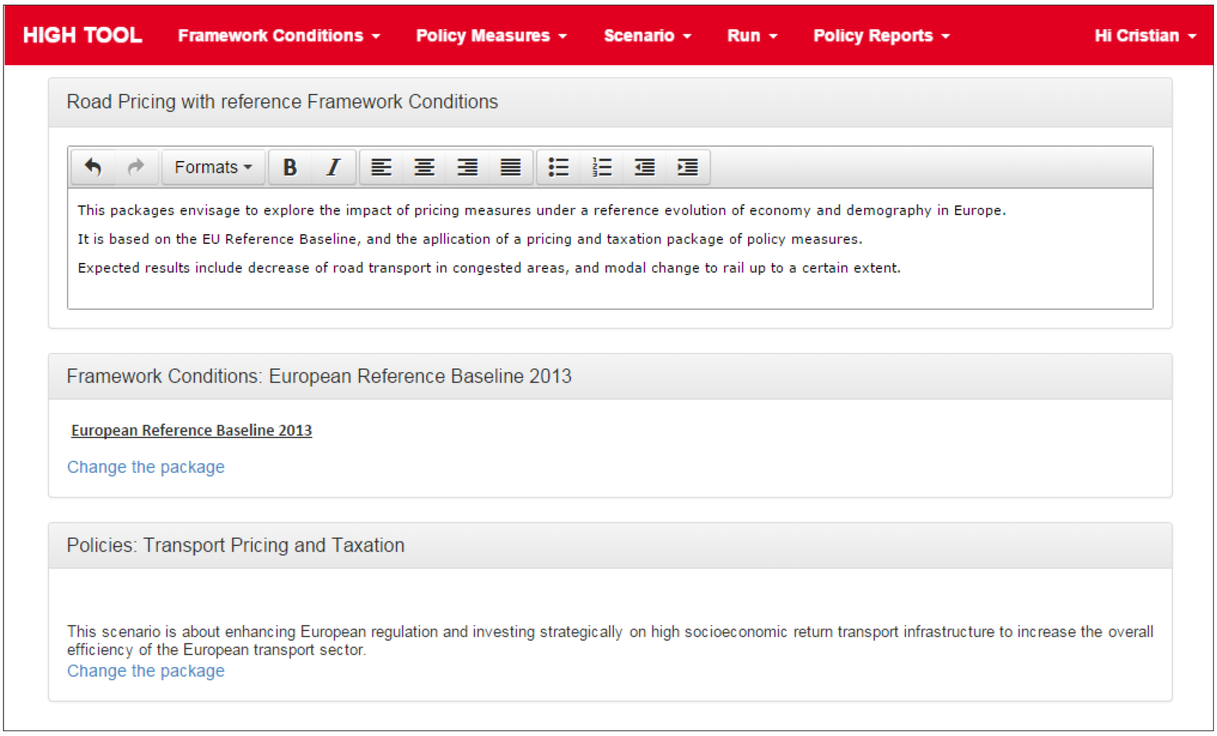


Figure 26: Definition of a Scenario to simulate a Transport Policy Package

6.10Running HIGH-TOOL

6.10.1 Running a Single TPM on Baseline Conditions

To edit and run a single TPM, users will select the option from the menu ***RUN >> Pre-defined Transport Policy Measure in baseline conditions***.

Once the TPM panel appears, users need to select a single TPM and edit its input variables (impact ranges are provided). Once this process is finalised, users will start the model by clicking on the red command ***RUN*** located on the right hand side of the TPM caption (see Figure 27). The system will then begin calculations and send an alert to the user once results become available.

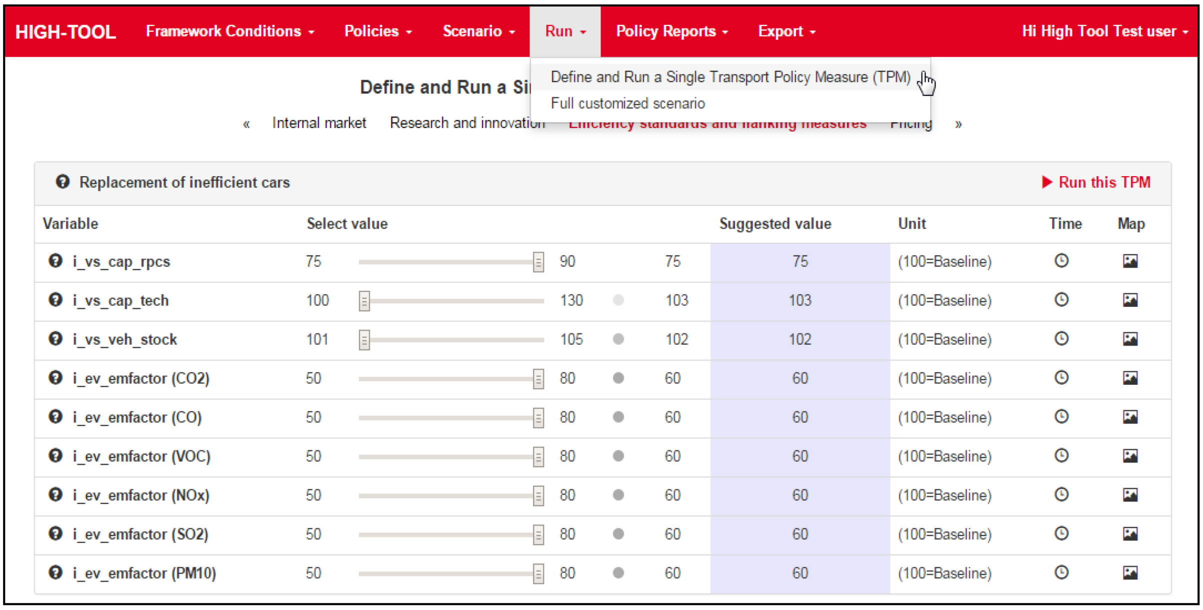


Figure 27: Running a single TPM

6.10.2Running a User-Defined Policy Scenario

To run a simulation of a User-defined Scenario, users will select the option from the menu ***RUN >> Full customised scenario.***

In the final version of the User Interface there might be different options to run a simulation with different computational and time efforts (e.g. consistency check, partial runs, full runs). Given that most HIGH-TOOL components are interlinked, partial runs are only possible for a number of defined combinations and will therefore only be considered if these combinations perform substantially better than full runs.

Specific options in the Run menu will be defined during the next weeks once the HIGH-TOOL modules are tested sufficiently and real computing times for HIGH-TOOL can be better apprehended. Ideas presented are conceptual approaches to possible solutions, as the module is under development.

Currently, the User Interface ***Run a customised scenario*** menu is just a mock up depicting such features. In any case, the user will have to select a Scenario and the kind of run envisaged.

Clicking the ***Run*** button will initiate the HIGH-TOOL model and when terminated, save results and produce the corresponding Policy Assessment Report. While the HIGH-TOOL model investigates the scenario, the user is allowed to work on other tasks simultaneously (e.g. specification of other Transport Policy Measures). An alert is sent to the user by e-mail once the run is completed.

6.11Policy Assessment Report

The full simulation results are available in a report on MS EXCEL format, downloadable from the server through the User Interface. This report is generated automatically by the HIGH-TOOL model User Interface and designed as an interactive Excel report. Tables, graphs or single values can be imported to a presentation tool (MS PowerPoint) or a text editor (MS Word, Open Office) by using the ordinary copy and paste functionalities.

The full Policy Assessment Report contains the following elements:

Contextual Information including the name of the Scenario and its abstract, as well as the names of the Transport Policy Measures being simulated and the underlying socioeconomic Framework Conditions.

Synthesis of assumptions and results. Contains synthesis information related to a transport policy measure applied, the corresponding socioeconomic framework conditions, and the main impact results obtained after the termination of the model run. The indicators displayed are identical with the ones on the User Interface.

Full Results by thematic area. Results of each thematic area (Demography, Economy and Resources, Passenger Demand, Freight Demand, Vehicle Stock, Environment, Safety) produced by the HIGH-TOOL model are presented by tables and charts, each on a separate page.

The following sequence of figures illustrates the assessment report configuration (Figure 28 to 31).

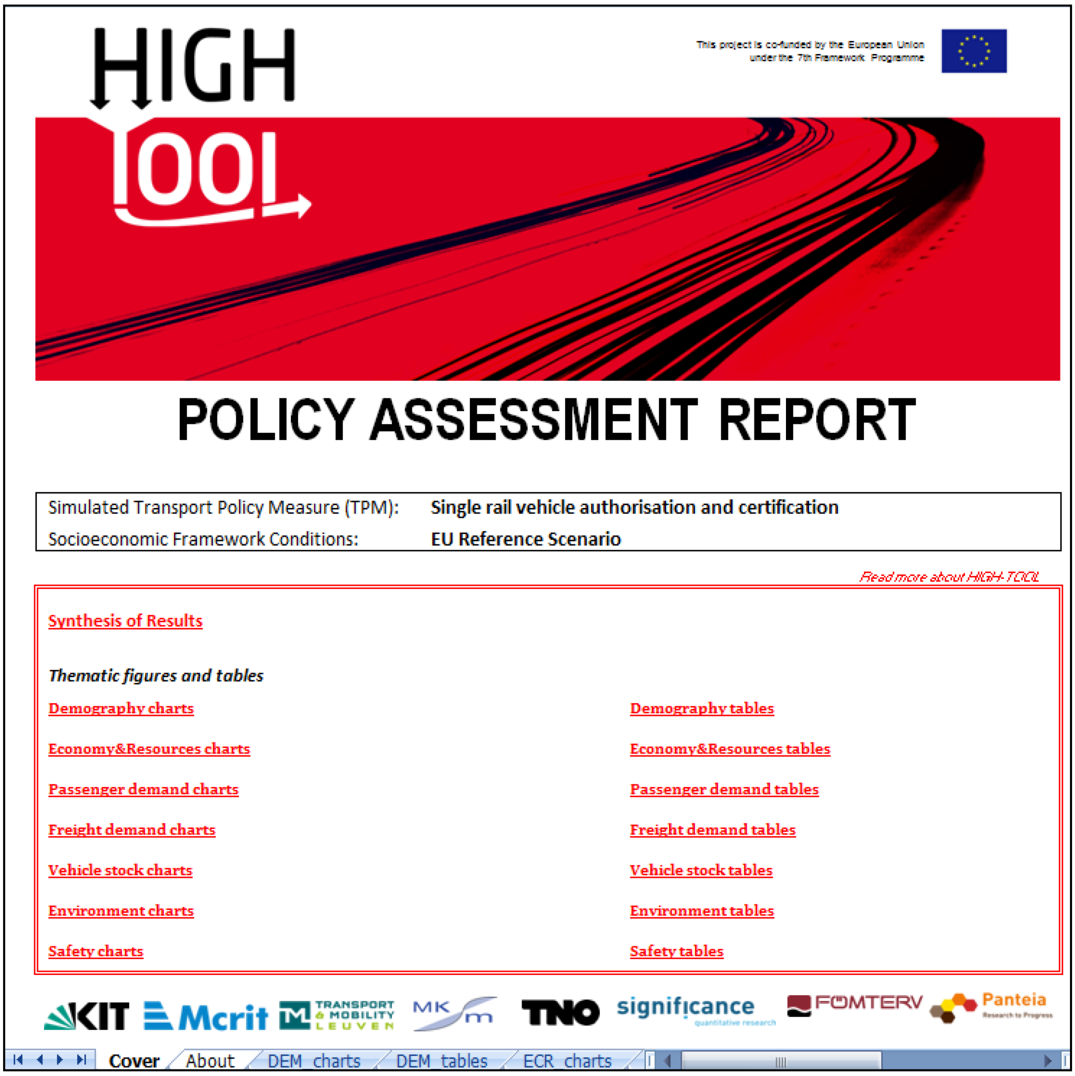


Figure 28: Main menu of the Policy Assessment Report in MS Excel

Policy Assessment: Pathways to fullfill European targets

Return to Main Page

	source	Reference Baseline			Regulation Scenario		
Framwork Conditions		**2015**	**2030**	**2050**	**2015**	**2030**	**2050**
European population (EU28) *(million inh.)*	user	507,2	518,8	525,5	507,2	518,8	525,5
Immigration *(immigrants / year)*	user	753013	1244100	1188300	753013	1244100	1188300
Labour force size *(million inh.)*	user	325,1	344,5	361,0	325,1	344,5	361,0
GDP growth *(% yearly average)*	user	1,10%	1,40%	1,45%	1,10%	1,40%	1,45%
Growth of international trade *(% yearly average)*	user	3,90%	3,37%	3,61%	3,90%	3,37%	3,61%
Growth of international travel (business, private, leisure) *(% yearly average)*	user	3,89%	4,38%	3,81%	3,89%	4,38%	3,81%
Energy: Oil price *(constant US$ 2005)*	user	108,7	145,0	160,0	108,7	145,0	160,0
Transport Policies		**2015**	**2030**	**2050**	**2015**	**2030**	**2050**
Average infrastructure pricing *(2015=100)*	user	100,0	190,7	238,0	100,0	210,4	264,9
Average vehicle and fuel taxation *(2015=100)*	user	100,0	106,4	112,3	100,0	113,0	119,5
Average subsidies to transport services (change)	user	100,0	88,6	68,6	100,0	90,4	71,4
Total infrastructure investment (% of GDP)	user	1,22%	1,28%	1,15%	1,22%	1,47%	1,32%
Cost of internal market fragmentation (regulation, standards, procedures...) *(2015=100)*	user	100,0	98,4	96,2	100,0	88,8	86,8
Maximum admited emissions from new vehicles (eurostandards) *(2015=100)*	user	100,0	53,9	35,2	100,0	53,9	35,2
Average vehicle capacity use in % (load factors) *(2015=100)*	user	100,0	110,9	125,9	100,0	110,9	125,9
Scenario Results		**2015**	**2030**	**2050**	**2015**	**2030**	**2050**
Passenger transport performance (Bn pkm)	model	5194	5481	5481	5194	5063,2	5037,4
Freight transport performance (Bn tkm)	model	2546	3245	4641	2546	2997,4	4265,2
Total aggregregated cost of the transport system (generalised costs) (Bn €)	model	375	628	859	375	631,6	866,3
Cost generated by inefficiencies (e.g. Congestion, market fragmentation unefficient prices ...)	meta-model	35%	32%	30%	35%	29%	27%
Energy consumption by the transport sector (Bn TOE)	model	470.114	586.884	604.820	470.114	542.184	555.906
Emissions of particulates (PM) (1990=100)	model	101,0	119,5	114,9	101,010101	110	106
Network extension (high speed rail network) (2010=100)	user	110	170	240	110	303	428
CO2 urban logistics *(% urban distribution served with RES)*	meta-model	5,0%	35,0%	85,0%	5,0%	35%	85%
Airport and port connections *(% of core airports&ports connected to rail)*	user	15%	25%	40%	15%	28,75%	46,00%
Rail Modal split - passengers for long-distance	model	8,8%	9,7%	11,3%	8,8%	29,74%	35,34%
Rail Modal split - freight for long-distance	model	15,8%	15,5%	15,1%	15,8%	31,61%	31,50%
Share of RES in transport	user	1,22%	1,16%	1,21%	1,22%	10,00%	20,00%
CO2 emissions from transport (1990=100)	model	102,0	64,1	51,6	102,0	59,2	47,4
Transport fatalities (total)	model	23.927	7.752	2.924	23.927	5.814	2.193

Cover / About / Main_interface / Synthesis Page / S_Reference / S_Regulation / S_Tecnology / Targets / TARGET DB / aux / EU2020 / d_charts / d_tables / e_charts / e_tables

Figure 29: Synthesis of key indicators compared to reference scenario

HIGH TOOL

Go to main page

DEMOGRAPHY CHARTS

European population by gender (in million)

Year	Male	Female	Total	*% Male*	*% Female*
2010	338	332	670	50,4%	49,6%
2015	332	349	681	48,8%	51,2%
2020	336	353	689	48,8%	51,2%
2025	340	356	696	48,9%	51,1%
2030	342	358	700	48,9%	51,1%
2035	344	359	703	48,9%	51,1%
2040	344	359	703	48,9%	51,1%
2045	345	359	704	49,0%	51,0%
2050	344	357	701	49,1%	50,9%

European population by age cohort (in millions)

Year	0-19	20-39	40-59	=>60	% (0-19)	% (20-39)	% (40-59)	% (=>60)
2010	153,5	189,0	184,1	142,3	23,0%	28,3%	27,5%	21,3%
2015	151,5	184,9	188,0	155,2	22,3%	27,2%	27,7%	22,8%
2020	151,3	178,5	189,7	168,7	22,0%	25,9%	27,6%	24,5%
2025	150,8	170,9	189,6	183,4	21,7%	24,6%	27,3%	26,4%
2030	147,5	165,6	188,5	197,5	21,1%	23,7%	27,0%	28,2%
2035	143,4	163,7	185,2	209,5	20,4%	23,3%	26,4%	29,8%
2040	140,5	163,6	179,5	219,4	20,0%	23,3%	25,5%	31,2%
2045	139,1	162,9	172,5	227,9	19,8%	23,2%	24,6%	32,4%
2050	138,5	159,4	167,6	234,5	19,8%	22,8%	23,9%	33,5%

Cover / About / DEM_charts / DEM_tables / ECR_charts / ECR_tables / PAD_charts / PAD_tables / FRE_c

Figure 30: Full results displayed in tables, on a per theme basis

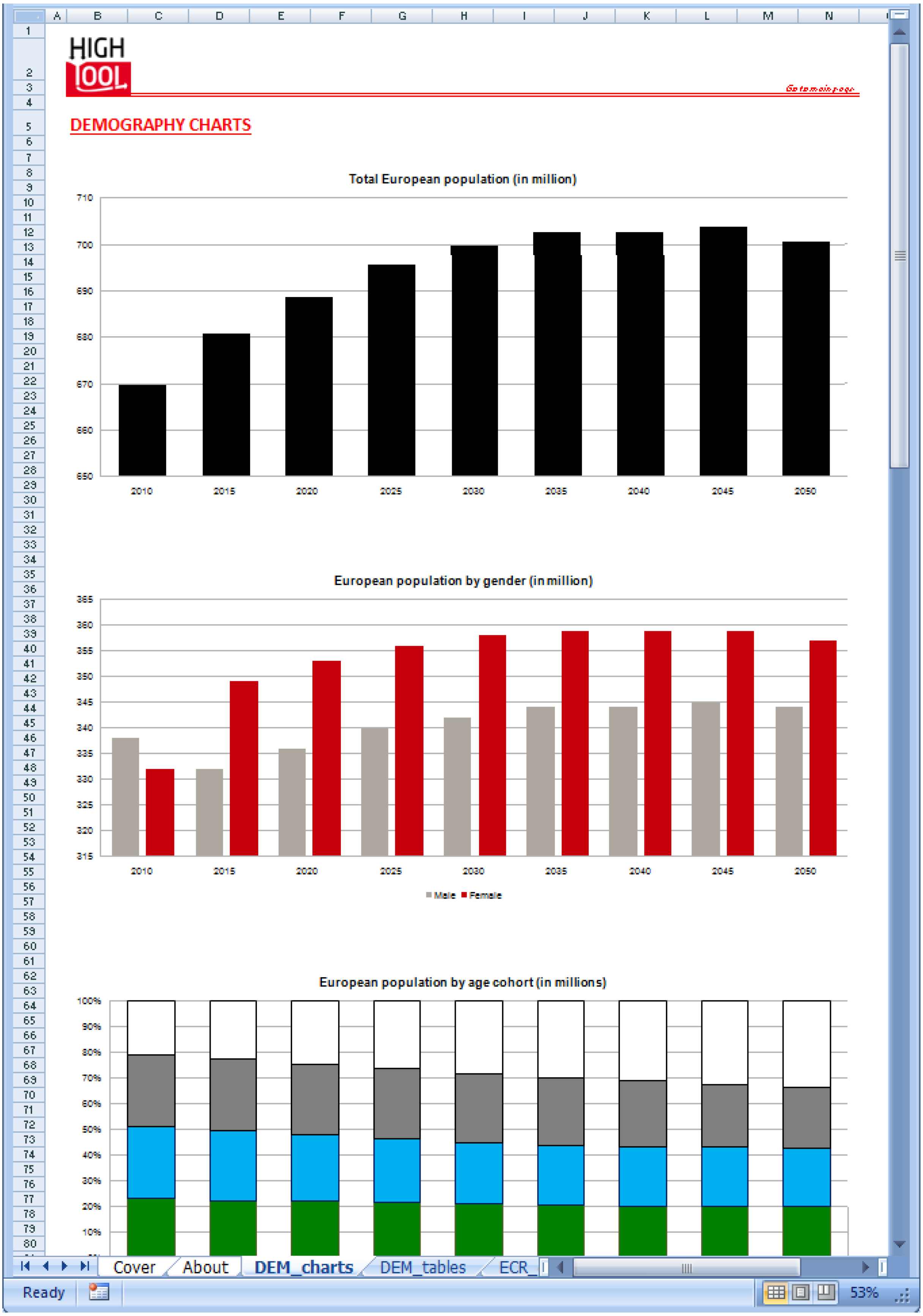


Figure 31: Full results displayed in charts, on a per theme basis

6.12 Export of Results

The export menu allows the exporting of raw data from tables contained in the HIGH-TOOL database. Exported data is sent to users in CSV format.

Users need to select a HIGH-TOOL database corresponding to an already existing model run. For a specific run, the User Interface shows all variables available. Users can filter such variables by restricting the search to specific strings in the variable name, the table name where the variable belongs to, or parts of the description of such variable.

Users can retrieve the variable contents by clicking on the rightmost icon ***Download.***

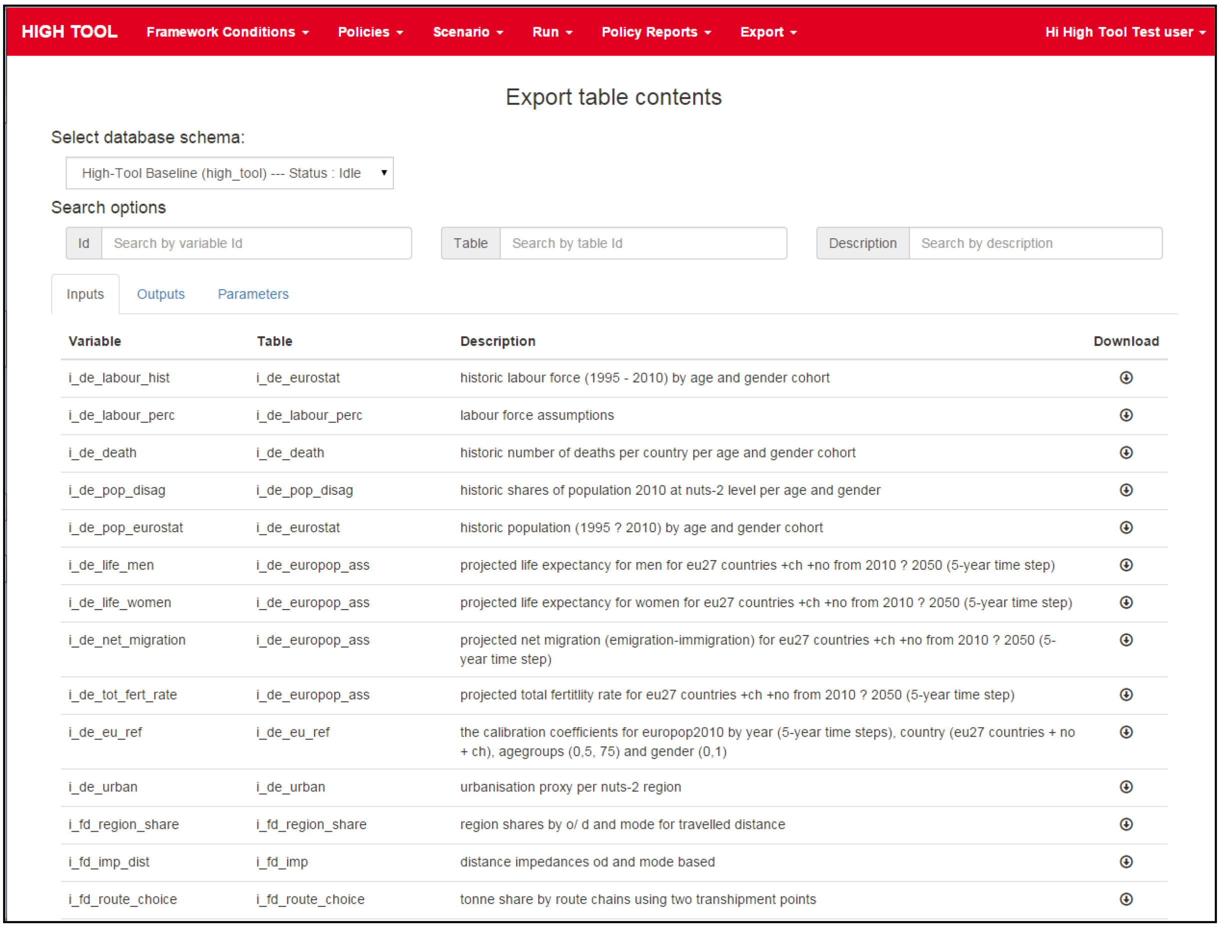


Figure 32: Export raw data control panel

7 References

Biosca O., Larrea E., Ulied A. (2015): HIGH-TOOL Deliverable D6.1, Design Criteria for the User Interface & Policy Assessment Reports, co-funded by the European Commission under the 7th Framework Programme, Karlsruhe.

European Commission (2011): Impact Assessment Working Paper Accompanying the White Paper, Roadmap to a Single European Transport Area – Towards a competitive and resource efficient transport system, Commission Staff Working Paper, SEC(2011) 358 final, Brussels.

European Commission (2012): The 2012 Ageing Report: Underlying Assumptions and Projection Methodologies. European Economy 4/2011, DG ECFIN, Brussels. http://ec.europa.eu/economy_finance/publications/european_economy/2012/pdf/ee-2012-2_en.pdf.

European Commission (2013): EU Energy, Transport and GHG Emissions Trends to 2050, Reference Scenario 2013. http://ec.europa.eu/transport/media/publications/doc/trends-to-2050-update-2013.pdf.

Eurostat (2014): Eurostat Population Projections 2013- (EUROPOP2013): Methodology and results of a long-term scenario of demographic convergence. http://epp.eurostat.ec.europa.eu/portal/page/portal/population/publications/population_projections.

Kiel J., Smith R., Laparidou K. (2014): Documentation: Input database for the HIGH-TOOL model, HIGH-TOOL Deliverable D3.1, project co-funded by the European Commission under the 7th Framework Programme, Zoetermeer/Karlsruhe.

Mandel B., Szimba E., Kraft M., Ihrig J., Smith R., Laparidou K., van Meijeren J., Zhang M., Vanherle K., Corthout R., Purwanto J. (2013): Structure of the HIGH-TOOL model, HIGH-TOOL Deliverable D2.1, project co-funded by the European Commission under the 7th Framework Programme, Karlsruhe.

Rich J., Bröcker J., Hansen C.O., Korchenewych A., Nielsen O.A., Vuk G. (2009): Report on Scenario, Traffic Forecast and Analysis of Traffic on the TEN-T, taking into Consideration the External Dimension of the Union – TRANS-TOOLS version 2; Model and Data Improvements, Funded by DG TREN, Copenhagen, Denmark.

Szimba E., Ihrig J., Kraft M. (2013): Inception Report, HIGH-TOOL Deliverable D10.2, project co-funded by the European Commission under the 7th Framework Programme, Karlsruhe.

Tognazzini B. (2013): AskTOG. http://www.asktog.com/basics/firstPrinciples.html. Referenced on 14 October 2015.

Van Grol R., De Bok M., De Jong G., Van Eck G., Ihrig J., Kraft M., Szimba E., Mandel B., Ivanova O., Corthout R., Purwanto J., Smith R., Laparidou K., Helder E., Grebe S. (2015): Elasticities and Equations, HIGH-TOOL Deliverable D4.2, project co-funded by the European Commission under the 7th Framework Programme, Karlsruhe.

Vanherle K., Corthout R., Szimba E., Meyer C., Kiel J., Ulied A., Biosca O., Török R. (2014): User Requirements, HIGH-TOOL Deliverable D1.1, project co-funded by the European Commission under the 7th Framework Programme, Karlsruhe.