TransimsNet *(version 4.0)*

Revision History
1/8/2010   Edited by AECOM Consult, Inc.
4/20/2010   Edited by RSG, Inc.

The **TransimsNet** program is used to:

1. Synthesize TRANSIMS network files from generic node and link information. Node, Link, Activity Location, Parking, Process Link, Lane Connectivity, and Pocket Lane files are generated by the program.
2. The program also generates Sign and Signal Warrant files that could be used as input to the IntControl program to synthesize the TRANSIMS Unsignalized Node, Signalized Node, Timing Plan, Phasing Plan, Detector and Signal Coordinator files.
3. Copy an existing network and update or delete network components associated with selected links or nodes.

TransimsNet is a consoled-based program that runs in a command window on either Windows or Linux. The command syntax is:

TransimsNet [-flag] [control_file]

The control_file is the file name of an ASCII file that contains the control strings expected by the program. The control_file is optional. If a file name is not provided, the program will prompt the user to enter a file name. The flag parameters are also optional. Any combination of the following flag parameters can be included on the command line:

-Q[uiet]   = execute without screen messages
-H[elp]   = show program syntax and control keys
-K[eyCheck]   = list unrecognized control file keys
-P[ause]   = pause before exiting
-N[oPause]   = never pause before exiting
-B[atch]   = execute in batch processing mode

The program automatically creates a printout file based on the control_file name. If the file name includes an extension (e.g., ".ctl"), the extension is removed and ".prn" is added. The printout file will be created in the current working directory and will overwrite an existing file with the same name.

Control File Parameters

Control parameters are defined using a control key followed by a string or number. The control parameters can be specified in any order. If a given key is defined more than once, the last instance of the key is used. The default value for each key is 0 or "Null". Null parameters do

not need to be included in the file. Note that comment lines or extraneous keys can be included in the file.   They will be ignored by the program.

A typical **TransimsNet** control file is shown below. These keys can be defined in a variety of different ways to perform different tasks.

```
TITLE                           Synthetic TRANSIMS Network
NET_DIRECTORY                   d:\Portland\Convert\Emme2
NET_NODE_TABLE                  Node
NET_LINK_TABLE                  Link
NET_ZONE_TABLE                  Zone_File.txt
NET_TURN_PROHIBITION_TABLE      Turn.out
KEEP_NODE_LIST                  d:\Portland\Convert\TransimsNet\Keep_Nodes.txt
TURN_FILE_FORMAT                AT-FROM-TO
LINK_NODE_EQUIVALENCE           d:\Portland\Convert\TransimsNet\Link_Node.txt
NEW_DIRECTORY                   d:\Portland\Convert\TransimsNet
NEW_NODE_TABLE                  Node
NEW_LINK_TABLE                  Link
NEW_ACTIVITY_LOCATION_TABLE     Activity_Location
NEW_PARKING_TABLE               Parking
NEW_PROCESS_LINK_TABLE          Process_Link
NEW_POCKET_LANE_TABLE           Pocket_Lane
NEW_LANE_CONNECTIVITY_TABLE     Lane_Connectivity
NEW_UNSIGNALIZED_NODE_TABLE     Sign_Warrants
NEW_SIGNALIZED_NODE_TABLE       Signal_Warrants
POCKET_RANGE_FOR_FACILITY_1     100, 400
POCKET_RANGE_FOR_FACILITY_2     60, 200
POCKET_RANGE_FOR_FACILITY_3     40, 100
POCKET_RANGE_FOR_FACILITY_4     30, 60
POCKET_RANGE_FOR_FACILITY_8     30, 60
SIGNAL_WARRANT_FOR_AREA_TYPE_1  COLLECTOR, LOCAL
SIGNAL_WARRANT_FOR_AREA_TYPE_2  COLLECTOR, COLLECTOR
SIGNAL_WARRANT_FOR_AREA_TYPE_3  MINOR, COLLECTOR
SIGNAL_WARRANT_FOR_AREA_TYPE_4  MINOR, MINOR
SIGNAL_WARRANT_FOR_AREA_TYPE_5  MAJOR, MINOR
SIGNAL_WARRANT_FOR_AREA_TYPE_6  MAJOR, MAJOR
SIGNAL_WARRANT_FOR_AREA_TYPE_7  PRINCIPAL, MAJOR
SIGNAL_WARRANT_FOR_AREA_TYPE_8  PRINCIPAL, PRINCIPAL
STOP_WARRANT_FOR_AREA_TYPE_1    LOCAL
STOP_WARRANT_FOR_AREA_TYPE_2    LOCAL
STOP_WARRANT_FOR_AREA_TYPE_3    COLLECTOR
STOP_WARRANT_FOR_AREA_TYPE_4    COLLECTOR
MAXIMUM_ACCESS_POINTS           4
MINIMUM_SPLIT_LENGTHS           100, 200, 200, 300, 300, 300, 300, 300
MINIMUM_LINK_LENGTH             37.5
MAXIMUM_LENGTH_TO_XY_RATIO      1.2
INTERSECTION_SETBACK_DISTANCE 0.0
```

This example generates the TRANSIMS network files from the generic link and the node files generated by the Emme2Net program. The keys recognized by the **TransimsNet** program are listed below. These keys can be defined in a variety of different ways to perform different tasks.

TRANSIMS
Open-Source

TITLE

Any text string can be used on this line. This text is printed on the top of each output page.

REPORT_FILE

The report file name is optional. If a file name is not provided, the program automatically creates a report file name based on the input control file name. The report file will overwrite an existing file with the same name if the Report Flag key is False or not specified.

REPORT_FLAG

The report flag key is optional. If it is specified as Yes or True, the report file or default printout file will be opened in "Append" mode rather than "Create" mode. This permits the user to consolidate the output of several programs into a single report file.

MAX_WARNING_MESSAGES

When the program generates a warning message, a counter is incremented and the total number of warning messages is reported and a warning return coded (2) is set at the end of the execution. By default the program prints up to 100,000 warning messages to the print-out file. If more than 100,000 warning messages are sent, the program stops printing additional messages to the file or terminates the program with an error message based on the MAX_WARNING_EXIT_FLAG. This parameter enables the user to modify the default warning limit.

MAX_WARNING_EXIT_FLAG

If the maximum number of warning messages is exceeded, this flag directs the program in what to do. If the flag is TRUE (the default), the program is terminated with an error message about the warning messages. If the flag is FALSE, the program continues execution, but no additional warning messages are sent to the screen or written to the printout file. The warning message counter continues to count the messages and reports the total at the end of the execution.

PROJECT_DIRECTORY

The project directory key is not required. If it is specified, it is added to all non-network file names required by the program. If it is not specified, all non-network file names should fully specify the file path.

NET_DIRECTORY

The network directory key is not required. If it is specified, it is added to all network table names. If it is not specified, the network table names should fully specify the file path.

NET_NODE_TABLE

The network node table key is required. It specifies the name of the input node file within the network directory. The full path and file name for the node table is constructed by appending the value of this key to the value of the NET_DIRECTORY key. This file can be created by running the GISNet, TPPlusNet or Emme2Net programs to convert the highway data into the input node file format. The node file contains the node number and X/Y coordinates in UTP meters.

TRANSIMS
Open-Source

NET_NODE_FORMAT

The node file format key can be used to specify the input file format. The default format is VERSION3; a tab delimited file compatible with the TRANSIMS Version 3.x software. Other options include BINARY, FIXED_COLUMN, COMMA_DELIMITED, SPACE_DELIMITED, TAB_DELIMITED, and DBASE.

NET_ZONE_TABLE

The network zone table key is optional. If provided, it specifies the name of the input zone file within the network directory. The full path and file name for the input zone file is constructed by appending the value of this key to the value of the NET_DIRECTORY key. The zone file contains the zone number, X/Y coordinates in UTM meters, and an area type code between 1 and 8. If a zone file is not provided, the zone centroids are extracted from the node file. In this case, all area types will be equal to 2.

NET_ZONE_FORMAT

The zone file format key can be used to specify the input file format. The default format is VERSION3; a tab delimited file compatible with the TRANSIMS Version 3.x software. Other options include BINARY, FIXED_COLUMN, COMMA_DELIMITED, SPACE_DELIMITED, TAB_DELIMITED, and DBASE.

NET_LINK_TABLE

The network link table key is required. It specifies the name of the input link file within the network directory. The full path and file name for the link table is constructed by appending the value of this key to the value of the NET_DIRECTORY key. This file can be created by running the GISNet, TPPlusNet, or Emme2Net programs to convert the highway data into the input link file format.

At a minimum, the header line should include nine fields in any order: node numbers, length, lanes, free speeds, facility type, and facility use. Capacity will also be processed if provided. The field names and coding conventions for these fields are consistent with the TRANSIMS network link table. This means that a standard TRANSIMS link and node file can be used as input to this program.

NET_LINK_FORMAT

The link file format key can be used to specify the input file format. The default format is VERSION3; a tab delimited file compatible with the TRANSIMS Version 3.x software. Other options include BINARY, FIXED_COLUMN, COMMA_DELIMITED, SPACE_DELIMITED, TAB_DELIMITED, and DBASE.

NET_SHAPE_TABLE

The network shape table key is optional. It specifies the name of the input shape file within the network directory. The full path and file name for the shape table is constructed by appending the value of this key to the value of the NET_DIRECTORY key. This file can be created by running the GISNet. The shape file contains the X/Y coordinates in UTP meters of points between the A and B nodes of a link.

NET_SHAPE_FORMAT

The shape file format key can be used to specify the input file format. The default format is VERSION3; a tab delimited file compatible with the TRANSIMS Version 3.x software. Other options include BINARY, FIXED_COLUMN, COMMA_DELIMITED, SPACE_DELIMITED, TAB_DELIMITED, and DBASE.

NET_ACTIVITY_LOCATION_TABLE

The activity location table key is optional. It specifies the name of the input activity location file within the network directory. The full path and file name for the activity location table is constructed by appending the value of this key to the value of the NET_DIRECTORY key. This file is used for update and delete applications.

NET_ACTIVITY_LOCATION_FORMAT

The activity location file format key can be used to specify the input file format. The default format is VERSION3; a tab delimited file compatible with the TRANSIMS Version 3.x software. Other options include BINARY, FIXED_COLUMN, COMMA_DELIMITED, SPACE_DELIMITED, TAB_DELIMITED, and DBASE.

NET_PARKING_TABLE

The parking table key is optional. It specifies the name of the input parking file within the network directory. The full path and file name for the activity location table is constructed by appending the value of this key to the value of the NET_DIRECTORY key. This file is used for update and delete applications.

NET_PARKING_FORMAT

The parking file format key can be used to specify the input file format. The default format is VERSION3; a tab delimited file compatible with the TRANSIMS Version 3.x software. Other options include BINARY, FIXED_COLUMN, COMMA_DELIMITED, SPACE_DELIMITED, TAB_DELIMITED, and DBASE.

NET_PROCESS_LINK_TABLE

The process link table key is optional. It specifies the name of the input process link file within the network directory. The full path and file name for the activity location table is constructed by appending the value of this key to the value of the NET_DIRECTORY key. This file is used for update and delete applications.

NET_PROCESS_LINK_FORMAT

The parking file format key can be used to specify the input file format. The default format is VERSION3; a tab delimited file compatible with the TRANSIMS Version 3.x software. Other options include BINARY, FIXED_COLUMN, COMMA_DELIMITED, SPACE_DELIMITED, TAB_DELIMITED, and DBASE.

NET_POCKET_LANE_TABLE

The pocket lane table key is optional. It specifies the name of the input pocket lane file within the network directory. The full path and file name for the activity location table is constructed by appending the value of this key to the value of the NET_DIRECTORY key. This file is used for update and delete applications.

NET_POCKET_LANE_FORMAT

The pocket lane file format key can be used to specify the input file format. The default format is VERSION3; a tab delimited file compatible with the TRANSIMS Version 3.x software. Other options include BINARY, FIXED_COLUMN, COMMA_DELIMITED, SPACE_DELIMITED, TAB_DELIMITED, and DBASE.

NET_UNSIGNALIZED_NODE_TABLE

The unsignalized node table key is optional. It specifies the name of the input unsignalized file within the network directory. The full path and file name for the activity location table is constructed by appending the value of this key to the value of the NET_DIRECTORY key. This file is used for update and delete applications.

NET_UNSIGNALIZED_NODE_FORMAT

The unsignalized lane file format key can be used to specify the input file format. The default format is VERSION3; a tab delimited file compatible with the TRANSIMS Version 3.x software. Other options include BINARY, FIXED_COLUMN, COMMA_DELIMITED, SPACE_DELIMITED, TAB_DELIMITED, and DBASE.

NET_SIGNALIZED_NODE_TABLE

The signalized node table key is optional. It specifies the name of the input signalized file within the network directory. The full path and file name for the activity location table is constructed by appending the value of this key to the value of the NET_DIRECTORY key. This file is used for update and delete applications.

NET_SIGNALIZED_NODE_FORMAT

The signalized lane file format key can be used to specify the input file format. The default format is VERSION3; a tab delimited file compatible with the TRANSIMS Version 3.x software. Other options include BINARY, FIXED_COLUMN, COMMA_DELIMITED, SPACE_DELIMITED, TAB_DELIMITED, and DBASE.

NET_TURN_PROHIBITION_TABLE

The turn prohibition key is optional and specifies the name of the input turn prohibitions within the network directory. The full path and file name for the input turn prohibition file is constructed by appending the value of this key to the value of the NET_DIRECTORY key. The **TransimsNet** program does not generate lane connections for the node combinations specified in this file. A sample turn prohibition file is shown below.

```
FROM    THRU    TO
8512    8520    8534
8512    8520    8536
8512    8523    8534
8512    8523    8536
8512    8534    8520
8512    8534    8523
8512    8536    8520
8512    8536    8523
8701    8705    9856
8705    8712    9864
```

Other file configurations include AT-FROM-TO and NODE-IN_LINK-OUT_LINK formats available in standard TRANSIMS file interfaces.

NET_TURN_PROHIBITION_FORMAT

The turn prohibition file format key can be used to specify the input file format. The default format is VERSION3; a tab delimited file compatible with the TRANSIMS Version 3.x software.   Other options include BINARY, FIXED_COLUMN, COMMA_DELIMITED, SPACE_DELIMITED, TAB_DELIMITED, and DBASE.

KEEP_NODE_LIST

The keep node list key is optional and specifies the full path and file name of the file that lists the node IDs that need to be retained in the highway network. The **TransimsNet** program removes node that are not required for the highway simulation. If transit routes will be included in the network, the nodes associated with transit stations and stops should not be removed from the network. The **TransimsNet** program retains all the nodes specified in the keep node list file even when they are not required for the highway simulation. A sample keep node list file is shown below.

```
15467
11634
11617
11625
11759
11601
11598
14489
11008
```

LINK_NODE_EQUIVALENCE

The Link Node Equivalence file is optional and specifies the full path and file name of the output link-node equivalency file. A sample equivalency file generated by the **TransimsNet** program is shown below. This file displays the sequence of nodes associated with each link ID. If links in the input link file are combined by the network collapsing process, the Link-Node Equivalence file provided a mechanism for associating the original node sequence with the newly created TRANSIMS link.

```
 1 = 1245 9089
 2 = 1246 8761
 3 = 1247 14945
 4 = 1248 15418
 7 = 1251 30125
18 = 8000 8056
19 = 8000 8716 8667
20 = 8045 8001 8011
22 = 8002 8041
23 = 8002 8048
24 = 8002 9883
25 = 8003 8004 8042
26 = 8003 8007
27 = 8003 9787 8048
29 = 8044 8005 8009
```

NEW_DIRECTORY

The new directory key is optional. It specifies the path to the output TRANSIMS network directory.

NEW_NODE_TABLE

The new node table key is required and specifies the filename of the output TRANSIMS node file generated by the **TransimsNet** program. The full path and file name for the output node table is constructed by appending the value of this key to the value of the NEW_DIRECTORY key.

NEW_NODE_FORMAT

The node file format key can be used to specify the output file format. The default format is VERSION3; a tab delimited file compatible with the TRANSIMS Version 3.x software. Other options include BINARY, FIXED_COLUMN, COMMA_DELIMITED, SPACE_DELIMITED, TAB_DELIMITED, and DBASE.

NEW_ZONE_TABLE

The new zone table key is optional and specifies the filename of the output TRANSIMS zone file generated by the **TransimsNet** program. The full path and file name for the output zone table is constructed by appending the value of this key to the value of the NEW_DIRECTORY key. If an input zone file is provided, all of the fields in the input zone file are copied to the output zone file. If an input zone file is not provided, a default zone file is generated using the nodes found in the node file with node numbers less than or equal to the highest zone number.

NEW_ZONE_FORMAT

The zone file format key can be used to specify the output file format. The default format is VERSION3; a tab delimited file compatible with the TRANSIMS Version 3.x software. Other options include BINARY, FIXED_COLUMN, COMMA_DELIMITED, SPACE_DELIMITED, TAB_DELIMITED, and DBASE.

NEW_LINK_TABLE

The new link table key is required and specifies the filename of the output TRANSIMS link file generated by the **TransimsNet** program. The full path and file name for the output link table is constructed by appending the value of this key to the value of the NEW_DIRECTORY key.

NEW_LINK_FORMAT

The link file format key can be used to specify the output file format. The default format is VERSION3; a tab delimited file compatible with the TRANSIMS Version 3.x software. Other options include BINARY, FIXED_COLUMN, COMMA_DELIMITED, SPACE_DELIMITED, TAB_DELIMITED, and DBASE.

NEW_SHAPE_TABLE

The new shape table key is optional. It specifies the filename of the output TRANSIMS shape file generated by the **TransimsNet** program. The full path and file name for the output node table is constructed by appending the value of this key to the value of the NEW_DIRECTORY key.

NEW_SHAPE_FORMAT

The shape file format key can be used to specify the output file format. The default format is VERSION3; a tab delimited file compatible with the TRANSIMS Version 3.x software. Other options include BINARY, FIXED_COLUMN, COMMA_DELIMITED, SPACE_DELIMITED, TAB_DELIMITED, and DBASE.

NEW_ACTIVITY_LOCATION_TABLE

The new activity location table key is required and specifies the filename of the output TRANSIMS activity location file generated by the **TransimsNet** program. The full path and file name for the output activity location table is constructed by appending the value of this key to the value of the NEW_DIRECTORY key.

NEW_ACTIVITY_LOCATION_FORMAT

The activity location file format key can be used to specify the output file format. The default format is VERSION3; a tab delimited file compatible with the TRANSIMS Version 3.x software. Other options include BINARY, FIXED_COLUMN, COMMA_DELIMITED, SPACE_DELIMITED, TAB_DELIMITED, and DBASE.

NEW_PARKING_TABLE

The new parking table key is required and specifies the filename of the output TRANSIMS parking file generated by the **TransimsNet** program. The full path and file name for the output parking table is constructed by appending the value of this key to the value of the NEW_DIRECTORY key.

NEW_PARKING_FORMAT

The shape file format key can be used to specify the output file format. The default format is VERSION3; a tab delimited file compatible with the TRANSIMS Version 3.x software. Other

options include BINARY, FIXED_COLUMN, COMMA_DELIMITED, SPACE_DELIMITED, TAB_DELIMITED, and DBASE.

NEW_PROCESS_LINK_TABLE

The new process link table key is required and specifies the filename of the output TRANSIMS process link file generated by the **TransimsNet** program. The full path and file name for the output process link table is constructed by appending the value of this key to the value of the NEW_DIRECTORY key.

NEW_PROCESS_LINK_FORMAT

The process link file format key can be used to specify the output file format. The default format is VERSION3; a tab delimited file compatible with the TRANSIMS Version 3.x software. Other options include BINARY, FIXED_COLUMN, COMMA_DELIMITED, SPACE_DELIMITED, TAB_DELIMITED, and DBASE.

NEW_POCKET_LANE_TABLE

The new pocket lane table key is required and specifies the filename of the output TRANSIMS pocket lane file generated by the **TransimsNet** program. The full path and file name for the output pocket lane table is constructed by appending the value of this key to the value of the NEW_DIRECTORY key.

NEW_POCKET_LANE_FORMAT

The pocket lane file format key can be used to specify the output file format. The default format is VERSION3; a tab delimited file compatible with the TRANSIMS Version 3.x software. Other options include BINARY, FIXED_COLUMN, COMMA_DELIMITED, SPACE_DELIMITED, TAB_DELIMITED, and DBASE.

NEW_LANE_CONNECTIVITY_TABLE

The new lane connectivity table key is required and specifies the filename of the output TRANSIMS lane connectivity file generated by the **TransimsNet** program. The full path and file name for the output lane connectivity table is constructed by appending the value of this key to the value of the NEW_DIRECTORY key.

NEW_LANE_CONNECTIVITY_FORMAT

The lane connectivity file format key can be used to specify the output file format. The default format is VERSION3; a tab delimited file compatible with the TRANSIMS Version 3.x software. Other options include BINARY, FIXED_COLUMN, COMMA_DELIMITED, SPACE_DELIMITED, TAB_DELIMITED, and DBASE.

NEW_UNSIGNALIZED_NODE_TABLE

The new unsignalized node table key is required and specifies the filename of the output sign warrants generated by the **TransimsNet** program. The full path and file name for the output sign file is constructed by appending the value of this key to the value of the NEW_DIRECTORY key. The sign warrants can be used as an input to the IntControl program to generate a complete TRANSIMS Unsignalized Node file. It can also be read as an Unsignalized Node file by the

ArcNet program to create an ArcView Shape file for reviewing and editing the sign warrants prior to final conversion with the IntControl program. Note that this file only includes records where a sign is recommended. A complete TRANSIMS Unsignalized Node file also identifies intersection approaches that are not controlled. A sample sign warrants file is shown below.

```
NODE    INLINK SIGN   NOTES
8078    185    S      STOP SIGN
8082    194    S      STOP SIGN
8085    201    S      STOP SIGN
8085    202    S      STOP SIGN
8087    185    Y      YIELD SIGN
8094    201    S      STOP SIGN
8130    261    S      STOP SIGN
8155    310    Y      YIELD SIGN
8165    324    S      STOP SIGN
8166    328    S      STOP SIGN
8178    286    S      STOP SIGN
8178    352    Y      YIELD SIGN
8178    354    S      STOP SIGN
8179    353    S      STOP SIGN
8179    356    S      STOP SIGN
8183    364    S      STOP SIGN
```

NEW_UNSIGNALIZED_NODE_FORMAT

The unsignalized file format key can be used to specify the output file format. The default format is VERSION3; a tab delimited file compatible with the TRANSIMS Version 3.x software. Other options include BINARY, FIXED_COLUMN, COMMA_DELIMITED, SPACE_DELIMITED, TAB_DELIMITED, and DBASE.

NEW_SIGNALIZED_NODE_TABLE

The new signalized node table key is required and specifies the filename of the output signal warrants file generated by the **TransimsNet** program. The full path and file name for the output signal table is constructed by appending the value of this key to the value of the NEW_DIRECTORY key. The signal warrants generated by the **TransimsNet** program could be used as an input to the **IntControl** program to generate complete TRANSIMS Signalized Node, Timing Plan, Phasing Plan, Detector and Signal Coordinator files. It can also be read as a Signalized Node file by the **ArcNet** program to create an ArcView Shape file for reviewing and editing the signal warrants prior to final conversion with the **IntControl** program. A sample signal warrants file is shown below.

```
NODE  TYPE   PLAN    OFFSET   STARTTIME    COORDINATR    RING   ALGORITHM     NOTES
8002  A      0       0        ALL00:00     8002          S      B             SIGNAL WARRANT
8003  A      0       0        ALL00:00     8003          S      B             SIGNAL WARRANT
8006  A      0       0        ALL00:00     8006          S      B             SIGNAL WARRANT
8007  A      0       0        ALL00:00     8007          S      B             SIGNAL WARRANT
8009  A      0       0        ALL00:00     8009          S      B             SIGNAL WARRANT
8010  A      0       0        ALL00:00     8010          S      B             SIGNAL WARRANT
8011  A      0       0        ALL00:00     8011          S      B             SIGNAL WARRANT
8015  A      0       0        ALL00:00     8015          S      B             SIGNAL WARRANT
8016  A      0       0        ALL00:00     8016          S      B             SIGNAL WARRANT
8017  A      0       0        ALL00:00     8017          S      B             SIGNAL WARRANT
```

NEW_SIGNALIZED_FORMAT

The signalized file format key can be used to specify the output file format. The default format is VERSION3; a tab delimited file compatible with the TRANSIMS Version 3.x software. Other options include BINARY, FIXED_COLUMN, COMMA_DELIMITED, SPACE_DELIMITED, TAB_DELIMITED, and DBASE.

NEW_TURN_PROHIBITION_TABLE

The new turn prohibition table key is optional and is only processed when an input turn prohibition file is included. This key specifies the filename of the output TRANSIMS turn prohibition file. The full path and file name for the output turn prohibition table is constructed by appending the value of this key to the value of the NEW_DIRECTORY key.

NEW_TURN_PROHIBITION_FORMAT

The turn prohibition file format key can be used to specify the output file format. The default format is VERSION3; a tab delimited file compatible with the TRANSIMS Version 3.x software. Other options include BINARY, FIXED_COLUMN, COMMA_DELIMITED, SPACE_DELIMITED, TAB_DELIMITED, and DBASE.

POCKET_RANGE_FOR_FACILITY_#

This key is optional and when provided specifies the maximum lengths (in meters) of the pocket lanes for different area types. The # at the end of facility keyword specifies the facility type. The number code associated with different roadway facility types is shown below

Number Code	Facility Type
1	Freeway
2	Expressway
3	Principal Arterial
4	Major Arterial
5	Minor Arterial
6	Collector
7	Local
8	Frontage Road
9	Ramp
10	Bridge
11	Walkway
12	Bikeway
13	Busway
14	Lightrail
15	Heavyrail
16	Ferry
17	External or Zone Connectors

Area types are defined in the input zone file. Each link is assigned to the area type of the closest zone centroid. The pocket lane values must be between 0 and 2000 meters. If a value is not

provided all area types, the last value in the key will be used for the remaining area types. The values are separated by a comma as shown in the examples below.

```
POCKET_RANGE_FOR_FACILITY_1        100, 400, 600, 1000
POCKET_RANGE_FOR_FACILITY_2        200, 300
POCKET_RANGE_FOR_FACILITY_3        100
POCKET_RANGE_FOR_FACILITY_4        60
```

SIGNAL_WARRANT_FOR_AREA_TYPE_#

This key is optional and when provided specifies the combinations of the roadway facilities that warrant a traffic signal in a given area type. The # at the end of keyword specifies the area type code. For example the SIGNAL_WARRANT_FOR_AREA_TYPE_1 specifies the roadway facility combination that warrants a traffic signal in area type 1. Each intersection is assigned to the area type of the closest zone centroid.

The facility codes correspond to the TRANSIMS functional class codes. Options include:

1. FREEWAY,
2. EXPRESSWAY or XPRESSWAY,
3. PRINCIPAL,
4. MAJOR or PRIARTER,
5. MINOR or SECARTER,
6. COLLECTOR,
7. LOCAL,
8. FRONTAGE,
9. RAMP, and
10. BRIDGE or OTHER,

Intersections that only contain freeways, expressways, and ramps are not considered by the signal warrant procedure. When a ramp is included in an arterial intersection, its classification is assumed equal to a minor arterial by the signal warrant logic.

The keys are in hierarchical order and should be interpreted as the minimum facility type where a signal is warranted. For example, MINOR means that all minor arterials and above (i.e., major and principal arterials) are included in the warrant. (Note: The BRIDGE area type is considered equal to a MAJOR arterial in the signal hierarchy). A warrant for a given area type is defined using two key words. The first value corresponds to the primary facility entering the intersection and the second value corresponds to the secondary facility entering the intersection. For example, "MAJOR, MINOR" means that all intersections that join a major arterial with a minor arterial will be assigned a traffic signal.

A typical set of signal warrants for area types that represent decreasing land-use densities (e.g., CBD thru rural) would look something like the example below.

```
SIGNAL_WARRANT_FOR_AREA_TYPE_1        COLLECTOR, LOCAL
SIGNAL_WARRANT_FOR_AREA_TYPE_2        COLLECTOR, COLLECTOR
SIGNAL_WARRANT_FOR_AREA_TYPE_3        MINOR, COLLECTOR
```

```
SIGNAL_WARRANT_FOR_AREA_TYPE_4      MINOR, MINOR
SIGNAL_WARRANT_FOR_AREA_TYPE_5      MAJOR, MINOR
SIGNAL_WARRANT_FOR_AREA_TYPE_6      MAJOR, MAJOR
SIGNAL_WARRANT_FOR_AREA_TYPE_7      PRINCIPAL, MAJOR
SIGNAL_WARRANT_FOR_AREA_TYPE_8      PRINCIPAL, PRINCIPAL
```

Two additional optional parameters can be included in each key. The first optional parameter is the signal type: ACTUATED or TIMED and the second optional parameter is the number of rings: SINGLE_RING or DUAL_RING. The default is ACTUATED, SINGLE_RING.

STOP_WARRANT_FOR_AREA_TYPE_#

This key is optional and when provided specifies the roadway facility type that warrant all-way stop signs in a given area type. The # at the end of keyword specifies the area type code. For example the STOP_WARRANT_FOR_AREA_TYPE_1 specifies the roadway facility type that warrants all-way stop signs in area type 1. Each intersection is assigned to the area type of the closest zone centroid.

By default, all intersections that are not assigned to a signal are considered for stop or yield signs. Yield signs are placed on ramps entering freeways and expressways. Stop signs are placed on non-signalized intersections with different primary and secondary facility types. The primary facility will be uncontrolled and the secondary facility will be stop controlled. This parameter determines how to control intersections between streets of the same facility type. All intersection types that are greater than or equal to the specified facility type are controlled with all-way stop signs. The default value is LOCAL which means that all LOCAL-LOCAL intersections and above are given all-way stop signs. If the value is set to COLLECTOR, a LOCAL-LOCAL intersection will be uncontrolled.

MAXIMUM_ACCESS_POINTS

This maximum access points key is required and specifies the maximum number of activity locations that can be generated on each side of a given link. The actual under of activity locations will depend on the link length. This value can range from 1 to 20. Note that activity locations are not created on freeways, ramps, zone connectors, "other", and transit only links.

MINIMUM_SPLIT_LENGTHS

This minimum split lengths key is required and specifies the minimum distance between activity locations assigned to a link. A minimum distance value specified in meters should be provided for each area type. For example, the following command:

```
MINIMUM_SPLIT_LENGTHS      100, 200, 300, 300, 300, 300, 300, 300
```

means that links in area type 1 will have activity locations every 100 meters, and links in area type 2 will have activity locations every 200 meters, etc.

MINIMUM_LINK_LENGTH

This minimum link length key is optional. This key defaults to 37.5 meters when the value is not provided. This key specifies the minimum length of the network links. When the network link

length is less than this value the length is reset to this value. The value can range from 7.5 to 100 meters. Note that TRANSIMS requires a link to be at least as long as the longest vehicle that can use the link. Vehicle sizes are defined in the vehicle prototype file and are defined as multiples of TRANSIMS cells (7.5 meters).

CELL_SIZE

This cell size key is optional. This key defaults to 7.5 meters when the value is not provided. The value can range from 4.5 to 9.0 meters. Note that the cell size must be less than or equal to the specified minimum link length.

KEEP_ZONE_CONNECTORS_AS_LOCALS

This key is optional and defaults to FALSE. If set to true or yes, the zone connectors are preserved and classified with "Local" as the facility type. If the key REPLICATE_MPO_NETWORK is set to false, additional lane connectivity is provided to allow trips through the zone centroids, otherwise, the trips through the zone centroids are prohibited.

REPLICATE_MPO_NETWORK

The replicate mpo network is optional and defaults to FALSE. If set to true or yes, it allows the program to convert a traditional travel demand forecasting network with zone connectors to TRANSIMS. One activity location is inserted for each zone centroid in order to replicate the way a traditional network loads trips between centroids. This network can also be used to conveniently generate zone-to-zone skim files. If the KEEP_ZONE_CENTROIDS_AS_LOCALS key is false, trips through the zone centroid are prohibited otherwise, additional lane connectivity is provided to allow trips through the zone centroids.

MAXIMUM_LENGTH_TO_XY_RATIO

The maximum length to X-Y ratio key is optional. This key defaults to 1.2 when the value is not specified. This key is used to compare the link length in the input network file to the straight-line distance between the coordinates of the nodes at either end of the link and any shape points that are provided. If the length is less than the coordinate-based length, the output length is set equal to the coordinate-based length. If the ratio between the length and the coordinate-based length is greater than the value specified by this key, the program reduces the link length to the ratio times the coordinate-based length. For example, if the length field indicates the link is 1.3 miles long and the straight-line distance between the nodes is 1.0 miles, and the maximum ratio is 1.2, the program will reset the link length to 1.2 miles. The output printout file reports the number of links that were changed as a result of the distance checks.

INTERSECTION_SETBACK_DISTANCE

This intersection setback distance key is optional and specifies the setback distance from the center of intersection. The default value for this key is 0.0 meters. TRANSIMS reduces the link length by the setback distance at each end of the link when defining cells in the Microsimulator.

UPDATE_NODE_RANGE

The update node range is optional and if specified defines a series of node numbers where the lane connectivity and traffic control warrants are recalculated. The range is a comma separated list of node ranges. A node range is specified using two period (e.g., 100..200). In update mode, the program reads existing network files and deletes the existing records for the node and adds new records at the end of the file.

UPDATE_NODE_FILE

The update node file is optional and if specified defines a series of node numbers where the lane connectivity and traffic control warrants are recalculated. Each record in the file is interpreted as a comma separated list of node ranges. A node range is specified using two period (e.g., 100..200). The file could also be a simple list of node numbers. The values in the node range and the node file are combined if both keys are provided. In update mode, the program reads existing network files and deletes the existing records for the node and adds new records at the end of the file.

UPDATE_LINK_RANGE

The update link range is optional and if specified defines a series of link numbers where the pocket lanes, activity locations, parking lots, and processing links are recalculated. The lane connectivity at both ends of the link is also updated. The range is a comma separated list of link ranges. A link range is specified using two period (e.g., 100..200). In update mode, the program reads existing network files and deletes the existing records for the link and adds new records at the end of the file.

UPDATE_LINK_FILE

The update link file is optional and if specified defines a series of link numbers where the pocket lanes, activity locations, parking lots, and processing links are recalculated. The lane connectivity at both ends of the link is also updated. Each record in the file is interpreted as a comma separated list of link ranges. A link range is specified using two period (e.g., 100..200). The file could also be a simple list of link numbers. The values in the link range and the link file are combined if both keys are provided. In update mode, the program reads existing network files and deletes the existing records for the link and adds new records at the end of the file.

DELETE_NODE_RANGE

The delete node range is optional and if specified defines a series of node numbers where the lane connectivity, traffic control warrants, and the node are deleted. The range is a comma separated list of node ranges. A node range is specified using two period (e.g., 100..200).

DELETE_NODE_FILE

The delete node file is optional and if specified defines a series of node numbers where the lane connectivity, traffic control warrants, and node are deleted. Each record in the file is interpreted as a comma separated list of node ranges. A node range is specified using two period (e.g., 100..200). The file could also be a simple list of node numbers. The values in the node range and the node file are combined if both keys are provided.

DELETE_LINK_RANGE

The delete link range is optional and if specified defines a series of link numbers where the pocket lanes, activity locations, parking lots, processing links and link are deleted. The lane connectivity at both ends of the link is also updated. The range is a comma separated list of link ranges. A link range is specified using two period (e.g., 100..200).

DELETE_LINK_FILE

The delete link file is optional and if specified defines a series of link numbers where the pocket lanes, activity locations, parking lots, processing links and link are deleted. The lane connectivity at both ends of the link is also updated. Each record in the file is interpreted as a comma separated list of link ranges. A link range is specified using two period (e.g., 100..200). The file could also be a simple list of link numbers. The values in the link range and the link file are combined if both keys are provided.

Algorithm Notes

The **TransimsNet** program synthesizes TRANSIMS node, link, activity locations, parking, process link, lane connectivity and pocket lane files from generic input node and link files. The software also generates signal and sign warrant files for input to the **IntControl** program.

The program using the ZONECONN functional class to identify external access links. A special external access link with a boundary parking and activity location is generated for each zone connector included in the input link file. The program assumes that the lower node number attached to a zone connector represents the external station zone number. Unless the user wants to include external access points for internal zone centroids, all zone connectors attached to internal zone centroids should be deleted from the input link file prior to running this program.

If the original network attached zone connectors to mid-block nodes, the input link file will include a significant number of mid-block nodes that are no longer needed. Since TRANSIMS has difficulty simulating short links, it is desirable to remove these extra nodes from the network. The **TransimsNet** program includes a collapsing algorithm to remove unnecessary nodes. This algorithm considers the number of links in to and out of a given node, the compass orientation of these links, and the functional class, lanes, and speeds of the links. The program will keep nodes that are in the keep node list file, represent a significant change in direction (i.e., a major shape point), or define a change in network attributes.

The intersection logic also examines the number of links in to and out of a given node and determines how to construct turn pockets, thru links, lane connectivity, and traffic controls. If the node includes freeway links, special algorithms are used to connect multiple freeways and/or ramps. Merge and diverge lanes are added to the right or left side of a freeway depending on the relative orientation of the links. Arterial intersections examine the relative orientation or each movement and the functional class of each link to determine when and where to include turn pockets and signals or signs. In general, if an approach has opposing traffic, a turn pocket is added to accommodate the movement. The signal warrants are determined based on the number of legs and the functional class by area type signal warrant parameters.

Parking lots, activity locations, and process links are added to links using the split length and access point parameters. The link must also permit auto traffic to be assigned parking lot access points. Activity locations are defined on each side of the link at the spacing specified by the split length. A parking lot and process links are included for each activity location. The zone number and area type of the closest zone centroid will be included as user fields on the activity location file. If trip table disaggregation or population synthesis and activity generation are to be performed, the zone number assigned to an activity location should probably be refined. The coordinates on the activity location file can be used to load the data into a GIS software package. The GIS software can be used to locate each activity location more accurately within zone boundary polygons and update the zone value. The **ArcNet** utilities are provided to assist with this process.

Update and Delete Applications

In addition to synthesizing a TRANSIMS network from basic link and node data, the **TransimsNet** program is also used to updated or delete records in an existing TRANSIMS network. The update and delete node and/or link ranges or files specify the records to be processed. Input network table keys are needed to define the existing network. The records in the input file are read and copied to the output network if not included one of the update or delete lists. If the record is associated with a link or a node that is updated or deleted, the record is not copied to the output file. The new records generated by the update option are added to the end of the output file. As a result, records for a given location (link or node) will have different record numbers in the output file than in the input file.

The link-related data types include the pocket lanes, activity locations, parking lots, and process links. Since the input link file may include changes to the number of lanes for an updated link, the lane connectivity records at both ends of the link are also regenerated. In all cases, the methods used to regenerate pocket lanes, activity locations, and other link attributes utilized the parameters specified for the application. In other words, the rules for generating synthetic data may be different than the rules used to generate the original network. This enables the user to customize the synthetic generate process for specific links or areas within the network.

The node-related data types include the lane connectivity and the traffic controls (sign and signal warrants). The node update option is typically used for apply different signal and sign warrant parameters for different areas within the region or to simply remove signals or signs that are not wanted.

The delete options are intended to physically remove link and nodes from the system. It is not possible to remove a node without also removing all of the links attached to the node. Deleting a link will automatically update the lane connectivity at the nodes at both ends of the deleted link.

Sample Printout

The printout file generated by **TransimsNet** will look something like the example below. It is an ASCII text file with a maximum of 95 characters per line and 65 lines per page. The file can be viewed or printed using a variety of text editors. For best results in a word processor, use a 10-point Courier font and 0.5 inch margins on all sides.

```
************************************************
|                                              |
|        TransimsNet – Version 4.0.25          |
|    Copyright (c) 2009 by AECOM Consult        |
|          Mon Apr 19 13:11:47 2010            |
|                                              |
************************************************

Control File = TransimsNet40.ctl
Report_File  = TransimsNet40.prn (Create)

Synthetic TRANSIMS Network

Network Directory = d:\test\case3\network
Node File = d:\test\case3\network\Input_Node.txt
Zone File = d:\test\case3\network\Input_Zone.txt
Link File = d:\test\case3\network\Input_Link.txt
Link File Format = TAB_DELIMITED

Default New Network Format = TAB_DELIMITED
New Network Directory = d:\test\case3\newnet
New Node File = d:\test\case3\newnet\Node.txt
New Link File = d:\test\case3\newnet\Link.txt
New Pocket Lane File = d:\test\case3\newnet\Pocket_Lane.txt
New Lane Connectivity File = d:\test\case3\newnet\Lane_Connectivity.txt
New Lane Connectivity File Format = VERSION3
New Parking File = d:\test\case3\newnet\Parking.txt
New Activity Location File = d:\test\case3\newnet\Activity_Location.txt
New Process Link File = d:\test\case3\newnet\Process_Link.txt
New Unsignalized Node File = d:\test\case3\newnet\Sign_Warrants.txt
New Signalized Node File = d:\test\case3\newnet\Signal_Warrants.txt

New Link Node Equivalence File = d:\test\case3\newnet\Link_Node.txt

Pocket Lengths for Facility Type 1 = 100, 200, 300, 400 meters
Pocket Lengths for Facility Type 2 = 75, 150, 200, 250 meters
Pocket Lengths for Facility Type 3 = 50, 100, 150, 200 meters
Pocket Lengths for Facility Type 4 = 25, 50, 100, 150 meters
Pocket Lengths for Facility Type 8 = 50, 100, 150, 200 meters

Signal Warrant for Area Type 1 = COLLECTOR, LOCAL, TIMED
Signal Warrant for Area Type 2 = COLLECTOR, COLLECTOR, ACTUATED
Signal Warrant for Area Type 3 = MINOR, COLLECTOR, ACTUATED
Signal Warrant for Area Type 4 = MINOR, MINOR, ACTUATED, DUAL_RING
Signal Warrant for Area Type 5 = MAJOR, MINOR, ACTUATED, DUAL_RING
Signal Warrant for Area Type 6 = MAJOR, MAJOR, ACTUATED, DUAL_RING
Signal Warrant for Area Type 7 = PRINCIPAL, MAJOR, ACTUATED
Signal Warrant for Area Type 8 = PRINCIPAL, PRINCIPAL, ACTUATED

Stop Warrant for Area Type 3 = COLLECTOR

Maximum Number of Access Points Per Link = 3

Minimum Split Lengths = 100, 200, 300, 300, 300, 300, 300, 300 meters

Minimum Link Length = 37.5 meters
```

```
Maximum Length to XY Ratio = 1.20

Maximum Connection Angle = 120 degrees
U-Turns will not be added to Dead-End Links

Intersection Setback Distance = 0.0 meters

Extra Nodes will be Removed from the Network


Number of Node File Records = 21

Number of Zone File Records = 5
Number of Zone Data Records = 4

Number of Link File Records = 24
Number of Directional Links = 48

Number of Input Node Records = 21
Number of Input Link Records = 24
Number of Input Zone Records = 4

Highest Zone Number = 0

Number of New Node Records = 21
Number of New Link Records = 24
Number of New Activity Location Records = 96
Number of New Parking Lot Records = 96
Number of New Process Link Records = 192
Number of New Pocket Lane Records = 44
Number of New Lane Connectivity Records = 216
Number of New Unsignalized Node Records = 2
Number of New Signalized Node Records = 7

Number of Stop Signs = 2

Number of Demand Actuated Single Ring Signals = 7

Mon Apr 19 13:11:47 2010 -- Process Complete (0:00:00)
```

TRANSIMS
Open-Source