SubareaPlans (version 4.0.33)

Revision History
20 October 2010 Created by Volpe Center

The SubareaPlans program:

1. Converts a regional plan file to a subarea plan file by creating trip origins and destinations on links that cross a subarea boundary polygon in ArcView format.
2. Adds boundary activity locations, process links, and parking lots to the regional network files.
3. Creates subarea transit routes and schedules from regional network files using a subarea boundary polygon in ArcView format.

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

```
SubareaPlans [-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 PLANS

The following **SubareaPlans** control file creates a subset of plans, 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 subarea plans
REPORT_FILE              SubareaPlans.prn
REPORT_FLAG              True
#PROJECT_DIRECTORY
DEFAULT_FILE_FORMAT                TAB_DELIMITED
#---- Input Files ----
```

```
NET_DIRECTORY                network
NET_NODE_TABLE                Node.txt
NET_LINK_TABLE                Link.txt
NET_PARKING_TABLE            Parking.txt
NET_LANE_CONNECTIVITY_TABLE           Lane_Connectivity.txt
NET_ACTIVITY_LOCATION_TABLE        Activity_Location.txt
NET_PROCESS_LINK_TABLE        Process_Link.txt
VEHICLE_FILE             Vehicle.txt
CREATE_NOTES_AND_NAME_FIELDS              YES
PLAN_FORMAT              TAB_DELIMITED
PLAN_FILE              8.TestBed.TimePlanA

#---- Parameter Files ----
SUBAREA_BOUNDARY_POLYGON              subareapolygon.shp
MAX_WARNING_MESSAGES        0
MAX_WARNING_EXIT_FLAG        False
RETIME_EARLY_ARRIVALS            False
TRAVELER_SCALING_FACTOR        1.0
NODE_LIST_PATHS        False

#---- Output Files ----
NEW_DIRECTORY              subarea    # Only applies to new Network files
NEW_BOUNDARY_SPEED_FILE             SpeedBdry.txt
NEW_PLAN_FORMAT              TAB_DELIMITED
NEW_PLAN_FILE               sub.TimePlanA
NEW_HOUSEHOLD_LIST              sub.Household.txt
NEW_VEHICLE_FILE             sub.Vehicle.txt
SUBAREAPLANS_REPORT_1          TRIP_SUMMARY_REPORT
SUBAREAPLANS_REPORT_2          PRINT_ZONE_EQUIVALENCIES
SUBAREAPLANS_REPORT_3          TIME_PERIOD_EQUIVALENCE
```

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 **SubareaPlans** program are listed below. These keys can be defined in a variety of different ways to perform different tasks. The first 10 keys specify the subarea polygon, input network, plan and vehicle files, and the output plan and vehicle files. 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.

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_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.

PLAN_FILE

The plan file key is required. It specifies the name of the TRANSIMS plan file within the project directory. The full path and file name for the plan file is constructed by appending the value of this key to the value of the PROJECT_DIRECTORY key.

VEHICLE_FILE

The vehicle file key is required. It specifies the name of the TRANSIMS vehicle file within the project directory. The full path and file name for the plan file is constructed by appending the value of this key to the value of the PROJECT_DIRECTORY key.

NEW_PLAN_FILE

The new plan file key is required. It specifies the name of the output TRANSIMS subarea plan file within the project directory. The full path and file name for the plan file is constructed by appending the value of this key to the value of the PROJECT_DIRECTORY key. If the partition number is '0', the '.tAA' extension is added. If the partition number is '1', the '.tAB' extension is added, etc. If the command line does not include a partition parameter and this key ends with '.t*'

or '.*', all of the plan files in the file group are processed sequentially. If the plan files have a companion *.def file, the PLAN_FORMAT and NODE_LIST_PATHS keys are not required.

NEW_VEHICLE_FILE

The vehicle file key is required. It specifies the name of the output TRANSIMS subarea vehicle file within the project directory. The full path and file name for the vehicle file is constructed by appending the value of this key to the value of the PROJECT_DIRECTORY key.

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.

LINK_DELAY_FILE

The link delay file key is optional. If the key is provided, the program uses the information in the link delay file to update the travel times on the legs of each plan. The header record in the link delay file is used to determine the size of each time period. The time periods are typically 15 minutes long.

LINK_DELAY_FORMAT

The file format keys can be used to specify the input or output file formats. 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.

PLAN_FORMAT

The plan format key is optional. If provided, it defines the file format for the input plan file. The default value is VERSION3 (unformatted text) format. This parameter enables the user to input plans in BINARY format.

NODE_LIST_PATHS

The node list paths key is optional and when provided specifies the way the path is identified in the input plan file. The key is "true" by default. This means that the input plans will include a list of the node ID numbers along the travel path. If the key is "false", the program interprets the path as a list of link ID numbers. If the first character of the key is "0", "N", "n", "F", or "f", the key is interpreted as "false".

NEW_PLAN_FORMAT

The new plan format key is optional. If provided, it defines the file format for the output plan file. The default value is VERSION3 (unformatted text) format. This parameter enables the user to output plans in BINARY format.

NEW_BOUNDARY_SPEED

The new boundary speed key is optional. If this file name is provided, SubareaPlans produces a boundary speed file. This file is used by the microsimulator to indicate the speed that vehicles enter or exit the subarea. An example appears below. The "Parking" lots represent the boundaries of the subarea.

PARKING	*START*	*END*	*SPEED*
1001	0:00	0:15	14.01
1001	0:15	0:30	14.01
…	..	…	…

1001	7:45	8:00	14.01
1001	8:00	8:15	12.95
1001	8:15	8:30	12.31
1001	8:30	8:45	12.49
1001	8:45	9:00	12.60
…			

NEW_HOUSEHOLD_LIST

The new household list key is optional. If used, SubareaPlans provides a list of those households whose travel plans enter, exit or cross the subarea.

RETIME_EARLY_ARRIVALS

The retime_early_arrivals key is optional, and defaults to False. If True, the arrival times of certain plans that terminate on boundary links are retimed. If False, the arrival times from the plans are retained.

ZONE_EQUIVALENCE_FILE

The zone equivalence file aggregates trips or skim values into summary districts.
The zone equivalence file is required for the trip adjustment factors. The key specifies the name of the file that defines a group of zones. Zone Groups typically represent large geographic areas or governmental entities (i.e., cities and counties). Each zone may only be associated with one Zone Group. The software generates warning messages if a zone is used more than once or appears to be missing from the sequence of zone numbers.

The zone equivalence file is a tab, space, or comma-delimited ASCII file with special format rules. A sample equivalence file is shown below.

```
1  0  Portland CBD - 1
1  1  1..16
2  0  West Suburbs - 2
2  1  79..307, 1248..1253
3  0  Southwest Suburbs - 3
3  1  308..403, 931..933
4  0  Southeast Suburbs - 4
4  1  404..557, 934..943, 1254..1258
5  0  East Portland - 5
5  1  561..563, 714..721, 731..738, 763..929, 949..961, 963..969
6  0  East Suburbs - 6
6  1  558..560, 564..713, 722..730, 739..762, 1259..1260
7  0  West Portland - 7
7  1  17..78, 930, 944..948, 962, 1247
8  0  Clark County - 8
8  1  970..1246
```

If the file contains a header record, it is ignored by the software. The first integer on each subsequent record is the district or zone group number. This number is followed by an index number that is used to associate multiple records with a given district. If the index number is zero, the software interprets everything that follows the index number as the district label. The first 25 characters of the label are printed in reports.

If the index number is not zero, the values that follow are interpreted as a range of zone numbers. Individual zone numbers and ranges of zone numbers can be specified on a given record. A range of zone numbers is specified using the first and last number in the sequence connected by two or more periods. For example, "79..307" represents all of the zone numbers between 79 and 307.

TIME_PERIOD_EQUIVALENCE

The zone equivalence file aggregates trips or skim values into summary districts.

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_LANE_CONNECTIVITY_TABLE

The network lane connectivity table key is optional. 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_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_PARKING_TABLE

The new network parking table key is optional. It specifies the name of the output TRANSIMS subarea parking 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. If the subarea plan file is to be simulated, a network file with the subarea boundary external stations is required. This can be achieved by creating new activity location, process link, and parking lot files in this program or by applying the SubareaNet program. SubareaNet creates the same external station records, but also removes all of the remaining records outside of the subarea. This program only adds the external stations to the regional network.

NEW_ACTIVITY_LOCATION_TABLE

The new network activity_location table key is optional. It specifies the name of the output TRANSIMS subarea activity_location 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_PROCESS_LINK_TABLE

The new network process_link table key is optional. It specifies the name of the output TRANSIMS subarea process_link 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_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. If the subarea transit plans are to be simulated, the subarea transit network must be created. This can be achieved by creating new transit file in this program or by applying the SubareaNet program. SubareaNet creates the same subarea transit network as this program. The regional transit network can not be updated to accommodate subarea transit plans in the same way as the highway network.

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

The following examples illustrates the functioning of SubareaPlans. Figure 1 shows a network with two plans. The first plan (in Blue) starts at Activity Location 39 on link 22, traverses links 7, 4, 5, 8 and 17, and ends at Activity Location 29 on link 15. The second plan (in Red), starts at activity location 30 on link 16, traverses links 13, 11 and 19, and ends at Activity Location 18 on link 7. Figure 2 shows the links in the subarea network.

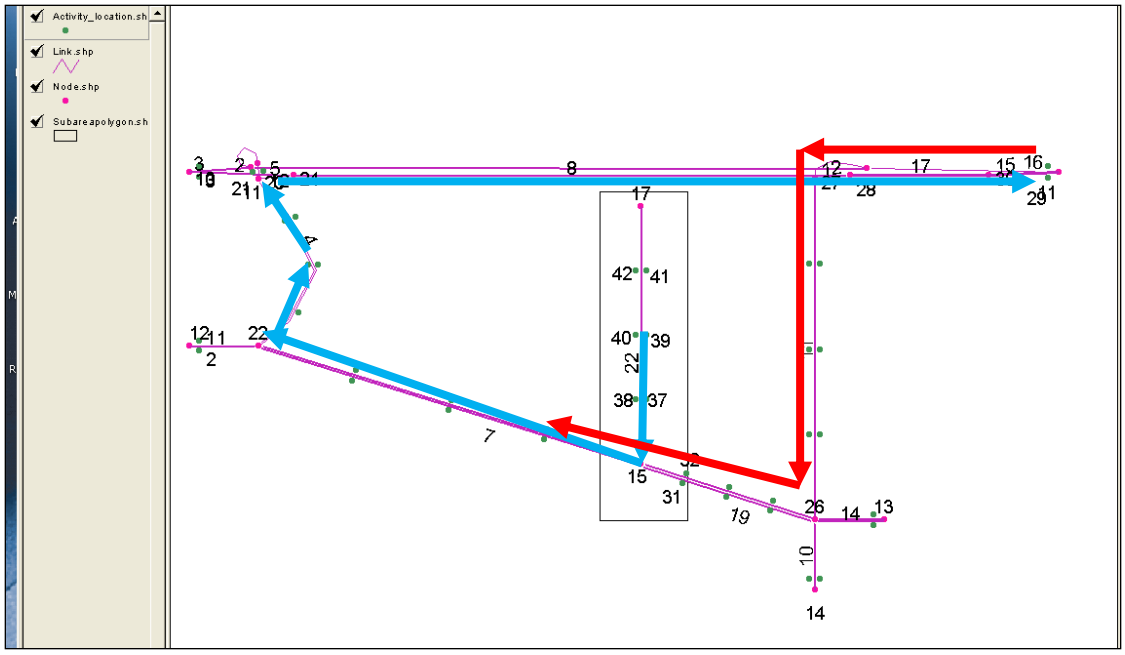

Figure 1 Test Network with Two Plans

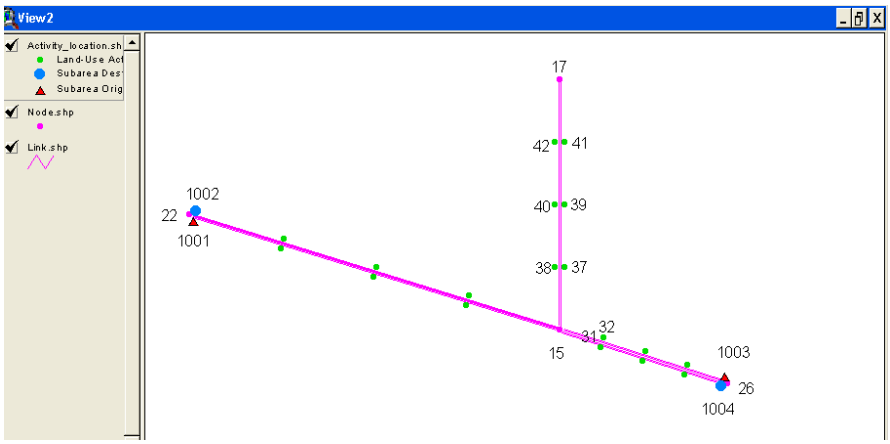

Figure 2 Subarea Network

The Blue plans are as follows

Leg	*Original Plan File*	*Original Description*	*Subarea Plan File*	*Subarea Description*
1	201 0 1 1 29341 39 1 39 2 30 29371 1 0 60 0 2 0	Leg 1 is the 30 second walk leg to the parking lot at activity location 39. Arrives at time 29371 (seconds after midnight)	201 0 1 1 29341 39 1 39 2 30 29371 1 0 60 0 2 0	Same as in the original plan file
2	201 0 1 2 29371 39 2 29 2 186 29557 1 0 186 1 0 9 2 0 -22 -7 4 5 8 17 -15	The drive leg from parking lot 39 to parking lot 29. The trip takes 186 seconds, and traverses links 22, 7, 4, 5, 8, 17, 15 (negative signs indicate traversal in B-A direction)	201 0 1 2 29371 39 2 1002 2 113 29484 1 0 113 1 0 4 2 0 -22 -7	The drive leg from parking lot 39 to lot 1002 (the exit from the subarea). The trip takes 113 seconds, and traverses links 22 and 7.
3	201 0 1 3 29557 29 2 29 1 30 29587 1 0 60 0 2 0	Final walk link from parking lot 29 to activity_location 29. Trip completes at time 29587	201 0 1 3 29484 1002 2 1002 1 1 29485 1 0 0 0 2 0	Dummy leg at the exit from the subarea. Trip exits at time 29485.

The Red plans are as follows

Leg	*Original Plan File*	*Original Description*	*Subarea Plan File*	*Subarea Description*
1	425501 0 1 1 28996 30 1 30 2 30 29026 1 0 60 0 2 0	Leg 1 is the 30 second walk leg to the parking lot at activity location 30. Arrives at time 29026	425501 0 1 1 29144 1003 1 1003 2 1 29145 1 0 0 0 2 0	Leg 1 is now at the entrance to the subarea, at location 1003, and time 29145.
2	425501 0 1 2 29026 30 2 18 2 141 29167 1 0 141 1 0 7 4255 0 16 13 -11 -19 -7	The drive leg from parking lot 30 to parking lot 18. The trip takes 141 seconds, and traverses links 16, 13, 11, 19, and 7.	425501 0 1 3 29145 1003 2 1002 2 120 29265 1 0 120 1 0 4 4255 0 -19 -7	The drive leg from parking lot 1003 to parking lot 1002. The trip takes 120 seconds, and traverses links 19 and 7. (The trip ends on a boundary link)
3	425501 0 1 3 29167 18 2 18 1 30 29197 1 0 60 0 2 0	Final walk leg at activity location 18. Trip completes at time 29197	425501 0 1 4 29265 1002 2 1002 1 1 29266 1 0 0 0 2 0	Dummy ending at parking lot 1002
4			425501 0 2 1 29167 18 2 18 1 30 29197 1 0 60 0 2 0	"Real" ending at activity location 18. The trip completes at time 29197[1]

[1] If RETIME_EARLY_ARRIVALS is set to true, this time would be 29026, somewhat earlier.

Sample Printouts

Sample printout files generated by the **SubareaPlans** 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

```
*****************************************
|                                       |
|      SubareaPlans - Version 4.0.33    |
|   Copyright (c) 2009 by AECOM Consult |
|         Fri Oct 22 10:48:42 2010      |
|                                       |
*****************************************

Control File = SubareaPlans.ctl
Report_File  = SubareaPlans.prn (Append)

Creates the subarea plans & link delays

Default File Format = TAB_DELIMITED

Maximum Warning Messages = 0
Maximum Warning Exit Flag = False

Traveler Scaling Factor = 1

Input Plan File = 8.TestBed.TimePlanA
Input Plan File Format = TAB_DELIMITED
Plan File contains Link List Paths

Network Directory = network
Node File = network\Node.txt
Link File = network\Link.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

New Network Directory = subarea

Vehicle File = Vehicle.txt
Vehicle File will be Sorted by Vehicle ID

New Vehicle File = sub.Vehicle.txt

New Boundary Speed File = SpeedBdry.txt

Subarea Boundary Polygon = subareapolygon.shp

New Plan File = sub.TimePlanA
New Plan File Format = TAB_DELIMITED

New Household List = sub.Household.txt

Activity Schedules will Not be Adjusted for Trips that Arrive Early
```

```
Time of Day Format = 24_HOUR_CLOCK

SubareaPlans Reports:  1. TRIP_SUMMARY_REPORT
                       2. PRINT_ZONE_EQUIVALENCIES
                       3. TIME_PERIOD_EQUIVALENCE


Number of Node File Records = 17

Number of Link File Records = 20
Number of Directional Links = 28

Number of Parking File Records = 42

Number of Activity Location File Records = 42

Number of Process Link File Records = 84

Number of Vehicle File Records = 9400

Number of Boundary Links = 2
Number of Short Links Skipped = 0
Number of New Activity Locations = 4
Number of New Parking Lots = 4
Number of New Process Links = 8

Number of Subarea Vehicles = 2471

Trip Summary Report

Period     Start        End      Trips

    1   Internal   External        745
    1   External   Internal        243
    1   External   External       1483

Number of Plan Files = 1
Number of Input Plans = 28200
Number of Input Records = 188000
Number of Input Travelers = 9400
Number of Input Trips = 9400

Number of Subarea Files = 1
Number of Subarea Plans = 7902
Number of Subarea Records = 52354
Number of Subarea Travelers = 2925
Number of Subarea Trips = 953

Number of Plans with Adjusted Times = 208
Average Travel Time Adjustment = 70 seconds (1.2 minutes)

Number of Internal-Internal Vehicle Trips = 0 (0.0%)
Number of Internal-External Vehicle Trips = 745 (30.1%)
Number of External-Internal Vehicle Trips = 243 (9.8%)
Number of External-External Vehicle Trips = 1483 (60.0%)

Fri Oct 22 10:48:42 2010 -- Process Complete (0:00:00)
```