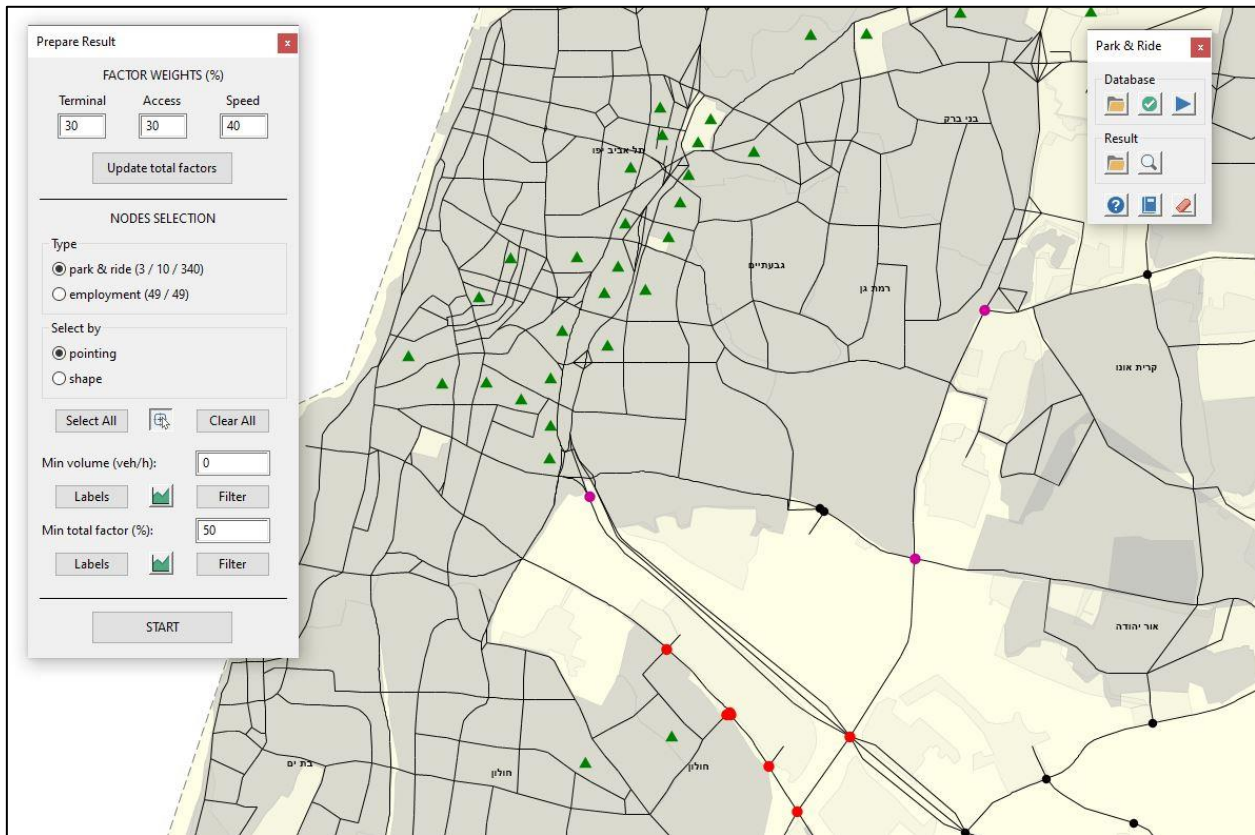


Park & Ride Model

User Guide



Ayalon Highways Co.

November 2021

Table of Contents

Installing	3
Park & Ride Desktop	6
Databases	8
Results	17

Installing

Welcome to Park & Ride Model (PRM), a TransCAD based application allowing to analyze different park and ride sites in Israel.

PRM is supported on 64-bit Microsoft Windows and requires Caliper TransCAD Full 8 software installed on the system.

To install the application:

1. Download the application archive *prm_public-<version_number>.zip*
2. Unarchive it into any preferred folder (*prm_folder*)
3. Launch *prm_folder/install.bat*

Note: Do not move *install.bat* file outside of *prm_folder*. If you move the whole *prm_folder* to a new location, you have to re-run *install.bat* script.

Console window will pop up with one of the statuses shown below:

- the installation successfully finished ([Figure 1](#))

You can move to step 4 from here.

```

C:\WINDOWS\system32\cmd.exe

Park & Ride Model was successfully installed.
You can launch the application by double clicking PRM shortcut on the Desktop.

Press any key to exit . . .
  
```

Figure 1. Successful installation

- the installation failed because environment variable *TCF* is not set ([Figure 2](#))
 Press *y* to open environment variables window ([Figure 3](#)) and click *New...* in user variables section. Type *TCF* as variable name and type TransCAD Full 8 installation folder path as variable value (for example, default path is *C:\Program Files\TransCAD 8.0*). Click *OK* and then again *OK* to close all windows. Repeat step 3.

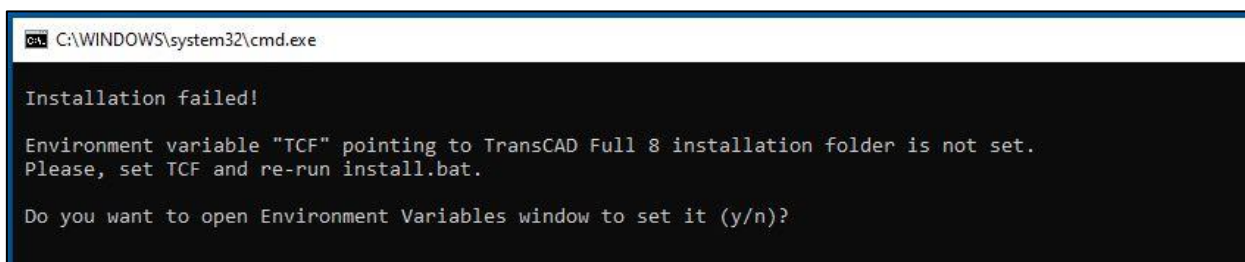


Figure 2. Installation failed

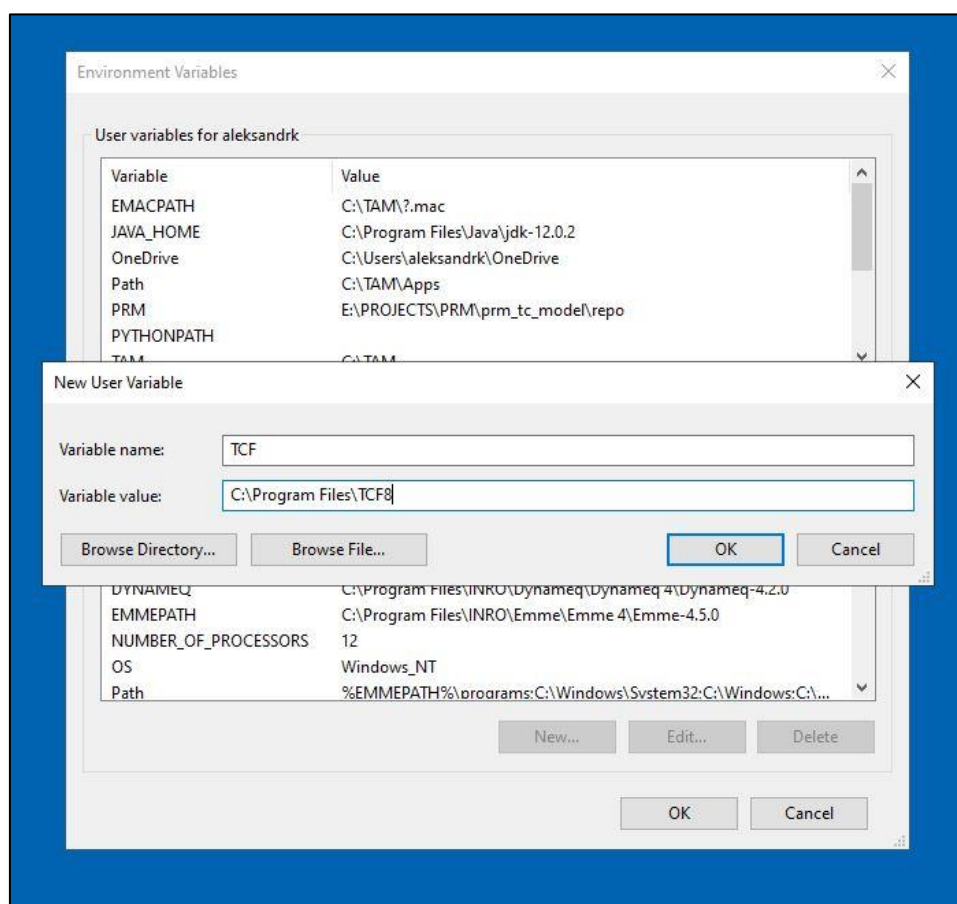


Figure 3. Creating TCF environment variable
for TransCAD Full 8 located in "C:\Program Files\TCF8"

4. Launch the application via PRM shortcut that is automatically placed by the installer to user Desktop ([Figure 4](#) and [Figure 5](#)).

Note: Every time you launch *install.bat*, the shortcut is re-created. If you accidentally deleted the shortcut, re-run the installer to place a new shortcut to the Desktop.



Figure 4. PRM shortcut

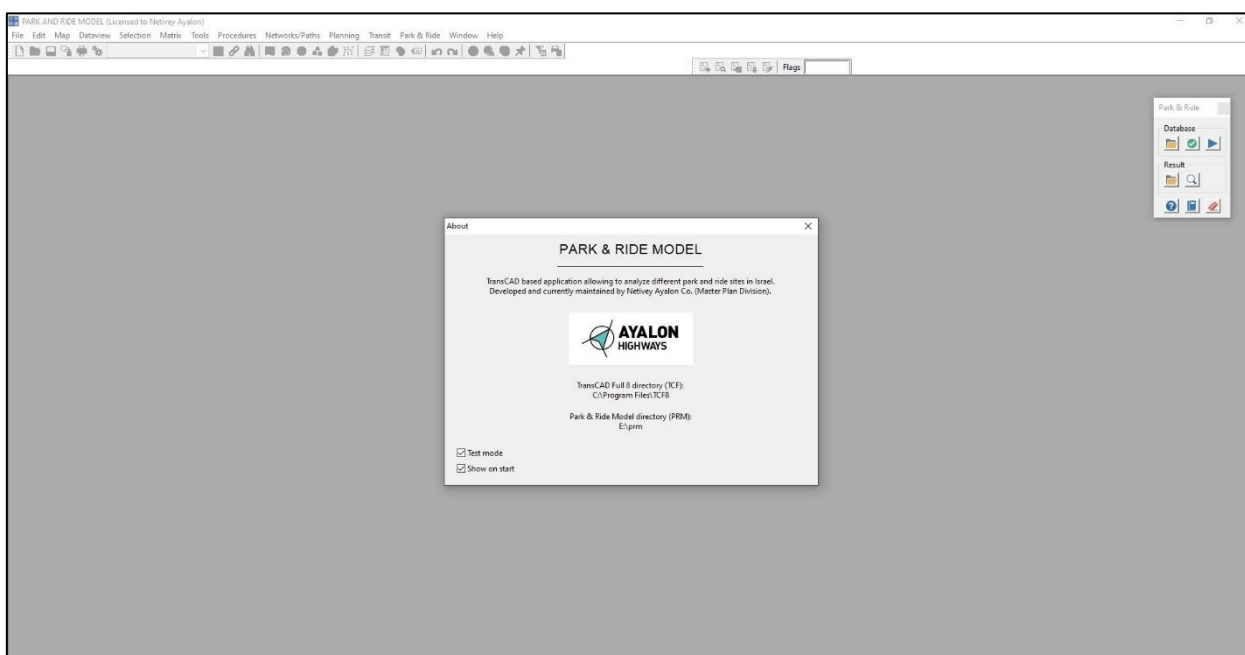


Figure 5. Welcome to Park & Ride Model

Park & Ride Desktop

When you open Park & Ride application for the first time, *About* window will pop up showing short description of the model, values stored in environment variables *TCF* (points to TransCAD Full 8 folder) and *PRM* (points to *prm_folder*) as well as allowing to adjust 2 settings:

- *Test mode* sets whether the application functions in *Test mode* (*On* by default) When *Test mode* is activated, the model executes traffic assignments with only 3 iterations. Although, the output of such assignments is worthless, *Test mode* is useful for playing with time consuming procedures.
- *Show on start* controls whether to show *About* window each time you open the application (*On* by default)

There is *Park & Ride menu* (Figure 6) at the top of the screen that allows to restore *Park & Ride toolbox*, create a database from default one, open this user guide (*Help*), and open *About* window.

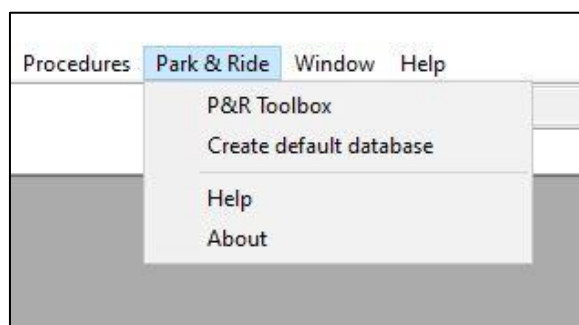


Figure 6. Park & Ride menu

There is *Park & Ride toolbox* (Figure 7) on the right top corner of the screen that has 3 sections:

- *Database* section provides methods to open a database, and launch 2 procedures: database preparation and result preparation
- *Result* section provides methods to open a result and post-process it (*Show Origins and Destinations* tool)
- section at the bottom provides methods to open *About* window, open this user guide and close all windows (TransCAD maps, editors, views, figures, etc. except for PRM windows) with option to save changes.

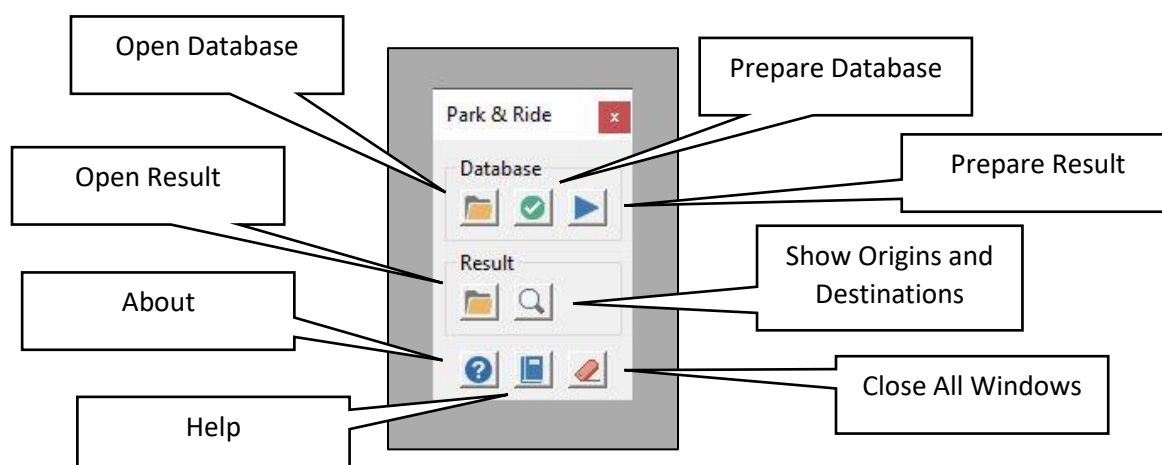


Figure 7. Park & Ride toolbox

Note: In order to launch application procedures and tools, PRM requires to open a database / result via *Park & Ride toolbox* only. It is possible to open only 1 database / result at a time via *Park & Ride toolbox*.

Databases

All databases are stored within *prm_folder/dbs* directory. The application distribution does not contain any databases after installation. In order to add a database, you can either create it from default one or copy your existing database to *prm_folder/dbs* directory.

You can create a new database from default one using *Create default database* method from *Park & Ride menu*. Click *Create default database* from *Park & Ride menu*, type new database name and click *Create* ([Figure 8](#)).

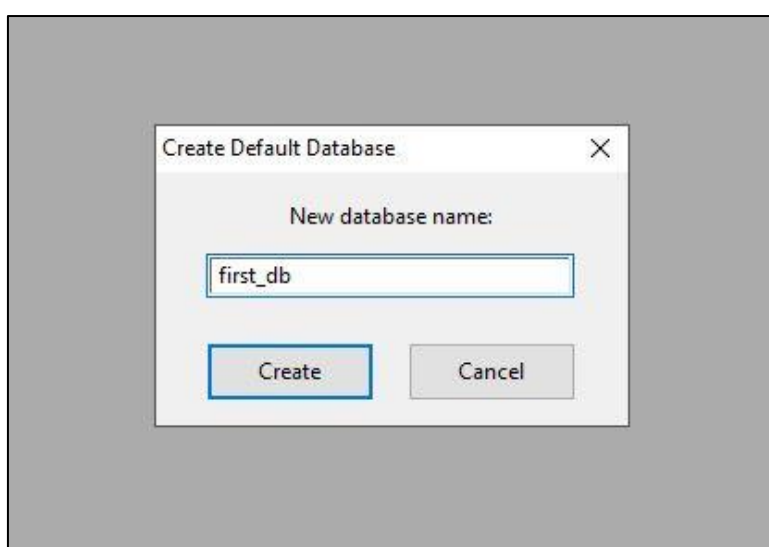


Figure 8. Creating default database called “first_db”

To open a database, click *Open Database* button from *Park & Ride toolbox*. Explorer window will open showing all existing databases. Double click on preferred database folder to go into it, select its *database.map* file, and click *Open* ([Figure 9](#)).

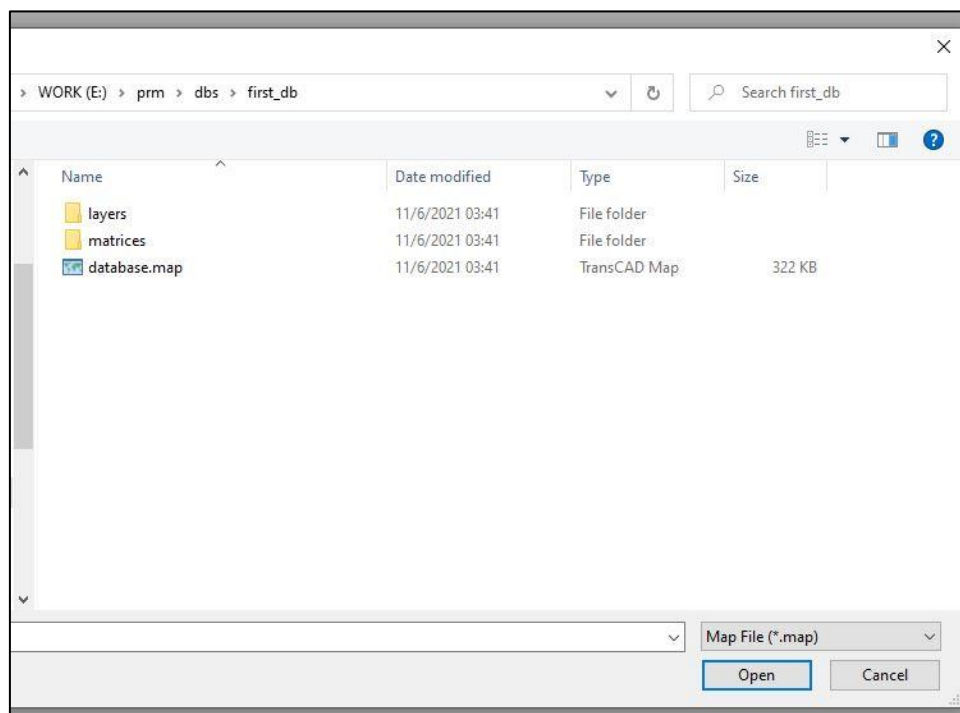


Figure 9. Opening database “first_db”

In order to prepare results, you first need to prepare a database. Click on *Prepare Database* button from *Park & Ride toolbox*. Currently open database will be automatically checked if the database is ready for preparation.

Note: The check inspects only the presense of layers and fields required by the database preparation procedure. It does not check if fields have acceptable values.

If some layer or field is missing, a message about problems ([Figure 10](#)) will pop up, and the application will not let you run the procedure until you fix the problems.

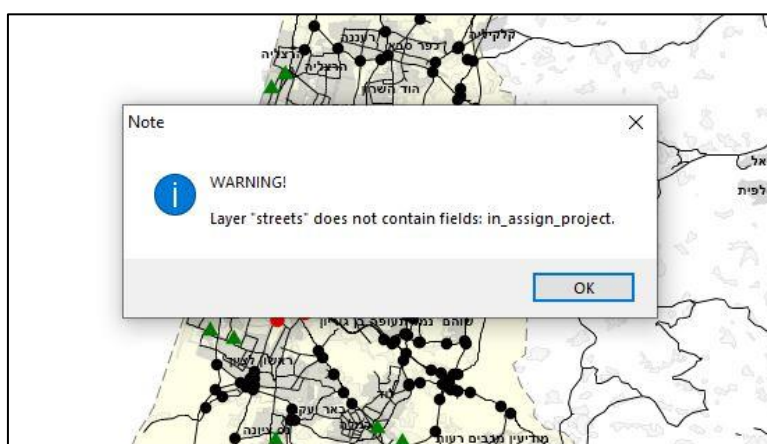


Figure 10. Warning about missing field

The following list shows database structure and layer fields required to launch database preparation procedure:

- layers
 - in_project_zone
 - *in_project_zone.dbd*

Contains *in_project_zone* layer with fields:
 - *ID* (integer, index)
 - *Area* (real, 2 decimals)
 - is_cty_1
 - *is_cty_1.dbd*

Contains *is_cty_1* layer with fields:
 - *ID* (integer, index)
 - *Area* (real, 2 decimals)
 - *cityname* (character)
 - is_cty_2
 - *is_cty_2.dbd*

Contains *is_cty_2* layer with fields:
 - *ID* (integer, index)
 - *Area* (real, 2 decimals)
- streets
 - *streets.dbd*

Contains *streets* layer with fields:
 - *ID* (integer, index)
 - *Dir* (integer)
 - *Length* (real, 2 decimals)
 - *Road_Number* (integer)
 - *[AB Type]* (integer)
 - *[BA Type]* (integer)
 - *[AB VDF]* (integer)
 - *[BA VDF]* (integer)
 - *[AB Lanes]* (real, 1 decimal)
 - *[BA Lanes]* (real, 1 decimal)
 - *AB_toll* (real, 2 decimals)

- *BA_toll* (real, 2 decimals)
- *[AB ci]* (real, 2 decimals)
- *[BA ci]* (real, 2 decimals)
- *[AB tk_ti]* (real, 2 decimals)
- *[BA tk_ti]* (real, 2 decimals)
- *in_project* (integer)

Contains *nodes* layers with fields:

- *ID* (integer, index)
- *Longitude* (integer)
- *Latitude* (integer)
- *Elevation* (real, 2 decimals)
- *Centroid* (character)
- *emp_area_id* (integer)
- *emp_area_name* (character)

- terminals

- *terminals.dbd*

Contains *terminals* layer with fields:

- *ID* (integer, index)
- *Longitude* (integer)
- *Latitude* (integer)
- *rating* (real, 2 decimals)

- matrices

- *nl_sum.mtx* (demand matrix)

- *database.map*

If all database layers and fields are present, *Prepare Database* window will pop up showing that database is ready for preparation. Click *Start* button to launch the procedure ([Figure 11](#)).

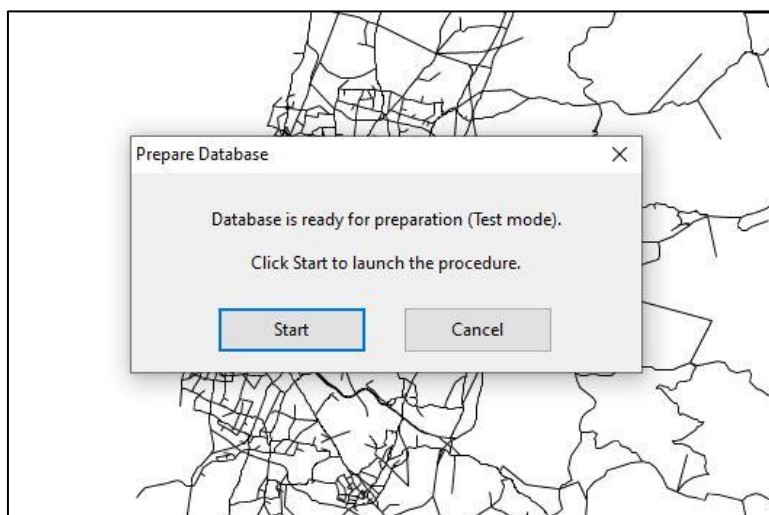


Figure 11. Prepare Database window

It will take no more than 20 minutes (depending on your computer performance) to finish in *Test mode*. If you run the procedure in *Real mode* (when *Test mode* setting is disabled), it can take around 24 hours to finish.

Note: The progress bar of TransCAD traffic assignments is constructed based on the maximum number of iterations defined. PRM assignments have it equal to 500 (in *Real mode*) but in fact, it takes around 250 iterations for default database to converge.

The following list describes all steps of database preparation procedure:

- creating new folder within *prm_folder/dbs* directory
The name of the folder (*preb_db_folder* further) consists of several parts: word *test* is pasted at the beginning only when launched in *Test mode*, the second part is the name of the source database, and the last part is datetime taken at the moment the procedure was launched.
- copying the source database into *preb_db_folder* and opening *preb_db_folder/database.map* file
- creating *preb_db_folder/extra* subfolder for storing new files produced by the procedure
All file paths are further provided relative to *preb_db_folder/extra* directory (if not explicitly defined).
- deleting all links of type 91 (if present)
- creating and filling link fields:
 - *A_node* (start node ID)
 - *B_node* (end node ID)

- *road_6* (value 1 if AB or BA link is part of road 6, otherwise 0)
- *AB_road* (value 1 if AB link is of type 1-6, 9, or 11-14, otherwise 0)
- *BA_road* (value 1 if BA link is of type 1-6, 9, or 11-14, otherwise 0)
- *AB_main_road* (value 1 if AB link is of type 1, 2, or 3, otherwise 0)
- *BA_main_road* (value 1 if BA link is of type 1, 2, or 3, otherwise 0)
- creating node fields (*initial_volume*, *terminal_factor*, *access_factor*, *weighted_speed*, and *speed_factor*)

The fields will be filled during next steps.

- creating selections for *is_cty_1* and *is_cty_2* layers
 - *area>2* (municipalities with area more than 2 km²)
- creating selections for *streets* layer
 - *in_project* (links with value 1 in *in_project* field)
 - *connectors* (subset of *in_project* links of type 90)
 - *type_3* (subset of *in_project* links of type 3)
 - *type_2* (subset of *in_project* links of type 2)
 - *type_1* (subset of *in_project* links of type 1)
 - *toll* (subset of *in_project* links with not NaN value in fields *AB_toll* or *BA_toll*)
 - *unused* (links not included in *in_project* selection)
- creating selections for *nodes* layer
 - *centroids* (nodes with value X in *Centroid* field)
 - *emp* (employment zones that are nodes with not NaN value in *emp_area_id* field)

Employment area is a geographical group of employment zones.

- *emp_virt* (employment zones that will be used during result preparation procedure)

The selection is set by user via *Prepare Result* tool.

- *ipz* (nodes not included in *centroids* selection and located within project zone area that is the first polygon from *in_project_zone* layer)
- *nic__ipz* (subset of *ipz* nodes located outside of municipality polygons from *is_cty_1* and *is_cty_2* layers)
- *nic__ipz__dummy* (subset of *nic__ipz* nodes that have 2 or less connections with links from *in_project* selection)
- *nic__ipz__dangling* (subset of *nic__ipz__dummy* nodes that have no connections with links from *in_project* selection)
- *nic__ipz__not_dummy__type>10* (subset of *nic__ipz* nodes that are not included in *nic__ipz__dummy* selection and have connections with *in_project* links of type higher than 10 only)

- *pr* (*nic_ipz* nodes except for nodes from *nic_ipz_dummy* and *nic_ipz_not_dummy_type > 10* selections)
- *pr_virt* (subset of *pr* nodes that will be used as possible park and ride locations during result preparation procedure)

The selection is set by user via *Prepare Result* tool.
- *pr_query* (subset of *pr_virt* nodes that are considered as points of interest during result preparation procedure)

The selection is set by user via *Prepare Result* tool.
- running assignment with queries (points of interest) for all *pr* nodes (defined in *queries.qry* file)

The assignment is launched with network file *car.net*. The output files are *link_flows.bin*, *movements.bin*, and *critical.mtx* (it has a separate core for each *pr* node). The assignment takes the most of the time during database preparation.
- filling node field *initial_volume* by taking *VOLUME* values from *movements.bin*, grouping them by field *NODE*, and summing up for each node
- creating matrix *shortest_paths.mtx* based on times obtained from *link_flows.bin* file (*AB_Time / BA_Time*) and with network file *shortest_paths.net*

The origins of the matrix are *pr* nodes, and destinations are *emp* zones.
- creating matrix *vht.mtx*

The origins of the matrix are *pr* nodes, and destinations are *emp* zones.

For each *pr* node, its critical matrix core is taken (*critical.mtx* file) and then for each *emp* zone, its column from the core is summed resulting in volume going from all centroids via a park & ride node to an employment zone.

For each *pr*-node-to-*emp*-zone pair, according time value is taken from *shortest_paths.mtx* file. Then, volume and time values are multiplied and divided by 60 resulting in vehicle-hours time (VHT) value.
- calculating park and ride node terminal factors (saved to *terminal_factor* field)

The factor shows accessibility to transit terminals. Layer *terminals* has ratings for each terminal (*rating* field): 1 for buses / BRT, 2 for metro / LRT, and 3 and 4 for trains.

The algorithm is the following:

if (no terminals in 600-meter radius):

factor is set to 0

elif (at least one of the terminals in 600-meter radius has rating 3 or 4):

factor is set to 3 or 4 (max rating is taken)

else: // there are terminals but all with rating 1 or 2 (no train)

if (no terminals in 300-meter radius):

factor is set to 0 // meaning that we do not want to go more than 300 meters if it is not a train station

else:

factor is set to 1 or 2 (max rating is taken)

- calculating park and ride node access factors (saved to *access_factor* field)

The factor shows accessibility to main roads (links of type 1, 2, or 3).

Only links included in *in_project* selection but not included in *toll* selection are considered for the factor.

The algorithm is the following:

if (node is directly connected to at least one main road):

factor is set to 2

elif (node is connected to at least one main road over 1 another node):

factor is set to 1

else:

factor is set to 0

- calculating park and ride node speed factors (saved to *speed_factor* field)

The factor shows how fast it is possible to reach employment zones from a park and ride location. In order to define speed factor, weighted speeds are calculated for each park and ride node (described below). Then factor is set to 0 if weighted speed is less than 30 km/h, 1 if weighted speed is no less than 30 km/h but less than 45 km/h, and 2 if it is no less than 45 km/h.

For calculating weighted speeds, the 3rd core *speed* within *shortest_paths.mtx* matrix is created as core *length* divided by core *time* and multiplied by 60 to get values in km/hour.

Weighted speed (stored in *weighted_speed* field) is calculated in the following way for each *pr* node:

- take a row from core *speed* resulting in vector of speeds from a park and ride node to *emp* zones
 - take a row from *vht.mtx* resulting in vector of VHT values from a park and ride node to *emp* zones
 - calculate VHT sum by summing vector of VHT values
- If the sum is 0, stop here and set park and ride node weighted speed to 0.
- calculate vector of weights as vector of VHT values divided by VHT sum

- multiply vector of speeds by vector of weights and sum it to get park and ride node weighted speed
- setting styles for layers and selections
- creating file *preb_db_folder/prepared*

After the procedure finished, status message will pop up ([Figure 12](#)) showing its runtime, and a newly created and prepared database is left open.

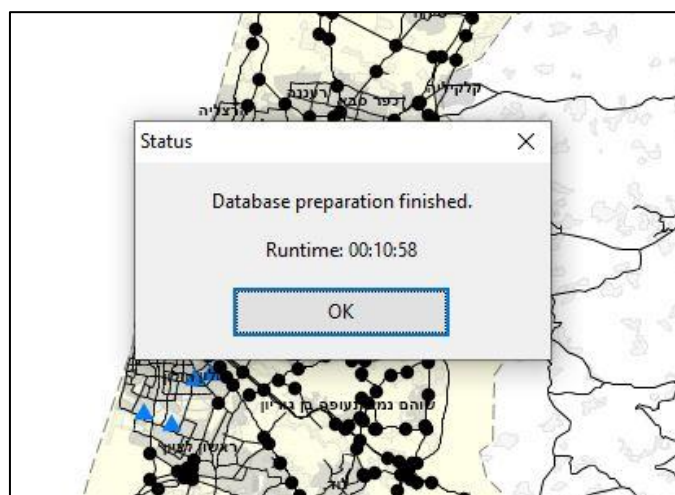


Figure 12. Status message after database preparation

Results

Once you have a prepared database, you can start to prepare results, i.e. run different experiments to examine park and ride locations. All results are produced by result preparation procedure and saved within *prm_folder/results* directory.

After you opened a prepared database, click on *Prepare Result* button from *Park & Ride toolbox*. The database will be automatically checked if it is ready for result preparation.

Note: The check inspects only the presense of layers and fields required by the result preparation procedure. It does not check if fields have acceptable values. It is assumed that user does not change a database since its preparation. Therefore, the presense of required selections is also not checked. Each time you change something in a database that can theoretically impact on assignment results or content of selections, you must re-prepare the database.

If some layer of field is missing, a message similar to the one shown in [Figure 10](#) will pop up, and you will not be able to run the procedure until you fix the problems.

If all fields and layers are present, *Prepare Result* tool will open ([Figure 13](#)).

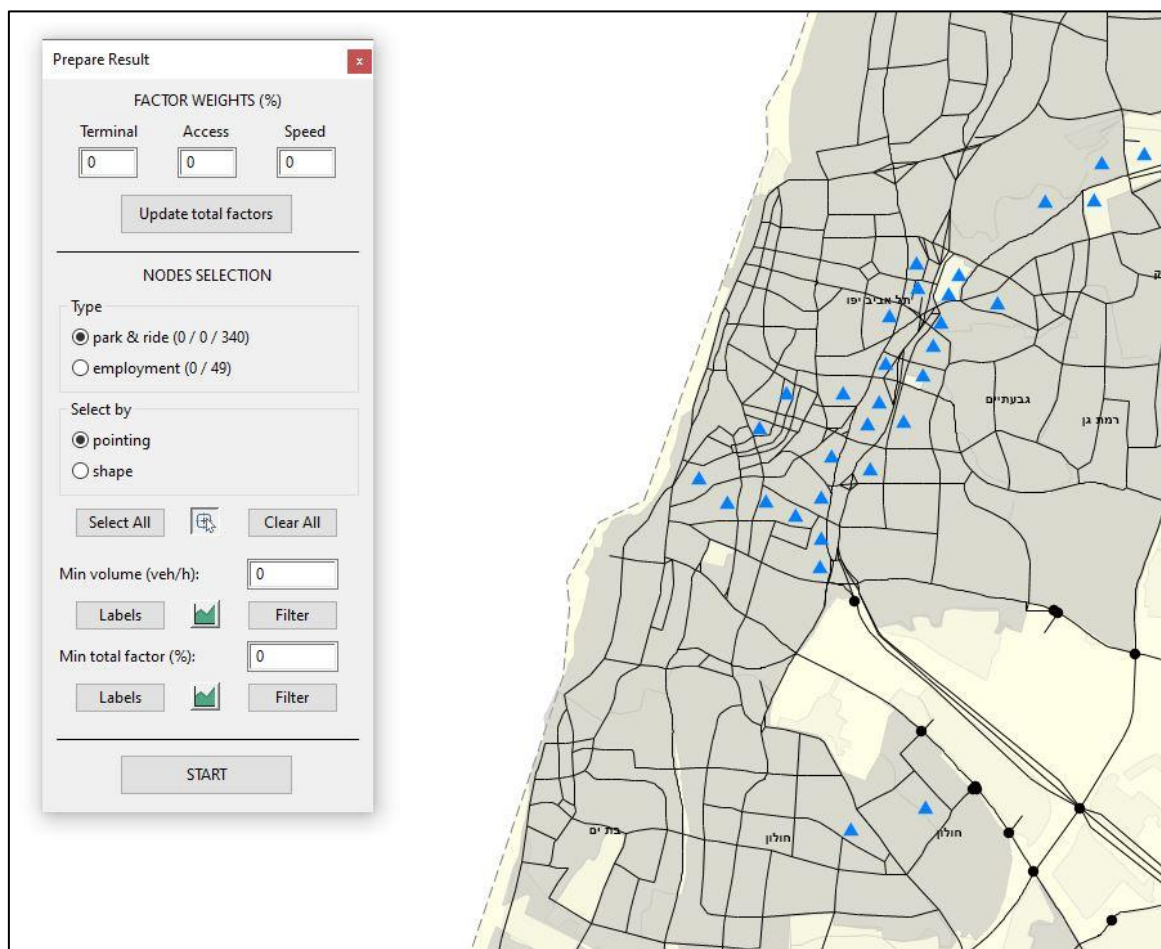


Figure 13. Prepare Result tool

When you open the tool, *total_factors* field is created (if not present) and total factors are re-calculated for *pr* nodes using factor weights. A park and ride node total factor is calculated as $(terminal_factor / 4 * terminal_factor_weight) + (access_factor / 2 * access_factor_weight) + (speed_factor / 2 * speed_factor_weight)$. All factor weights must be defined, and their sum must be equal to 100.

If you would like to change factor weights, type their values and click *Update total factors*.

Right after database preparation, there is no park and ride nodes included in *pr_virt* selection. These nodes will be used as possible park and ride locations during result preparation procedure. Technically, only these nodes will be used as start nodes for building virtual links.

Make sure that *Type* is set to *park & ride* and select nodes from map. Black nodes represent all possible park & ride locations currently not selected. After selection, the nodes will be included in *pr_virt* selection and change their color to red (Figure 14).

You can select nodes one by one with *Select by pointing*, or by shape with the according option. If you want to deselect node, press *Ctrl* button while making pointing or shape selection. If you want to include or clear all nodes at once, you can click *Select All* or *Clear All* buttons.

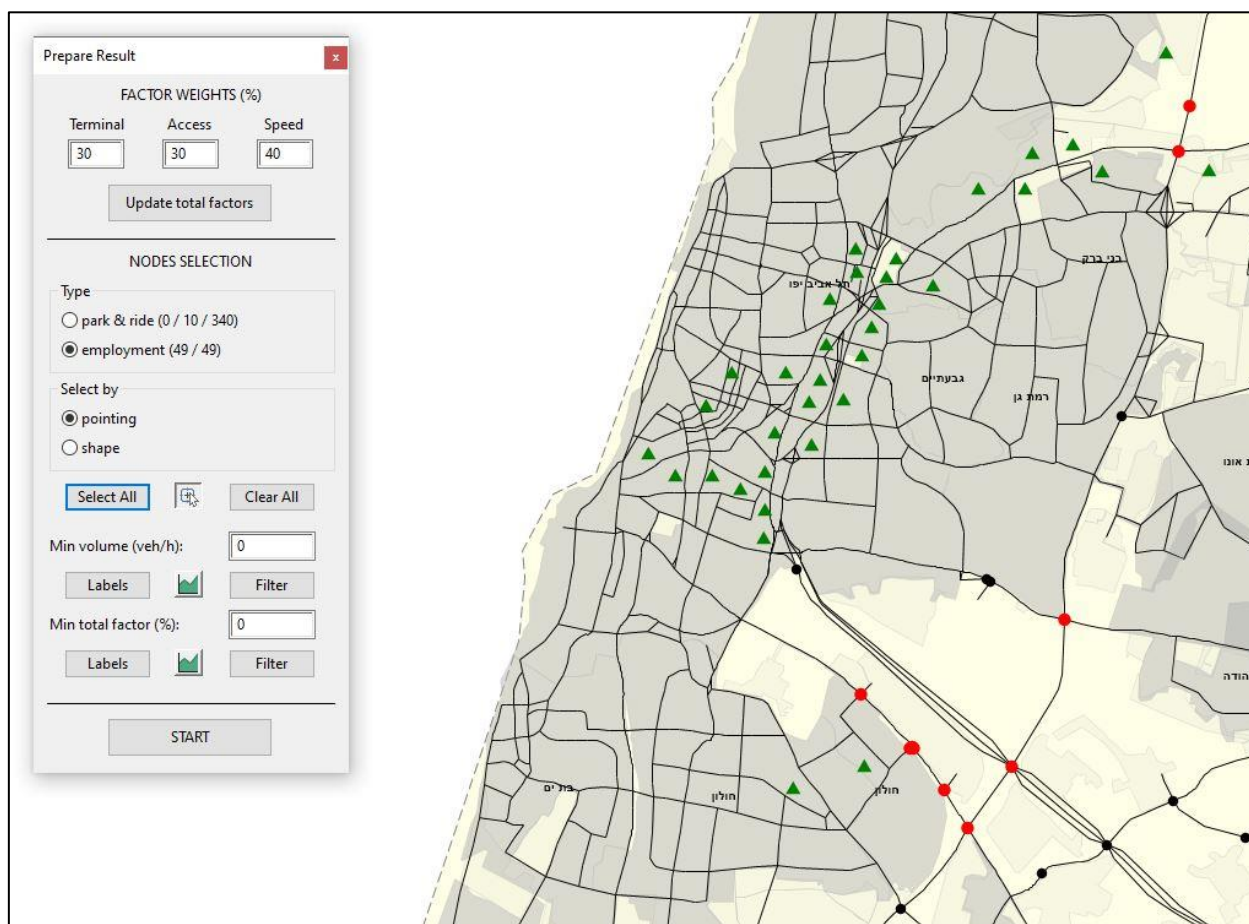


Figure 14. Prepare Result tool

after selecting park and ride nodes and employment zones

There is another selection called *pr_query* that is a subset of *pr_virt* nodes. If there is at least one node in *pr_query* selection, the procedure will create critical matrix with separate core for each query node (point of interest) that can be further investigated via *Show Origins and Destinations* tool. In order to include a node not only to *pr_virt* but also to *pr_query* selection, press *Shift* button while selecting it from map. Nodes selected as points of interest will change color to purple. If you want to deselect a node only from *pr_query* but save it in *pr_virt* selection, press both *Ctrl* and *Shift* button while selecting.

You can see number of nodes in the selections to the right from the according *Type* option (“*pr_query / pr_virt / pr*”).

Park and ride node volumes (from field *initial_volume*) and total factors (from field *total_factor*) can be used for filtering and plotting graphs. Type minimum volume or minimum total factor in the according fields and click *Filter* (for example, if minimum volume is set to 200, then filtering will exclude all nodes among those in *pr_virt* selection that have volume less than 200). If you click on the middle button with graph icon, a chart window ([Figure 15](#)) will open showing all unselected nodes sorted by the according feature (in black color), and all selected nodes sorted as well (in red color).

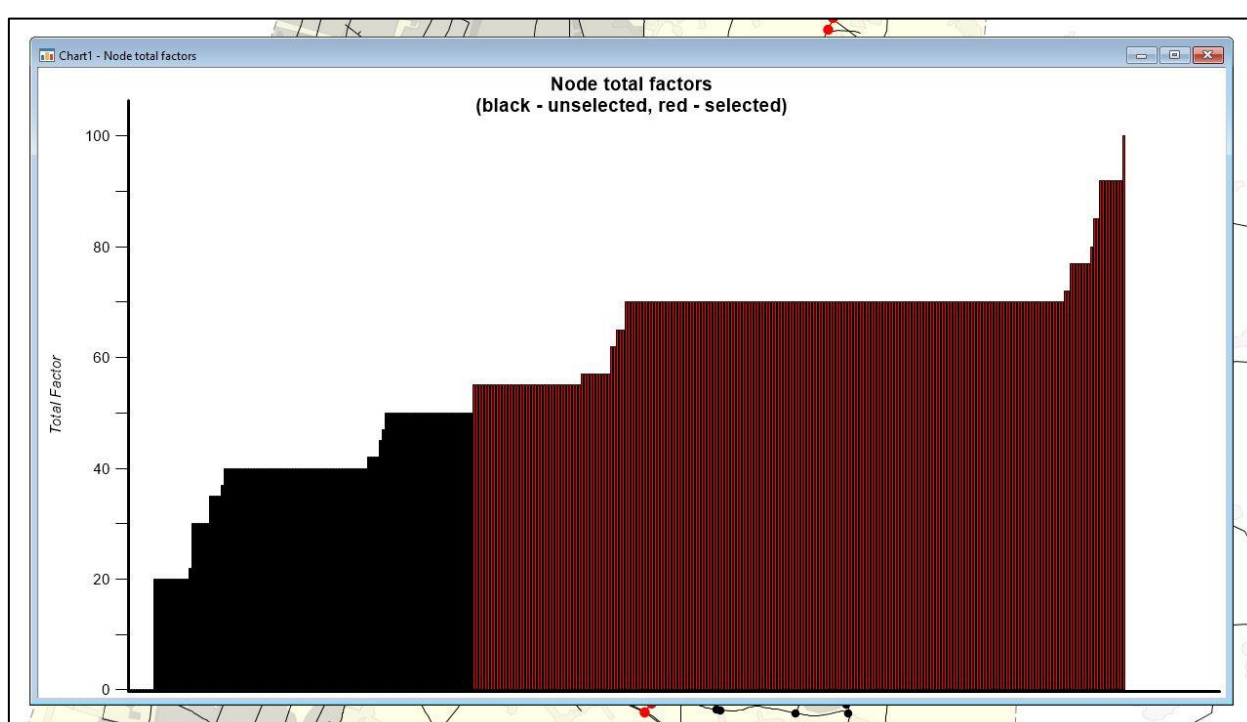


Figure 15. Plotting park and ride nodes based on total factors

If you want to examine park and ride node volumes or total factors, you can click *Labels* button of the according feature to show values on map.

The same selection must be done for employment zones (*emp_virt* selection). Only *emp_virt* zones will be used as employment zones during result preparation. Technically, only these zones will be used as end nodes for building virtual links.

Make sure that *Type* is set to *employment* and select zones from map. Blue triangles represent all employment zones currently not selected. After selection, the zones will be included in *emp_virt* selection and change their color to green ([Figure 14](#)). You can select zones by pointing or shape, deselect while pressing *Ctrl* button (*Shift* is not

relevant in this case), select and clear all of them, and see number of zones in the selections to the right from the according *Type* option (“*emp_virt / emp*”).

Once you finished defining the selections, click *Start* to open *Start* window ([Figure 16](#)) from where you can define additional settings for park and ride nodes (minimum volume to employment area) and for virtual links (minimum length and 3 speeds for each terminal factor). The settings are described below.

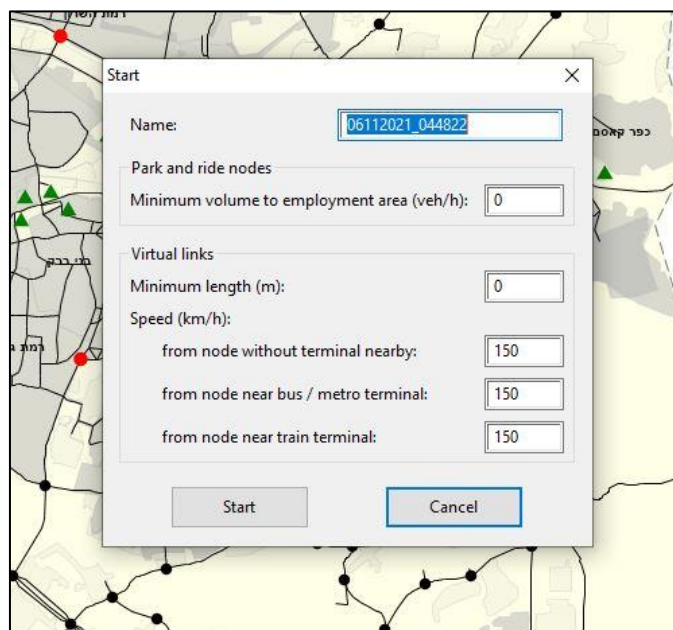


Figure 16. Launching result preparation

After you defined the settings, you can run the procedure by clicking *Start* button. It will take no more than 3 minutes to finish in *Test mode*. If you run it in *Real mode* (when *Test mode* setting is disabled), it can take around 25 minutes to finish for experiments without points of interest.

Note: The actual time will be longer if you define some nodes as points of interest since most of the time in TransCAD traffic assignments is spent on queries.

The following list describes all steps of the procedure:

- creating new folder within *prm_folder/results* directory
The name of the folder (*res_db_folder* further) consists of several parts: word *test* is pasted at the beginning only when launched in *Test mode*, the second part is the result name defined in settings and being a datetime by default.
- creating *res_db_folder/db* subfolder, copying the source database into it, and opening *res_db_folder/db/database.map* file

- creating *res_db_folder/extra* subfolder for storing new files produced by the procedure
All file paths are further provided relative to *res_db_folder/extra* directory (if not explicitly defined).
- saving procedure settings to *res_db_folder/settings.txt* file
- deleting all links of type 91 (if present)
- creating link fields (*virtual*, *to_emp_area_id*, and *from_node_to_emp_area*)
The fields will be filled during next steps.
- creating node fields (*vht_saved*, *rank*, and *position*)
The fields will be filled during next steps.
- creating virtual links from each *pr_virt* node to each *emp_virt* zone
Virtual link fields are filled with the following values:
 - *[AB Type]* – 91
 - *[AB VDF]* – 4
 - *[AB Lanes]* – 9
 - *[AB tk_ti]* – $virtual_link_length * 60 / virtual_link_speed$
Value for *virtual_link_speed* is chosen among 3 speeds from settings based on terminal factor. For example, for all virtual links going from park and ride nodes with terminal factor 2 (i.e. there is a train station nearby), train speed defined by user will be applied in the formula.
 - *[AB ci]* – 99999
 - *in_project* – 0
 - *virtual* – 0 or 1
Virtual links with length less than defined in setting *Minimum length* are created but not used in assignments at all. They are gathered in selection *virt_short*, and get value 0 in the field. Links with acceptable length are gathered in selection *virt_assign_1*, and get value 1 in the field.
 - *A_node* – start node ID
 - *B_node* – end node ID
 - *to_emp_area_id* - employment area ID of B node
 - *from_node_to_emp_area* - concatenation of field *A_node* and *to_emp_area_id*
- running the 1st assignment
The assignment is launched with network file *car_1.net*. The output files are *link_flows_1.bin* and *movements_1.bin*. Links used in the assignment are union of *in_project* and *virt_assign_1* selections (gathered in selection *in_project__virt_assign_1*).

- filtering virtual links

File *link_flows_1.bin* is joined with layer *streets*. The joined table is filtered to leave only *virt_assign_1* links (*streets_and_link_flows_1.bin* file).

The rows of *streets_and_link_flows_1.bin* table are grouped by *from_node_to_emp_area* field, and for each group (represents all virtual links going from a park and ride node to all employment zones of the same area), volumes are summed up (field *AB_Flow_PCE*) resulting in file *groups.bin*.

File *groups.bin* is joined with layer *streets*. The joined table is filtered to leave only *virt_assign_1* links (*streets_and_groups.bin* file).

Virtual links having volume in *AB_Flow_PCE* field of *streets_and_groups.bin* table (the field represents group volume in this case) less than defined in setting *Minimum volume to employment area* are excluded from the 2nd assignment. These virtual links are gathered in *virt_deactivated* selection. Virtual links with acceptable group volumes are included in *virt_assign_2* selection, and their *virtual* field is overwritten with value 2.

- running the 2nd assignment with queries (points of interest) for *pr_query* nodes (defined in *queries.qry* file)

The assignment is launched with network file *car_2.net*. The output files are *link_flows_2.bin*, *movements_2.bin*, and *critical.mtx* (it has a separate core for each *pr_query* node). Links used in the assignment are union of *in_project* and *virt_assign_2* selections (gathered in *in_project__virt_assign_2* selection).

- creating matrix *volumes.mtx*

The matrix origins are *pr_virt* nodes, and destinations are *emp_virt* zones.

It is filled with virtual link volumes obtained from the 2nd assignment results (field *AB_Flow_PCE* from file *link_flows_2.bin*).

- creating matrix *vht.mtx*

The matrix origins are *pr_virt* nodes, and destinations are *emp_virt* zones.

For each pair of *pr_virt* node and *emp_virt* zone, time value is taken from *res_db_folder/db/extra/shortest_paths.mtx* matrix (produced by database preparation procedure), and volume value is taken from *volumes.mtx*. Then, time and volume values are multiplied and divided by 60 resulting in vehicle-hours (VHT) value.

- calculating VHT saved for each *pr_virt* node (stored in *vht_saved* field)

For each *pr_virt* node, a row from *vht.mtx* matrix is taken resulting in vector of VHT values. The vector is summed up to get VHT saved value.

- calculating node rank (stored in *rank* field)

For each *pr_virt* node its VHT saved value is multiplied by its total factor to get node rank. Although, total factors are stored in its field as percentage, in this calculation they are divided by 100 to get values from 0 to 1. All nodes with positive rank are included in *pr_positive_rank* selection. These nodes are sorted based on their rank, assigned a position (field *position*), and up to 10 best nodes are included in *pr_best* selection.

- setting styles for layers and selections
- saving map to *res_db_folder/result.map* file
- creating file *success*

After the procedure finished, status message will pop up showing its runtime ([Figure 17](#)), and a newly created result is left open.

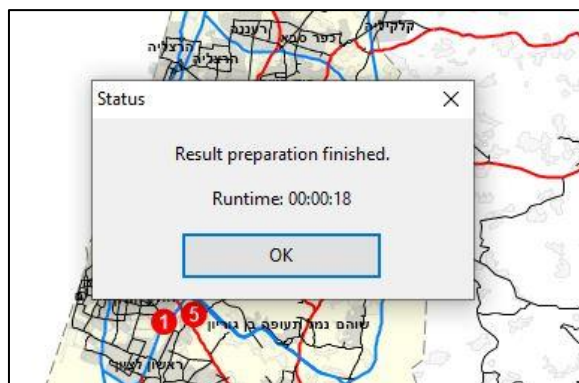


Figure 17. Status message after result preparation

If the procedure failed, and you try to open its result via *Park & Ride toolbox*, you will be notified about it ([Figure 18](#)).

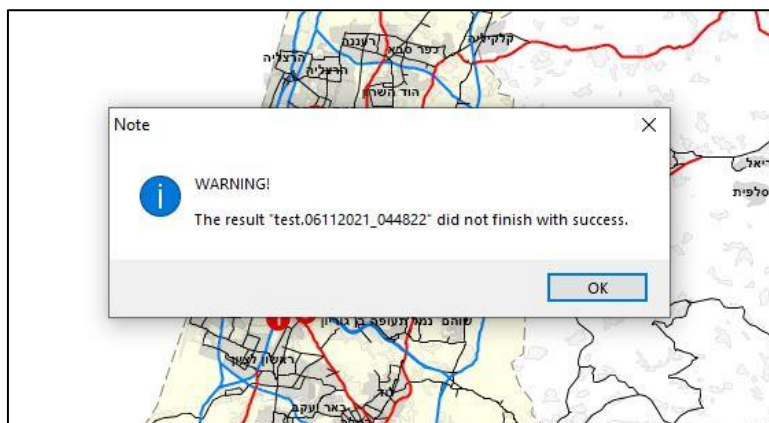


Figure 18. Warning after opening result

The result can be investigated with post-processing tool *Show Origins and Destinations*.

After you opened a result, click *Show Origins and Destinations* button from *Park & Ride* toolbox. If the result was prepared without points of interest, warning window will pop up ([Figure 19](#)).

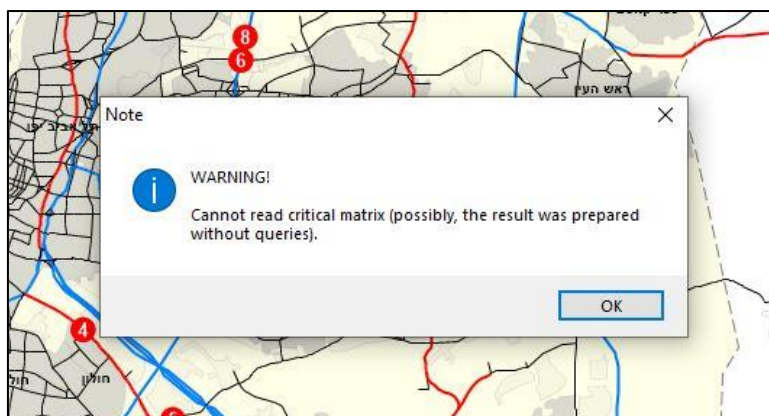


Figure 19. Warning after launching *Show Origins and Destinations* tool

If the result was prepared with points of interest, *Show Origins and Destinations* tool will open ([Figure 20](#)).

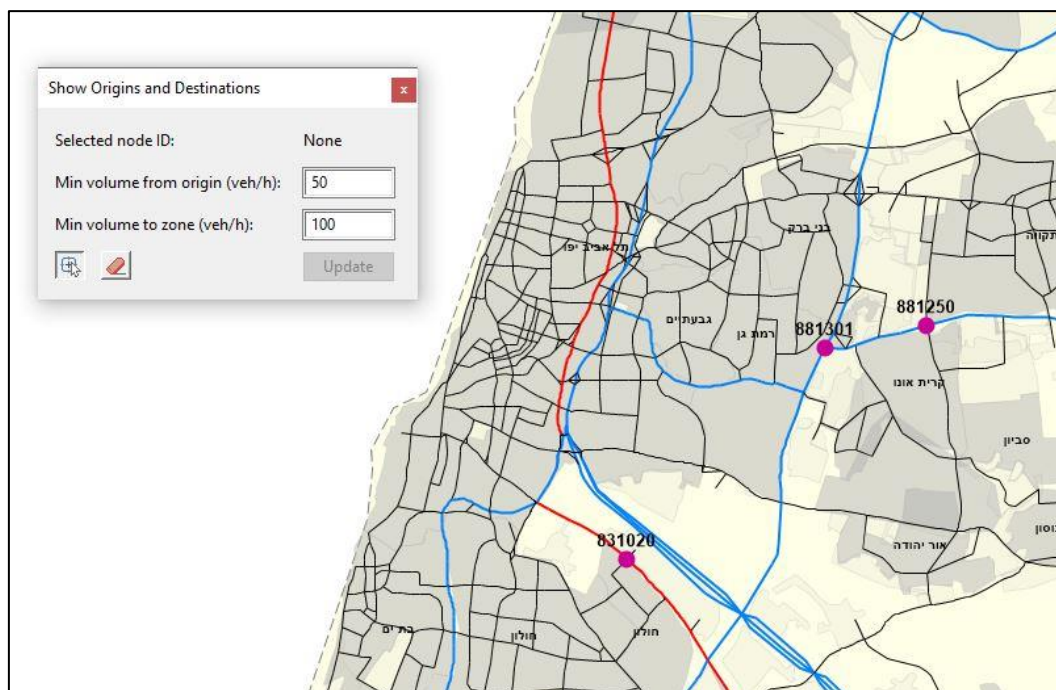


Figure 20. *Show Origins and Destinations* tool

The tool will automatically create additional node fields (*show_od__origins* and *show_od__zones*) and selections (*show_od__nodes*, *show_od__selected*, *show_od__origins*, and *show_od__zones*) where it saves its intermediate data. The styles are also adjusted to show only points of interest. Click on some node to examine volumes going via the selected node from all centroids and to employment zones (Figure 21).

Note: You cannot see volumes going from employment zones, but they can be checked in field *show_od__origins* for *emp_virt* selection (the field is filled after you selected a point of interest).

You can adjust minimum volume from origin and to zone in according fields and click *Update* to show only origins and zones with volumes no less than defined.

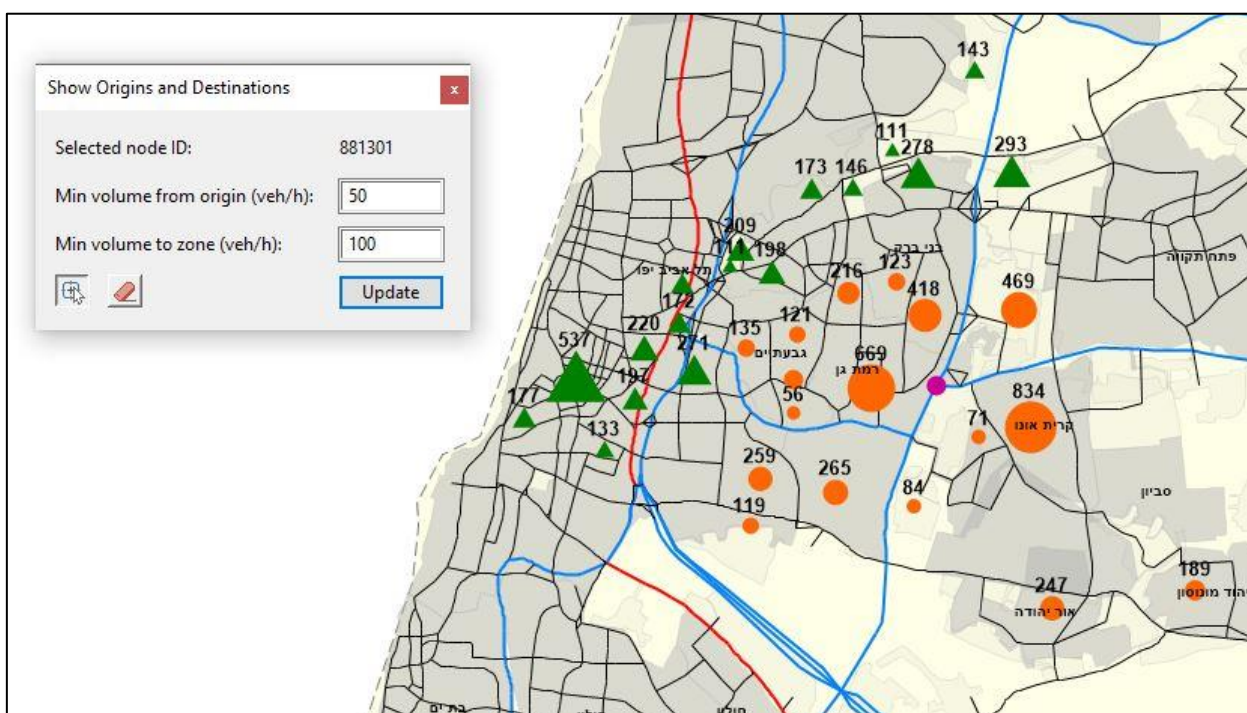


Figure 21. Show Origins and Destinations tool with selected node