



JPI URBAN EUROPE

URBAN EUROPE



Smart Cities  
Member States Initiative

## *D3.3 Tool for evaluating real-time strategies*

*Project acronym:* TRANS-FORM

*Project title:* Smart transfers through unravelling urban form and travel flow dynamics

*Funding Scheme:* ERA-NET call on Smart Cities and Communities (ENSCC)

### *Author*

Nikola Obrenovic

[nikola.obrenovic@epfl.ch](mailto:nikola.obrenovic@epfl.ch)

### *Internal Reviewer*

Oded Cats

[o.cats@tudelft.nl](mailto:o.cats@tudelft.nl)

*State:*

*Draft version*

*Distribution:*

*Confidential*



***Deliverable History***

<i>Date</i>	<i>Author</i>	<i>Changes</i>
16-03-2018	<i>Nikola Obrenovic</i>	<i>Draft version</i>
13-08-2018	<i>Nikola Obrenovic</i>	<i>Initial version</i>
30-09-2018	<i>Menno Yap</i>	<i>Internal review</i>
13-03-2019	<i>Nikola Obrenovic</i>	<i>Final version</i>

## Contents

---

Contents.....	3
Summary.....	4
1. Introduction.....	5
2. The Integration Tool Functionalities.....	5
2.1. Main Window.....	5
2.2. Sequences Dialog.....	7
2.3. Properties.....	8
3. Data transformations.....	9
3.1. Urban to Hub level.....	10
3.2. Hub to Urban level.....	11
3.3. Regional to Urban level.....	11
1. Bibliography.....	13

## Summary

---

*The objective of the TRANS-FORM project is to enable the monitoring and analysis of passenger activities and flows within regional and urban public transport networks, and transportation hubs that connect these transportation networks. For this purpose, three separate models have been developed, each focusing on a different public transport network level: hub, urban and regional.*

*The goal of this project is to provide an analysis of real case studies consisting of regional and urban networks connected via hubs. For this purpose of considering the interactions between the different network levels, the models are required to be run in a (predefined) sequence and to exchange data between them. One such sequence will represent a case study.*

*To integrate the models, a software tool named TransFormInt is developed in the scope of this deliverable. The role of the tool is to enable easy definition of the model sequences and control their execution. Also, the tool performs the necessary data transformations of the model outputs into the input data format required by the other models.*

## 1. Introduction

---

In Task 2.2 “Linking hubs and urban networks” and Task 2.3 “Moving between networks”, the models are developed to represent public transport system dynamics on a specific level, i.e. hub, urban and regional level. In addition to that, the required integration of these models has been described with the data flows between individual models.

The specific focus of each model is the following:

- Hub level: The interaction between passengers and their movements within a hub.
- Urban level: The interaction between passengers and vehicles within an urban public transport network operated by one or multiple transport service providers including the passenger flow dynamics.
- Regional level: The interaction between primarily passenger trains, but also regional buses, operated by various service providers in regional/national railway traffic systems.

In this task, these models are integrated by enabling that the results of one model are used as the input data of the other. By this, we enable analysis of the whole multi-level transportation system behaviour, in terms of passenger activities and flow.

To perform the model integration, a software tool named TransFormInt is developed which allows for the definition of different sequences of model executions, and the executions of the sequences themselves. Each sequence represents a scenario, i.e. a case study, of the integrated model usage. The tool is also responsible for converting models’ input and output data between different formats required by the models.

This document presents the results from Task 3.4, i.e. the developed integration tool. In Section 2, we present the tool’s functionalities. In Section 3, the data transformations of model outputs are given. In this document we present the model integration and data transformations from a technical point of view. For the theoretical aspect of the model integration, the reader is directed to the deliverable D2.2: Multi-layer passenger flow network model.

## 2. The Integration Tool Functionalities

---

From the user perspective, the functionalities of the TransFormInt integration tool are the following:

- execution of each individual model (regional, hub or urban) separately;
- execution of predefined and user-defined sequences of models, where results of one model are used as the input for the following models, without user interaction;
- definition and update of the sequences of model executions.

In the following subsections, we present the integration tool GUI elements which allow execution of the above-listed functionalities.

### 2.1. Main Window

The main window of the integration tool is given in Figure 1. The main window is divided into 4 sections:

- Hub model,
- Urban model,

- Regional model, and
- Sequence of models.

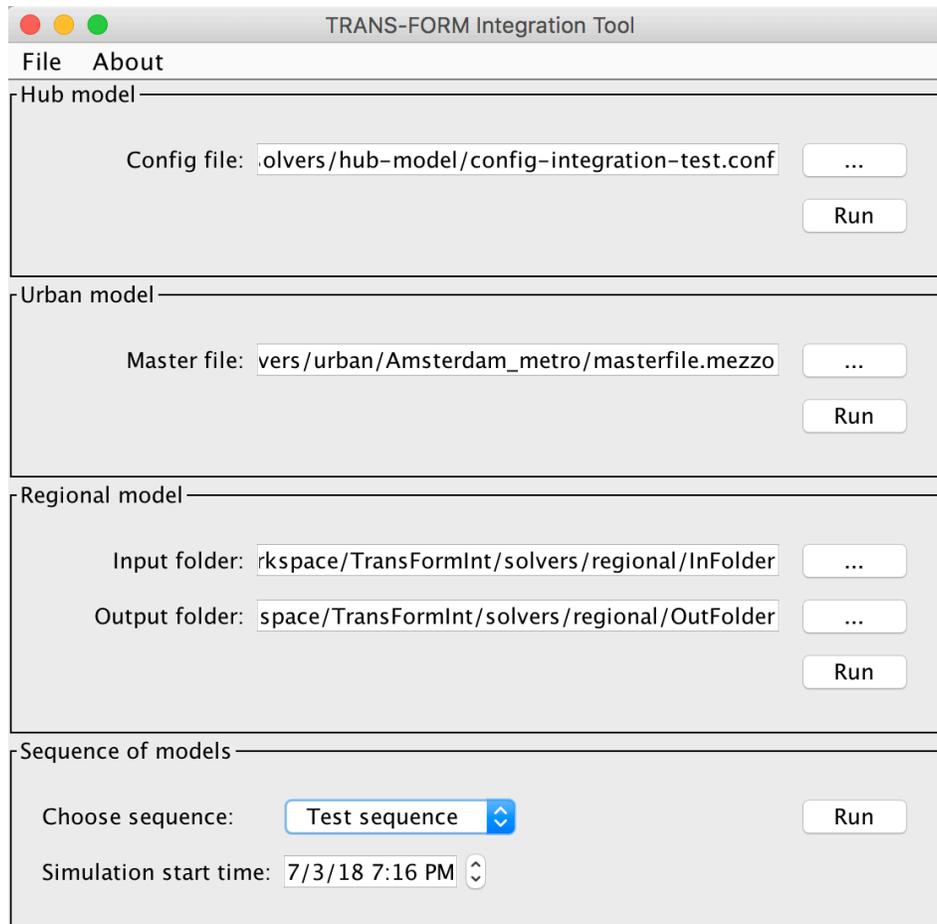


Figure 1. Integration tool main window

In order to run the hub model, the user needs to select the hub configuration file and click the Run button. For running the urban model, the user should select the main BusMezzo masterfile (with .mezzo extension). Finally, for the regional model, the user is expected to select the folder which contains input data of the regional solver, as well as the folder where the output data will be placed. For more information on input/configuration data of the mentioned model, the reader is referred to the corresponding user guides and/or other material.

In the lowest section of the GUI, the user can select one of the defined model sequences, i.e. scenarios, and execute it. In order to start execution, the user must enter the simulation start time, as well as the input or configuration data for all models. The simulation start time will be used as a reference time for aligning internal clock of different model solvers.

## 2.2. Sequences Dialog

The Sequences dialog can be started from the File menu of the main window. It allows creation, update and deleting of the case studies, i.e. the sequences of model executions which are run automatically, without any user interaction required. The Sequences dialog is presented in Figure 2.

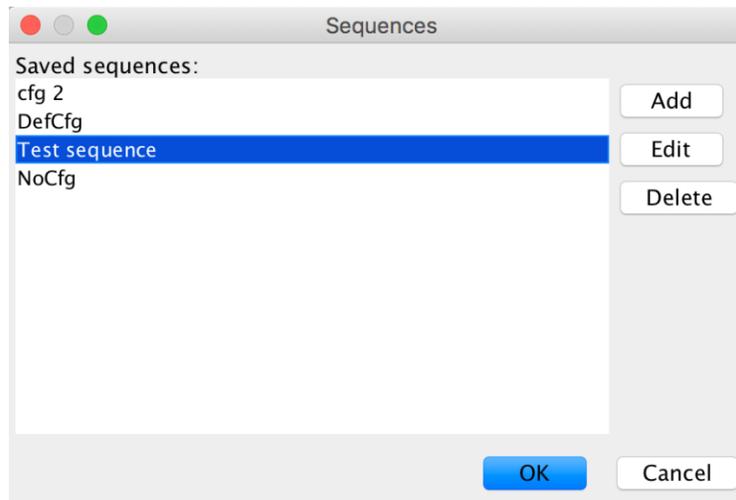


Figure 2. Sequences dialog

When the user clicks on Add or Edit button, another dialog (Figure 3) opens enabling definition of a new model sequence or update of an existing one, respectively. Here, the user can add a sequence step, i.e. a model execution, by clicking the Add step button. This action will create a new combo-box with the Delete button next to it. In the combo-box, the user can select the model she or he wants to be executed in that step. Each step can be deleted by clicking the Delete button next to it.

The user can define a sequence with any order of model executions. However, data is transferred only between certain pairs of models which is described in Section 3.

For each sequence, the user can override the executable file of each model, configuration files for urban and hub models, and input and output folders for the regional model. For this, the user needs to click the Configure button which opens the Sequence Configuration dialog, presented in Figure 4.

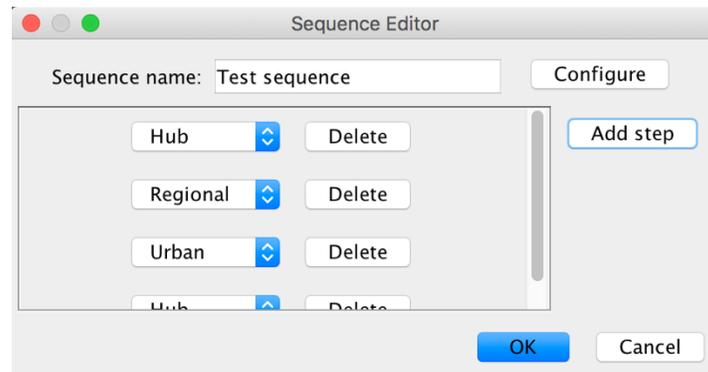


Figure 3. Sequence Editor dialog

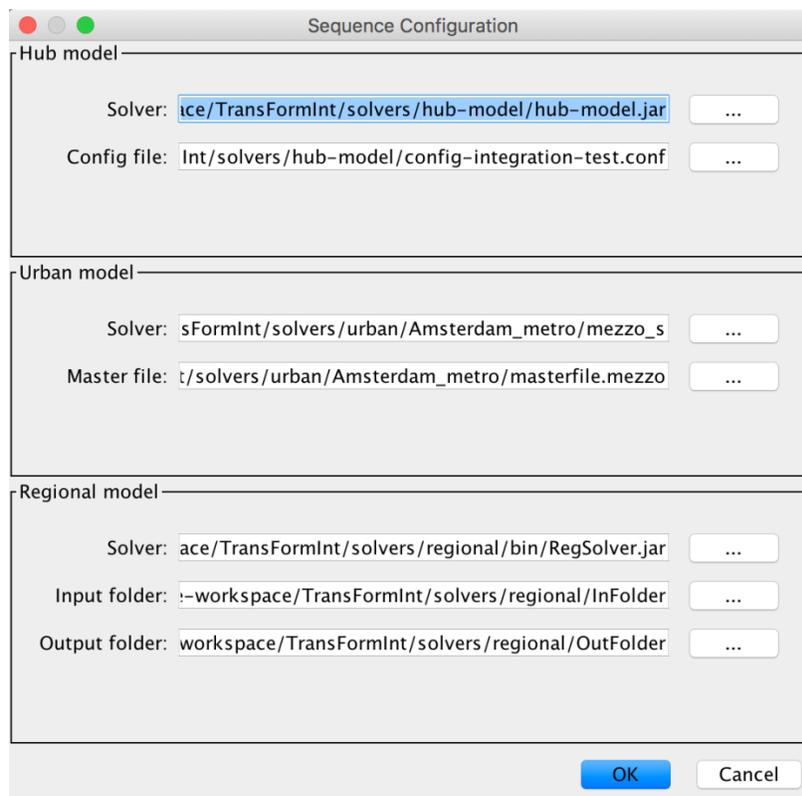


Figure 4. Setting override for a sequence

In the Sequence Configuration dialog, the user can override paths to the default executable files, input, output and configuration data. The override is valid only for the current sequence, i.e. for the case study which is presented by the sequence.

### 2.3. Properties

In the Settings dialog (Figure 5), the user can setup default paths to each of the model solvers.

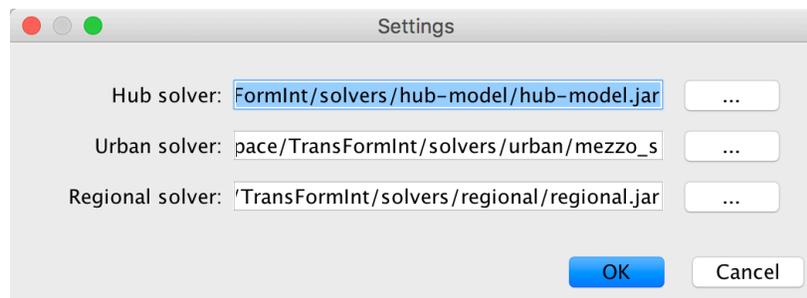


Figure 5. Settings dialog

### 3. Data transformations

The TransFormInt tool performs the following data transformations for the purpose of integrating hub, regional and urban network level models:

1. Urban to hub level: the pedestrian flows and timetables, produced by BusMezzo, are transformed into the hub model input data; and
2. Hub to urban level: the walking time distributions, created by the hub model, are transformed into the BusMezzo input format.
3. Regional to urban level: the revised regional trains timetable, created by the regional model, is transformed into the BusMezzo input data;

In Figure 6, the data flows between the models are illustrated, while the following subsections contain details about the transformations.

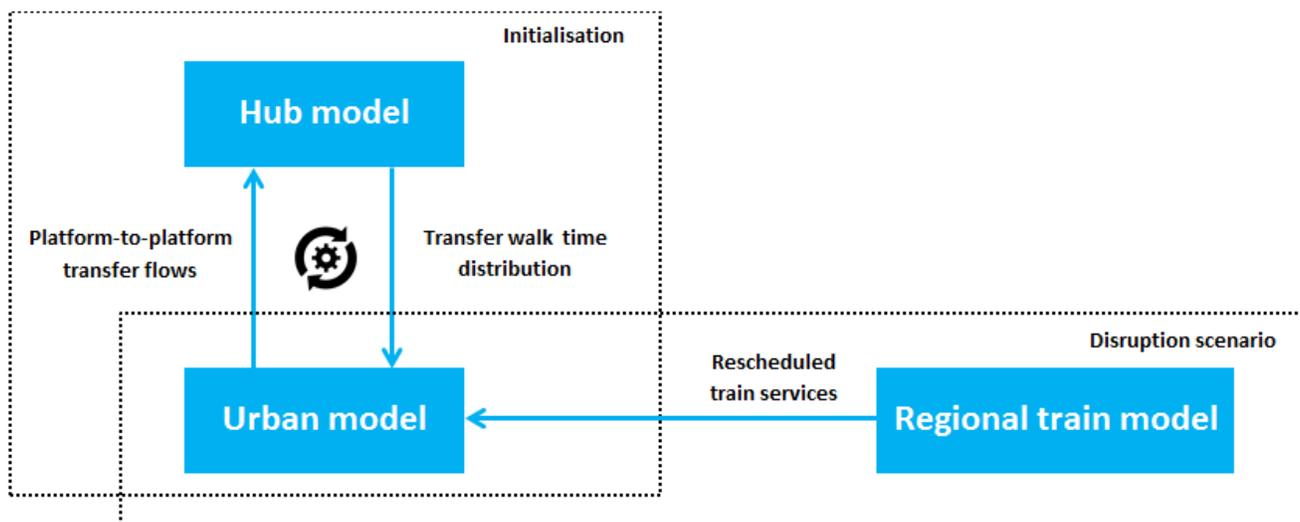


Figure 6. Inter-model data flows

### 3.1. Urban to Hub level

The urban model produces data about the realized arrival and departure times of urban transportation services at the hubs, and pedestrian flows that enter or exit the hubs to and from these services. These data are given in files `transitlog_out.dat` and `selected_paths.dat`, respectively, whose format can be found in [1].

These files are transformed into the json files, specified in the fields `timetable_TF` and `flows_TF` of the hub model configuration file. For the description of hub model data format, the reader is referred to [2]. The data transformations are performed in the following manner.

Once the passenger assignment is performed by the urban model, the integration tool searches the `transitlog_out.dat` file for the entries where field `stop_ID` is equal to one of the hub stops, specified in the hub model input data. Each entry is transformed into a hub model entry using the field mapping given in **Error! Reference source not found.**

Table 1. Timetable entry mapping between urban and hub models

Hub model field	Population rule
<code>stop_id</code>	Field <code>stop_id</code> of the transit log entry from the urban model.
<code>trip_id</code>	Field <code>trip_id</code> of the transit log entry from the urban model.
<code>arrival_time</code>	Field <code>entering_time</code> of the transit log entry from the urban model.
<code>departure_time</code>	Field <code>exit_time</code> of the transit log entry from the urban model.

For the pedestrian flow, the integration tool searches in the file `selected_paths.dat` the passenger path entries which contain a hub stop. Here, we identify the following characteristic scenarios that are of interest to the hub model:

1. The entry contains two subsequent and different hub stops. This means that the passenger has arrived to the hub with one regional or urban trip and left the hub with another.
2. The hub stop is the second stop of the observed passenger path entry. It means that the passenger came by foot to the station and took a mean of urban or regional transportation.
3. The hub stop is one but last stop of the observed passenger path entry. It means that the passenger came to the station by a mean of urban or regional transportation and left the station by foot.

The pedestrian flow data transformation between the urban and hub modes depends on the identified scenario, which can be one of the above-mentioned. Assuming that  $i$  is the index of the first found hub stop in the passenger path entry, the data mapping is given in **Error! Reference source not found.**

Table 2. Selected path entry mapping between urban and hub models

Scenario	Hub model field	Population rule
<b>ALL</b>	<code>pass_id</code>	Field <code>passenger_ID</code> of selected path entry from the urban model.
1	<code>origin</code>	Value at index $i$ of array field <code>list_of_trips</code> of selected path entry from the urban model, prefixed with "t_".
	<code>destination</code>	Value at index $i+1$ of array field <code>list_of_trips</code> of selected path entry from the urban model, prefixed with "t_".
2	<code>origin</code>	Value at index $i-1$ of array field <code>list_of_stops</code> of selected path entry from the urban model, prefixed with "z_".
	<code>destination</code>	Value at index $i$ of array field <code>list_of_trips</code> of selected path entry from the urban

		<i>model, prefixed with “t_”.</i>
3	<i>origin</i>	<i>Value at index i of array field list_of_trips of selected path entry from the urban model, prefixed with “t_”.</i>
	<i>destination</i>	<i>Value at index i+1 of array field list_of_stops of selected path entry from the urban model, prefixed with “z_”.</i>

The prefixes have been introduced for the purpose of differentiating trip and stop identifiers when the data is processed by the hub model.

### 3.2. Hub to Urban level

The hub model output file *walking-times.json*, containing pedestrian walking times distributions, is transformed in the section *stops\_walking\_times* of the BusMezzo input file *transit\_network.dat*. Each field of the destination file is populated according to the mapping given in **Error! Reference source not found.**

Table 3. Mapping walking time distributions from the hub output to BusMezzo data format

Urban model field	Population rule
<i>Orig_stop_name</i>	<i>Field “o” in the hub walking-times.json file.</i>
<i>Dest_stop_name</i>	<i>Field “d” in the hub walking-times.json file.</i>
<i>Time_interval_start</i>	<i>Field “start_timestamp” in the hub walking-times.json file.</i>
<i>Time_interval_end</i>	<i>Field “end_timestamp” in the hub walking-times.json file.</i>
<i>Num_quantiles</i>	<i>Number of elements of the array “quantiles” in the hub walking-times.json file.</i>
<i>Quantile_n</i>	<i>Value of the n<sup>th</sup> element of the array “quantiles” in the hub walking-times.json file.</i>
<i>Walking_time_n</i>	<i>Value of the n<sup>th</sup> element of the array “values” in the hub walking-times.json file.</i>

### 3.3. Regional to Urban level

Between the regional and urban models, TransFormInt is responsible for transforming the revised timetable of regional trains, created by the regional model, into the input data of BusMezzo.

The regional model format of the revised timetable is nearly the same as the urban model format, described in Section “B.5.1 Complete time-table (format 1)” of [1]. The two formats differ only in the following:

1. BusMezzo trip ID is obtained by removing the last digit from the train number used by the regional model;
2. BusMezzo line id is obtained by removing the first and last digits from the train number used by the regional model;
3. the times in the regional output file are given as the number of seconds from the beginning of the day, while the urban model expects them to be the number of seconds from the simulation start; and
4. in the regional model format, the braces are missing.

Hence, the necessary data transformations between the regional and urban levels are only to add the missing braces and transform the regional level times  $t_{reg}$  by using the reference time  $t_{ref}$ , specified in the main window, with the following formula:

$$t_{urb} = t_{reg} - t_{ref}.$$

*In this transformation, TransFormInt will only accept revised timetable entries which already exist in the BusMezzo input data, and update the corresponding entries in the transit\_network.dat file. If the trip identifier from the regional model output is unknown, such entry will be ignored by the integration tool.*

## 1. Bibliography

---

- [1] O. Cats and D. Leffler, *BusMezzo Input/Output Formats v2.1*, 2018.
- [2] N. Molyneaux, *Documentation for the hub model*, 2018.
- [3] J. Törnquist Krasemann, *Outline of the current data structure and file specification for the optimization approach*, 2017.
- [4] R. Scarinci, O. Cats, J. Törnquist Krasemann, F. Hänseler and N. Molyneaux, *D2.2: Multi-layer passenger flow network model*, 2017.