SubareaNet (version 4.0.11)

Revision History
20 October 2010 Created by Volpe Center

The SubareaNet program:

1. Creates subarea network files from regional network files using a subarea boundary polygon in ArcView format.
2. Creates subarea transit routes and schedules from regional network files using a subarea boundary polygon in ArcView format.

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

```
SubareaNet [-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, 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 Examples

EXAMPLE 1 CREATE SUBAREA NETWORK

The following **SubareaNet** control file creates a smaller network, based on a shapefile that is specified in the control file. The shapefile is an ArcView shapefile with a single polygon record defining the boundary of the subarea.

```
    TITLE        Creates a subarea network
    REPORT_FILE        subareanet.prn
    REPORT_FLAG         True
    PROJECT_DIRECTORY

    DEFAULT_FILE_FORMAT        TAB_DELIMITED

#---- Input Files ----
```

```
    NET_DIRECTORY         network
    NET_NODE_TABLE        Node.txt
    NET_ZONE_TABLE       Zone.txt

    NET_SHAPE_TABLE      Shape.txt
    NET_LINK_TABLE        Link.txt
    NET_POCKET_LANE_TABLE        Pocket_Lane.txt
    NET_LANE_USE_TABLE       Lane_Use.txt
    NET_TOLL_TABLE        Toll.txt
    NET_LANE_CONNECTIVITY_TABLE        Lane_Connectivity.txt
    NET_PARKING_TABLE        Parking.txt
    NET_ACTIVITY_LOCATION_TABLE        Activity_Location.txt
    NET_PROCESS_LINK_TABLE        Process_Link.txt

    NET_UNSIGNALIZED_NODE_TABLE        Unsignalized_Node.txt
    NET_SIGNALIZED_NODE_TABLE        Signalized_Node.txt
    NET_PHASING_PLAN_TABLE        Phasing_Plan.txt
    NET_TIMING_PLAN_TABLE        Timing_Plan.txt
    NET_DETECTOR_TABLE        Detector.txt
    NET_SIGNAL_COORDINATOR_TABLE        Signal_Coordinator.txt

#---- Output Files ----
    NEW_DIRECTORY       subnet
    NEW_NODE_TABLE        Node.txt
    NEW_ZONE_TABLE      Zone.txt
    NEW_SHAPE_TABLE      Shape.txt
    NEW_LINK_TABLE        Link.txt
    NEW_POCKET_LANE_TABLE        Pocket_Lane.txt
    NEW_LANE_USE_TABLE      Lane_Use.txt
    NEW_TOLL_TABLE      Toll.txt
    NEW_LANE_CONNECTIVITY_TABLE        Lane_Connectivity.txt
    NEW_TURN_PROHIBITION_TABLE         Turn_Prohibition.txt
    NEW_PARKING_TABLE        Parking.txt
    NEW_ACTIVITY_LOCATION_TABLE        Activity_Location.txt
    NEW_PROCESS_LINK_TABLE        Process_Link.txt
    NEW_UNSIGNALIZED_NODE_TABLE        Unsignalized_Node.txt
    NEW_SIGNALIZED_NODE_TABLE        Signalized_Node.txt
    NEW_PHASING_PLAN_TABLE        Phasing_Plan.txt
    NEW_TIMING_PLAN_TABLE        Timing_Plan.txt
    NEW_DETECTOR_TABLE        Detector.txt
    NEW_SIGNAL_COORDINATOR_TABLE        Signal_Coordinator.txt
    CREATE_NOTES_AND_NAME_FIELDS      TRUE

    NET_DEFAULT_FORMAT        TAB_DELIMITED
    NET_NODE_FORMAT        TAB_DELIMITED
    NET_ZONE_FORMAT        TAB_DELIMITED
    NET_SHAPE_FORMAT        TAB_DELIMITED
    NET_LINK_FORMAT        TAB_DELIMITED
    NET_POCKET_LANE_FORMAT        TAB_DELIMITED
    NET_LANE_USE_FORMAT        TAB_DELIMITED
    NET_TOLL_FORMAT        TAB_DELIMITED
    NET_LANE_CONNECTIVITY_FORMAT        TAB_DELIMITED
    NET_TURN_PROHIBITION_FORMAT        TAB_DELIMITED
    NET_PARKING_FORMAT        TAB_DELIMITED
    NET_ACTIVITY_LOCATION_FORMAT        TAB_DELIMITED
    NET_PROCESS_LINK_FORMAT        TAB_DELIMITED
```

```
    NET_UNSIGNALIZED_NODE_FORMAT        TAB_DELIMITED
    NET_SIGNALIZED_NODE_FORMAT        TAB_DELIMITED
    NET_PHASING_PLAN_FORMAT        TAB_DELIMITED
    NET_TIMING_PLAN_FORMAT        TAB_DELIMITED
    NET_DETECTOR_FORMAT        TAB_DELIMITED
    NET_SIGNAL_COORDINATOR_FORMAT        TAB_DELIMITED
    NEW_DEFAULT_FORMAT        TAB_DELIMITED
    NEW_NODE_FORMAT        TAB_DELIMITED
    NEW_ZONE_FORMAT        TAB_DELIMITED
    NEW_SHAPE_FORMAT        TAB_DELIMITED
    NEW_LINK_FORMAT        TAB_DELIMITED
    NEW_POCKET_LANE_FORMAT        TAB_DELIMITED
    NEW_LANE_USE_FORMAT        TAB_DELIMITED
    NEW_TOLL_FORMAT        TAB_DELIMITED
    NEW_LANE_CONNECTIVITY_FORMAT        TAB_DELIMITED
    NEW_TURN_PROHIBITION_FORMAT        TAB_DELIMITED
    NEW_PARKING_FORMAT        TAB_DELIMITED
    NEW_ACTIVITY_LOCATION_FORMAT        TAB_DELIMITED
    NEW_PROCESS_LINK_FORMAT        TAB_DELIMITED
    NEW_UNSIGNALIZED_NODE_FORMAT        TAB_DELIMITED
    NEW_SIGNALIZED_NODE_FORMAT        TAB_DELIMITED
    NEW_PHASING_PLAN_FORMAT        TAB_DELIMITED
    NEW_TIMING_PLAN_FORMAT        TAB_DELIMITED
    NEW_DETECTOR_FORMAT        TAB_DELIMITED
    NEW_SIGNAL_COORDINATOR_FORMAT        TAB_DELIMITED

#---- Parameters ----

    SUBAREA_BOUNDARY_POLYGON    subareapolygon.shp
```

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.

The keys recognized by the **SubareaNet** program are listed below. These keys can be defined in a variety of different ways to perform different tasks. The first 15 keys specify the subarea polygon, input network, and output network. They are required; other keys are optional.

Required Keys

SUBAREA_BOUNDARY_POLYGON

The boundary polygon is required. It is an arcview shapefile that contains a single polygon record defining the boundary of the subarea. The polygon should not have any holes, and should minimize the number of significant links crossed by its boundary.

The polygon can be created in any GIS. Here are some steps, taken from the TRANSIMS training course developed at Argonne National Labs, for creating a polygon using uDig:

- Zoom in to an area slightly bigger than the polygon to be created

- Choose "Layer" and then "Create Layer" from the menu bar
  - Under "geometry", change the type from "LineString" to "Polygon"
  - Change the coordinate system from "WGS84" (or some other possible default) to the appropriate UTM zone of the TRANSIMS model (TRANSIMS does not convert projections automatically):
    - NAD83 / UTM zone 16N (EPSG:26916) for Chicago
    - NAD83 / UTM zone 18N (EPSG:26918) for Alexandria
- From the tool bar, select the "Create Polygon Tool" (make sure that the newly created layer stays selected when using the polygon tool)
  - Click once for each shape point of the polygon, and double-click to finish polygon creation (create only one polygon)
- In the layer list, right-click on the newly created polygon layer and choose "Rename", then choose a meaningful name (e.g. "SubArea")
- In the layer list, right-click on the newly created polygon layer and choose "Export", then "Layer Export", then choose file name "SubArea.shp"

NET_NODE_TABLE

The node table key is required. It specifies the name of the TRANSIMS 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.

NET_LINK_TABLE

The link table key is required. It specifies the name of the TRANSIMS 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.

NET_POCKET_LANE_TABLE

The network pocket lane table key is required. It specifies the name of the TRANSIMS pocket lane file within the network directory. The full path and file name for the pocket lane table is constructed by appending the value of this key to the value of the NET_DIRECTORY key.

NET_LANE_CONNECTIVITY_TABLE

The network lane connectivity table key is required. It specifies the name of the TRANSIMS lane connectivity file within the network directory. The full path and file name for the lane connectivity table is constructed by appending the value of this key to the value of the NET_DIRECTORY key.

NET_PARKING_TABLE

The network parking table key is required. It specifies the name of the TRANSIMS parking table file within the network directory. The full path and file name for the parking table is constructed by appending the value of this key to the value of the NET_DIRECTORY key.

NET_ACTIVITY_LOCATION_TABLE

The activity location table key is required. It specifies the name of the TRANSIMS 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. The activity location file is a primary input file for the ActGen process. It should contain

one or more data fields used as the attraction weight for the activity location in the location choice model.

NET_PROCESS_LINK_TABLE

The process link table key is required. It specifies the name of the TRANSIMS process file within the network directory. The full path and file name for the process link table is constructed by appending the value of this key to the value of the NET_DIRECTORY key. The process link data are used to assign vehicles to parking lots attached to activity locations.

NEW_NODE_TABLE

The node table key is required. It specifies the name of the TRANSIMS subarea node file within the new 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 NEW_DIRECTORY key.

NEW_LINK_TABLE

The link table key is required. It specifies the name of the new TRANSIMS subarea link file within the new 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 NEW_DIRECTORY key.

NEW_POCKET_LANE_TABLE

The network pocket lane table key is required. It specifies the name of the new TRANSIMS subarea pocket lane file within the new directory. The full path and file name for the pocket lane table is constructed by appending the value of this key to the value of the NEW_DIRECTORY key.

NEW_LANE_CONNECTIVITY_TABLE

The network lane connectivity table key is required. It specifies the name of the new TRANSIMS subarea lane connectivity file within the new directory. The full path and file name for the lane connectivity table is constructed by appending the value of this key to the value of the NEW_DIRECTORY key.

NEW_PARKING_TABLE

The network parking table key is required. It specifies the name of the new TRANSIMS subarea parking table file within the new directory. The full path and file name for the parking table is constructed by appending the value of this key to the value of the NEW_DIRECTORY key.

NEW_ACTIVITY_LOCATION_TABLE

The activity location table key is required. It specifies the name of the new TRANSIMS subarea activity location file within the new 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 NEW_DIRECTORY key. The activity location file is a primary input file for the ActGen process. It should contain one or more data fields used as the attraction weight for the activity location in the location choice model.

NEW_PROCESS_LINK_TABLE

The process link table key is required. It specifies the name of the new TRANSIMS subarea process file within the new directory. The full path and file name for the process link table is constructed by appending the value of this key to the value of the NEW_DIRECTORY key. The process link data are used to assign vehicles to parking lots attached to activity locations.

Optional Keys

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. Its default is FALSE. 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.

DEFAULT_FILE_FORMAT

Default format for files other than network files. Default is VERSION3. Other possible values include BINARY, FIXED_COLUMN, COMMA_DELIMITED, SPACE_DELIMITED, TAB_DELIMITED, CSV_DELIMITED, DBASE, LANL and SQLITE3.

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_ZONE_TABLE

The network zone table key is optional. It specifies the name of the TRANSIMS zone file within the network directory. The full path and file name for the zone table is constructed by appending the value of this key to the value of the NET_DIRECTORY key

NET_SHAPE_TABLE

The network shape table key is optional. It specifies the name of the TRANSIMS 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

NET_LANE_USE_TABLE

The network lane use table key is optional. It specifies the name of the TRANSIMS lane use file within the network directory. The full path and file name for the lane use table is constructed by appending the value of this key to the value of the NET_DIRECTORY key.

NET_TURN_PROHIBITION_TABLE

The network turn prohibition table key is optional. It specifies the name of the TRANSIMS turn prohibition file within the network directory. The full path and file name for the turn prohibition table is constructed by appending the value of this key to the value of the NET_DIRECTORY key.

NET_TOLL_TABLE

The network toll table key is optional. It specifies the name of the TRANSIMS toll file within the network directory. The full path and file name for the toll table is constructed by appending the value of this key to the value of the NET_DIRECTORY key

NET_UNSIGNALIZED_NODE_TABLE

The network unsignalized node table key is optional. It specifies the name of the TRANSIMS unsignalized node file within the network directory. The full path and file name for the unsignalized node table is constructed by appending the value of this key to the value of the NET_DIRECTORY key

NET_SIGNALIZED_NODE_TABLE

The network signalized node table key is optional. It specifies the name of the TRANSIMS signalized node file within the network directory. The full path and file name for the signalized node table is constructed by appending the value of this key to the value of the NET_DIRECTORY key

NET_TIMING_PLAN_TABLE

The network timing plan table key is optional. It specifies the name of the TRANSIMS timing plan file within the network directory. The full path and file name for the timing plan table is constructed by appending the value of this key to the value of the NET_DIRECTORY key

NET_PHASING_PLAN_TABLE

The network phasing plan table key is optional. It specifies the name of the TRANSIMS phasing plan file within the network directory. The full path and file name for the phasing plan table is constructed by appending the value of this key to the value of the NET_DIRECTORY key

NET_DETECTOR_TABLE

The network detector table key is optional. It specifies the name of the TRANSIMS detector file within the network directory. The full path and file name for the detector table is constructed by appending the value of this key to the value of the NET_DIRECTORY key

NET_SIGNAL_COORDINATOR_TABLE

The network signal coordinator table key is optional. It specifies the name of the TRANSIMS signal coordinator file within the network directory. The full path and file name for the signal coordinator table is constructed by appending the value of this key to the value of the NET_DIRECTORY key

NET_TRANSIT_STOP_TABLE

The transit stop table is optional. If the stop table is not provided, transit paths cannot be built. This key specifies the name of the TRANSIMS transit stop file within the network directory. The full path and file name for the transit stop table is constructed by appending the value of this key to the value of the NET_DIRECTORY key.

NET_TRANSIT_ROUTE_TABLE

The transit route table is required if the transit stop file is provided. This key specifies the name of the TRANSIMS transit route file within the network directory. The full path and file name for the transit route table is constructed by appending the value of this key to the value of the NET_DIRECTORY key.

NET_TRANSIT_SCHEDULE_TABLE

The transit schedule table is required if the transit stop file is provided. This key specifies the name of the TRANSIMS transit schedule file within the network directory. The full path and file name for the transit schedule table is constructed by appending the value of this key to the value of the NET_DIRECTORY key.

NET_TRANSIT_DRIVER_TABLE

The transit driver table is required if the transit stop file is provided. This key specifies the name of the TRANSIMS transit driver file within the network directory. The full path and file name for the transit driver table is constructed by appending the value of this key to the value of the NET_DIRECTORY key.

NEW_DIRECTORY

The new 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.

NEW_ZONE_TABLE

The network zone table key is optional. It specifies the name of the output TRANSIMS subarea zone file within the new directory. The full path and file name for the zone table is constructed by appending the value of this key to the value of the NEW_DIRECTORY key

NEW_SHAPE_TABLE

The network shape table key is optional. It specifies the name of the output TRANSIMS subarea shape file within the new 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 NEW_DIRECTORY key

NEW_LANE_USE_TABLE

The network lane use table key is optional. It specifies the name of the output TRANSIMS subarea lane use file within the new directory. The full path and file name for the lane use table is constructed by appending the value of this key to the value of the NEW_DIRECTORY key.

NEW_TURN_PROHIBITION_TABLE

The network turn prohibition table key is optional. It specifies the name of the output TRANSIMS subarea turn prohibition file within the new directory. The full path and file name for the turn prohibition table is constructed by appending the value of this key to the value of the NEW_DIRECTORY key.

NEW_TOLL_TABLE

The network toll table key is optional. It specifies the name of the output TRANSIMS subarea toll file within the new directory. The full path and file name for the toll table is constructed by appending the value of this key to the value of the NEW_DIRECTORY key

NEW_UNSIGNALIZED_NODE_TABLE

The network unsignalized node table key is optional. It specifies the name of the output TRANSIMS subarea unsignalized node file within the new directory. The full path and file name for the unsignalized node table is constructed by appending the value of this key to the value of the NEW_DIRECTORY key

NEW_SIGNALIZED_NODE_TABLE

The network signalized node table key is optional. It specifies the name of the output TRANSIMS subarea signalized node file within the new directory. The full path and file name for the signalized node table is constructed by appending the value of this key to the value of the NEW_DIRECTORY key

NEW_TIMING_PLAN_TABLE

The network timing plan table key is optional. It specifies the name of the output TRANSIMS subarea timing plan file within the new directory. The full path and file name for the timing plan table is constructed by appending the value of this key to the value of the NEW_DIRECTORY key

NEW_PHASING_PLAN_TABLE

The network phasing plan table key is optional. It specifies the name of the output TRANSIMS subarea phasing plan file within the new directory. The full path and file name for the phasing plan table is constructed by appending the value of this key to the value of the NEW_DIRECTORY key

NEW_DETECTOR_TABLE

The network detector table key is optional. It specifies the name of the output TRANSIMS subarea detector file within the new directory. The full path and file name for the detector table is constructed by appending the value of this key to the value of the NEW_DIRECTORY key

NEW_SIGNAL_COORDINATOR_TABLE

The network signal coordinator table key is optional. It specifies the name of the output TRANSIMS subarea signal coordinator file within the new directory. The full path and file name for the signal coordinator table is constructed by appending the value of this key to the value of the NEW_DIRECTORY key

NEW_TRANSIT_STOP_TABLE

The transit stop table is optional. If the stop table is not provided, transit paths cannot be built. This key specifies the name of the output TRANSIMS subarea transit stop file within the new directory. The full path and file name for the transit stop table is constructed by appending the value of this key to the value of the NEW_DIRECTORY key.

NEW_TRANSIT_ROUTE_TABLE

The transit route table is required if the transit stop file is provided. This key specifies the name of the output TRANSIMS subarea transit route file within the new directory. The full path and file name for the transit route table is constructed by appending the value of this key to the value of the NEW_DIRECTORY key.

NEW_TRANSIT_SCHEDULE_TABLE

The transit schedule table is required if the transit stop file is provided. This key specifies the name of the output TRANSIMS subarea transit schedule file within the new directory. The full path and file name for the transit schedule table is constructed by appending the value of this key to the value of the NEW_DIRECTORY key.

NEW_TRANSIT_DRIVER_TABLE

The transit driver table is required if the transit stop file is provided. This key specifies the name of the output TRANSIMS subarea transit driver file within the new directory. The full path and file name for the transit driver table is constructed by appending the value of this key to the value of the NEW_DIRECTORY key.

NET_DEFAULT_FORMAT

Default format for network files. The default file format is set by DEFAULT_FILE_FORMAT. Other options include VERSION3, BINARY, FIXED_COLUMN, COMMA_DELIMITED, SPACE_DELIMITED, TAB_DELIMITED, CSV_DELIMITED, DBASE, LANL..

NET_*_FORMAT

The file format key enables the user to specify the input format for an input network file. Replace the * with any of the network file types: node, link, pocket_lane, etc. The default file format is set by NET_DEFAULT_FORMAT. Other options include VERSION3, BINARY, FIXED_COLUMN, COMMA_DELIMITED, SPACE_DELIMITED, TAB_DELIMITED, CSV_DELIMITED, DBASE, LANL.

NEW_DEFAULT_FORMAT

Default format for new output (subarea) files. The default file format is set by DEFAULT_FILE_FORMAT. Other options include VERSION3, BINARY, FIXED_COLUMN, COMMA_DELIMITED, SPACE_DELIMITED, TAB_DELIMITED, CSV_DELIMITED, DBASE, LANL.

.

NEW_*_FORMAT

The file format key enables the user to specify the input format for a new subarea network file. Replace the * with any of the network file types: node, link, pocket_lane, etc. The default file format is set by NEW_DEFAULT_FORMAT. The format options include VERSION3, BINARY, FIXED_COLUMN, COMMA_DELIMITED, SPACE_DELIMITED, TAB_DELIMITED, CSV_DELIMITED, DBASE, LANL.

Algorithmic Notes

A rectangular subarea is drawn around link 22, and includes portions of links 7 and 19 (Figure 1).

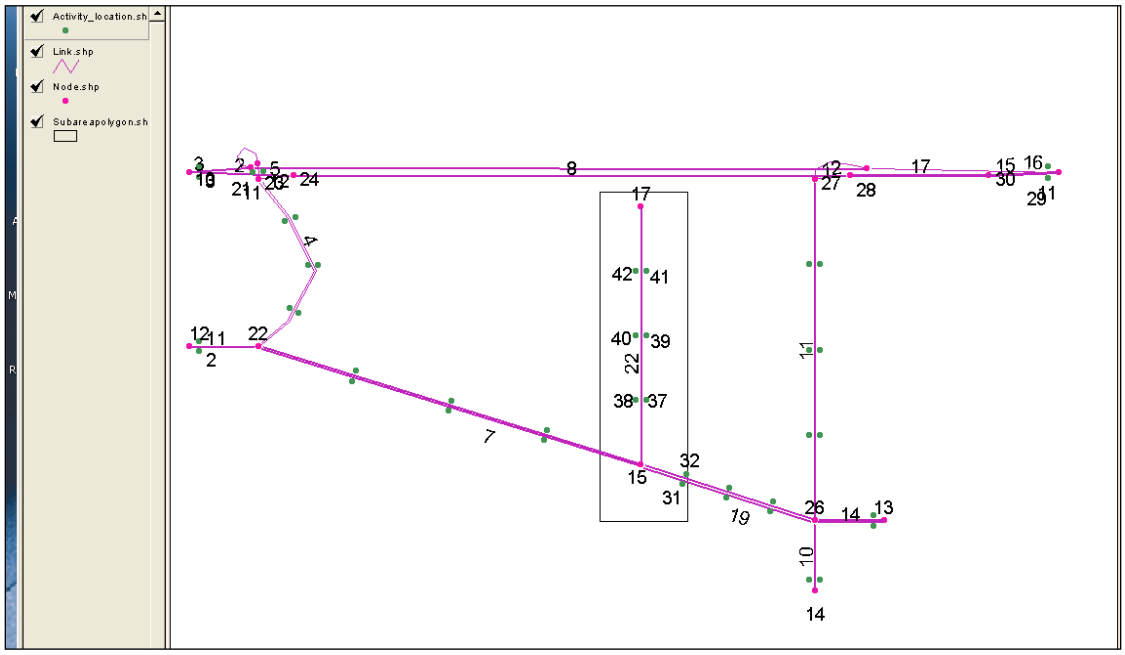

Figure 1 Example Network with Subarea Indicated

Figure 2 shows the subarea. If any part of a link is within the subarea polygon, that link becomes part of the subarea. New subarea origin and subarea destination activity locations (numbered 1001 – 1004) were created at nodes 22 and 26 (the boundary of the subarea). Trips that start or end outside of the subarea will enter or exit the subarea at these locations.

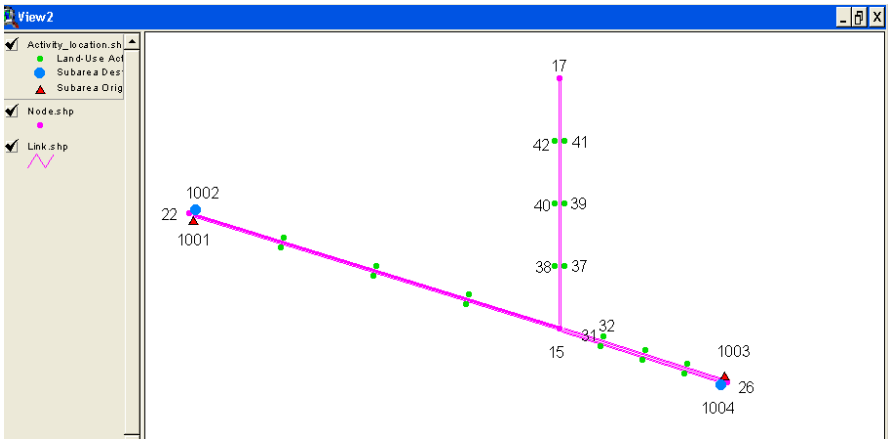

Figure 2 Subarea

Sample Printouts

Sample printout files generated by the **SubareaNet** program are shown below. Each printout 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.

Example 1

```
*****************************************
|                                       |
|       SubareaNet - Version 4.0.11     |
|   Copyright (c) 2009 by AECOM Consult |
|         Thu Oct 21 10:42:37 2010      |
|                                       |
*****************************************

Control File = SubareaNet.ctl
Report_File  = subareanet.prn (Append)

Creates a subarea network

Default File Format = TAB_DELIMITED

Network Directory = network
Node File = network\Node.txt
Zone File = network\Zone.txt
Shape File = network\Shape.txt
Link File = network\Link.txt
Pocket Lane File = network\Pocket_Lane.txt
Lane Use File = network\Lane_Use.txt
Toll File = network\Toll.txt
Lane Connectivity File = network\Lane_Connectivity.txt
Parking File = network\Parking.txt
Activity Location File = network\Activity_Location.txt
Process Link File = network\Process_Link.txt
Unsignalized Node File = network\Unsignalized_Node.txt
Signalized Node File = network\Signalized_Node.txt
Phasing Plan File = network\Phasing_Plan.txt
Timing Plan File = network\Timing_Plan.txt
Detector File = network\Detector.txt
Signal Coordinator File = network\Signal_Coordinator.txt

New Network Directory = subnet
New Node File = subnet\Node.txt
New Zone File = subnet\Zone.txt
New Shape File = subnet\Shape.txt
New Link File = subnet\Link.txt
New Pocket Lane File = subnet\Pocket_Lane.txt
New Lane Use File = subnet\Lane_Use.txt
New Toll File = subnet\Toll.txt
New Lane Connectivity File = subnet\Lane_Connectivity.txt
New Turn Prohibition File = subnet\Turn_Prohibition.txt
New Parking File = subnet\Parking.txt
New Activity Location File = subnet\Activity_Location.txt
New Process Link File = subnet\Process_Link.txt
New Unsignalized Node File = subnet\Unsignalized_Node.txt
New Signalized Node File = subnet\Signalized_Node.txt
```

```
New Phasing Plan File = subnet\Phasing_Plan.txt
New Timing Plan File = subnet\Timing_Plan.txt
New Detector File = subnet\Detector.txt
New Signal Coordinator File = subnet\Signal_Coordinator.txt

Notes and Name Fields will be Created

Subarea Boundary Polygon = subareapolygon.shp

External Offset Length = 15 meters


Number of Node File Records = 3490

Number of Zone File Records = 577

Number of Shape File Records = 314860
Number of Shape Data Records = 3005

Number of Link File Records = 4797
Number of Link Data Records = 1537
Number of Directional Links = 2570

Number of Pocket Lane File Records = 3315

Number of Lane Use File Records = 83

Number of Toll File Records = 12

Number of Lane Connectivity File Records = 20929

Number of Parking File Records = 18140

Number of Activity Location File Records = 18136
Number of Activity Location Data Records = 5878

Number of Process Link File Records = 36256

Number of Unsignalized Node File Records = 5288

Number of Timing Plan File Records = 2655

Number of Signal Coordinator File Records = 802

Number of Signalized Node File Records = 804
Number of Signalized Node Data Records = 281

Number of Detector File Records = 4748

Number of Phasing Plan File Records = 10954

Number of Subarea Node Records = 1149
Number of Subarea Link Records = 1537
Number of Subarea Pocket Lane Records = 1145
Number of Subarea Lane Connectivity Records = 6900
Number of Subarea Activity Locations = 6005
Number of Subarea Parking Lots = 6009
Number of Subarea Process Links = 11994
```

```
Number of Subarea Zone Records = 157
Number of Subarea Shape Records = 991
Number of Subarea Lane Use Records = 8
Number of Subarea Turn Prohibition Records = 0
Number of Subarea Toll Records = 8
Number of Subarea Unsignalized Node Records = 1603
Number of Subarea Signalized Node Records = 281
Number of Subarea Timing Plan Records = 927
Number of Subarea Phasing Plan Records = 3800
Number of Subarea Detector Records = 1614
Number of Subarea Signal Coordinator Records = 278

Number of Boundary Links = 75
Number of Short Links Skipped = 0
Number of New Activity Locations = 127
Number of New Parking Lots = 127
Number of New Process Links = 254

Thu Oct 21 10:42:42 2010 -- Process Complete (0:00:05)
```