INTERMODEL EU

Simulation using Building Information Modelling Methodology of Multimodal, Multipurpose and Multiproduct Freight Railway Terminal Infrastructures

Grant agreement: 690658

D6.1 – EXTERNAL MOBILITY SIMULATION MODEL	

Authors	Francisco Rodero (CENIT)
Status	Final Version
Dissemination	Confidential

Revision history:

Revision	Date	Author	Organization	Description
0.1	24/02/2018	Francisco Rodero	CENIT	Initial Version

Statement of originality:

This deliverable contains original unpublished work except where clearly indicated otherwise. Acknowledgement of previously published material and of the work of others has been made through appropriate citation, quotation or both.

The information set out in this publication are those of the author(s) and do not necessary reflect the official opinion of neither INEA nor the Commission. Neither INEA nor the Commission is responsible for the use that may be made of the information contained therein.

Executive Summary

The **INTERMODEL project** aims at establishing a methodology to design and alternative appraisal of multimodal freight terminals taking the most of the BIM tool and their capacity for providing multi-dimensional models. The dimensional models are to be combined with different simulation models resulting in an aggregated *decision-making tool* to be used during the project-planning phase and thorough its life cycle.

One of these simulation models has as main purpose to **analyse the impact of a terminal on the surrounding mobility**, focusing on the interaction with private vehicles and mainly assessed in terms of environmental KPIs.

From the work plan point of view, the developments related to the External Mobility Simulation are placed inside the WP6. This document shows the **results of the work done during the M6-M18 period**, which whole covers the task T6.1 and provides this deliverable D6.1. More in depth, the task T6.1 is composed of the following main subtasks:

- An initial benchmarking of traffic simulation environments and software in order to define the framework that better fits with the objectives of the project.
- Definition of an input data structure allowing the modelling of a wide variety of access networks to the terminal along with setting up interfaces between both BIM and TOS modules (M7-M14).
- Designing a methodology to estimate arrivals of trucks into the simulation and how it should be paired with the terminal simulation developed in WP5.
- Finally, KPIs integration in terms of environment (such as emission or pollutants) and traffic (congestion or LOS).

This deliverable contains a description of how the previous items have been developed, including performed works and objectives achieved. Although the design considers the necessary functionalities for the pending tasks, it is possible that minor changes have to be implemented but will not alter the fact of having avoided the risk identified as #18 (*failure in the simulation techniques and framework*).

A visual exhibition of the current progress will be done during the WP6 presentation that will take place in the 3rd Plenary Meeting scheduled for middle March 2018 in Barcelona.

Table of contents

List of Figures

List of Tables

1 Introduction

1.1 Scope

This deliverable contains a description of the work done in the context of the WP6 during the M6-M18 period. In other words, the document includes performed tasks related to the definition of the EMS model and corresponding data setting.

From the whole system perspective, the existing dataflow between the BIM module and the building of the access network simulation can be found in the sections below, describing, step by step, how information is transformed and which processes take part.

1.2 Audience

Since this document describes internal details of how information coming from the BIM module is processed in order to generate a valid input for the traffic simulation environment and how this environment has been configured to automatically build a traffic network, the **target audience is the Consortium that composes the INTERMODEL project and the Commission Services**.

It is very important to remark that the dataflow described in the next sections includes some cutting edge developments that may result a key factor in terms of future exploitation.

1.3 Definitions / Glossary

Application Programming Interface: Set of definitions (routines, protocols or tools) for building application software.

Calibration - In the context of simulation models, the process of checking that obtained outputs are consistent respect to given inputs.

Comma-separated values - way of storing tabular data in plain text, one record for each line where each field is separated by a separator, usually a comma.

Highway Capacity Manual - publication of the Transportation Research Board of the National Academies of Science in the United States. It contains concepts, guidelines, and computational procedures for computing the capacity and quality of service of various highway facilities, including freeways, highways, arterial roads, roundabouts, signalized and unsignalized intersections, rural highways, and the effects of mass transit, pedestrians, and bicycles on the performance of these systems.

JavaScript Object Notation (JSON) - open-standard file format that uses human-readable text to transmit data objects consisting of attribute–value pairs and array data types. Sometimes used as an alternative for XML.

Python - Python is an interpreted high-level programming language for general-purpose programming.

Script - programs written for a special run-time environment that automate the execution of tasks that could alternatively be executed one-by-one by a human operator.

Shapefile (SHP) - The shapefile format is a popular geospatial data format for GIS software, describing vector features: points, lines and polygons, developed and regulated by ESRI as open specification for data interoperability.

Validation - Confirmation, through provision of objective evidence that the requirements for a specific intended use or application have been fulfilled.

1.4 Abbreviations

API: Application programming interface

CO_2: Carbon dioxide

CSV: Comma-separated Values

DB: Database

DLL: Dynamic-Link Libraries

EMS: External Mobility Simulation

GIS: Geographic Information System

GUI: Graphical User Interface

HCM: Highway Capacity Manual

JSON: JavaScript Object Notation

KPI: Key Performance Indicator

LOS: Level of Service

NO_X: Nitrogen oxides

1.5 Structure

- **Section 1:** contains an overview of this document, providing its Scope, Audience, and this Structure itself.
- **Section 2, Traffic Simulation Environment Benchmarking**: describes the features considered to select the environment to use as traffic simulation software.
- **Section 3, Input Data Structure for Modelling**: explains how data flows from BIM to become a simulation model and which additional configuration files are necessary.
- **Section 4, Linking EMS and TOS:** contains information about how both EMS and TOS become consistent in terms of truck activity.

2. Traffic Simulation Environment benchmarking

The INTERMODEL project has some requirements that must be covered to ensure that objectives are fully achieved. In that sense, a benchmarking of traffic simulation environments and software has been done in order to define the framework that better fits with these objectives.

In order to easily filter or discard non interesting products, a set of features has been identified as mandatory since, from the technical point of view, they are necessary as shown in the following table:

Expected features
Accuracy
Environmental and traffic KPIs are calculated using models and/or formulas that directly rely in how vehicles move. That is, the environment must offer the chance of running simulations at microscopic level where instantaneous speed and acceleration are available, with proven models faithfully recreating the behaviour of drivers, such as *lane-changing* or *car-following.*
Automation
Traffic simulation in the context of the project acts as one internal module that receives inputs from BIM and, after running a scenario, generates a set of known outputs. Since this should be user-transparent and able to run different replicas with the aim of considering stochastic behaviour, the process must be non-interactive.
API
Although many environments offer a wide set of features, some specific situations may require extend them using API or scripting. The more powerful they are, the more variety of developments will be possible.
Miscellaneous
The abovementioned features are very important but there are others that, depending on the context, may imply a significant reduction in effort and therefore, improving the project life cycle in several ways. Among them, we can find: user-friendly interface, licensed or free, etc.

Table 1. Expected features of the traffic simulation environment

Taking into account these features, the table below summarizes the main results obtained after matching them with the most currently used and existing environments in the market.

Main remarks about simulation environments
Dynameq
Intended to support large-scale urban planning in metropolitan areas.
Vissim
Offers an API package to integrate own or external applications by means of *DLL* but only for: • Traffic signals • Driver behavior model (*car-following* and *lane-change*) • Emissions (limited to some vehicle attributes and dynamic parameters). BIM importer but for pedestrian simulation purposes.
Transmodeler
Traffic microscopic simulator to take advantage from if already using of *TransCAD* transport planning software, which is designed for purposes other than the INTERMODEL ones.
Transyt
Traffic simulator specific for signalized junctions.
Sumo
Open source microscopic package composed of several applications, including the following emission

Main remarks about simulation environments
models: • Derivation of the versions 2.1 and 3.1 of the *Handbook Emission Factors for Road Transport* data base *(HBEFA)*   • A <u>non open source</u> derivation of the *Passenger Car and Heavy Duty Emission Model (PHEM)*
Aimsun

• Unattended batch mode and automation using built-in support for *Python* scripting
 • Includes fuel consumption model and environmental models:
  • QUARTET
  • Panis et al, *"Modelling instantaneous traffic emission and the influence of traffic speed limits"*
  • Initial release of London Emissions Model for measuring pollutants in both microscopic and mesoscopic simulations for modeling CO_2 and NOX.
 • Possibility to extend features by means of C++ or *Python* custom API

Table 2. Summary of simulation environments

Two of the previous environment highlight over the others: *Vissim* and ***Aimsun***. Nevertheless, the second one has no limits when extending built-in features using API, **becoming a very interesting choice to reach one of the main goals of the simulation model: to be adapted to any kind of terminal**.

In addition, the Consortium can take advantage of the many years of expertise that CENIT has using *Aimsun* in several projects, including EU-funded projects such as STRAIGHTSOL (*Strategies and Measures for Smarter Urban Freight Solutions*), and the fact that Aimsun headquarters are also located in Barcelona, making easier business relationships in case of needing specific technical support.

3. Input Data Structure for Modelling

3.1 Overview

The bigger effort of the task T6.1 focuses on the data flow that transforms a SHP file in a valid network model to be used in the traffic simulation environment. The next diagram visually depicts this process for a better understanding:
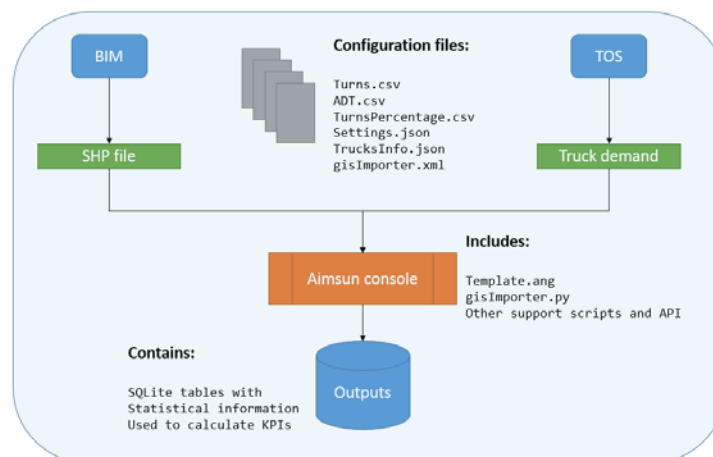
Figure 1. Overview of the EMS architecture

3.2 Geographic information of the network

The network is generated from BIM module in a SHP format file, including information about the road sections, understanding a section as the segment between two junctions or without changes in its characteristics (e.g. speed or number of lanes). The file includes a set of additional project-specific attributes used to better describe the topology:

- **ExternalID**: Each object created in Aimsun has its own identifier; this field is intended to preserve identifier used in BIM and link later results.
- **FromNode / ToNode**: These two attributes are used to virtually link sections between them. Once the network is loaded in Aimsun, this information is used to create specific objects representing nodes (junctions).

Figure 2. Example of FromNode/ToNode attributes

- **DIR**: This attribute is used to distinguish single sections that represent two-ways or to warn if topology (how geographic line has been drawn) does not match with driving direction.
- **NumLanes**: Number of lanes of a road section.
- **Speed**: Maximum speed allowed on the road section (km/h).
- **LaneWidth**: Width of the lane (meters).
- **RoadType**: Can be *Motorway*, *Primary*, *Secondary*, *AcessRoundabout* or *Roundabout*. These types match with previously defined types in an Aimsun template in order to adapt some behaviours related to driving, such as reaction time or aggressiveness.

ExternalID	FromNode	ToNode	DIR	NumLanes	Speed	LaneWidth	RoadType
0	7	8	1	2	15	3	Motorway
1	7	9	1	1	15	3	Motorway
2	12	13	1	2	90	3	Motorway
3	19	6	1	1	40	3	Motorway
4	11	64	1	2	90	3	Motorway
5	84	12	1	2	90	3	Motorway
6	8	9	1	2	15	3	Motorway
7	9	53	1	2	80	3	Motorway
8	10	11	1	3	90	3	Motorway
9	6	7	1	2	15	3	Motorway

Figure 3. Sample of attributes in a SHP file

As mentioned in D1.4, *Internal Progress Report*, developments related to T6.1 have been mostly tested using La Spezia or Melzo actual networks. The picture below shows the considered surrounding network of La Spezia (in yellow colour) once loaded from the SHP and then, exporting it from Aimsun.
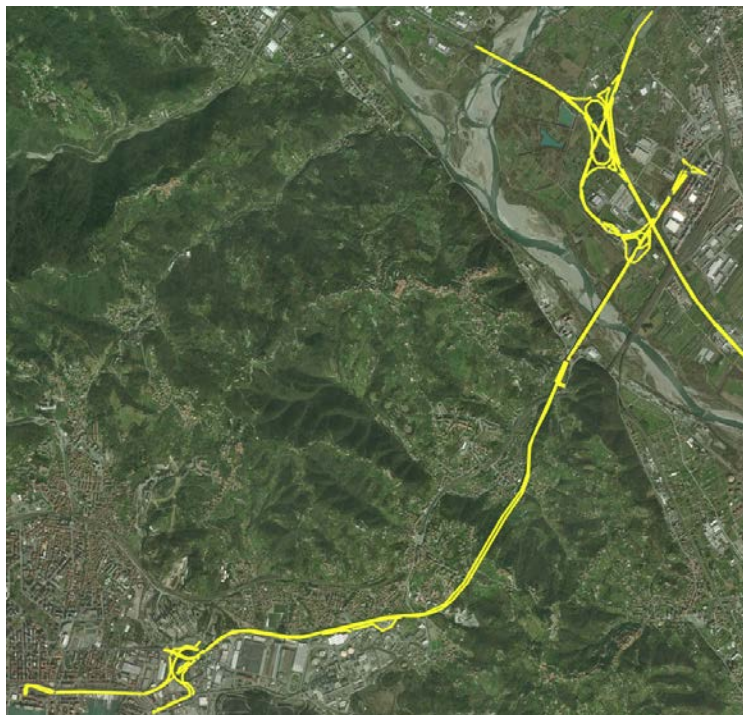
Figure 4. Surrounding network of La Spezia

3.3 GIS Importer

Aimsun supports interactive import from a lot of sources (AutoCAD ®, *OpenStreetMap*, Visum, GTFS, other network formats such as CONTRAM, Synchro, Paramics, RoadXML, Vissim) through its GUI.

On the other hand, the process of automatically importing the network geometry using a custom *python* script requires two inputs, the SHP mentioned in the previous section and a settings file in XML format containing a list of preferences used to customize how the SHP file is translated to the simulation environment.

Aimsun can import a SHP file in several ways: as just one more layer (internal or external), as a centroid configuration, as bus stops, etc. The GIS Importer imports all the information of SHP file as network; that is, considering each geographical line as a section object and where the number of lanes, speed and so on will be set according the attributes described in the previous section of this document.

As an example, the figure above partially shows contents of a valid XML file:

```
<GIS_IMPORTER>
        <importer>network</importer>
        <ColumnLanes>NumLanes</ColumnLanes>
        <ColumnSpeed>Speed</ColumnSpeed>
        <ColumnEID>ExternalID</ColumnEID>
        <ColumnLaneWidth>LaneWidth</ColumnLaneWidth>
        <ColumnRoadType>
                <ColumnRoadTypeValue>
                        RoadType
                </ColumnRoadTypeValue>
                <RoadTypeValue>
                        <Value>AcessRoundabout</Value>
                        <RoadType>180</RoadType>
                </RoadTypeValue>
                <RoadTypeValue>
                        <Value>Motorway</Value>
                        <RoadType>180</RoadType>
                </RoadTypeValue>
                <RoadTypeValue>
                        <Value>Primary</Value>
                        <RoadType>184</RoadType>
                </RoadTypeValue>
                <RoadTypeValue>
                        <Value>Roundabout</Value>
                        <RoadType>180</RoadType>
                </RoadTypeValue>
                <RoadTypeValue>
                        <Value>Secondary</Value>
                        <RoadType>190</RoadType>
                </RoadTypeValue>
        </ColumnRoadType>
        <LaneWidth>3</LaneWidth>
        <ColumnFNode>FromNode</ColumnFNode>
        <ColumnTNode>ToNode</ColumnTNode>
        <SingleDirValue>1</SingleDirValue>
        <SingleInversedDirValue>-1</SingleInversedDirValue>
        <DualDirValue>2</DualDirValue>
        <DualInversedDirValue>3</DualInversedDirValue>
</GIS_IMPORTER>
```
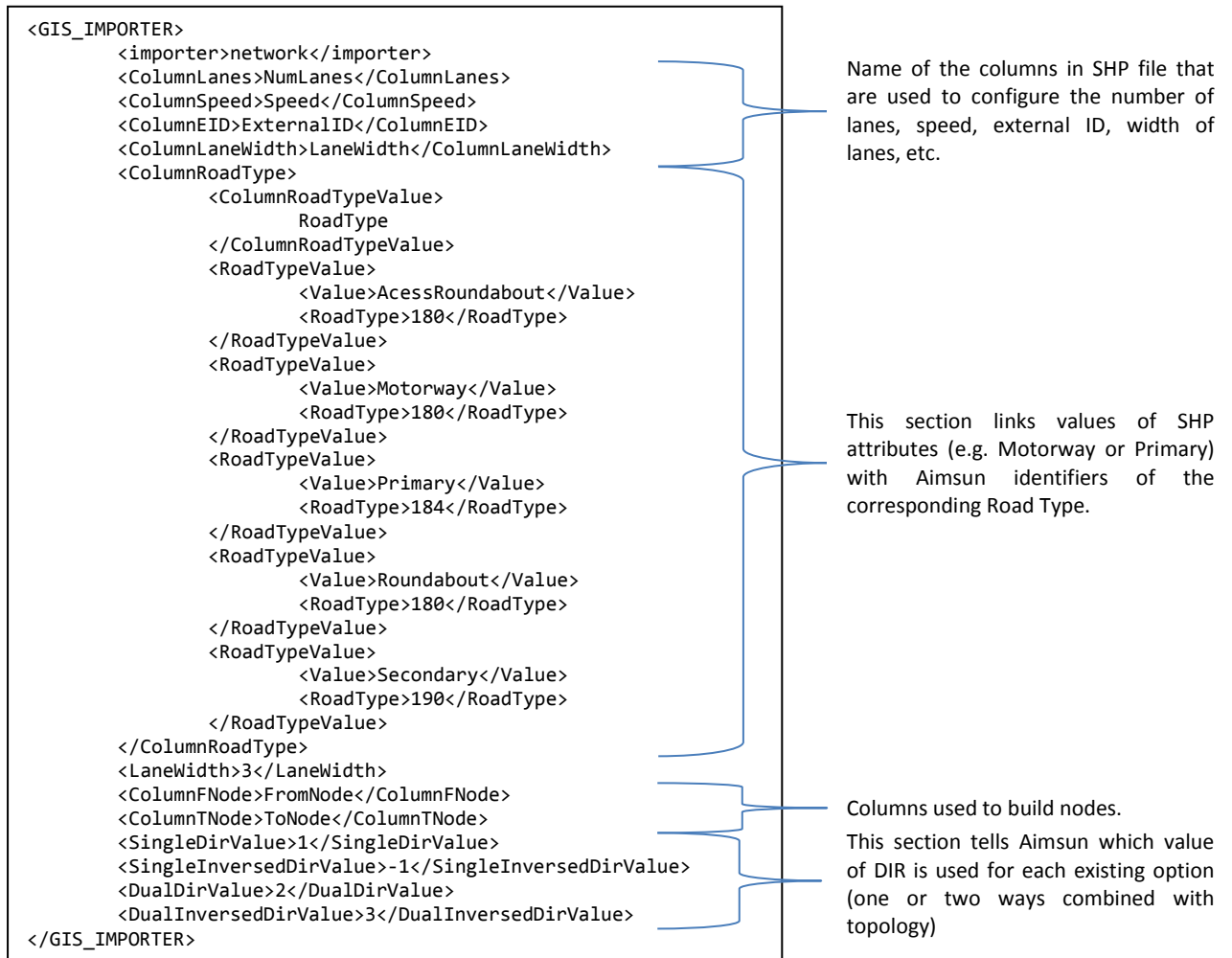
Name of the columns in SHP file that are used to configure the number of lanes, speed, external ID, width of lanes, etc.

This section links values of SHP attributes (e.g. Motorway or Primary) with Aimsun identifiers of the corresponding Road Type.

Columns used to build nodes.

This section tells Aimsun which value of DIR is used for each existing option (one or two ways combined with topology)

Figure 5. Example of XML file with setting for the GIS importer

The next picture illustrates two examples of how Aimsun builds the network, both sections and nodes:
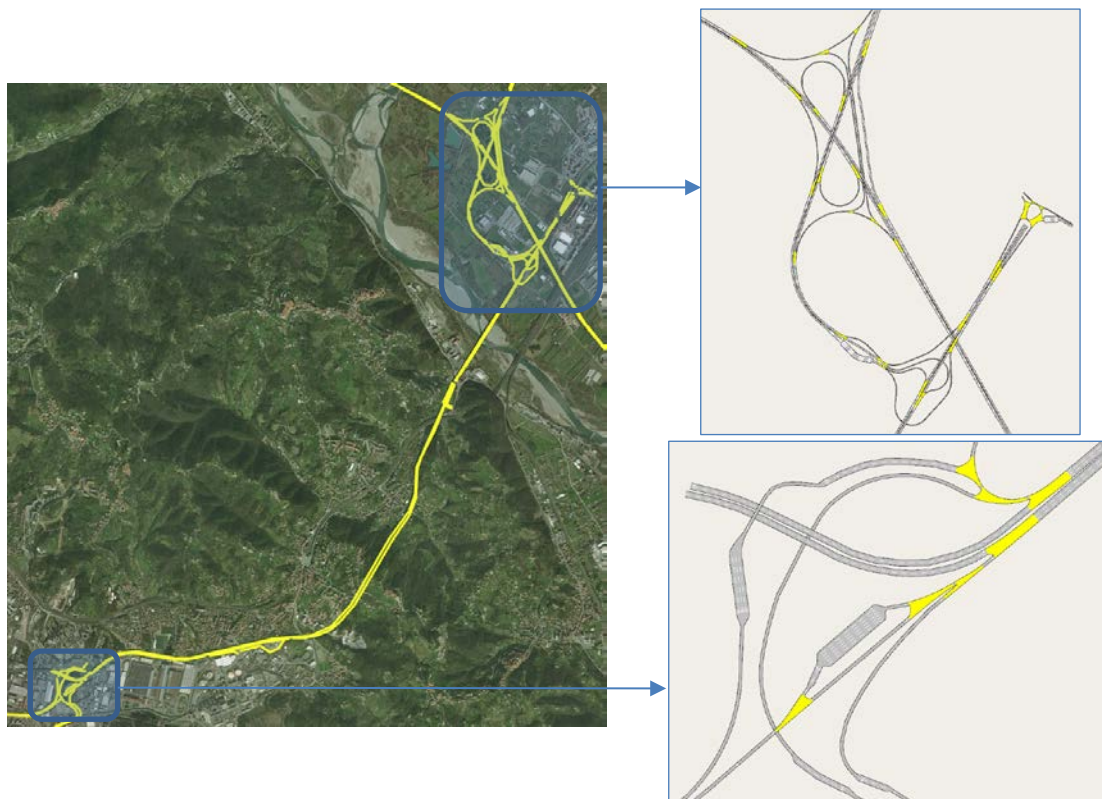

Figure 6. Building nodes in Aimsun

3.4 Fine Tuning of the network

As shown in previous section, attributes *FromNode* and *ToNode* are used to virtually connect sections of the network. By default, Aimsun implements an all-to-all connection between the sections that share a node, which may be invalid for certain complex junctions and especially for roundabouts.

Because of that, a fine tuning mechanism has been implemented to ensure that forbidden turns are not available during simulation and, therefore, avoiding non-realistic vehicle movements.

There is a CSV file supporting this feature, having the next fields for each data record:

- **Origin Section**: External ID of the origin section.
- **Destination Section**: External ID of the destination section.
- **Origin Lanes**: Range of lanes, from 0 to N, from the origin section that takes part of the turn (non-consecutive lanes are not allowed).
- **Destination Lanes**: Range of lanes, from 0 to N, from the destination section that takes part of the turn (non-consecutive lanes are not allowed).

- **Connections**: Sets which combinations lane-to-lane are permitted.
- **Priority Signal**: Traffic signals that may exist in the road (0 = None; 1 = Give-way; 2 = Stop).
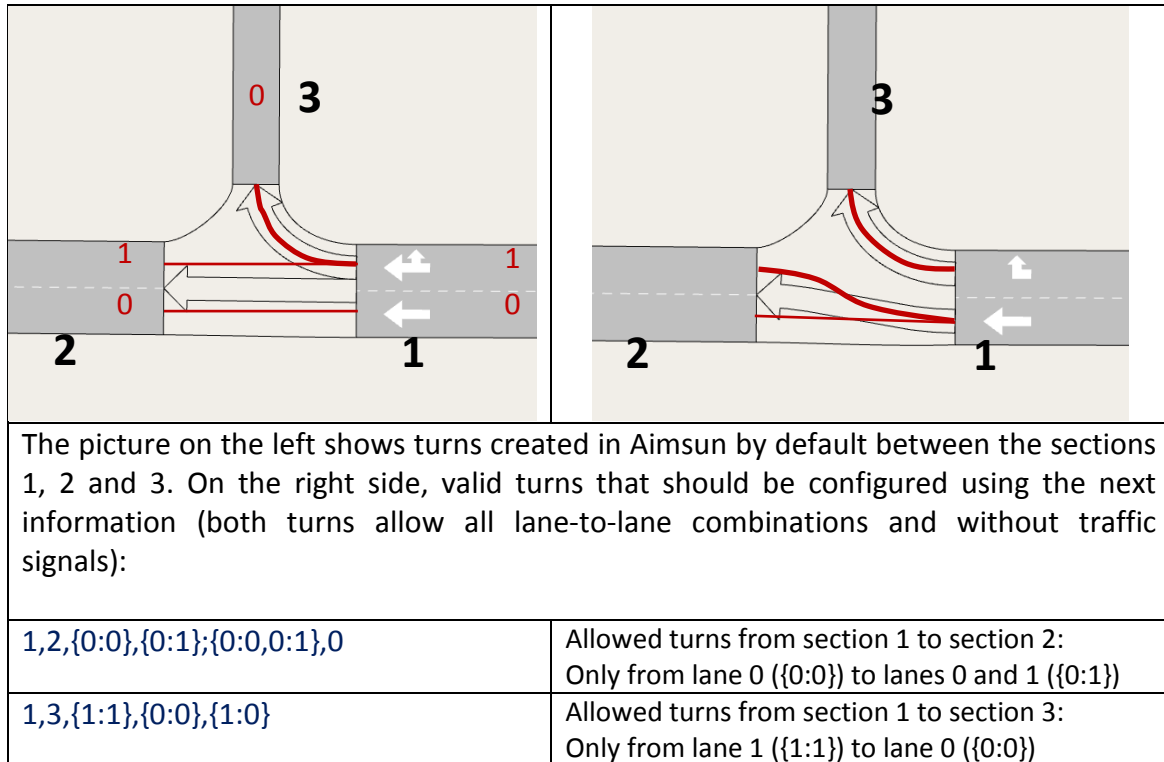

The picture on the left shows turns created in Aimsun by default between the sections 1, 2 and 3. On the right side, valid turns that should be configured using the next information (both turns allow all lane-to-lane combinations and without traffic signals):	
1,2,{0:0},{0:1};{0:0,0:1},0	Allowed turns from section 1 to section 2: Only from lane 0 ({0:0}) to lanes 0 and 1 ({0:1})
1,3,{1:1},{0:0},{1:0}	Allowed turns from section 1 to section 3: Only from lane 1 ({1:1}) to lane 0 ({0:0})

Figure 7. Sample of CSV file for advanced edition of turns

3.5 Model Demand

Demand refers to vehicles that must be injected to the network at entry points using one of the next alternatives:

a) By means of centroid configuration attracting and/or generating trips and O-D matrices between these centroids.
b) By means of the vehicle flow for each section used to enter to the network and turn percentages for all nodes (so called *traffic states*).
c) Modelling known paths as public transport lines

Both alternatives need a previous definition of major features for each considered vehicle type: size, acceleration and speed parameters, fuel consumption, emissions; this information can be included in the Aimsun template.

Private Vehicles

OD matrices are no available from the case studies of La Spezia and Melzo; thus, working with traffic states is the better approach since it may require less effort during calibration of the models.

To support that, there are two configuration files:

Private vehicles demand
ADT.csv

Contains the Average Daily Traffic for each network entry point, where each data record has the next fields:
 	
• **Section Identifier:** External ID of the section
• **Flow:** Average daily traffic (vehicles / hour)
 	
Example:
 	
`5,9250`
`13,8800`
`12,8500`

TurnPercentage.csv

Contains the percentage of vehicles that make a specific turn for each existing node, where each data record has the next fields:
 	
• **Origin Identifier**: External ID of the origin section
• **Destination Identifier**: External ID of the destination section
• **Percentage**: Percentage of vehicles that takes the turn between origin section and destination section. The total sum of percentages for a given origin must be 100.
 	
Example:
 	
`17,10,80`
`17,21,20`
`16,24,90`
`16,25,10`

Table 3. Input files for private vehicles demand

Private demand is created using hourly traffic states, so it is necessary to define which percentage over the ADT corresponds to each hour interval. This information, along with other parameters, is stored in a configuration file of parameters called *settings*, using *JSON* format. To be more specific, there are two fields that work in combination with the ADT:

• **Initial_Time**: Sets the initial time of the day of the simulation.
• **Intervals**: Array of hourly intervals to be taken into account from the initial time on. Each item of the array represents a percentage over the ADT (this percentage applies over all entry sections in the same way).

E.g. Considering a simulation from 05am during 5 hours, setting files must contain the following information:

"Initial_Time" : "05:00:00",
"Intervals" : {
        "Interval_1" : 4,
        "Interval_2" : 6.5,
        "Interval_3" : 8.5,
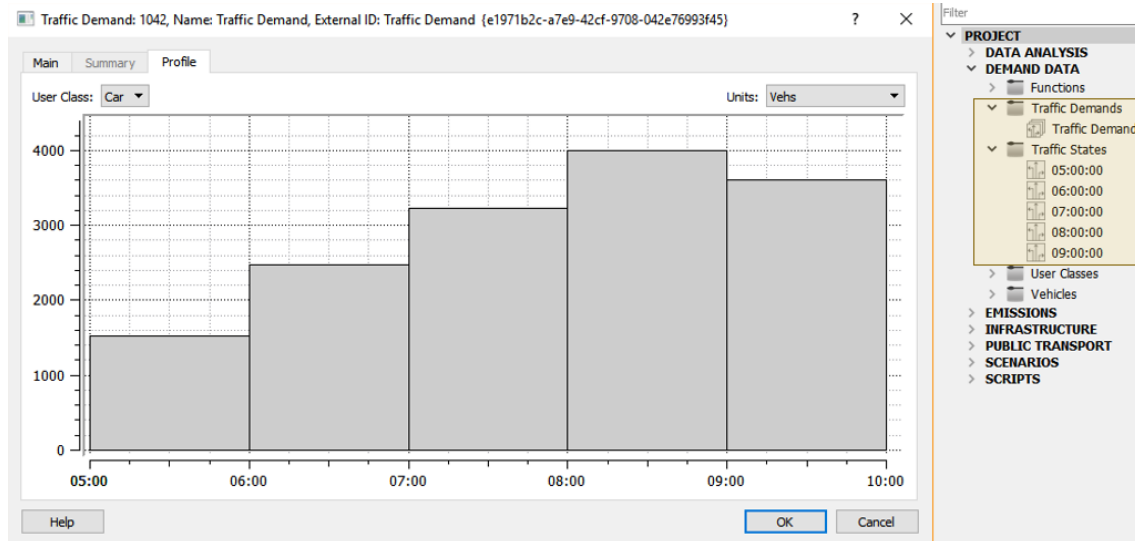   "Interval_4" : 10.5,
        "Interval_5" : 9.5}

Figure 8. Sample of hourly demand for private vehicles

Trucks

Regarding the trucks that go to or leave the terminal, they enter the access network as defined in a CSV file generated by the TOS module, from which the next fields are used:

- **ID:** Identifier of a truck in the TOS.
- **Arrival time:** time a truck is created at a source (expected arrival time to the last section of the surrounding network before the terminal).
- **Departure gate exit:** time a truck has left the gate (time when is injected in the surrounding network).
- **Unloaded TEU / Loaded TEU:** The maximum between these two values will define the length of a truck, considering only three different values:
    a. **1TEU =** 7.6m
    b. **2TEU =** 13.7m
    c. **2.5TEU =** 15.2m

	A	B	C	D	E	F	G	H	I	J	K
1	ID	Arrival time	Arrival gate entry	Arrival gate exit	Last handling finished	Departure gate entry	Departure gate exit	Unloaded boxes	Unloaded TEU	Loaded boxes	Loaded TEU
2	1	20/09/2017 05:03	20/09/2017 05:05	20/09/2017 05:09	20/09/2017 05:28	20/09/2017 05:31	20/09/2017 05:35	2	2	1	2
3	2	20/09/2017 05:08	20/09/2017 05:10	20/09/2017 05:16	20/09/2017 05:19	20/09/2017 05:22	20/09/2017 05:25	1	2	0	0
4	3	20/09/2017 05:40	20/09/2017 05:41	20/09/2017 05:45	20/09/2017 05:58	20/09/2017 06:02	20/09/2017 06:06	1	1	1	2.25
5	4	20/09/2017 05:53	20/09/2017 05:55	20/09/2017 05:58	20/09/2017 06:08	20/09/2017 06:12	20/09/2017 06:13	0	0	1	1
6	5	20/09/2017 06:04	20/09/2017 06:05	20/09/2017 06:10	20/09/2017 06:43	20/09/2017 06:47	20/09/2017 06:50	1	1	1	2
7	6	20/09/2017 06:16	20/09/2017 06:18	20/09/2017 06:23	20/09/2017 06:57	20/09/2017 07:00	20/09/2017 07:05	1	2	1	1
8	7	20/09/2017 06:20	20/09/2017 06:22	20/09/2017 06:27	20/09/2017 06:35	20/09/2017 06:39	20/09/2017 06:41	2	2	0	0
9	8	20/09/2017 06:21	20/09/2017 06:22	20/09/2017 06:26	20/09/2017 07:02	20/09/2017 07:06	20/09/2017 07:10	0	0	1	2

Figure 9. Sample of truck demand generated by TOS

From the Aimsun point of view, paths for every single truck can be modelled as two public transport lines, one going to the terminal and the other leaving the terminal.

Also, as for private vehicles, it is necessary to define which sections are used by the trucks to enter the model. To support that, there is another configuration file in *JSON* format, called *truckInfo*, containing the following information (all of them have the same data structure):

- **NetEntrances:** Array of pairs {section identifier, %} defining which percentage of trucks use a specific section to enter the surrounding network.
- **NetExits:** Array of pairs {section identifier, %} defining which percentage of trucks use a specific section to leave the surrounding network.
- **TerminalEntrances:** Array of pairs {section identifier, %} defining which percentage of trucks use a specific section to enter the terminal.
- **TerminalExits:** Array of pairs {section identifier, %} defining which percentage of trucks use a specific section to leave the terminal.

The picture below shows a sample of the *truckInfo.json* file together with a simplified version of a network and terminal connection (please note that the total sum of percentages of an array must be 100):
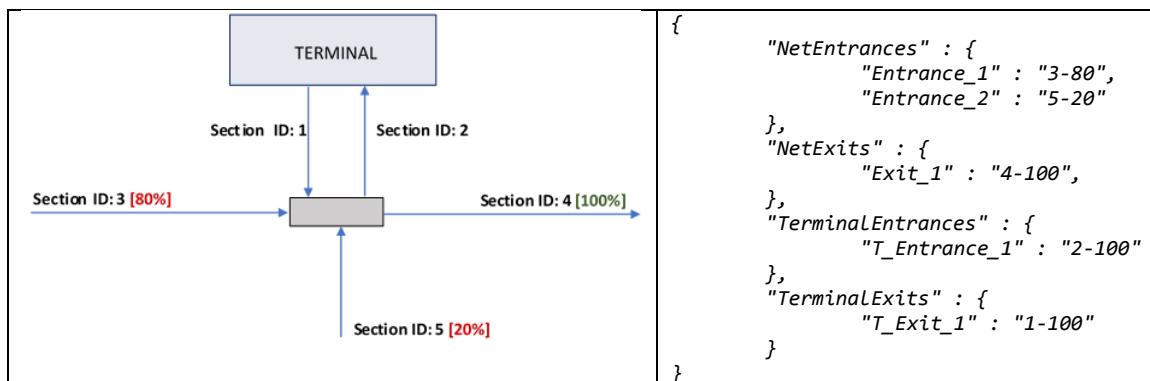

Figure 10. Sample of *truckInfo.json* file based on a network

3.6 Running a simulation

Once the network has been loaded, the Aimsun script included in the template file runs as many replicas as they are configured in the abovementioned *settings.json* file using these fields:

- **Nb_Replications:** Set the maximum number of replicas to simulate.
- **WarmUp_Time:** Sets a warm up period ("hh:mm:ss") used to populate the network with random private vehicles in order to avoid non-realistic situations at the beginning of the simulation time (roads are not empty in normal conditions).

3.7 Outputs

By default, Aimsun provides as output some statistics as they are previously configured in the scenario settings, e.g. which measures, output target, level of aggregation or time interval.

The *settings.json* file will be modified to define which time interval must be used for outputs (5', 15' or 60'), the output target will be a relational database in SQLite format and all measures will be available. The following sections show the most relevant measures that can be found:

Network Statistics

- **Delay Time**: average delay time per vehicle per kilometre. This is the difference between the expected travel time (the time it would take to traverse the system under ideal conditions) and the travel time. It is calculated as the average of all vehicles and then converted into time per kilometre. It does not include the time spent in virtual queue.
- **Density**: average number of vehicles per kilometre for the whole network.
- **Flow**: average number of vehicles per hour that have passed through the network during the simulation period. The vehicles are counted when leaving the network via an exit section.
- **Harmonic Speed**: harmonic mean speed for all vehicles that have left the system.
- **Input Count**: number of vehicles that have entered the network during the simulation period. The vehicles are counted when entering the network via an entrance section.
- **Input Flow**: number of vehicles per hour that have entered the network during the simulation period. The vehicles are counted when entering the network via an entrance section.
- **Max Virtual Queue**: maximum virtual queue in the network during the simulation period. It is measured in number of vehicles.
- **Mean Queue**: average queue in the network during the simulation period. It is measured in vehicles.

- **Mean Virtual Queue**: average virtual queue in the network during the simulation period. It is measured in number of vehicles which are blocked from entering the network.
- **Missed Turns**: total number of missed turns.
- **Number of Lane Changes**: total number of lane changes per network km.
- **Number of Stops**: average number of stops per vehicle per kilometre.
- **Speed**: average speed for all vehicles that have left the system. This is calculated using the mean journey speed for each vehicle.
- **Travel Time**: average time a vehicle needs to travel one kilometre inside the network. This is the mean of all the single travel times (exit time - entrance time) for every vehicle that has crossed the network, converted into time per kilometre.
- **Stop Time**: average time at standstill per vehicle per kilometre.
- **Total Number of Stops**: total number of stops in the network during the simulation period.
- **Total Travel Time**: total travel time experienced by all the vehicles that have crossed the network. It does not include the time spent in virtual queue.
- **Total Travelled Distance**: total number of kilometres travelled by all the vehicles that have crossed the network.
- **Vehicles Inside**: number of vehicles inside the network when the simulation finishes.
- **Vehicles Lost Inside**: if the route-based mode is being used, the number of lost vehicles that are currently in the network, that mean a vehicles that has lost at least one of its pre-assigned turns and ais in a location from where its original destination can no longer be reached.
- **Vehicles Lost Outside**: if the route-based mode is being used, the number of lost vehicles that have left the network during the simulation period.
- **Vehicles Outside**: number of vehicles that have left the network during the simulation.
- **Vehicles Waiting to Enter**: number of vehicles that are waiting to enter the network.
- **Waiting Time Virtual Queue**: average time in seconds that vehicles remain in the virtual queue.

Environmental Models Network Statistics

- **Fuel Consumption**: total litres of fuel consumed by all the vehicles that have crossed the network.
- **IEM Emission**: for each pollutant, the total grams of pollution emission emitted by all the vehicles that have exited the network (Panis et al, 2006).
- **Pollutant Emission**: for each pollutant, total kilograms of pollution emitted by all the vehicles that have crossed the network (QUARTET, 1992).

Section Statistics

- **Count**: number of vehicles that have crossed the section during the simulation period.
- **Delay Time**: average delay time per vehicle. This is the difference between the expected travel time (time it would take to traverse the section under ideal conditions) and the travel time. It is calculated as the average of all vehicles.
- **Delay Time / Travel Time**: the delay time percentage from the travel time measured during the simulation period.
- **Density**: average number of vehicles per kilometre in the section.
- **Flow**: average number of vehicles per hour that have crossed the section during the simulation period.
- **Harmonic Speed**: harmonic mean speed for all vehicles that have traversed the section.
- **Input Count**: number of vehicles that have entered the section during the simulation period.
- **Input Flow**: average number of vehicles per hour that have entered the section.
- **Max Queue**: maximum length of the queue in this section, expressed as number of vehicles per lane.
- **Max Virtual Queue**: maximum virtual queue length in this section measured in vehicles. Only sections where vehicles enter the network may have values on this statistic.
- **Mean Queue**: average length of the queue in that section, expressed as the number of vehicles per lane. It is calculated as a time average.
- **Mean Virtual Queue**: average virtual queue in this section measured in number of vehicles. Only sections where vehicles enter the network may have values on this statistic.
- **Number of Lane Changes**: average number of lane changes during the simulation period.
- **Number of Stops**: average number of stops per vehicle while travelling in the section.
- **Speed**: average speed for all vehicles that have traversed the section. This is calculated using the mean speed for the section journey for each vehicle.
- **Stop Time**: average time at a standstill per vehicle while travelling in the section.
- **Total Number of Lane Changes**: the total amount of number of lane changes during the simulation period.
- **Total Travel Time**: total travel time experienced by all the vehicles in this section.
- **Total Travelled Distance**: total number of kilometres travelled by all the vehicles in this section.
- **Travel Time**: average time a vehicle needs to cross the section. This is the mean of all the single travel times (section exit time - section entrance time) of every vehicle that has left the section.
- **V/C Ratio**: the flow divided by the section capacity.

- **Virtual Queue**: virtual queue length measured in vehicles. Only sections where vehicles enter the network might have values on this statistic.
- **Waiting Time Virtual Queue**: average waiting time in virtual queue.

Environmental Models Section Statistics

- **Fuel Consumption**: total litres of fuel consumed inside the section by all the vehicles that have crossed it.
- **Instant Emission**: for each pollutant, the total grams of pollution emission emitted by all the vehicles that have exited the section (Panis et al, 2006).
- **Pollution Emitted:** for each pollutant, total kilograms of pollution emitted by all the vehicles that have crossed the section (QUARTET, 1992).

Node Statistics

- **Avg Approach Delay**: This is calculated as the sum of delay time of each link weighted by the link flow and then divided by the total flow.
- **Lost Vehicles**: number of vehicles that become lost in any of the turns in the node.
- **Missed Turns**: number of vehicles that miss any turn in the node.

Environmental Models Node Statistics

- **Instant Emission**: for each pollutant, the total grams of pollution emission emitted by all the vehicles that have exited any turn in the node (Panis et al, 2006).

Public Transport Statistics (trucks)

- **Count:** number of vehicles that have completed the trip.
- **Delay Time:** average delay time per vehicle to make the trip. This is the difference between the expected travel time (time it would take to go from the origin to the destination under ideal conditions) and the actual travel time. It does not include the time performing stops nor the time spent in virtual queue.
- **Dwell Time:** average time per vehicle at public transport stops.
- **Flow:** number of vehicles that have reached the end of the public transport line during the simulation period. The vehicles are counted when going out of the network via the last section of the line.
- **Harmonic Speed:** harmonic mean speed for all vehicles that have reached the end of the public transport line.
- **Input Count:** number of vehicles that have started the trip.
- **Input Flow:** number of vehicles per hour that have started the trip.
- **Max Virtual Queue:** maximum virtual queue length in this trip measured in vehicles.
- **Number of Stops:** average number of stops per vehicle during the trip.

- **Speed:** average speed for all vehicles that have reached the end of the public transport line. This is calculated using the mean journey speed for each vehicle.
- **Stop Time:** average time spent stopped outside public transport stops per vehicle during the trip.
- **Total Travel Time:** total travel time experienced by all the vehicles that have made the trip.
- **Total Travelled Distance:** total number of kilometres travelled by all the vehicles that have made the trip.
- **Travel Time:** average time it takes for a vehicle to travel along a public transport line. This is the mean of all the single travel times (exit time - entrance time) for each vehicle.
- **Waiting Time Virtual Queue**: average waiting time in virtual queue.

Environmental Models Public Transport Statistics

- **Fuel Consumption**: total litres of fuel consumed by all the vehicles that have made the trip.
- **Instant Emission**: for each pollutant, the total grams of pollution emission emitted by all the vehicles that have made the trip (Panis et al, 2006).
- **Pollution Emitted**: for each pollutant, total kilograms of pollution emitted by all the vehicles that have made the trip (QUARTET, 1992).

In addition, similar structures are available for *Section Lane, Turn and Link, Subpaths, Traffic Demand, Centroids, Traffic Management* and other HCM 2010 compliant statistics.

All these outputs can be accessed through the corresponding SQLite DB tables (please, refer to the Aimsun manual for further information about field names and data types) in order to calculate environmental and traffic KPIs:

Aimsun microscopic output tables	
MISYS	statistical information of the whole system for each period
MISECT	statistical information of the sections for each period
MINODE	statistical information of the nodes for each period
MIPT	statistical information of public transport lines for each period
*PO	There is a table with the same name and the suffix PO storing pollution statistical information based on QUARTET
*IEM	There is a table with the same name and the suffix IEM storing emissions statistical information based on Panis Pollution Model

Table 4. SQLite tables for microscopic simulations

4. Linking EMS and TOS

As explained before, the input file of trucks generated by TOS defines both arrival time (time a truck is created at a source) and departure gate exit (time a truck leaves the terminal).

Departure gate exit (time) is used directly by EMS since this time matches exactly with the time when the truck enters the surrounding network. On the other hand, the arrival time does not match with the actual time when the truck must enter the surrounding network because the scope of this network is wider than the scope of the access network considered in TOS.

So, as a result of task T6.1.4, it is necessary to implement some convergence criteria to match truck arrival times between EMS and TOS.
At design level, script included in the Aimsun template will support reading specific settings from configuration files in order to apply execution control somehow to stop running replicas if acceptance criteria has been achieved. Also, a specific API in *python* has been developed to provide specific outputs for each replica in order to support this convergence.

This methodology will be tested during task T6.2 (calibration and validation), but the overview of the initial approach is as follows:

1. Run a simulation without private vehicle demand
2. Get, for each truck, its (free-flow) travel time
3. Run as many simulation replicas as configured (as a maximum)
    a. Modify, for each truck, the arrival time from the input file according to the difference between its expected time and the simulated one
    b. Run the simulation
    c. Assess configured criteria
        i. If accepted, then STOP
        ii. Otherwise, go to step 3

Criteria may depend on the calibration process but some examples can be:

- At least a percentage of trucks have an arrival time that differs less than a given amount of time.
- The average difference of expected time and simulated time is less than a given amount of time.
- At least a percentage of replicas fulfil any of the previous criteria.

References

Aimsun 8.2 Help (2015). TSS-Transport Simulation Systems, SL