PopSyn *(version 4.0)*

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

The **PopSyn** program synthesizes households, persons and vehicles, and assigns the households to activity locations and the vehicles to parking lots. The Public Use Microdata Samples (PUMS) data from the U.S. Census are used to define the household characteristics. Forecast distributions of household attributes for zones or block groups are used to select household types allocated to geographic areas within the region. The **PopSyn** program performs the following tasks:

1. Extract the household attribute classifications for each PUMA (Public Use Microdata Area);
2. Construct the PUMA cross-classification table for the selected household attributes;
3. Retrieve the zone marginal totals for each identified attribute;
4. Aggregate the marginal totals for all zones within a PUMA;
5. Apply an IPF process to adjust the PUMA cross-classification table based on the aggregate marginal totals;
6. Apply a second IPF process to estimate the cross-classification table for each zone using zone marginal totals and the adjusted PUMA cross-classification table as an additional marginal constraint;
7. Randomly select a corresponding PUMS household for each household in the zone cross-classification table;
8. Randomly assign the household to an activity location within the zone using the specified location weights; and
9. Randomly select a vehicle type for each household vehicle and assign the vehicle to the parking lot attached to the household's activity location.

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

PopSyn [-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 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 **PopSyn** control file is shown below:

```
TITLE                           Population Synthesizer Test
NET_DIRECTORY                   ..\network
NET_ACTIVITY_LOCATION_TABLE     Activity_Location.txt
NET_PROCESS_LINK_TABLE          Process_Link.txt

PUMS_HOUSEHOLD_FILE             ..\pums\Household.txt
PUMS_POPULATION_FILE            ..\pums\Population.txt
ZONE_DATA_FILE                  ..\stf3\Zone_Data.txt
VEHICLE_TYPE_DISTRIBUTION       ..\vehicle\Type_Distribution.txt

NEW_HOUSEHOLD_FILE              ..\population\Household.txt
NEW_POPULATION_FILE             ..\population\Population.txt
NEW_VEHICLE_FILE                ..\vehicle\Vehicle.txt
NEW_PROBLEM_FILE                ..\results\Problem.txt

RANDOM_NUMBER_SEED              12332
MAXIMUM_IPF_ITERATIONS          10000
MAXIMUM_IPF_DIFFERENCE          0.0000001

STATE_PUMA_LIST                 OR1200, OR1300, OR1400, OR1500, WA1901, WA1902
PUMS_WEIGHT_FIELD               WEIGHT
PUMS_VEHICLE_FIELD              AUTOS
ZONE_DATA_ID_FIELD              ZONE
LOCATION_ZONE_FIELD             TAZ

ZONE_TOTAL_FIELD_1              TOTAL
LOCATION_WEIGHT_FIELD_1         USER1

PUMS_ATTRIBUTE_FIELD_1_1        HHSIZE
PUMS_ATTRIBUTE_BREAKS_1_1       1, 2
ZONE_FIELD_GROUP_1_1            HHSIZE

PUMS_ATTRIBUTE_FIELD_1_2        AUTOS
PUMS_ATTRIBUTE_BREAKS_1_2       1
ZONE_FIELD_GROUP_1_2            AUTOS

STARTING_HOUSEHOLD_ID           100
STARTING_VEHICLE_ID             100
```

```
OUTPUT_HOUSEHOLD_FIELDS          STATE, PUMA, WEIGHT, HHSIZE, AUTOS
OUTPUT_POPULATION_FIELDS         AGE, GENDER

POPSYN_REPORT_1                  PUMA_CROSS_CLASSIFICTION
POPSYN_REPORT_2                  PUMS_HOUSEHOLD_SUMMARY
POPSYN_REPORT_3                  PUMS_POPULATION_SUMMARY
```

The keys recognized by the **PopSyn** 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

This key 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

This 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

This key is optional. 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

This key is optional. 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_ACTIVITY_LOCATION_TABLE

This 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 **PopSyn** process. At a minimum, it needs a zone field that links activity locations to the zones defined in the zone data file (e.g., a traffic analysis zone or a block group). It also can include fields for weighting the allocation of households to activity locations within the zone. These fields are typically used to influence the distribution of single family, multi-family, and group quarter households within the zone.

NET_PROCESS_LINK_TABLE

This 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.

DEMAND_FILE_FORMAT

This 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, DBASE, COMMA_DELIMITED, SPACE_DELIMITED, TAB_DELIMITED, and FIXED_COLUMN, and SQLITE3.

PUMS_HOUSEHOLD_FILE

This key value is appended to the PROJECT_DIRECTORY key to specify the file name for the input PUMS household file. This file includes household characteristics from the U.S. Census Public Use Microdata Samples or similar household survey file.

PUMS_HOUSEHOLD_FORMAT

This key enables the user to specify the input format for the PUMS household file. The format options include VERSION3, BINARY, FIXED_COLUMN, COMMA_DELIMITED, SPACE_DELIMITED, TAB_DELIMITED, DBASE, and SQLITE3.

PUMS_POPULATION_FILE

This key value is appended to the PROJECT_DIRECTORY key to specify the file name for the input PUMS population file. This file includes the person characteristics for each household included in the PUMS household file.

PUMS_POPULATION_FORMAT

This key enables the user to specify the input format for the PUMS population file. The format options include VERSION3, BINARY, FIXED_COLUMN, COMMA_DELIMITED, SPACE_DELIMITED, TAB_DELIMITED, DBASE, and SQLITE3.

ZONE_DATA_FILE

This key value is appended to the PROJECT_DIRECTORY key to specify the file name for the input zone data file. This file includes a forecast of the distribution of household attributes for zones associated with Public Use Microdata Areas (PUMAs).

ZONE_DATA_FORMAT

This key enables the user to specify the input format for the zone data file. The format options include VERSION3, BINARY, FIXED_COLUMN, COMMA_DELIMITED, SPACE_DELIMITED, TAB_DELIMITED, DBASE, and SQLITE3.

VEHICLE_TYPE_DISTRIBUTION

This key is optional. If provided, the key value is appended to the PROJECT_DIRECTORY key to specify the file name for the input vehicle type distribution. The vehicle type distribution is used to specify the probability of various Vehicle Type and Subtype values on the vehicle file.

HOUSEHOLD_FILE

An input household file is optional. This key value is appended to the PROJECT_DIRECTORY key to specify the file name for the input household file. These records are copied to the output household file at the beginning of the process.

HOUSEHOLD_FORMAT

This key enables the user to specify the format for the input household file. By default the file format is specified by the *.def file. The format options include VERSION3, BINARY, FIXED_COLUMN, COMMA_DELIMITED, SPACE_DELIMITED, TAB_DELIMITED, DBASE, and SQLITE3.

POPULATION_FILE

An input population file is optional. This key value is appended to the PROJECT_DIRECTORY key to specify the file name for the input population file. These records are copied to the output population file at the beginning of the process.

POPULATION_FORMAT

This key enables the user to specify the format for the input population file. By default the file format is specified by the *.def file. The format options include VERSION3, BINARY, FIXED_COLUMN, COMMA_DELIMITED, SPACE_DELIMITED, TAB_DELIMITED, DBASE, and SQLITE3.

VEHICLE_FILE

An input vehicle file is optional. This key value is appended to the PROJECT_DIRECTORY key to specify the file name for the input vehicle file. These records are copied to the output vehicle file at the beginning of the process.

VEHICLE_FORMAT

This key enables the user to specify the format for the input vehicle file. By default the file format is specified by the *.def file. The format options include VERSION3, BINARY, FIXED_COLUMN, COMMA_DELIMITED, SPACE_DELIMITED, TAB_DELIMITED, DBASE, and SQLITE3.

NEW_HOUSEHOLD_FILE

This key value is appended to the PROJECT_DIRECTORY key to specify the file name for the output household file.

NEW_HOUSEHOLD_FORMAT

This key enables the user to specify the output 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.

NEW_POPULATION_FILE

This key value is appended to the PROJECT_DIRECTORY key to specify the file name for the output population file.

NEW_POPULATION_FORMAT

This key enables the user to specify the output 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.

NEW_VEHICLE_FILE

This key value is appended to the PROJECT_DIRECTORY key to specify the file name for the output vehicle file.

NEW_VEHICLE_FORMAT

This key enables the user to specify the output 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.

NEW_PROBLEM_FILE

This key value is appended to the PROJECT_DIRECTORY key to specify the file name for the output problem file created by the program. Problems with the PUMS or zone data will be flagged in this file.

NEW_PROBLEM_FORMAT

This 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.

RANDOM_NUMBER_SEED

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

MAXIMUM_IPF_ITERATIONS

This key is optional. It defaults to 10,000. If specified, it defines the maximum number of iterations that is permitted for the iterative proportional fit process applied to the PUMA and each zone within the PUMA. The IPF process terminates when the convergence criteria is met (i.e., MAXIMUM_IPF_DIFFERENCE) or the maximum number of iterations are reached. The acceptable range is 100 to 100,000.

MAXIMUM_IPF_DIFFERENCE

This key is optional. It defaults to 0.0000001. If specified, it defines the maximum difference between the sum of the distribution and the target marginals for IPF convergence. If convergence is not achieved, the IPF process terminates when the maximum number of iterations (i.e., MAXIMUM_IPF_ITERATIONS) is reached. The acceptable range is greater than zero and less than or equal to 0.001.

STATE_PUMA_LIST

The State-PUMA list identifies the PUMAs that will be processed by this application of the population synthesizer. A comma or space delimited list of PUMA codes can be provided. The PUMA's will be processed in the order they appear in the list. Each PUMA code is a combination of the two letter state code followed by the PUMA number. For example, WA1901 refers to PUMA 1901 within the State of Washington. If the PUMS data were constructing using State FIPS codes rather than abbreviations, the State-PUMA list needs to use these codes. For example, the FIPS code for Virginia is 51. The State-PUMA list value for Alexandria, Virginia is 51200.

PUMS_WEIGHT_FIELD

This key is optional. If provided, the key references a field name within the PUMS household file that contains the household weighting factor. This factor defines the number of times the surveyed household should be replicated to account for its share of total households within the PUMA. The default is to treat each sample equally (i.e., a factor of one).

PUMS_VEHICLE_FIELD

This key is optional. If provided, the key references a field name within the PUMS household file that contains the number of vehicles owned by the household. This value is used to synthesize vehicles for the household. If a field is not provided, the program checks if a field name is provided for the PUMS_AGE_FIELD key. If the age field is not specified, the program

will check for a field called “AGE” in the PUMS population file. If either field is found, a vehicle will be synthesized for each person who is 16 years of age or older. If neither field is found, a vehicle will be synthesized for each person in the household.

PUMS_AGE_FIELD

This key is optional and is only read if PUMS_VEHICLE_FIELD has not been specified. If provided, the key references a field name within the PUMS population file that contains the age in years of each person in the household. The default value for this key is “AGE”. If the PUMS_VEHICLE_FIELD is not provided, a vehicle will be synthesized for each person who is 16 years of age or older.

ZONE_DATA_ID_FIELD

This key is optional. If provided, the key references a field name within the zone data file that contains ID value for each zone. The default value for this key is “ZONE”, "ZONE_ID", "ZONEID", "ZID", or "ID". A unique ID value is expected for each record within the zone data file.

LOCATION_ZONE_FIELD

This key is required and references a field name within the activity location file that contains a zone ID value that links each activity location to a record in the zone data file. Households synthesized for each zone are randomly assigned to activity locations based on the location weighting factor.

Household Models

The following five keys make up a household model group. The first “#” at the end of the each key identifies the model number. All keys with the same model number combine to define how the households are synthesized. Any number of model groups can be defined. Each of the State-PUMA codes identified by the STATE_PUMA_LIST is included in each of the household models. Household models are used to define and distribute households of different types using different household attributes, target marginals, and activity location weighting factors. For example, different models could be used to synthesize and distribute single family, multi-family, and group quarters households within each zone.

LOCATION_WEIGHT_FIELD_#

This key is optional. If provided, it references a field name within the activity location file that contains a weighting factor for a given activity location within the zone. The default is an equal distribution of households to all activity locations within the zone. Since a separate field can be specified for each household model, the modeler can influence how households of different types are distributed to activity locations. For example, the weighting factor for the single family household model can force the single family households to be distributed to activity locations within single family neighborhoods.

ZONE_TOTAL_FIELD_#

This key is optional. If provided, it references a field name within the zone data file that contains the total number of households that should be synthesized for the zone. The default is to use the sum of the attribute distribution fields as the total number of households. Providing a total field enables the modeler to increase or decrease the total number of households assigned to a zone without changing or adjusting the attribute distributions. In this way, the attribute distributions can be based on actual survey data that is automatically factored to reflect the total households estimated for a forecast year. Each household type can have its own growth factor.

Attribute Groups

Each household model can include any number of cross-classification attributes for synthesizing and selecting PUMS households. The following three keys make up an attribute group for a given household model. The last "#" at the end of each key identifies the attribute group. All keys with the same attribute number combine to define how the attribute is processed. Each model must have at least two attribute groups.

PUMS_ATTRIBUTE_FIELD_#_#

This key is required. It references a field name within the PUMS household file that is used to classify the household. The number of classifications associated with each attribute is defined by the PUMS_ATTRIBUTE_BREAKS key.

PUMS_ATTRIBUTE_BREAKS_#_#

This key is required. It defines the values of the corresponding PUMS attribute field that define the break points between classification types. The key is interpreted as a comma or space delimited list of integers. For example, if the PUMA attribute field is income and the income attribute in the PUMS household file is defined in dollars, a breaks key like:

20000, 50000, 100000

would create four income classification categories defined as:

1 = $0 to $20,000
2 = $20,000 to $50,000
3 = $50,000 to $100,000
4 = more than $100,000

ZONE_FIELD_GROUP_#_#

This key is required. It references a base text for a group of field names within the zone data file. The number of fields included in the field group is based on the number of classification categories defined by the corresponding PUMS_ATTRIBUTE_BREAKS key. Given the example above, the value of this key might be "INCOME". The program would then look for four fields in the zone data file that start with "INCOME" and end with the category number (i.e., INCOME1, INCOME2, INCOME3, and INCOME4). These four fields define the income distribution for each zone. The field values may be the number of households, percentages, or

other floating point values. The values are automatically normalized to total 1.0 within the population synthesizer.

STARTING_HOUSEHOLD_ID

This key is optional. If provided, it defines the number assigned to the first household synthesized by the program. The default value is one.

STARTING_VEHICLE_ID

This key is optional. If provided, it defines the number assigned to the first vehicle synthesized by the program. The default value is one.

ZONE_WEIGHT_FACTOR_#

This key 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.

OUTPUT_HOUSEHOLD_FIELDS

This key is optional. If provided, it defines a comma or space delimited list of field names in the PUMS household file that will be copied to the output household file. If this key is not provided, all fields in the PUMS household file are copied to the output household file. This key provides a way to select or restrict the number of data fields saved to the output file.

OUTPUT_POPULATION_FIELDS

This key is optional. If provided, it defines a comma or space delimited list of field names in the PUMS population file that will be copied to the output population file. If this key is not provided, all fields in the PUMS population file are copied to the output population file. This key provides a way to select or restrict the number of data fields saved to the output file.

LOCATION_WARNING_FLAG

This key is optional and defaults to false which keeps the **PopSyn** program from writing warning messages for each activity location with zone numbers that are outside of the TRANSIMS modeling area. This key is useful when generating a synthetic population for a subarea.

POPSYN_REPORT_#

Reports are optional. The "#" at the end of the report keyword represents the report number (e.g., **POPSYN**_REPORT_1). The key can be provided with additional numbers to specify additional reports. The string parameter associated with a report keyword is limited to the following options:

PUMA_CROSS_CLASSIFICATION
PUMS_HOUSEHOLD_SUMMARY
PUMS_POPULATION_SUMMARY

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

PUMA_Cross_Classification

The results of the iterative proportional fit for each PUMA can be displayed in a report like the example shown below:

```
PUMA DC200 Cross Classification

      AUTOS     HHSIZE        Share
          1          1     0.000000
          1          2     0.436000
          1          3     0.224000
          2          1     0.268000
          2          2     0.000000
          2          3     0.072000
```

PUMS Household Summary

The PUMS household summary report calculates the average, minimum, and maximum values for all fields and data records selected from the PUMS household file. Household, location, state, and PUMA fields, plus any other text fields, are not summarized. An example of the report is shown below:

```
PUMS Household Summary

Attribute Name            Average      Minimum      Maximum

HHSIZE                       2.25         1.00         3.00
AUTOS                        1.50         1.00         2.00
WEIGHT                     132.13        11.00       500.00
```

PUMS Population Summary

The PUMS population summary report calculates the average, minimum, and maximum values for all fields and data records selected from the PUMS population file. Household and person, plus any other text fields, are not summarized. An example of the report is shown below:

```
PUMS Population Summary

Attribute Name            Average      Minimum      Maximum

AGE                         19.89         2.00        40.00
GENDER                       1.56         1.00         2.00
```

Algorithm Notes

The **PopSyn** program synthesizes households, persons and vehicles, and assigns the households to activity locations and the vehicles to parking lots. The Public Use Microdata Samples (PUMS) data from the U.S. Census are used to define the household characteristics. Forecast distributions of household attributes for zones or block groups are used to select household types allocated to geographic areas within the region.

Household Models and Attributes

The process begins by defining household models, attributes, and attribute classifications for each PUMA specified in the control file. Household models are used to synthesize and distribution households that have different geographic patterns, attributes, or forecasted totals. Models are often used to distinguish single family, multi-family, and group quarters household types. These types are typically located in specific subareas within a zone, are often classified using different household attributes, and are frequently included in urban land-use forecasts.

Within each household model, the modeler will identify the set of cross-classification attributes used to select an appropriate household from the PUMS sample to replicate for a given zone. Each attribute is classified into several categories using data ranges or break points. These same attributes and attribute classifications need to be defined for each zone within each PUMA. The zone data defines the attribute distribution or marginal totals for the zone.

For example, the following control keys define a model with two attributes (household size and auto ownership). The household size data is defined into three categories (1, 2 and 3+). The auto ownership is defined with two categories (0-1 and 2+). The program will expect to find six fields in the zone data file to support this model. A field called "SFHH20" defines the total number of households of this type synthesized for this zone. Fields "HHSIZE1", "HHSIZE2", and "HHSIZE3" define the household size distribution (or marginal totals) for the zone. Fields "AUTOS1" and "AUTOS2" define the auto ownership distribution for the zone. Finally, a field called "SFHHOLD" in the activity location file defines the weighting factors for distributing the households synthesized for each zone to specific locations within the zone.

```
ZONE_TOTAL_FIELD_1              SFHH20
LOCATION_WEIGHT_FIELD_1         SFHHOLD

PUMS_ATTRIBUTE_FIELD_1_1        PERSONS
PUMS_ATTRIBUTE_BREAKS_1_1       1, 2
ZONE_FIELD_GROUP_1_1            HHSIZE

PUMS_ATTRIBUTE_FIELD_1_2        VEHICLES
PUMS_ATTRIBUTE_BREAKS_1_2       1
ZONE_FIELD_GROUP_1_2            AUTOS
```

PUMA Classification Totals

The modeler specified list of PUMAs are then added to each model. The PUMS household and population files are read and aggregated based on the attribute classifications and PUMS household weighting factor. The program expects to find several fields in the household and population files to complete this task. The household file should include the following fields:

STATE, PUMA, and
HOUSEHOLD, HHOLD, HH_ID, HHID, or HH.

The population file should include the following fields:

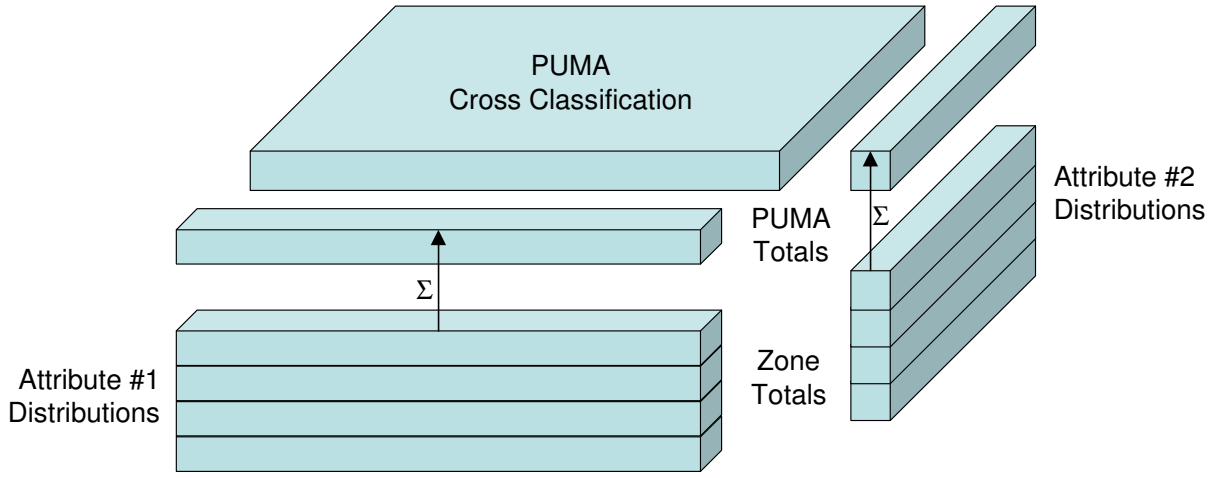
HOUSEHOLD, HHOLD, HH_ID, HHID, or HH, and
PERSON, MEMBER, PER_ID, PERID, or PER.

The state and PUMA fields link the household to a specific PUMA. The household ID fields link the person records to a specific household. The number of persons in the household is equal to the number of person records in the population file.

The zone data file is then read and the attribute distributions and control totals for each zone are added to the zone list for the corresponding PUMA. In order to make this linkage, the program expects to find the following two fields in the zone data file:

STATE and PUMA

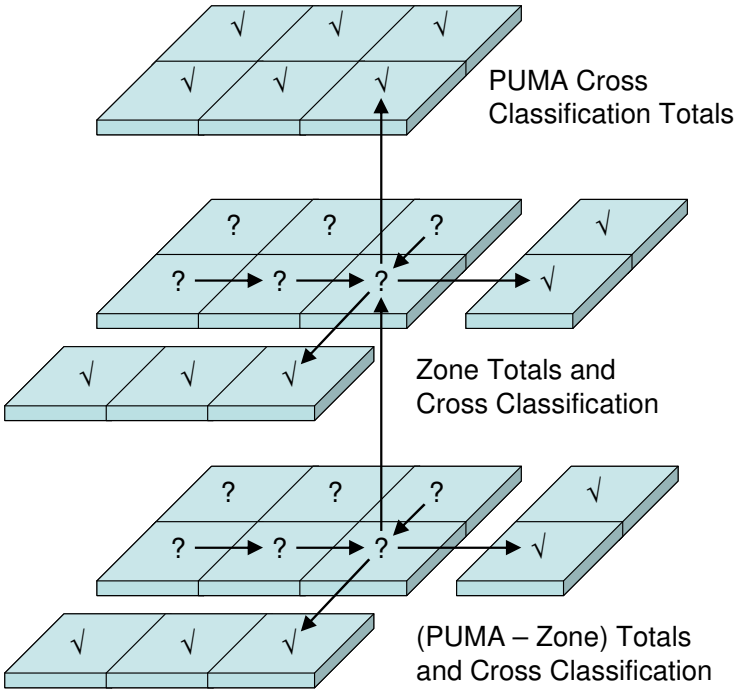
The attribute distributions for each zone are then normalized to 1.0, multiplied by the zone control total, and summed to the PUMA control totals. The PUMA control totals become the target marginals for the iterative proportional fit process that adjusts the PUMS cross classification. This results in a PUMA control total for each cell in the cross classification table. This process for two attributes is depicted in the following graphic.



Zone Classification Totals

The cross classification of household types assigned to each zone is calculated using a second iterative proportional fit (IPF) procedure based on the attribute distributions for the zone and the PUMA cross classification totals. In this process, the PUMA cross classification totals become an additional dimension in the zone IPF process. The number of categories for each of the household attributes is also doubled. The marginals for the original set of categories are the same as the attribute distribution provided for the zone. The new categories are based on the difference between the PUMA marginals for that attribute and the corresponding zone value.

The following graphic may help depict this process. This example shows how a simple cross classification table with two attributes is calculated for a given zone. Note that one attribute has two categories and the other attribute has three categories. The first step in the process generates the six cell matrix of control totals for the whole PUMA. The control totals for each attribute are provided in the zone data file. The control totals for all other zones can then be calculated as the difference between the PUMA attribute totals and the zone attribute totals. The IPF process uses these three sets of control totals to estimate the cross classification matrix for the zone (and the total of all other zones).



Integer Household Distribution

The cross classification table that is generated for each zone by the two stage IPF process has floating point numbers in each cell. These values need to be converted to integer households before they are distributed to activity locations and saved to the output file. This is achieved by applying a bucket rounding process to each cell in the cross classification table.

The cross classification cells at the PUMA level are first set to 0.5. Then for each zone, the bucket rounding value for each cell is added to the floating point households in that cell and truncated to an integer. The remainder is stored in the cell bucket and added to the next zone. This process preserves the total number of households of each household classification, but may change the total number of households assigned to a given zone slightly. If the zone numbering is reasonably uniform, this will cause minimal distortion.

The process then selects households assigned to each cell from the PUMS database based on the number of times a given household must be replicated. Since the PUMS sample represents

approximately five percent of the actual households, most PUMS households will need to be replicated approximately 20 times. The number of replications for a given cell is calculated based on the total number of households in the zone cell to the total number of PUMS household records in the PUMA cell. A bucket rounding process is used to determine the specific number of times a given PUMS household is replicated.

Locating Households and Vehicles

Once a PUMS household is selected, the PUMS person records associated with the household are retrieved and counted. If the number of vehicles is to be set equal to the number of adults in the household, the number of persons 16 years of age or older is also counted. The household data fields are then copied to the output household file, the household ID and number of vehicles fields are updated, and the household location is set. The person records for each PUMS household member are copied to the output population file using the new household ID and person counts.

Households are randomly assigned to an activity location within the zone based on the location weighting factor and zone equivalence. Activity locations are linked to a specific zone through the ZONE_DATA_ID_FIELD and LOCATION_ZONE_FIELD keys. The weights assigned to each location within a zone are normalizes and converted to a cumulative probability distribution. A random number is generated for each household assigned to the zone. This number is used to select the activity location.

Once the household is located, the number of vehicles owned by the household can be synthesized and located. The number of vehicles may be based on the number of vehicles coded on the PUMS household record or based on the number of adult persons included in the household. Generating a vehicle for each adult is one way of avoiding problems in the activity generator caused by a mismatch between the scheduling of auto trips and the number of autos available. Each vehicle is located at the parking lot attached to the home activity location with a process link.

If a vehicle type distribution is provided, the program also randomly selects a vehicle type and subtype from the distribution to assign to each vehicle. Otherwise the default vehicle type is one with a subtype of zero.

Sample Printout

A sample printout file generated by the **PopSyn** 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.

```
*****************************************
|                                       |
|          PopSyn - Version 4.0.7       |
|   Copyright (c) 2009 by AECOM Consult |
|         Mon Apr 19 13:11:46 2010      |
|                                       |
*****************************************

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

Population Synthesizer Test

PUMS Household File = ..\pums\Household.txt

PUMS Population File = ..\pums\Population.txt

Zone Data File = ..\stf3\Zone_Data.txt

Vehicle Type Distribution = ..\vehicle\Type_Distribution.txt

Network Directory = ..\network
Activity Location File = ..\network\Activity_Location.txt
Process Link File = ..\network\Process_Link.txt

New Household File = ..\population\Household.txt
New Household File Format = TAB_DELIMITED

New Population File = ..\population\Population.txt
New Population File Format = TAB_DELIMITED

New Vehicle File = ..\vehicle\Vehicle.txt
New Vehicle File Format = TAB_DELIMITED

New Problem File = ..\results\Problem.txt
New Problem File Format = TAB_DELIMITED

Random Number Seed = 12332
Maximum IPF Iterations = 10000
Maximum IPF Difference = 0.000000100

State PUMA List = DC200, AZ600

PUMS Weight Field Name = WEIGHT, Number = 4
PUMS Vehicle Field Name = AUTOS, Number = 6
Zone Data ID Field Name = ZONE, Number = 1
Location Zone Field Name = TAZ, Number = 9

Household Model #1
Location Weight Field Name = USER1, Number = 13
Zone Data Total Field Name = TOTAL, Number = 4
PUMS Attribute Field #1 Name = HHSIZE, Number = 5
PUMS Attribute Breaks #1 = 1, 2 (3 Types)
Zone Field Group #1 = HHSIZE
       Attribute Type #1 Field = HHSIZE1, Number = 5
       Attribute Type #2 Field = HHSIZE2, Number = 6
       Attribute Type #3 Field = HHSIZE3, Number = 7
PUMS Attribute Field #2 Name = AUTOS, Number = 6
PUMS Attribute Breaks #2 = 1 (2 Types)
Zone Field Group #2 = AUTOS
```

```
       Attribute Type #1 Field = AUTOS1, Number = 8
       Attribute Type #2 Field = AUTOS2, Number = 9

Starting Household ID = 100
Starting Vehicle ID = 100
Output Household Fields = STATE, PUMA, WEIGHT, HHSIZE, AUTOS
Output Population Fields = AGE, GENDER


Number of Activity Location File Records = 96

Number of Process Link File Records = 224

Number of Vehicle Type Distribution Records = 5

Number of PUMS Household File Records = 8
Number of Household Database Records = 8

Number of PUMS Population File Records = 18
Number of Population Database Records = 18

Number of Zone Data File Records = 5


Number of New Household Records = 2000
Number of New Population Records = 4064
Number of New Vehicle Records = 2720
```