

## **ActGen (version 4.0)**

### **Revision History**

1/8/2010 Edited by AECOM Consult, Inc.

4/20/2010 Edited by RSG, Inc.

The **ActGen** program generates, schedules and locates activities for each person in a household. The **ActGen** program is used to:

1. Match synthetic households to survey households.
2. Assign survey activity patterns to individual members of the synthetic household.
3. Coordinate household activities and vehicle usage.
4. Apply location choice models to locate activities by purpose
5. Adjust activity schedules based on travel times and schedule constraints.
6. Create an activity file and a summary of activity generation problems.
7. Updated an activity file by regenerating selected household.
8. Output an activity pattern file for editing with ActivityPattern and then relocating with LocationChoice.
9. Apply attraction balancing factors by zone and trip purpose.

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

***ActGen [-flag] [control\_file] [partition]***

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 partition parameter is optional. It is used to specify the partition number for a particular execution. Partitions are used in the TRANSIMS process for parallel execution on multiple CPUs. If a partition number is provided, the program will generate activities for the households included in the corresponding household file and store the results in the corresponding activity and problem files. A partition number of "0" corresponds to partition files ".tAA", "1" equals ".tAB", etc.

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 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 **ActGen** control file is shown below:

TITLE	ActGen Test
PROJECT_DIRECTORY	d:\software\test\actgen
NET_DIRECTORY	d:\software\test\actgen\network
NET_NODE_TABLE	Node.txt
NET_LINK_TABLE	Link.txt
NET_ZONE_TABLE	Zone.txt
NET_ACTIVITY_LOCATION_TABLE	Activity_Location.txt
NET_PARKING_TABLE	Parking.txt
NET_PROCESS_LINK_TABLE	Process_Link.txt
HOUSEHOLD_FILE	demand\Household.txt
HOUSEHOLD_TYPE_SCRIPT	demand\Household_Type.txt
POPULATION_FILE	demand\Population.txt
VEHICLE_TYPE_FILE	demand\Vehicle_Type.txt
VEHICLE_FILE	demand\Vehicle.txt
NEW_ACTIVITY_FILE	demand\Activity.txt
NEW_ACTIVITY_FORMAT	TAB_DELIMITED
RANDOM_NUMBER_SEED	1122
TIME_OF_DAY_FORMAT	24_HOUR_CLOCK
DISTANCE_CALCULATION	RIGHT_ANGLE
AVERAGE_TRAVEL_SPEED	1.0, 15.0
ADDITIONAL_TRAVEL_TIME	600, 180
SURVEY_HOUSEHOLD_FILE	survey\Household.txt
SURVEY_HOUSEHOLD_WEIGHTS	survey\Survey_Weights.txt
SURVEY_TYPE_SCRIPT	survey\Survey_Type.txt
SURVEY_POPULATION_FILE	survey\Population.txt
SURVEY_ACTIVITY_FILE	survey\Activity.txt
SURVEY_ACTIVITY_FORMAT	TAB_DELIMITED
ACTGEN_REPORT_1	HOUSEHOLD_TYPE_SCRIPT
ACTGEN_REPORT_2	HOUSEHOLD_TYPE_STACK
ACTGEN_REPORT_3	HOUSEHOLD_TYPE_SUMMARY
ACTGEN_REPORT_4	SURVEY_TYPE_SCRIPT
ACTGEN_REPORT_5	SURVEY_TYPE_STACK
ACTGEN_REPORT_6	SURVEY_TYPE_SUMMARY
ACTGEN_REPORT_7	LOCATION_CHOICE_SCRIPT
ACTGEN_REPORT_8	LOCATION_CHOICE_STACK
ACTIVITY_PURPOSE_RANGE_1	1, 7, 8, 11, 17, 18

ACTIVITY_ANCHOR_FLAG_1	TRUE
SCHEDULE_CONSTRAINT_1	FIXED_TIME
ZONE_BASED_METHOD_1	YES
LOCATION_CHOICE_SCRIPT_1	demand\Anchor_Model.txt
ACTIVITY_PURPOSE_RANGE_2	6, 16
ACTIVITY_ANCHOR_FLAG_2	FALSE
SCHEDULE_CONSTRAINT_2	PASSENGER
ZONE_BASE_METHOD_2	NO
LOCATION_WEIGHT_FIELD_2	USER1
LOCATION_WEIGHT_FACTOR_2	0.5
MODE_DISTANCE_FACTORS_2	-0.05, -0.0006, -0.07
ACTIVITY_PURPOSE_RANGE_3	2..5, 12..15
ACTIVITY_ANCHOR_FLAG_3	FALSE
SCHEDULE_CONSTRAINT_3	NO_CONSTRAINT
ZONE_BASED_METHOD_3	YES
LOCATION_CHOICE_SCRIPT_3	demand\Other_Model.txt
ZONE_SKIM_FILE_1	demand\Skim.txt
ZONE_SKIM_FORMAT_1	TAB_DELIMITED
TIME_PERIOD_EQUIVALENCE_1	demand\Period.txt

The keys recognized by the **ActGen** program are listed below. These keys can be defined in a variety of different ways to perform different tasks.

### 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 plus the partition number. 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 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\_ZONE\_TABLE**

The zone table key is optional. If provided, 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. This file is used for zone-based location choice models. It should contain one or more data fields used as the attraction weight for the zone in the location choice model.

### **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\_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\_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\_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.

**NET\_TRANSIT\_STOP\_TABLE**

The transit stop table key is optional. It 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. The transit stop data are used to assign transit accessibility to activity locations.

**DEMAND\_FILE\_FORMAT**

The demand file format key can be used to change the default 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, DBASE, and SQLITE3.

**HOUSEHOLD\_LIST**

The household list file is optional. If it is not provided, activities will be generated for all households in the household file. If it is provided, the key is appended to the value of the PROJECT\_DIRECTORY key to identify the full path to one or more household list files. A household list file is a simple list of the household ID numbers that will be processed by a particular ActGen application. A sample household list is shown below.

```
3
20
32
49
100
120
```

The household list key can be the path to a specific file or the root path to a group of partitioned files. If the command line includes a partition parameter, the program will add “.t\*” to the household list key. If the partition number is “0”, the household list will include the “.tAA” extension. If the partition number is “1”, the “.tAB” extension is used....

**ACTIVITY\_FILE**

The activity file is optional. If it is provided, all activities for households not included in the household list file will be copied to the new activity file with change. This file is typically used when updating or regenerating activities for household with problems. The key is appended to the PROJECT\_DIRECTORY key to specify the file name for the input activity file.

**ACTIVITY\_FORMAT**

The activity format key enables the user to specify the input format for the activity file. The default file format is set by DEMAND\_FILE\_FORMAT. The format options include VERSION3, BINARY, FIXED\_COLUMN, COMMA\_DELIMITED, SPACE\_DELIMITED, TAB\_DELIMITED, DBASE, and SQLITE3.

**HOUSEHOLD\_FILE**

The household file key is appended to the PROJECT\_DIRECTORY key to specify the file name for the input household file.

**HOUSEHOLD\_FORMAT**

The household format key enables the user to specify the input format for the household file. The default file format is set by DEMAND\_FILE\_FORMAT. The format options include VERSION3, BINARY, FIXED\_COLUMN, COMMA\_DELIMITED, SPACE\_DELIMITED, TAB\_DELIMITED, DBASE, and SQLITE3.

**HOUSEHOLD\_TYPE\_SCRIPT**

The household type script is optional. If provided, the key is appended to the PROJECT\_DIRECTORY key to specify the file name for a user program script. This script enables the user to set the household type based on data fields found on the input household file. If the key is not provided, the program needs to find a field labeled “TYPE” on the input household file. Refer to the Algorithm section of this document for more details about how the script file is used.

**POPULATION\_FILE**

The population file key is appended to the PROJECT\_DIRECTORY key to specify the file name for the input population file.

**POPULATION\_FORMAT**

The population format key enables the user to specify the input format for the population file. The default file format is set by DEMAND\_FILE\_FORMAT. The format options include VERSION3, BINARY, FIXED\_COLUMN, COMMA\_DELIMITED, SPACE\_DELIMITED, TAB\_DELIMITED, DBASE, and SQLITE3.

**VEHICLE\_FILE**

The vehicle file key is appended to the PROJECT\_DIRECTORY key to specify the file name for the input vehicle file.

**VEHICLE\_FORMAT**

The vehicle format key enables the user to specify the input format for the vehicle file. The default file format is set by DEMAND\_FILE\_FORMAT. The format options include VERSION3, BINARY, FIXED\_COLUMN, COMMA\_DELIMITED, SPACE\_DELIMITED, TAB\_DELIMITED, DBASE, and SQLITE3.

**VEHICLE\_TYPE\_FILE**

The vehicle type file is appended to the PROJECT\_DIRECTORY key to specify the file name for the input vehicle type file. The vehicle type file is used to validate the Vehicle Type and Subtype values on the vehicle file.

**NEW\_ACTIVITY\_FILE**

The new activity file is appended to the PROJECT\_DIRECTORY key to specify the file name for the output activity file created by the program. The program generates multiple activity records for each household person.

**NEW\_ACTIVITY\_FORMAT**

The new activity format key enables the user to specify the output format for the activity file. The default file format is set by DEMAND\_FILE\_FORMAT. The format options include VERSION3, BINARY, FIXED\_COLUMN, COMMA\_DELIMITED, SPACE\_DELIMITED, TAB\_DELIMITED, and DBASE.

**NEW\_PROBLEM\_FILE**

The new problem file is appended to the PROJECT\_DIRECTORY key to specify the file name for the output problem file created by the program. The program may generate multiple problem messages for each household. The PROBLEM field identifies the problem type for a given record. The message corresponding to each problem code is listed below.

- |                     |                       |                        |
|---------------------|-----------------------|------------------------|
| 1. Path Building    | 13. Bike Distance     | 25. Access Restriction |
| 2. Time Schedule    | 14. Departure Time    | 26. Transit Stop       |
| 3. Zero Node        | 15. Arrival Time      | 27. Activity Location  |
| 4. Vehicle Type     | 16. Link Access       | 28. Vehicle Passenger  |
| 5. Path Circuitry   | 17. Lane Connectivity | 29. Activity Duration  |
| 6. Travel Mode      | 18. Parking Access    | 30. Kiss-&-Ride Lot    |
| 7. Vehicle Access   | 19. Lane Merging      | 31. Vehicle ID         |
| 8. Walk Distance    | 20. Lane Changing     | 32. Data Sort          |
| 9. Wait Time        | 21. Turning Speed     | 33. Walk Location      |
| 10. Walk Access     | 22. Pocket Merge      | 34. Bike Location      |
| 11. Path Size       | 23. Vehicle Spacing   | 35. Transit Location   |
| 12. Park-&-Ride Lot | 24. Traffic Control   |                        |

**NEW\_PROBLEM\_FORMAT**

The new problem format key enables the user to specify the output format for the problem file. The default file format is VERSION3. The format options include FIXED\_COLUMN, COMMA\_DELIMITED, SPACE\_DELIMITED, TAB\_DELIMITED, BINARY, DBASE, and SQLITE3.

**NEW\_HOUSEHOLD\_MATCH\_FILE**

The new household match file is appended to the PROJECT\_DIRECTORY key to specify the file name for outputting the match between each person in a synthetic household and a person in

the survey household. This is a tab-delimited text file that contains the data shown in the HOUSEHOLD\_MATCH\_REPORT.

### **NEW\_HOUSEHOLD\_PERSON\_COUNT**

The new household person count is appended to the PROJECT\_DIRECTORY key to specify the file name for outputting the number of times a survey household person was replicated. This is a tab-delimited text file that contains the household, person, and count fields.

### **NEW\_ACTIVITY\_PATTERN\_FILE**

The new activity pattern file is appended to the PROJECT\_DIRECTORY key to specify the file name for outputting the household activity patterns. This is standard Activity file with negative numbers included for the locations of out-of-home activities. This file can be edited by the **ActivityPattern** program and read by the **LocationChoice** program.

### **NEW\_ACTIVITY\_PATTERN\_FORMAT**

The activity pattern format key enables the user to specify the output format for the activity pattern file. The default file format is set by DEMAND\_FILE\_FORMAT. The format options include VERSION3, BINARY, FIXED\_COLUMN, COMMA\_DELIMITED, SPACE\_DELIMITED, TAB\_DELIMITED, DBASE, and SQLITE3.

### **NEW\_TRIP\_TIME\_FILE**

The new trip time file is appended to the PROJECT\_DIRECTORY key to specify the file name for outputting the distribution of trip travel times by mode and origin and destination purpose. The number of trips or tour legs in 60 second increments of trip length is saved to the tab-delimited text file.

### **NEW\_TRIP\_DISTANCE\_FILE**

The new trip distance file is appended to the PROJECT\_DIRECTORY key to specify the file name for outputting the distribution of trip lengths by mode and origin and destination purpose. The number of trips or tour legs in 500 meter increments of trip length is saved to the tab-delimited text file.

### **TIME\_OF\_DAY\_FORMAT**

The time of day format defines how the activity start and end times are written to the activity file. The default format will display values in hours. The format options include HOURS, SECONDS, 24\_HOUR\_CLOCK, and 12\_HOUR\_CLOCK.

### **SURVEY\_HOUSEHOLD\_FILE**

The household file key is appended to the PROJECT\_DIRECTORY key to specify the file name for the input survey household file.



**SURVEY\_HOUSEHOLD\_FORMAT**

The survey household format key enables the user to specify the input format for the survey household file. The format options include VERSION3, BINARY, FIXED\_COLUMN, COMMA\_DELIMITED, SPACE\_DELIMITED, TAB\_DELIMITED, DBASE, and SQLITE3.

**SURVEY\_HOUSEHOLD\_WEIGHTS**

The survey household weights key is optional. If provided, the key is appended to the PROJECT\_DIRECTORY key to specify the file name for the input survey household weight file. The weights file is used to increase or decrease the probability that a given survey household will be selected for a given household type.

**SURVEY\_TYPE\_SCRIPT**

The survey type script is optional. If provided, the key is appended to the PROJECT\_DIRECTORY key to specify the file name for a user program script. This script enables the user to set the survey household type based on data fields found on the input survey household file. If the key is not provided, the program needs to find a field labeled “TYPE” on the input survey household file. Refer to the Algorithm section of this document for more details about how the script file is used.

**SURVEY\_POPULATION\_FILE**

The survey population file key is appended to the PROJECT\_DIRECTORY key to specify the file name for the input survey population file.

**SURVEY\_POPULATION\_FORMAT**

The survey population format key enables the user to specify the input format for the survey population file. The format options include VERSION3, BINARY, FIXED\_COLUMN, COMMA\_DELIMITED, SPACE\_DELIMITED, TAB\_DELIMITED, DBASE, and SQLITE3.

**SURVEY\_ACTIVITY\_FILE**

The survey activity file is appended to the PROJECT\_DIRECTORY key to specify the file name for the input survey activity file. The program uses the survey activity patterns to generate activity records for each household person.

**SURVEY\_ACTIVITY\_FORMAT**

The survey activity format key enables the user to specify the input format for the survey activity file. The format options include VERSION3, BINARY, FIXED\_COLUMN, COMMA\_DELIMITED, SPACE\_DELIMITED, TAB\_DELIMITED, DBASE, and SQLITE3.

**DISTANCE\_CALCULATION**

The program calculates the distance between activity locations as part of the location choice process. The distance values are provided to the location choice scripts and used to estimate the trip travel time for the activity scheduling process. By default, this distance is the straight-line distance between the activity location coordinates. This key can be used to select other distance calculation methods. The key values include STRAIGHT\_LINE, RIGHT\_ANGLE,

**SIMPLE\_AVERAGE**, and **WEIGHTED\_AVERAGE**. Right angle distance is the sum of the absolute difference of the X and Y coordinates. Simple average is the average of the straight line distance and the right angle distance. Weighted average uses the right angle distance for near-by locations and straight line distance for far-away locations.

### **MAXIMUM\_WALK\_DISTANCE**

Maximum walk distance is optional with units in meters. Acceptable values are integers ranging from 100 to 10,000. The default value is 2,000. The program uses this key to modify the way activity locations are selected within a selected destination zone. If the travel mode is “walk”, the algorithm gives preference to activity locations closer to the origin location rather than further way as it does by default for all motorized travel models. Activity locations that are further away than the maximum walk distance are not considered. Activity locations that are less than the maximum are weighted by subtracting their distance from the maximum walk distance.

### **AVERAGE\_TRAVEL\_SPEED**

The average travel speed and additional travel time keys are combined with distance to estimate the travel time between activity locations. The key can accepted a comma delimited list of speed values by mode of travel. The mode order is defined as:

Walk, Drive, Bus, Rail, Park-&-Ride Outbound, Park-&-Ride Inbound, Bicycle, Magic Move, School Bus, 2 Person Carpool, 3 Person Carpool, 4 Person Carpool, Kiss-&-Ride Outbound, and Kiss-&-Ride Inbound

The default value is 10 meters per second for all modes. If the user provides data for fewer than 14 modes, the last value provided with the key will be used for all subsequent modes. The resulting travel time is calculated as:

$$\text{Travel Time} = \text{Distance} / \text{Average\_Travel\_Speed}_{\text{mode}} + \text{Additional\_Travel\_Time}_{\text{mode}}$$

### **ADDITIONAL\_TRAVEL\_TIME**

The additional travel time and average travel speed keys are combined with distance to estimate the travel time between activity locations. The key can accepted a comma delimited list of time values in seconds by mode of travel. The mode order is defined as:

Walk, Drive, Bus, Rail, Park-&-Ride Outbound, Park-&-Ride Inbound, Bicycle, Magic Move, School Bus, 2 Person Carpool, 3 Person Carpool, 4 Person Carpool, Kiss-&-Ride Outbound, and Kiss-&-Ride Inbound

The default value is 600 seconds (10 minutes) for all modes. If the user provides data for fewer than 14 modes, the last value provided with the key will be used for all subsequent modes. This value can be interpreted as the vehicle access and parking time or a time buffer to account for congestion or distance errors. The resulting travel time is calculated as:

$$\text{Travel Time} = \text{Distance} / \text{Average\_Travel\_Speed}_{\text{mode}} + \text{Additional\_Travel\_Time}_{\text{mode}}$$

**RANDOM\_NUMBER\_SEED**

This key specifies the random number seed used by the **ActGen** program. If the key is not provided or the key value is zero, the random number seed will be set by the computer clock.

**TIME\_SCHEDULE\_VARIANCE**

Time schedule variance is optional with units in minutes. The default value is 0. It includes random variance in the activity schedule assigned to each synthetic household.

***Purpose Groups***

The following 12 keys make up an activity purpose group. The “#” at the end of the each key identifies the purpose group number. All keys with the same purpose group number combine to define how the activity location is identified. Any number of purpose groups can be defined. If there are gaps in the purpose codes, the program will warn the user of these missing values. If an activity is found in the survey activity file with a purpose code that is not included in a purpose group, the activity will be deleted from the activity pattern.

**ACTIVITY\_PURPOSE\_RANGE\_#**

The activity purpose range is required. It defines which activity purpose codes associated with the purpose group. The key is a comma delimited list of integer codes and code ranges. A code range is defined by the first and last values of the range joined by two or more periods (e.g., 2..6). Typical TRANSIMS activity purpose codes include:

0	Home
1	Work
2	Shop
3	Visit
4	Social/Recreation
5	Other
6	Serve Passenger
7	School
8	College

The user can change the meaning of these codes or add additional codes as needed. The only code that can not be changed is zero. Zero must be used to define at-home activities. Home will always be the beginning or end of travel tours. Zero should not be included in this key.

**ACTIVITY\_ANCHOR\_FLAG\_#**

The activity anchor flag is optional. If it is not provided, the program defaults to FALSE which means that the activities in this purpose group are not processed as tour anchors. The tour anchor is located first and then activities before or after the anchor activity are located given the anchor and home locations. If the first character of the key is “0”, “N”, “n”, “F”, or “f”, the key is interpreted as “false”.

**SCHEDULE\_CONSTRAINT\_#**

Schedule constraints are optional. If it is not provided, the program does not constrain the activity start time, end time or duration. The activity can therefore be rescheduled to coordinate with other activities and the travel times between activities. The options for this key include: NO\_CONSTRAINT, FIXED\_TIME, START\_TIME, END\_TIME, DURATION, and PASSENGER. (The program also recognizes NONE, FIXED, START, and END). The fixed, start, and end time constraints force the activity to keep the start and/or end time specified in activity survey. Duration maintains the activity duration, but can adjust the start and end times based on travel times. The passenger constraint is used when the activity schedule is defined by the schedule constraints of an auto passenger (e.g., a parent dropping their child off at school). In this case the activity purpose of the driver is typically “serve passenger”.

**ZONE\_BASED\_METHOD\_#**

The zone based method flag is optional. If it is not provided, the program defaults to TRUE which means that the location choice model for this purpose group will select the destination zone first and then an activity location within the zone. Zone related data is required to implement this method. If the first character of the key is “0”, “N”, “n”, “F”, or “f”, the key is interpreted as “false”.

**LOCATION\_CHOICE\_SCRIPT\_#**

The location choice script is optional. If provided, the key is appended to the PROJECT\_DIRECTORY key to specify the file name for a user program script. This script enables the user to calculate the relative probability of selecting a given activity location based on its proximity to the anchor activities. A random number is used to select the activity from the cumulative probability distribution. Scripts can be used for both zone-based and location-based location choice models as defined by the corresponding ZONE\_BASED\_METHOD\_# key. Refer to the Algorithm section of this document for more details about how the script file is used.

**ZONE\_WEIGHT\_FIELD\_#**

The zone weight field is optional. It is only processed when the ZONE\_BASED\_METHOD\_# key is true. This key specifies the field name in the zone file that contains the attraction weight value for the zone. This value is multiplied by the zone weight factor in the default zone-based location choice model. A location weight field is used to select an activity location within the selected zone. If a location choice script is provided, the value of the field is passed to the location choice script using the Tour.UTILITY variable.

**ZONE\_WEIGHT\_FACTOR\_#**

The zone weight factor is optional and is only processed when the zone weight field is specified. By default, the factor is 1.0. This value is multiplied by the value of the zone weight field in the default zone-based location choice model.

**BALANCING\_FACTOR\_FIELD\_#**

The balancing factor field is optional. It is used when the ZONE\_BASED\_METHOD\_# key is true. This key specifies the field name in the balancing factor file that contains the attraction adjustment factor for a given zone. This value is multiplied by the zone weight factor in the

default zone-based location choice model. A location weight field is used to select an activity location within the selected zone.

### **LOCATION\_WEIGHT\_FIELD\_#**

The location weight field is optional for most applications. It is only required for the default location-based location choice model. For zone-based models, this key defines the attraction weight for selecting a location within the zone. By default, the location weight is 1.0 which means that all activity locations have equal weights. This key enables the user to specify the field name in the activity location file that contains the attraction weight value that is used for the purpose group. This value is multiplied by the location weight factor in the default location choice model.

### **LOCATION\_WEIGHT\_FACTOR\_#**

The location weight factor is optional and is only processed when the default location-based location choice model is applied. By default, the factor is 1.0. This value is multiplied by the value of the location weight field in the default location choice model.

### **SKIM\_TIME\_FIELD\_#**

The skim time field is optional. It is only processed when location choice scripts are not provided. If this key is specified, it changes the default location choice model from a distance-based model to a time-based (or impedance-based) model. The key identifies a field in one of the zone skim files used to define the travel time or impedance between zones. The field name must identify the zone skim file number in addition to the field name within the corresponding skim file. The syntax is

*Skim#.Field\_Name*

Where # corresponds to the number specified in the ZONE\_SKIM\_FILE\_# key and *Field\_Name* is the name of one of the fields in that file. For example, if ZONE\_SKIM\_FILE\_2 contains the highway travel time in a field named DRIVE, this key would be

Skim2.DRIVE

### **MODE\_TIME\_FACTORS\_#**

The mode time factors are optional and are only processed when the location choice script is not provided and a skim time field is provided. The key can accepted a comma delimited list of time factors by mode of travel. The mode order is defined as:

Walk, Drive, Bus, Rail, Park-&-Ride Outbound, Park-&-Ride Inbound, Bicycle, Magic Move, School Bus, 2 Person Carpool, 3 Person Carpool, 4 Person Carpool, Kiss-&-Ride Outbound, and Kiss-&-Ride Inbound

By default, the factors are -0.001 for all modes. If the user provides data for fewer than 14 modes, the last value provided with the key will be used for all subsequent modes. Refer to the Algorithm section of this document for more details about how this parameter is used.

**MODE\_DISTANCE\_FACTORS\_#**

The mode distance factors are optional and are only processed when the location choice script and a skim time field are not provided. The key can accepted a comma delimited list of distance factors by mode of travel. The mode order is defined as:

Walk, Drive, Bus, Rail, Park-&-Ride Outbound, Park-&-Ride Inbound, Bicycle, Magic Move, School Bus, 2 Person Carpool, 3 Person Carpool, 4 Person Carpool, Kiss-&-Ride Outbound, and Kiss-&-Ride Inbound

By default, the factors are -0.001 for all modes. If the user provides data for fewer than 14 modes, the last value provided with the key will be used for all subsequent modes. Refer to the Algorithm section of this document for more details about how this parameter is used.

**Zone Skim Groups**

The following 3 keys make up a zone skim group. The “#” at the end of the each key identifies the skim group number. All keys with the same skim group number combine to define the zone-to-zone skim data available for each location choice script. Zone skims are not processed if none of the activity purpose groups include a location choice script or a skim zone field. Any number of skim groups can be defined. Multiple groups are typically used to specify the travel characteristics of different modes of travel and/or different time periods.

**ZONE\_SKIM\_FILE\_#**

The zone skim file key is appended to the PROJECT\_DIRECTORY key to specify the file name for a zone-to-zone travel skim file. If the zone skim format is not specified and a Definition file is not found, the program assumes the file is in Version 3 format. The default Version 3 format is a tab-delimited text file with seven data fields and no header record. The fields are assumed to have the following meaning:

Origin Zone Number  
 Destination Zone Number  
 Travel Mode Code  
 Time Period or Interval  
 Travel Time in Seconds  
 Number of Observations  
 Travel Time Variance in Seconds

For other formats any number of fields can be specified. The file must include two (or three) fields that the program needs for indexing the skim data. These are:

Origin Zone Number → ORG, ORIGIN, FROM, or FROM\_ZONE  
 Destination Zone Number → DES, DESTINATION, TO, or TO\_ZONE  
 Time Period → PERIOD or INTERVAL

If the file includes data for only one time period the time period field is not necessary. If time periods are defined, the time period equivalence file is required to associate the time period codes in the skim file with time-of-day ranges in the activity schedule.

### **ZONE\_SKIM\_FORMAT\_#**

The zone skim format key enables the user to specify the input format for the zone skim file. The default file format is VERSION3. Other options include BINARY, FIXED\_COLUMN, COMMA\_DELIMITED, SPACE\_DELIMITED, TAB\_DELIMITED, DBASE, and SQLITE3. This key is only used if a Definition file is not found. It is primarily used to override the default Version 3 processing and have the program construct a Definition file based on the file header and field types.

### **TIME\_PERIOD\_EQUIVALENCE\_#**

The time period equivalence file is optional. It is required when time periods greater than one are included in the zone skim file. The time period equivalence key is appended to the PROJECT\_DIRECTORY key to specify the file name for a time period equivalence. This is an unformatted text file with one line of time ranges corresponding to each time period value. A sample time equivalence file for six time periods is show below:

```
0:00..6:00, 9:00..12:00
6:00..9:00
12:00..13:00
13:00..15:30, 18:30..21:00
15:30..18:30
21:00..24:00
```

### **SKIM\_MEMORY\_FACTOR\_#**

The skim memory factor key is optional. The default value 1.0, which means that the program will attempt to allocate sufficient memory for zones \* zones \* time periods skim data. If the matrix is sparse, memory allocation can be reduced by specifying a value less than 1.0. This parameter reduces the initial memory allocation, but does not restrict the ultimate size of the skim table. It is used primarily to improve the performance of the skim reading process.

### **TRAVEL\_TIME\_FIELD\_MODE\_#**

By default, the travel times for each trip are calculated using the DISTANCE\_CALCULATION and AVERAGE\_TRAVEL\_SPEED keys. These parameters estimate travel time for a given mode of travel based on the coordinate distance between the origin and destination activity locations. If zone-based skims are provided, the program can override these calculations based on the value of a field in one of the zone skim file. The program uses the zone associated with the origin and destination activity location and the start time of the trip to retrieve the appropriate data record from the skim file. The value of the specified field is assumed to be the travel time in seconds.



This key specifies the skim file and field name for a given travel mode. The syntax is

*Skim#.Field\_Name*

Where # corresponds to the number specified in the ZONE\_SKIM\_FILE\_# key and *Field\_Name* is the name of one of the fields in that file. For example, if ZONE\_SKIM\_FILE\_1 contains the highway travel time in a field named DRIVE, this key would be

Skim1.DRIVE

The # at the end of the key identified the travel mode where:

- 1 = Walk
- 2 = Drive
- 3 = Bus
- 4 = Rail
- 5 = Park-&-Ride Outbound
- 6 = Park-&-Ride Inbound
- 7 = Bicycle
- 8 = Magic Move
- 9 = School Bus
- 10 = 2 Person Carpool
- 11 = 3 Person Carpool
- 12 = 4 Person Carpool
- 13 = Kiss-&-Ride Outbound
- 14 = Kiss-&-Ride Inbound

For the example above, the TRAVEL\_TIME\_FIELD\_MODE\_2 would be used to set the travel time field for driving trips. Note if a field is not specified for a given mode, the default distance and speed calculations are used to estimate the travel time. In either case, the time specified by the ADDITIONAL\_TRAVEL\_TIME key is added to the calculated or skimmed travel time.

### **BALANCING\_FACTOR\_FILE**

The balancing factor file is appended to the PROJECT\_DIRECTORY key to specify the file name for the input balancing factors for each zone. The program uses the factor field specified in by the BALANCING\_FACTOR\_FIELD\_# key for a given purpose group to adjust the attraction weights for the zone in order to balance the trips attracted to the zone by trip purpose.

### **BALANCING\_FACTOR\_FORMAT**

The balancing factor form key enables the user to specify the input format for the balancing factor file. The format options include VERSION3, BINARY, FIXED\_COLUMN, COMMA\_DELIMITED, SPACE\_DELIMITED, TAB\_DELIMITED, DBASE, and SQLITE3.



**ACTGEN\_REPORT\_#**

Reports are optional. The “#” at the end of the report keyword represents the report number (e.g., ACTGEN\_REPORT\_1). The key can be provided with additional numbers to specify additional reports. The reports are generated in numerical order (i.e., 1, 2, 3...) for each purpose or zone skim group.

The string parameter associated with a report keyword is limited to the following options:

```
HOUSEHOLD_TYPE_SCRIPT
HOUSEHOLD_TYPE_STACK
HOUSEHOLD_TYPE_SUMMARY
SURVEY_TYPE_SCRIPT
SURVEY_TYPE_STACK
SURVEY_TYPE_SUMMARY
LOCATION_CHOICE_SCRIPT
LOCATION_CHOICE_STACK
LOCATION_CHOICE_DETAILS_*
TIME_PERIOD_EQUIVALENCE
HOUSEHOLD_MATCH_REPORT
TRIP_LENGTH_SUMMARY
TOUR_LENGTH_SUMMARY
TRIP_PURPOSE_SUMMARY
TOUR_PURPOSE_SUMMARY
MODE_LENGTH_SUMMARY
MODE_PURPOSE_SUMMARY
ACTIVITY_PROBLEM_REPORT
```

The above reports are printed in the “\*.prn” file that is generated in the same directory as the control file used to run the **ActGen** program. Each of above reports is described below:

Household Type Script

If a household type script is provided for the household file, this report lists the commands found in the script file. An example is shown below.

Household Type Script

```
IF (Household.Persons > 2) THEN
  IF (Household.Persons > 3) THEN
    RETURN (13)
  ELSE
    IF (Household.AgeLT5 == 1) THEN
      RETURN (12)
    ELSE
      IF (household.Age5to17 == 1) THEN
        RETURN (11)
      ELSE
        RETURN (10)
      ENDIF
    ENDIF
  ENDIF
```

```

ENDIF
ENDIF
ELSE
  IF (Household.Persons == 2) THEN
    IF (Household.AgeLT5 == 1) THEN
      RETURN (6)
    ELSE
      RETURN (5)
    ENDIF
  ELSE
    RETURN (3)
  ENDIF
ENDIF
ENDIF

```

### Household Type Stack

The program compiles the household type script into a command processing stack for execution. This report shows how the script was converted to processing commands. The stack commands corresponding to the script listed above are shown below.

#### Household Type Stack

1) Integer	Household.PERSONS
2) Integer	2
3) Relation	GT
4) Logical	If False, Jump to 29
5) Integer	Household.PERSONS
6) Integer	3
7) Relation	GT
8) Logical	If False, Jump to 12
9) Integer	13
10) Return	Integer
11) Logical	Jump to 28
12) Integer	Household.AGELT5
13) Integer	1
14) Relation	EQ
15) Logical	If False, Jump to 19
16) Integer	12
17) Return	Integer
18) Logical	Jump to 28
19) Integer	Household.AGE5TO17
20) Integer	1
21) Relation	EQ
22) Logical	If False, Jump to 26
23) Integer	11
24) Return	Integer
25) Logical	Jump to 28
26) Integer	10
27) Return	Integer
28) Logical	Jump to 45
29) Integer	Household.PERSONS
30) Integer	2
31) Relation	EQ
32) Logical	If False, Jump to 43
33) Integer	Household.AGELT5

```

34) Integer      1
35) Relation     EQ
36) Logical      If False, Jump to 40
37) Integer      6
38) Return       Integer
39) Logical      Jump to 42
40) Integer      5
41) Return       Integer
42) Logical      Jump to 45
43) Integer      3
44) Return       Integer
45) End

```

### Household Type Summary

This report summarizes the number of households assigned to each of the household types returned by the household type script. An example is shown below.

#### Household Type Summary

Type	Households	Households/Survey
3	10102	123.3
5	24555	289.0
10	145351	503.1
13	50889	424.0

### Survey Type Script

If a survey type script is provided for the survey household file, this report lists the commands found in the script file. This report is similar to the Household Type Script.

### Survey Type Stack

The program compiles the survey type script into a command processing stack for execution. This report shows how the script was converted to processing commands. This report is similar to the Household Type Stack.

### Survey Type Summary

This report summarizes the number of survey households assigned to each of the household types returned by the survey type script. An example is shown below.

#### Survey Type Summary

Type	Households	Cumulative Weight
3	82	1.100000
5	85	1.100000
10	289	1.500000
13	120	1.100000

### Location Choice Script

If a location choice script is provided for a given activity purpose group, this report lists the commands found in the script file. An example is shown below.

```
Location Choice Script #1

INTEGER distance, time ENDDEF

IF (Tour.DISTANCE1 == 0) THEN
  RETURN (0)
ENDIF

distance = Tour.DISTANCE1 + Tour.DISTANCE2

If (Tour.MODE == 1 && distance > 3000) THEN
  RETURN (0)
ENDIF

time = Skim1A.DRIVE + Skim1B.DRIVE

Tour.UTILITY = 0.05 * Location.USER1 *
               EXP (-0.0005 * time - 0.0001 * distance)
RETURN (1)
```

### Location Choice Stack

The program compiles each location choice script into a command processing stack for execution. This report shows how the script was converted to processing commands. This report is similar to the Household Type Stack.

### Location Choice Details \*

This report dumps the intermediate calculations during the location choice process. It is intended for debugging the location choice scripts, attraction weights, and travel time skims used in the process. The \* at the end of the report name refers to a specific household ID. Hundreds of lines of output may be generated for each household selected. If the \* is set to zero, all households in the household list are dumped. This should only be done for a few households. Since any number of these reports can be specified, the user can dump several select households by providing multiple instances of this report with different household IDs.

### Time Period Equivalence

If a zone skim file includes a time period equivalence file, this report lists the time period range records. An example is shown below.

```
Time Period Equivalence

Period 1 = 0:00..6:00, 9:00..12:00
Period 2 = 6:00..9:00
Period 3 = 12:00..13:00
Period 4 = 13:00..15:30, 18:30..21:00
Period 5 = 15:30..18:30
```

Period 6 = 21:00..24:00

### Household Match Report

The program matches a synthetic household type to a survey household type. This report shows the results of this match for the household included in the household list. In general, this report should not be selected if all of the households in the region are being processed.

Household Match Report												
----- Household -----							----- Survey -----					
Type	Household	Person	Age	Group	Worker	Gender	Household	Person	Age	Group	Worker	Gender
3	1	1	29	5	1	2	200107	1	74	6	2	2
3	2	1	82	6	2	2	200107	1	74	6	2	2
5	3	1	25	5	1	1	200060	2	63	5	2	1
5	3	2	28	5	2	2	200060	1	31	5	2	2
3	4	1	82	6	2	1	200102	1	66	6	1	2
3	5	1	77	6	2	1	200107	1	74	6	2	2
10	6	1	66	6	1	2	200029	1	66	6	1	2
10	6	2	66	6	1	1	200029	2	66	6	1	1
10	6	3	45	5	1	1	200029	3	45	5	1	1
3	7	1	54	5	1	1	200107	1	74	6	2	2
13	8	1	40	5	1	1	200058	2	46	5	1	1
13	8	2	3	1	2	2	200058	4	3	1	2	2
13	8	3	7	2	2	2	200058	3	6	2	2	2
13	8	4	44	5	1	2	200058	2	46	5	1	1
11	9	1	25	5	1	2	200045	1	25	5	1	2
11	9	2	15	3	2	1	200045	2	15	3	2	1
11	9	3	39	5	1	1	200045	3	39	5	1	1
5	10	1	52	5	1	2	200060	1	31	5	2	2
5	10	2	51	5	1	1	200060	2	63	5	2	1
13	11	1	42	5	1	2	200067	1	41	5	1	2
13	11	2	17	4	1	1	200067	2	42	5	1	1

### Trip Length Summary

The program generates household tours based on activity patterns found in the household survey. This report summarizes the number, trip length, and travel time of each travel leg (i.e., trip) within a tour by trip purpose associated with the activity.

#### Trip Length Summary

Purpose	Trips	----- Distance (meters) -----				----- Time (minutes) -----			
		Minimum	Maximum	Average	StdDev	Minimum	Maximum	Average	StdDev
1	34194	0	11646	779	705	0.00	34.75	0.62	1.25
2	20384	0	8522	809	787	0.00	28.40	0.74	1.71
3	12428	0	9906	833	864	0.00	30.82	0.80	1.85
4	18917	0	8808	674	772	0.00	31.70	0.72	1.77
5	31097	0	8596	482	656	0.00	28.18	0.58	1.06
6	27512	0	11520	852	968	0.00	34.03	0.87	2.36
7	11187	0	7449	678	599	0.00	17.02	0.68	0.68
8	6567	0	9060	4178	2498	0.00	34.42	10.70	6.86
9	46678	0	8836	677	539	0.00	26.37	0.53	0.46
11	5480	0	8400	811	657	0.50	18.82	0.57	0.96
12	30048	0	10081	791	561	0.00	17.95	0.52	0.42
13	13850	0	8115	804	637	0.00	23.82	0.57	0.80
14	34008	0	10361	713	544	0.00	18.87	0.52	0.31
15	37277	0	8578	723	585	0.00	32.70	0.54	0.69

16	24408	0	8831	791	701	0.00	34.42	0.60	1.16
17	4174	0	4276	718	507	0.47	6.43	0.50	0.11
18	7211	0	6181	671	535	0.00	12.28	0.59	0.47
19	369	0	2789	725	428	0.50	1.35	0.50	0.04
Total	365789	0	11646	791	880	0.00	34.75	0.79	2.00

### Tour Length Summary

The program generates household tours based on activity patterns found in the household survey. This report summarizes the number, length, and travel time of each tour by trip purpose associated with the primary or anchor activity in the tour. A tour is defined as a sequence of trips that starts and ends at home.

#### Tour Length Summary

Purpose	Tours	Distance (meters)				Time (minutes)			
		Minimum	Maximum	Average	StdDev	Minimum	Maximum	Average	StdDev
1	24455	0	32875	2203	2328	0.00	69.52	1.94	4.25
2	671	60	12329	1826	1240	0.50	15.85	1.29	0.82
3	1218	30	12706	2221	1557	0.50	3.03	1.62	0.55
4	3512	30	12570	1839	1218	0.50	35.68	1.50	2.47
5	1256	0	10979	1837	1246	0.50	9.27	1.24	0.62
6	220	266	7296	1730	1033	0.50	8.65	1.11	0.59
7	8779	0	15665	1253	1153	0.00	31.47	1.29	1.24
8	5196	0	36355	6840	4109	0.00	77.35	16.48	10.36
9	34790	0	23659	1534	1477	0.00	70.85	1.34	1.99
12	7499	30	24422	1586	1404	0.05	27.90	1.13	1.08
13	4096	0	25486	1646	1379	0.05	11.37	1.23	0.83
14	19786	0	22680	1728	1355	0.00	38.62	1.50	2.50
15	12986	0	26107	2837	2231	0.00	82.78	2.35	3.48
16	295	30	13021	2059	1565	0.50	13.72	1.32	0.94
18	5623	0	15823	1634	1345	0.00	28.57	1.52	1.53
Total	130382	0	36355	2043	2148	0.00	82.78	2.17	4.47

### Trip Purpose Summary

The program generates household tours based on activity patterns found in the household survey. This report summarizes the number, trip length, and travel time of each travel leg (i.e., trip) within a tour by the trip purpose associated with the origin and destination activities.

#### Trip Purpose Summary

Purpose	Trips	Distance (meters)				Time (minutes)			
		Minimum	Maximum	Average	StdDev	Minimum	Maximum	Average	StdDev
0-0	822	0	0	0	0	0.00	0.00	0.00	0.00
0-1	348	333	3887	1895	881	0.37	33.30	4.08	5.78
0-2	80	333	2975	1159	538	0.37	3.30	1.28	0.60
0-3	15	0	3606	2372	933	0.00	4.00	2.63	1.04
0-4	58	333	2404	1054	523	0.37	27.78	8.71	8.46
0-5	35	333	3565	1475	841	0.37	3.95	1.63	0.94
0-6	70	335	3606	1802	878	0.37	4.00	2.00	0.98
0-7	71	333	3771	2027	977	0.55	6.28	2.89	1.60
1-0	309	333	3887	1909	891	0.37	33.30	4.37	6.07
1-1	33	0	3300	1010	1056	0.00	3.67	1.12	1.17
1-2	28	30	2731	1329	736	0.03	3.03	1.47	0.82
1-3	30	725	3606	1996	894	0.80	4.00	2.21	0.99

Tour Purpose Summary

The program generates household tours based on activity patterns found in the household survey. This report summarizes the number, length, and travel time of each tour by trip purpose associated with the primary or anchor activity in the tour and the number of stops on the tour. A tour is defined as a sequence of trips that starts and ends at home.

## Tour Purpose Summary

Purpose -Stops	Tours	Distance (meters)				Time (minutes)			
		Minimum	Maximum	Average	StdDev	Minimum	Maximum	Average	StdDev
1-1	148	450	3793	1836	877	0.50	33.30	6.46	8.19
1-2	85	335	6307	2555	1489	0.52	28.03	5.61	5.91
1-3	127	472	7955	2965	1796	0.52	49.12	6.55	9.06
1-4	16	5033	10219	6979	1563	5.58	11.33	7.73	1.74
1-5	38	1900	16574	7117	3319	2.10	18.38	7.88	3.69
1-7	5	6927	12987	9860	2376	31.42	57.10	49.60	10.40
2-1	80	333	4484	1648	1150	0.37	4.97	1.82	1.28
3-2	19	1238	6211	3177	1200	1.37	6.90	3.52	1.34
4-1	18	335	2370	1101	555	0.37	2.63	1.22	0.62
5-1	12	333	3565	1360	914	0.37	3.95	1.50	1.02
7-1	38	333	3771	2108	981	0.55	6.28	2.81	1.50
7-2	9	333	3606	1991	1288	0.55	6.00	3.31	2.15
7-3	24	667	3565	1911	869	1.10	5.93	2.87	1.58
7-4	26	755	5301	2414	1035	2.12	40.32	15.64	13.52
7-7	18	3372	7181	4952	1143	33.63	51.47	44.70	6.34
Total	663	333	16574	2747	2174	0.37	57.10	6.95	10.36

Mode Length Summary

The program generates household tours based on activity patterns and modes found in the household survey. This report summarizes the number, trip length, and travel time of each travel leg (i.e., trip) within a tour by travel mode to the associated the activity.

## Mode Length Summary

Mode	Trips	Distance (meters)				Time (minutes)			
		Minimum	Maximum	Average	StdDev	Minimum	Maximum	Average	StdDev
1	34194	0	11646	779	705	0.00	34.75	0.62	1.25
2	20384	0	8522	809	787	0.00	28.40	0.74	1.71
3	12428	0	9906	833	864	0.00	30.82	0.80	1.85
4	18917	0	8808	674	772	0.00	31.70	0.72	1.77
5	31097	0	8596	482	656	0.00	28.18	0.58	1.06
6	27512	0	11520	852	968	0.00	34.03	0.87	2.36
7	11187	0	7449	678	599	0.00	17.02	0.68	0.68
8	6567	0	9060	4178	2498	0.00	34.42	10.70	6.86
9	46678	0	8836	677	539	0.00	26.37	0.53	0.46
11	5480	0	8400	811	657	0.50	18.82	0.57	0.96
Total	130382	0	36355	2043	2148	0.00	82.78	2.17	4.47

### Mode Purpose Summary

The program generates household tours based on activity patterns and modes found in the household survey. This report summarizes the number, length, and travel time of each tour leg (i.e., trip) by mode and trip purpose associated with the activity.

#### Mode Purpose Summary

Mode- Purpose	Trips	Distance (meters)				Time (minutes)			
		Minimum	Maximum	Average	StdDev	Minimum	Maximum	Average	StdDev
1-1	148	450	3793	1836	877	0.50	33.30	6.46	8.19
1-2	85	335	6307	2555	1489	0.52	28.03	5.61	5.91
1-3	127	472	7955	2965	1796	0.52	49.12	6.55	9.06
1-4	16	5033	10219	6979	1563	5.58	11.33	7.73	1.74
1-5	38	1900	16574	7117	3319	2.10	18.38	7.88	3.69
1-7	5	6927	12987	9860	2376	31.42	57.10	49.60	10.40
2-1	80	333	4484	1648	1150	0.37	4.97	1.82	1.28
3-2	19	1238	6211	3177	1200	1.37	6.90	3.52	1.34
4-1	18	335	2370	1101	555	0.37	2.63	1.22	0.62
5-1	12	333	3565	1360	914	0.37	3.95	1.50	1.02
7-1	38	333	3771	2108	981	0.55	6.28	2.81	1.50
7-2	9	333	3606	1991	1288	0.55	6.00	3.31	2.15
7-3	24	667	3565	1911	869	1.10	5.93	2.87	1.58
7-4	26	755	5301	2414	1035	2.12	40.32	15.64	13.52
7-7	18	3372	7181	4952	1143	33.63	51.47	44.70	6.34
Total	663	333	16574	2747	2174	0.37	57.10	6.95	10.36

### Algorithm Notes

The **ActGen** program performs activity generation and location choice through a multi-step process. The process begins by reading and compiling all of the script files. This includes checking that the requested files and fields are defined by the specified input files. The program then reads the survey household, population, and activity files into memory and applies the survey type script and survey weights. It is not until all of these data are validated that the program reads the network, household, and vehicle data. The household type script is applied to the household data records and the vehicles are linked to the households.

This same functionality is duplicated by two other TRANSIMS programs. **ActivityPattern** implements and edits the household matching and activity pattern generation step. The **LocationChoice** program can then select or update the activity locations and generate the complete Activity file.

### Household Type and Survey Type Scripts

The household and survey type scripts apply a series of logical statements to data fields in the household file to return an integer type value. If the fields in the synthetic and survey household files are consistent, the same script can be used for both applications. The script references the fields in either household file using the syntax:

Household.*Field\_Name*



Where “Household” represents the file name and “*Field\_Name*” is one of the user-defined fields in the household file. In Linux applications, the case of the field and file names is important. The sample script shown below assigns type 13 to households with more than three persons and type 10 to households with three persons and no children less than 18 years of age.

```

IF (Household.Persons > 2) THEN
  IF (Household.Persons > 3) THEN
    RETURN (13)
  ELSE
    IF (Household.AgeLT5 == 1) THEN
      RETURN (12)
    ELSE
      IF (household.Age5to17 == 1) THEN
        RETURN (11)
      ELSE
        RETURN (10)
      ENDIF
    ENDIF
  ENDIF
ELSE
  IF (Household.Persons == 2) THEN
    IF (Household.AgeLT5 == 1) THEN
      RETURN (6)
    ELSE
      RETURN (5)
    ENDIF
  ELSE
    RETURN (3)
  ENDIF
ENDIF

```

## Zone Skims

If zone skim files are provided, these files are read into memory at this point. Each record in the skim file is indexed by the origin zone number, destination zone number, and time period. The combination of these three values must fit within an integer key. This creates a tradeoff between the zone numbers and the number of time periods. The following table shows the maximum combinations of zone numbers and time periods that can be processed.

Time Periods	Zone Numbers
1	32767
7	16383
31	8191
127	4095
255	2047

For example, if the zone skim file includes a travel time estimate for every 15-minutes within a day (i.e., 96 time periods), the origin and destination zone numbers must be less than or equal to 4095.

## Matching Household Members

The activity generation process is then applied to each household one at a time. Given the household type, the program randomly selects a household from the survey dataset with the same type. It then gathers the members of the survey household for the person matching process. The program matches each person in the synthetic household to the most appropriate person in the survey household. This is done using age, work status, and gender. Each person is assigned to an age group. Age groups are defined as follows:

Age Group	Age Range
1	0 – 4
2	5 – 11
3	12 – 15
4	16 – 20
5	21 – 64
6	65+

These grouping roughly correspond to school types and stages of adulthood.

The order of matching household persons is first age group, then work status, and then gender. If more than one household member matches a given combination of age group, work status and gender, the program selects the first “unselected” household member. If this fails to find a reasonable match, the program expands the age match and tries again. The expanded age group logic is shown in the following table.

Original Group	Expanded Match
1	1 or 2
2	2 or 3
3	2 or 3
4 and non-work	3 or 4
4 and worker	4 or 5
5	5 or 6
6	5 or 6

The program then copies the activity patterns, activity times, and travel modes from the survey and assigns it to the corresponding household person and all at-home activities are collapsed and located at the household activity location.

The program then assigns household vehicles to drivers and passengers within the household. Each vehicle is scheduled based on the time of day it is away from home. If the household does not own an adequate number of vehicles to satisfy the number of simultaneous activities that required a vehicle, a problem message is generated.

## Activity Location

All non-home activities then need to be located. This is done based on the utility of the travel to and from a given activity location by time of day and the attraction weight of the activity location. The location is selected using a random number and a cumulative probability distribution. This location is then assigned to all other persons and activities that share the same household location code in the survey activity file.

The probability of selecting a given activity location is based the activity schedule and the previous and next anchor locations. Initially the only anchor location is the home location. The program first scans all of the activities on tours that start and end at home to select the best anchor location for the tour. If the activity purpose is identified as an anchor location by the purpose group, it is considered first. If the tour has multiple anchor activities, the program will select the activity with the longest duration. If the tour does not include anchor activities, it will use the activity with the longest duration as the anchor for the tour.

Once the anchor is identified, the program uses the home location as the previous and next activity location for locating the anchor activity. The time the trip leaves home is used to select the travel conditions for the outbound trip and the time when the trip leaves the anchor activity is used to select the travel conditions for the return trip. In other words, the travel conditions for the round trip to and from each activity location are used in the location choice process.

Once the anchor activity is located, the process locates the activities between home and the anchor activity based on the distance to home and the distance to the anchor location. After the first intermediate stop is located, it becomes the anchor location for locating the next stop. The process continues until all of the activities are located.

## Location Choice Model

The **ActGen** program supports six types of location choice models. The model that is implemented for any given trip purpose is defined by the combination of keys that are specified in the purpose group. Different models can be used for different trip purposes. The first model distinction is between zone-based and location-based models. A location-based model selects an activity location based on the attractiveness and travel characteristics to all activity locations. A zone-based model first assigns an activity to a traffic analysis zone and then selects an activity location within that zone. A network zone file must be provided in order to implement any of the zone-based models. This file must include the X and Y coordinates of the zone centroid and one or more zone attraction fields. The selection of an activity location within a given zone is based on the attraction weight of the activity location times the sum of the coordinate distance to each anchor location.

The next distinction for location choice models is if it is or is not script-based. Any purpose group that specified a location choice script is script-based. This means that all of the utility calculations are controlled by the modeler within a modeling script or program subroutine. If a script is not provided, the default location choice model is applied. These two model types are described in more detail in the following sections.

The third distinction is made for default location choice models. They can be either distance-based or time-based. Distance-based models calculate the travel characteristics between activity locations based on X and Y coordinates. Time-based models use data from a zone-to-zone skim file to estimate the travel time or impedance between two zones by time of day and mode.

The following table outlines the control keys that are used to determine which model type is used for a given purpose group.

ActGen Keys	Location Choice Methods					
	Default-Model				Script-based	
	Distance-based		Time-based			
	Zone based	Location based	Zone based	Location based	Zone based	Location based
ZONE_BASED_METHOD_#	Yes	No	Yes	No	Yes	No
NET_ZONE_TABLE	X		X		X	
ZONE_SKIM_FILE_#			X	X	Option	Option
LOCATION_CHOICE_SCRIPT_#					X	X
ZONE_WEIGHT_FIELD_#	X		X		Option	
ZONE_WEIGHT_FACTOR_#	X		X		Option	
BALANCING_FACTOR_FIELD_#	X		X		Option	
LOCATION_WEIGHT_FIELD_#	X	X	X	X	X	Option
LOCATION_WEIGHT_FACTOR_#		X		X		Option
SKIM_TIME_FIELD_#			X	X		
MODE_TIME_FACTORS_#			X	X		
DISTANCE_CALCULATION	X	X	X		X	
MODE_DISTANCE_FACTORS_#	X	X				

### Default Choice Model

The default location choice model uses the distances or times between the anchor locations and each activity location or zone to calculate the probability that the activity location or zone will be selected. The alternate approach uses a user program script to calculate each location's utility based on tour, household, and zone skim data.

If the default model is used, the travel calculation method, the purpose group weights, and the mode specified in the survey are important. The probability of selecting a given destination for a given purpose is based on the relative utility of each destination to the sum of the utilities to all locations:

$$\text{Probability}_n = \text{Utility}_n / \sum \text{Utility}_i$$

Where utility is defined as follows:

$$\text{Utility}_n = \text{Attraction\_Weight}_n * \text{Weight\_Factor} * \text{Balancing\_Factor} * \\ \text{exponential} ( \text{Mode\_Factor}_{\text{mode}} * (\text{Travel}_{\text{in}} + \text{Travel}_{\text{nj}}))$$

Where “n” is the destination, “i” is the previous activity location, and “j” is the next activity location. For zone-based models Attraction Weight is defined by the zone weight field and Weight Factor is defined by the zone weight factor. For location-based models Attraction Weight is the location weight field and the Weight Factor is the location weight factor. For time-based models, Mode Factor for a given mode is defined by mode time factors and Travel is defined by the skim time field. For distance-based models, Mode Factor is defined by the mode distance factors and Travel is the calculated distance in meters.

For distance-based models and the distribution step of zone-based models, the distance between the anchor locations and each potential activity location is defined by the Distance Calculation key. The options include the straight line distance, the right angle distance, or some combination of the two.

## Location Choice Scripts

Location choice scripts enable the modeler to define models that are significantly more complex than the default location choice model. The script is provided with data from three or more files to calculate the utility of selecting a given activity location or zone. The first file is called a Tour file. This file includes basic information about the current location relative to the overall tour. It is also where the modeler stores the calculations for the target location. The fields included in the Tour file are listed below:

START_TIME	The time when the trip to the activity begins (seconds)
END_TIME	The time when the trip from the activity begins (seconds)
DURATION	The difference between the activity end and start times (seconds)
PURPOSE	The purpose of the activity
MODE	The mode from the survey or the user-defined tour mode
STOPS1	The number of stops between the previous anchor and this activity
STOPS2	The number of stops between this activity and the next anchor
SUBTOURS	The number of subtour activities while at this location
DISTANCE1	The distance between the previous anchor and this location
DISTANCE2	The distance between this location and the next anchor (meters)
BUDGET1	The survey travel time from the previous anchor (seconds)
BUDGET2	The survey travel time to the next anchor (seconds)
UTILITY	The utility or probability that this locations is selected
MIN_FAC1	The minimum distance ratio from the origin to the target zone
MAX_FAC2	The maximum distance ratio from the origin to the target zone
MIN_FAC2	The minimum distance ratio from the target zone to the destination
MAX_FAC2	The maximum distance ratio from the target zone to the destination

The program expects the modeler to return the UTILITY field and potentially the MODE field. The MODE field is only needed when the modeler wishes to change the tour mode. If the tour mode limits the mode options available to the traveler between the two anchor locations, the mode specified for these activities will be changed to the tour mode. The modeler can also use the RETURN function in the script to remove a given location from further consideration. This has the same affect as setting the UTILITY field to zero.

The model script also has access to all of the data fields in the location and household files. The household fields are accessible using the variable “Household.*Field\_Name*”. The location fields depend on the model type. If the model is a location-based model, fields from the activity location files are accessible using the variable “Location.*Field\_Name*”. If the model is a zone-based model, fields from the zone file are accessible using the variable “Zone.*Field\_Name*”.

If the user provided a zone weight field for the trip purpose, the program reads the field from the zone file. If the field value is less than or equal to zero, the zone is skipped. If it is greater than zero, the value is stored in the Tour.UTILITY field and passed to the model script. The script can use the field in utility calculations such as:

$$\text{Tour.UTILITY} = \text{Tour.UTILITY} * \text{EXP} (-0.01 * (\text{time1} + \text{time2}))$$

or it can be ignored and replaced by accessing other fields in the zone file.

If the user provided zone skim files, the appropriate records from these files are sent to the location choice script. In this case, two records are provided for each zone skim file. The first record corresponds to the trip between the previous anchor and the target location at the start time of the trip. The second record corresponds to the trip from the target location to the next anchor at the departure time of the trip. The data fields in each record can be accessed using the following convention:

$$\text{Zone\_Skim\_File\_}\# \rightarrow \text{Skim}\#\text{A.}\textit{Field\_Name} \text{ and } \text{Skim}\#\text{B.}\textit{Field\_Name}$$

Where “#” is the zone skim group number (e.g., Skim1A.Time).

The zone number included in the activity location file for the previous anchor location is used as the origin of skim A. The zone number of the next anchor is the destination of skim B. The zone number of the target location is the destination of skim A and the origin of skim B. The time period used for skim A is based on the departure time from the previous anchor activity. The time period used for skim B is based on the departure time from the target activity.

A sample script is show below.

```

INTEGER distance, time ENDDF

#---- force a new activity location ----

IF (Tour.DISTANCE1 == 0) THEN
  RETURN (0)
ENDIF

distance = Tour.DISTANCE1 + Tour.DISTANCE2

#---- limit walk trips ----

IF (Tour.MODE == 1 && distance > 3000) THEN
  RETURN (0)
ENDIF

#---- round trip travel time ----

time = Skim1A.TIME + Skim1B.TIME

Tour.UTILITY = 0.05 * Location.WORK *
              EXP (-0.0005 * time - 0.0001 * distance)
RETURN (1)

```

Refer to the User Program documentation for more details about script syntax.

## Travel Time Estimates

After all of the activities are located, the program estimates the travel time between each location based on the distance between locations, the travel mode, and the travel speed and additional travel time by mode. The default calculation is as follows:

$$\text{Travel Time} = \text{Distance} / \text{Average\_Travel\_Speed}_{\text{mode}} + \text{Additional\_Travel\_Time}_{\text{mode}}$$

where the distance is based on the method specified by the Distance Calculation key. If a Travel Time Field Mode key is provided for the mode number, the skim record corresponding to the zones where the origin and destination activity locations are located and the trip start time are used to override the travel time estimate. In this case the travel time is calculated as:

$$\text{Travel Time} = \text{Skim}_{\text{mode}}(\text{Org Zone}, \text{Des Zone}, \text{Start Time}).\text{Field}_{\text{mode}} + \text{Additional\_Travel\_Time}_{\text{mode}}$$

In either case the travel time estimate is further constrained by the available time between the end of the previous activity and the start of the next activity. If the schedule time is less than the travel time, the additional travel time is reduced to better accommodate the activity schedule.

## Scheduling Activities

The last step in the activity generation process is to adjust the activity schedules based on the relative proximity of the activities and the travel time. This process identifies each tour and the anchor activity within that tour. It then uses the travel times to and from the anchor activity to estimate the departure time from the previous activity or the arrival time at the next activity.

If the departure time from the previous activity is different from the scheduled end time of that activity, the program attempts to adjust the activity schedule. The types of adjustments that are permitted are controlled by the Schedule Constraints parameter assigned to the purpose group. If the purpose is unconstrained (i.e., NO\_CONSTRAINT), the end time of the activity will be rescheduled to the departure time and the activity duration will be used to set the activity start time.

If the activity has a fixed time constraint, the program will work within this constraint to adjust the activity, nearby activities, and the travel time estimate to maintain as much consistency as possible. If a simple solution is not possible, the program will proportionally distribute the error to the affected travel times and activity durations. This process continues until all activities are adjusted before and after the anchor activity location. Note that Home activities are always considered to be unconstrained.

If the tour includes shared rides between household members, the travel portion of a shared activity is synchronized. This includes the departure time of the previous activity and the arrival time of the shared activity. If the purpose of the driver is “serve passenger”, the schedule constraint for synchronizing the activities will be controlled by the passenger.

If the total time for the previous tour ends after the next tour is scheduled to leave home, each activity that was scheduled to take place before the end of the previous tour is flag as a problem and deleted from the output activity file.

## Sample Printout

A sample printout file generated by the **ActGen** program is shown 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.

```

*****
|
|           ActGen - Version 4.0.36
|   Copyright (c) 2009 by AECOM Consult
|           Mon Apr 19 13:11:46 2010
|
*****

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

Activity Generator

Network Directory = d:\test\actgen\network

```



```

Node File = d:\test\actgen\network\Node.txt
Zone File = d:\test\actgen\network\Zone.txt
Link File = d:\test\actgen\network\Link.txt
Parking File = d:\test\actgen\network\Parking.txt
Activity Location File = d:\test\actgen\network\Activity_Location.txt
Process Link File = d:\test\actgen\network\Process_Link.txt

Household File = d:\test\actgen\demand\Household.txt

Population File = d:\test\actgen\demand\Population.txt

Vehicle Type File = d:\test\actgen\demand\Vehicle_type.txt

Vehicle File = d:\test\actgen\demand\Vehicle.txt
Vehicle File will be Sorted by Vehicle ID

New Activity File = d:\test\actgen\demand\Activity.txt
New Activity File Format = TAB_DELIMITED

Time of Day Format = 24_HOUR_CLOCK

Household Type Script = d:\test\actgen\demand\Household_Type.txt

Survey Household File = d:\test\actgen\survey\Household.txt
Survey Household Weights = d:\test\actgen\survey\Survey_Weights.txt
Survey Type Script = d:\test\actgen\survey\Survey_Type.txt

Survey Population File = d:\test\actgen\survey\Population.txt

Survey Activity File = d:\test\actgen\survey\Activity.txt
Survey Activity File Format = TAB_DELIMITED

Distance Calculation = STRAIGHT_LINE
Average Travel Speed = 1.0, 15.0, 10.0 ... meters/second

Random Number Seed = 1234

Activity Purpose Range #1 = 1, 3, 7
Activity Anchor Flag #1 = True
Schedule Constraint #1 = NONE
Location Choice Script #1 = d:\test\actgen\demand\Zone_Anchor_Model.txt (Zone-Based)
Location Weight Field Name = USER1, Number = 13

Activity Purpose Range #3 = 2, 4..5
Activity Anchor Flag #3 = False
Schedule Constraint #3 = NONE
Location Choice Script #3 = d:\test\actgen\demand\Other_Model.txt (Location-Based)

Zone Skim File #1 = d:\test\actgen\demand\Skim.txt
Zone Skim File #1 Format = TAB_DELIMITED
Time Period Equivalence #1 = d:\test\actgen\demand\Period.txt

ActGen Reports: 3. HOUSEHOLD_TYPE_SUMMARY
                6. SURVEY_TYPE_SUMMARY
                7. LOCATION_CHOICE_SCRIPT

```