Location Data Version 5 – Program Reference

Version 5.0.3

Revision History:

March 2013 – Created by Volpe Center

The **LocationData V5** program is used to:

1. Create or update fields in the activity location file.

2. Assign activity locations to a zone number based on the point-in-polygon equivalence to an ArcView zone boundary file.

3. Create transit accessibility weights based on the number of transit runs within a specified distance of each activity location.

4. Create trip distribution flags based on the use codes of the link attached to the activity location.

5. Create trip distribution weights based on the location of subzone centroids and a subzone data field.

6. Copy data fields from a zone file based on a zone number in the activity location file.

7. Apply custom data processing scripts to manipulate and calculate fields in the activity location file based on inputs from several related files.

8. Access fields in an ArcView polygon boundary file based on a point-in-polygon match to the activity location coordinates.

Syntax is LocationData [-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:

Optional Flags:

-Q[uiet] = execute without screen messages

-H[elp] = show program syntax and control keys

-C[ontrol] = create/update a default control file

-K[eyCheck] = list unrecognized control file keys

-P[ause] = pause before exiting

-N[oPause] = never pause before exiting

-D[etail] = execute with detailed status messages

-X[ML] = write an XML file with control keys

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

# Version 5 Features

# Control Key List

The list of control file keys appears in the table below:

* Req / Opt indicates whether the key is **req**uired or **opt**ional
* The types include **Text**, Input **File**name, **New** file, **Bool**ean, **Path** (to a file), **Time**, **Int**eger, **Dec**imal, and **List** of items
* The Default is the default value, used if the key does not appear in the control file.
* I/O/P indicates Input, Output or Parameter.

For a more detailed description of the Parameter control keys, refer to the Parameter Reference. For a more detailed description of the Input or Output control keys, refer to the File Reference. These two documents also provide the possible values or range of values allowed for each control key listed below. For instance, files can usually be output to numerous formats beyond TAB\_DELIMITED for additional post-processing / file manipulation actions.

## Configuration Keys

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control File Keys:** | **Req/Opt** | **Type** | **Default** | **I/O/P** |
| TITLE | Opt | Text |  | P |
| REPORT\_FILE | Opt | File |  | O |
| REPORT\_FLAG | Opt | Bool | FALSE | P |
| PROJECT\_DIRECTORY | Opt | Path |  | P |
| DEFAULT\_FILE\_FORMAT 1 | Opt | Text | TAB\_DELIMITED | P |
| TIME\_OF\_DAY\_FORMAT 1 | Opt | Text | DAY\_TIME | P |
| MODEL\_START\_TIME | Opt | Time | 0:00 | P |
| MODEL\_END\_TIME | Opt | Time | 24:00:00 | P |
| UNITS\_OF\_MEASURE | Opt | Text | METRIC | P |
| RANDOM\_NUMBER\_SEED | Opt | Int. | 0 | P |
| MAX\_WARNING\_MESSAGES | Opt | Int. | 100000 | P |
| MAX\_WARNING\_EXIT\_FLAG | Opt | Bool | TRUE | P |
| MAX\_PROBLEM\_COUNT | Opt | Int. | 0 | P |
| NUMBER\_OF\_THREADS | Opt | Int. | 1 | P |
| LOCATIONDATA\_REPORT\_\* | Opt | Text |  | P |

## System File Keys

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control File Keys:** | **Req/Opt** | **Type** | **Default** | **I/O/P** |
| NODE\_FILE | Req | File |  | I |
| NODE\_FORMAT | Opt | Text | TAB\_DELIMITED | P |
| LINK\_FILE | Req | File |  | I |
| LINK\_FORMAT | Opt | Text | TAB\_DELIMITED | P |
| LOCATION\_FILE | Req | File |  | O |
| LOCATION\_FORMAT | Opt | Text | TAB\_DELIMITED | P |
| NEW\_LOCATION\_FILE | Req | File |  | O |
| NEW\_LOCATION\_FORMAT | Opt | Text | TAB\_DELIMITED | P |
| ZONE\_FILE | Req | File |  | I |
| ZONE\_FORMAT | Opt | Text | TAB\_DELIMITED | P |
| SHAPE\_FILE | Opt | File |  | I |
| SHAPE\_FORMAT | Opt | Text | TAB\_DELIMITED | P |
| ACCESS\_FILE | Opt | File |  | I |
| ACCESS\_FORMAT | Opt | Text | TAB\_DELIMITED | P |
| TRANSIT\_STOP\_FILE |  |  |  |  |
| TRANSIT\_ROUTE\_FILE |  |  |  |  |
| TRANSIT\_SCHEDULE\_FILE |  |  |  |  |

## File Service Keys

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control File Keys:** | **Req/Opt** | **Type** | **Default** | **I/O/P** |
| NOTES\_AND\_NAME\_FIELDS | Opt | Bool | FALSE | P |

## Projection, Smooth Data, Difference Data Keys

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control File Keys:** | **Req/Opt** | **Type** | **Default** | **I/O/P** |
| INPUT\_COORDINATE\_SYSTEM | Opt | List |  | P |
| INPUT\_COORDINATE\_ADJUSTMENT | Opt | List |  | P |
| OUTPUT\_COORDINATE\_SYSTEM | Opt | List |  | P |
| OUTPUT\_COORDINATE\_ADJUSTMENT | Opt | List |  | P |
| OUTPUT\_XYZ\_SHAPES | Opt | Bool | FALSE | P |
| OUTPUT\_XYM\_SHAPES | Opt | Bool | FALSE | P |

## Control Keys

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Control File Keys:** | **Req/Opt** | **Type** | **Default** | **I/O/P** |
| COPY\_EXISTING\_FIELDS | Opt | Bool | FALSE | P |
| NEW\_WALK\_ACCESS\_FIELD | Opt | Text |  | P |
| MAX\_WALK\_DISTANCE | Opt | Int. | 1000 meters | P |
| WALK\_ACCESS\_TIME\_RANGE | Opt | Text |  | P |
| NEW\_USE\_FLAG\_FIELD\_\* | Opt | Text |  | P |
| LINK\_USE\_FLAG\_TYPES\_\* | Opt | Text |  | P |
| SUBZONE\_ZONE\_FACTOR\_FILE | Opt | File |  | I? |
| NEW\_SUBZONE\_FIELD\_\* | Opt | Text |  | P |
| MAX\_SUBZONE\_DISTANCE\_\* | Opt | Int. |  | P |
| SUBZONE\_DATA\_FILE\_\* | Opt | File |  | I |
| SUBZONE\_DATA\_FORMAT\_\* | Opt | Text |  | P |
| SUBZONE\_DATA\_FIELD\_\* | Opt | Text |  | P |
| SUBZONE\_ZONE\_FIELD\_\* | Opt | Text |  | P |
| NEW\_LOCATION\_FIELD\_\* | Opt | Text |  | P |
| CONVERSION\_SCRIPT | Opt | File |  | I |
| DATA\_FILE\_\* | Opt | File |  | I |
| DATA\_FORMAT\_\* | Opt | Text |  | P |
| DATA\_JOIN\_FIELD\_\* | Opt | Text |  | P |
| LOCATION\_JOIN\_FIELD\_\* | Opt | Text |  | P |
| BOUNDARY\_POLYGON\_FILE\_\* | Opt | File |  | I |
| NEW\_ZONE\_LOCATION\_MAP\_FILE | Opt | File |  | O |
| MINIMUM\_ZONE\_LOCATIONS | Opt | Int. | 4 | P |
| ZONE\_BOUNDARY\_FILE | Opt | File |  | I |
| ZONE\_FIELD\_NAME | Opt | Text |  | P |
| ZONE\_UPDATE\_RANGE | Opt | List | ALL | P |

#### Notes

Each '\_FILE' key has a corresponding '\_FORMAT' key. The following file formats can be used for input and output files:

TEXT, BINARY, FIXED\_COLUMN, COMMA\_DELIMITED, SPACE\_DELIMITED, TAB\_DELIMITED, CSV\_DELIMITED, DBASE, SQLITE3, VERSION3

1-Control key is listed in the –h command line listing generated by LocationData, but the key is not listed in the associated Quick Reference document.

# *Control Key Changes in LocationData Version 5*

## New Key NEW\_ZONE\_LOCATION\_MAP\_FILE

This is the name of a file that contains a list of zones along with nearby activity locations that are not currently assigned to the zones. The zones that are listed include those zones that are currently assigned to fewer activity locations than the number specified in MINIMUM\_ZONE\_LOCATIONS. For each such zone, one or more locations, near to but not currently assigned to the zone, are listed. An example appears below.

ZONE LOCATIONS

2 48, 9

5 46

10 13, 56

11 57, 14

12 51, 1

13 29, 37

14 58, 41

## New Key MINIMUM\_ZONE\_LOCATIONS

This key is an integer, with a default value of 4, and a valid range of 2 to 20. It indicates the minimum number of locations that should be associated with each zone. It is used with the NEW\_ZONE\_LOCATION\_MAP\_FILE key to suggest additional location assignment to those zones that are currently assigned to fewer than MINIMUM\_ZONE\_LOCATIONS.

## Possibly Unused Keys

The following keys are inherited from the Execution Service, but do not appear to be used:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| MODEL\_START\_TIME | Opt | Time | 0:00 | P |
| MODEL\_END\_TIME | Opt | Time | 24:00:00 | P |
| UNITS\_OF\_MEASURE | Opt | Text | METRIC | P |
| RANDOM\_NUMBER\_SEED | Opt | Int. | 0 | P |
| MAX\_PROBLEM\_COUNT | Opt | Int. | 0 | P |
| NUMBER\_OF\_THREADS | Opt | Int. | 1 | P |
| OUTPUT\_XYZ\_SHAPES | Opt | Bool | FALSE | P |
| OUTPUT\_XYM\_SHAPES | Opt | Bool | FALSE | P |

## Keys Needing Definition

The following keys need to be defined:

SUBZONE\_ZONE\_FACTOR\_FILE

BOUNDARY\_POLYGON -> BOUNDARY\_POLYGON\_FILE\_\*

# Network File Name and Control Key Name Changes (TransimsNet 4.0 🡪 5.0)

Additional control key changes are present in TransimsNet 5.0 that are not described in this section. Changes have been made to select control key and file and parameter names in many instances as can been seen from the list below. Additionally, file structures have been modified in some instances. New conceptual constructs have been introduced to improve overall performance and ease of use and robustness of TRANSIMS 5.0 as well. Lastly, new data fields have been added within some input and output files (e.g., SubArea). Refer to the File Reference and Parameter Reference documents for additional details. Of particular importance to LocationData is the Location\_File:

## ACTIVITY\_LOCATION\_TABLE 🡪 LOCATION\_FILE

The ACTIVITY\_LOCATION\_TABLE file control key (and associated file format key) from LocationData 4.0 is replaced by the LOCATION\_FILE key in LocationData 5.0. Refer to the File Reference mentioned above for additional details on the programmatic changes made to the LOCATION\_FILE control key in TRANSIMS 5.0. These changes also apply to the NEW\_ACTIVITY\_LOCATION\_FILE control key and associated file output format control key.

Other specific examples include the following control key and file name changes (V4 🡪 V5):

* NET\_NODE\_TABLE 🡪 NODE\_FILE
* NET\_ZONE\_TABLE 🡪 ZONE\_FILE
* NET\_SHAPE\_TABLE 🡪 SHAPE\_FILE
* NET\_LINK\_TABLE 🡪 LINK\_FILE
* NET\_POCKET\_LANE\_TABLE 🡪 POCKET\_FILE
* NET\_LANE\_USE\_TABLE 🡪 LANE\_USE\_FILE
* NET\_TOLL\_TABLE 🡪 LANE\_USE\_FILE
* NET\_LANE\_CONNECTIVITY\_TABLE 🡪 CONNECTION\_FILE
* NET\_TURN\_PROHIBITION\_TABLE 🡪 TURN\_PENALTY\_FILE
* NET\_PARKING\_TABLE 🡪 PARKING\_FILE
* NET\_ACTIVITY\_LOCATION\_TABLE 🡪 LOCATION\_FILE
* NET\_PROCESS\_LINK\_TABLE 🡪 ACCESS\_FILE
* NET\_UNSIGNALIZED\_NODE\_FILE 🡪 SIGN\_FILE
* NET\_SIGNALIZED\_NODE\_TABLE 🡪 SIGNAL\_FILE

# Examples

## Using a TAZ shape file to assign locations to zones

This example reads a list of zones (Input\_Zone.txt), and a zone shape file (TAZ.shp) where the id field contains the zone number. It generates a new location file with location-zone assignments as well as a location map file (the location map file is optional). The control file is as follows.

TITLE Tiny Example - Zones from Shape File

LINK\_FILE ../network/Link.txt

NODE\_FILE ../network/Node.txt

SHAPE\_FILE ../network/Shape.txt

LOCATION\_FILE ../network/Location.txt

ZONE\_FILE ../input/Input\_Zone.txt

NEW\_LOCATION\_FILE ../network/Location1.txt

ZONE\_FIELD\_NAME id

ZONE\_UPDATE\_RANGE ALL

ZONE\_BOUNDARY\_FILE ../input/TAZ.shp

NEW\_ZONE\_LOCATION\_MAP\_FILE zonemap1\_8.txt

MINIMUM\_ZONE\_LOCATIONS 8

Outputs include a .prn file, a new location file and a zonemap file.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

| |

| LocationData - Version 5.0.3 |

| Copyright 2012 by TRANSIMS Open-Source |

| Fri Mar 15 14:04:48 2013 |

| |

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Control File = LocationData1.ctl

Report File = LocationData1.prn (Create)

Tiny Example - Zones from Shape File

Project Directory = ./

Default File Format = TAB\_DELIMITED

Time of Day Format = HOUR\_CLOCK

Model Start Time = 0:00

Model End Time = 27:00

Units of Measure = METRIC

Random Number Seed = 1363370688

Number of Threads = 1

Input System Network Files:

Node File = ./../network/Node.txt

Zone File = ./../input/Input\_Zone.txt

Shape File = ./../network/Shape.txt

Link File = ./../network/Link.txt

Location File = ./../network/Location.txt

Output System Network Files:

New Location File = ./../network/Location1.txt

Notes And Name Fields = TRUE

Data Service Controls:

LocationData Control Keys:

Warning: No New Location Fields

New Zone Location Map File = ./zonemap1\_8.txt

Minimum Zone Locations = 8

Zone Boundary File = ./../input/TAZ.shp

Zone Field Name = id

Zone Update Range = ALL

Number of Node File Records = 23

Number of Zone File Records = 10

Highest Zone Number = 14

Number of Shape File Records = 13

Number of Link Shape Records = 3

Number of Link File Records = 24

Number of Directional Links = 37

Number of Zone Boundary File Records = 5

Warning: Location 21 was not within a Zone Polygon

Warning: Location 51 was not within a Zone Polygon

Warning: Location 52 was not within a Zone Polygon

Warning: Location 53 was not within a Zone Polygon

Warning: Location 54 was not within a Zone Polygon

Warning: Location 55 was not within a Zone Polygon

Warning: Location 56 was not within a Zone Polygon

Warning: Location 57 was not within a Zone Polygon

Warning: Location 58 was not within a Zone Polygon

Warning: Location 59 was not within a Zone Polygon

Warning: Location 60 was not within a Zone Polygon

Number of Location File Records = 60

New Zone Location Map File contains 7 Zones and 18 Locations

Fri Mar 15 14:04:48 2013 -- Process Complete with 13 Warnings (0:00:00)

The zonemap file is as follows:

ZONE LOCATIONS

2 32, 20, 9

5 50, 40, 9

10 13, 2

11 31

12 51, 8, 26

13 54, 38, 29

14 21, 58, 19

See Figure 1 and Table 1 (later in this document), for a map and a zone-location assignment.

## Adding fields to the Location table

TITLE LocationData Tiny Example

LINK\_FILE ../network/Link.txt

NODE\_FILE ../network/Node.txt

SHAPE\_FILE ../network/Shape.txt

LOCATION\_FILE ../network/Location.txt

ZONE\_FILE ../input/Input\_Zone.txt

NEW\_LOCATION\_FILE ../network/Location2.txt

CONVERSION\_SCRIPT ../input/LocationData\_Script2.txt

NEW\_LOCATION\_FIELD\_1 ORIG\_WGT, I, 2

NEW\_LOCATION\_FIELD\_2 DEST\_WGT, I, 2

LOCATIONDATA\_REPORT\_1 CONVERSION\_SCRIPT

The script, LocationData\_Script2.txt, is as follows:

NewLocation.Orig\_Wgt = 1

NewLocation.Dest\_Wgt = 1

if (Location.Notes == "External Origin") then

NewLocation.Orig\_Wgt = 1

NewLocation.Dest\_Wgt = 0

endif

if (Location.Notes == "External Destination") then

NewLocation.Orig\_Wgt = 0

NewLocation.Dest\_Wgt = 1

endif

return(1)

The output .prn file is as follows:

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

| |

| LocationData - Version 5.0.3 |

| Copyright 2012 by TRANSIMS Open-Source |

| Fri Mar 15 14:21:57 2013 |

| |

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Control File = LocationData2.ctl

Report File = LocationData2.prn (Create)

LocationData Tiny Example

Project Directory = ./

Default File Format = TAB\_DELIMITED

Time of Day Format = HOUR\_CLOCK

Model Start Time = 0:00

Model End Time = 27:00

Units of Measure = METRIC

Random Number Seed = 1363371717

Number of Threads = 1

Input System Network Files:

Node File = ./../network/Node.txt

Shape File = ./../network/Shape.txt

Link File = ./../network/Link.txt

Location File = ./../network/Location.txt

Output System Network Files:

New Location File = ./../network/Location2.txt

Notes And Name Fields = TRUE

Data Service Controls:

LocationData Control Keys:

New Location Field #1 = ORIG\_WGT, I, 2

New Location Field #2 = DEST\_WGT, I, 2

Conversion Script = ./../input/LocationData\_Script2.txt

LocationData Reports: 1. CONVERSION\_SCRIPT

Compiling Conversion Script

Conversion Script

NewLocation.Orig\_Wgt = 1

NewLocation.Dest\_Wgt = 1

if (Location.Notes == "External Origin") then

NewLocation.Orig\_Wgt = 1

NewLocation.Dest\_Wgt = 0

endif

if (Location.Notes == "External Destination") then

NewLocation.Orig\_Wgt = 0

NewLocation.Dest\_Wgt = 1

endif

return(1)

Number of Node File Records = 23

Number of Shape File Records = 13

Number of Link Shape Records = 3

Number of Link File Records = 24

Number of Directional Links = 37

Number of Location File Records = 60

Fri Mar 15 14:21:57 2013 -- Process Complete (0:00:00)

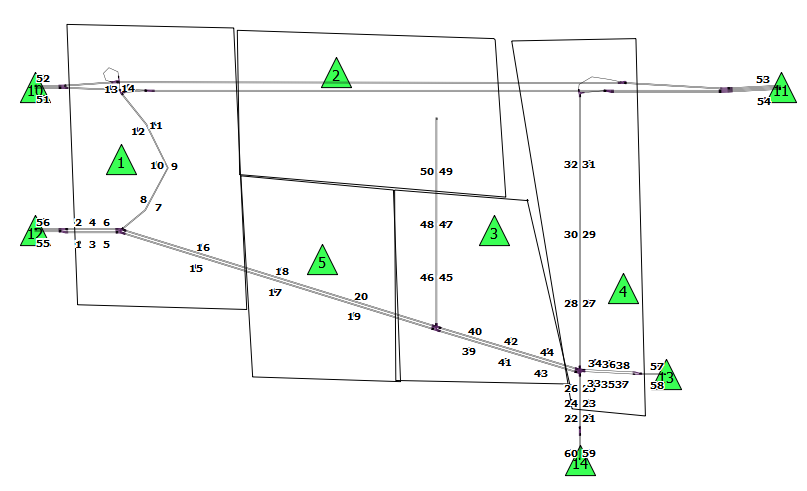


Figure 1 Zones and Locations for the Two Examples

Table 1 Locations and zone assignments for the two examples

|  | INPUT | After running example 1 | After running example 2 | | |  |
| --- | --- | --- | --- | --- | --- | --- |
| LOCATION | ZONE | ZONE | ZONE | ORIG\_WGT | DEST\_WGT | NOTES |
| 1 | 1 | 1 | 1 | 1 | 1 | Activity Location |
| 2 | 1 | 1 | 1 | 1 | 1 | Activity Location |
| 3 | 1 | 1 | 1 | 1 | 1 | Activity Location |
| 4 | 1 | 1 | 1 | 1 | 1 | Activity Location |
| 5 | 1 | 1 | 1 | 1 | 1 | Activity Location |
| 6 | 1 | 1 | 1 | 1 | 1 | Activity Location |
| 7 | 1 | 1 | 1 | 1 | 1 | Activity Location |
| 8 | 1 | 1 | 1 | 1 | 1 | Activity Location |
| 9 | 1 | 1 | 1 | 1 | 1 | Activity Location |
| 10 | 1 | 1 | 1 | 1 | 1 | Activity Location |
| 11 | 1 | 1 | 1 | 1 | 1 | Activity Location |
| 12 | 1 | 1 | 1 | 1 | 1 | Activity Location |
| 13 | 1 | 1 | 1 | 1 | 1 | Activity Location |
| 14 | 1 | 1 | 1 | 1 | 1 | Activity Location |
| 15 | 5 | 1 | 5 | 1 | 1 | Activity Location |
| 16 | 5 | 1 | 5 | 1 | 1 | Activity Location |
| 17 | 5 | 5 | 5 | 1 | 1 | Activity Location |
| 18 | 5 | 5 | 5 | 1 | 1 | Activity Location |
| 19 | 5 | 5 | 5 | 1 | 1 | Activity Location |
| 20 | 5 | 5 | 5 | 1 | 1 | Activity Location |
| 21 | 4 | 4 | 4 | 1 | 1 | Activity Location |
| 22 | 4 | 4 | 4 | 1 | 1 | Activity Location |
| 23 | 4 | 4 | 4 | 1 | 1 | Activity Location |
| 24 | 4 | 4 | 4 | 1 | 1 | Activity Location |
| 25 | 4 | 4 | 4 | 1 | 1 | Activity Location |
| 26 | 4 | 4 | 4 | 1 | 1 | Activity Location |
| 27 | 4 | 4 | 4 | 1 | 1 | Activity Location |
| 28 | 4 | 4 | 4 | 1 | 1 | Activity Location |
| 29 | 4 | 4 | 4 | 1 | 1 | Activity Location |
| 30 | 4 | 4 | 4 | 1 | 1 | Activity Location |
| 31 | 3 | 4 | 3 | 1 | 1 | Activity Location |
| 32 | 3 | 4 | 3 | 1 | 1 | Activity Location |
| 33 | 4 | 4 | 4 | 1 | 1 | Activity Location |
| 34 | 4 | 4 | 4 | 1 | 1 | Activity Location |
| 35 | 4 | 4 | 4 | 1 | 1 | Activity Location |
| 36 | 4 | 4 | 4 | 1 | 1 | Activity Location |
| 37 | 4 | 4 | 4 | 1 | 1 | Activity Location |
| 38 | 4 | 4 | 4 | 1 | 1 | Activity Location |
| 39 | 3 | 3 | 3 | 1 | 1 | Activity Location |
| 40 | 3 | 3 | 3 | 1 | 1 | Activity Location |
| 41 | 3 | 3 | 3 | 1 | 1 | Activity Location |
| 42 | 3 | 3 | 3 | 1 | 1 | Activity Location |
| 43 | 4 | 3 | 4 | 1 | 1 | Activity Location |
| 44 | 4 | 3 | 4 | 1 | 1 | Activity Location |
| 45 | 3 | 3 | 3 | 1 | 1 | Activity Location |
| 46 | 3 | 3 | 3 | 1 | 1 | Activity Location |
| 47 | 3 | 3 | 3 | 1 | 1 | Activity Location |
| 48 | 3 | 3 | 3 | 1 | 1 | Activity Location |
| 49 | 3 | 2 | 3 | 1 | 1 | Activity Location |
| 50 | 3 | 2 | 3 | 1 | 1 | Activity Location |
| 51 | 10 | 10 | 10 | 1 | 0 | External Origin |
| 52 | 10 | 10 | 10 | 0 | 1 | External Destination |
| 53 | 11 | 11 | 11 | 1 | 0 | External Origin |
| 54 | 11 | 11 | 11 | 0 | 1 | External Destination |
| 55 | 12 | 12 | 12 | 1 | 0 | External Origin |
| 56 | 12 | 12 | 12 | 0 | 1 | External Destination |
| 57 | 13 | 13 | 13 | 1 | 0 | External Origin |
| 58 | 13 | 13 | 13 | 0 | 1 | External Destination |
| 59 | 14 | 14 | 14 | 1 | 0 | External Origin |
| 60 | 14 | 14 | 14 | 0 | 1 | External Destination |

## Using Data files to allocate supplemental information to locations

Suppose you have a data file that contains population by zone, and you want to calculate population by location, and create a new field called “population” in the location file. The control file is as follows:

TITLE LocationData Tiny Example

LINK\_FILE ../network/Link.txt

NODE\_FILE ../network/Node.txt

SHAPE\_FILE ../network/Shape.txt

LOCATION\_FILE ../network/Location.txt

ZONE\_FILE ../input/Input\_Zone.txt

ZONE\_FIELD\_NAME id

ZONE\_UPDATE\_RANGE ALL

ZONE\_BOUNDARY\_FILE ../input/TAZ.shp

NEW\_LOCATION\_FILE ../network/Location3.txt

CONVERSION\_SCRIPT ../input/LocationData\_Script3.txt

NEW\_LOCATION\_FIELD\_1 ORIG\_WGT

NEW\_LOCATION\_FIELD\_2 DEST\_WGT

NEW\_LOCATION\_FIELD\_3 POPULATION

DATA\_FILE ../input/Zone\_Population.txt

DATA\_JOIN\_FIELD TAZ

LOCATION\_JOIN\_FIELD ZONE

The conversion script (LocationData\_Script3.txt) is

NewLocation.POPULATION = DATA.POPULATION / DATA.AL\_COUNT

NewLocation.POPULATION = DATA.POPULATION / DATA.AL\_COUNT

NewLocation.ORIG\_WGT = 1

NewLocation.DEST\_WGT = 1

IF (Location.NOTES == "External Destination") THEN

NewLocation.ORIG\_WGT = 0

NewLocation.DEST\_WGT = 1

ENDIF

IF (Location.NOTES == "External Origin") THEN

NewLocation.ORIG\_WGT = 1

NewLocation.DEST\_WGT = 0

ENDIF

RETURN (1)

AL\_COUNT is a built-in TRANSIMS variable that refers to the number of activity locations in a zone.

The input Zone\_Population file is

TAZ POPULATION

1 2500

2 400

3 1300

4 1200

5 800

The LocationData program does three things:

- Assigns locations to zones in accordance with the TAZ.shp file (as in Example 1)

- Assigns ORIG\_WGT and DEST\_WGT to the external zones (as in Example 2)

- Adds a field called POPULATION to the output location table, and fills it in.

In this example, values for POPULATION range from 66 (zone 4 where a population of 1200 is split among 18 activity locations) to 200 (zone 2 where a population of 400 is split among 2 activity locations).

The important file and parameter names for this example are

### DATA\_FILE

The name of the file (typically a text file) that contains supplemental information for each zone. It is used by LocationData in conjunction with a conversion script to add supplemental information to each activity location.

### DATA\_FORMAT OR DATA\_FORMAT\_#

Format for the Data File. Defaults to TAB\_DELIMITED. Options include VERSION3, BINARY, FIXED\_COLUMN, COMMA\_DELIMITED, SPACE\_DELIMITED, TAB\_DELIMITED, CSV\_DELIMITED, DBASE, LANL, SQLITE3

### DATA\_JOIN\_FIELD

The name of the field in the DATA\_FILE (above) that identifies the zone.

### LOCATION\_JOIN\_FIELD

The name of the field in the Location file that identifies the zone.