

**New York Metropolitan Transportation Council
Best Practice Model Regional Transit Network**

**Metropolitan Transportation Authority
Regional Transit Forecasting Model**

PARTIAL DRAFT Technical Memorandum

Transit Network Data Coding

**Partial Draft
September 18, 2001**

1. Introduction

This document presents a description of the procedures used for building the transit networks incorporated in the NYMTC Best Practices Model (BPM) and for the MTA Regional Transit Demand Forecasting Model (RTFM). By design, these networks for the BPM and RTFM are identical and can be used without modification in each model. This is possible because both agencies have selected TransCAD version 4.0 as the platform for their ongoing demand forecasting activities and the use of identical network data and procedures ensures consistency and minimizes confusion on the proper data preparation and coding techniques.

This document describes the procedures used to setup the transit network data and processing procedures on a new computer, the required input data files, and the steps used to run the transit network processing procedures for the NYMTC BPM or the full MTA RTFM.

2. Setup Procedures

This chapter describes the procedures for setting up the transit network processing procedures on a new computer.

2.1 Computer Requirements

The transit network processing procedures have been developed for computers running TransCAD 4.0 (Beta) under the Windows 2000 or Windows 98 Operating Systems. At least 500MB of system memory (RAM) is required and at least 17.4GB of free disk space is required for storage of input data and the results of one simulation run. Each additional run stored on-line requires 10.3 GB. Data storage requirements are as follows:

- Alternative-independent data and procedures: (6.4GB)
 - Procedures and programs: 0.064 GB
 - Trip tables (for MTA procedures and assignment testing only): 4.4GB
 - Highway skims (1.787GB)
 - Other data (0.075GB)
- Alternative-specific information (repeated for each scenario): (10.275 GB)
 - Input data: (0.275 GB)
 - Output data (10.0 GB)
- Work (temporary) space (0.722 GB)

2.2 Installation Procedures

Procedures for Installing Transit Network Processing Procedures

1. Install Current Version of TransCAD 4.0 and note directory for later use in defining the environment variables (e.g., c:\program files\ TransCAD4).
 - Current Version is 5/10/01
2. Add PKZIP25 to a “pathed” directory
3. Load Procedure Directories
 - Load All Subdirectories of the TransCAD Procedures (e.g., e:\tcwutils)
4. Load Data File
 - Input Alternative Data (e:\model\input3)
 - Transit Configuration Information (e:\model\trn)
 - Zone Data (e:\model\zones)
 - County Data (e:\model\nymtcco)
 - Highway Skims (e:\model\hwynet)
 - Highway Zone-PTZs (e:\model\hwypztz)
 - NJ Transit hard coded park-and-ride information: (e:\model\njtpnr)
5. Copy Selected Executables to Transcad Directory specified in step 1
 - TCMX4.exe
 - CONVERT.EXE

- TCONVERT.EXE
 - DFORMD.DLL
 - CALIPERMTX.DLL
6. Create Working and Output Directories
 - Work Directory (e.g., e:\model\work)
 - Output Directory (e.g., e:\model\newnet4)
 7. Start TransCAD and create “Disable Progress Bars” Add in:
 - Name: Disable Progress Bars
 - Macro: Batch Macro
 - UI: <utils directory>disable.dbd (e.g., e:\tcwutils\disable.dbd)

3. Alternative-Specific Input Data Files

The following describe the input data files required for processing the transit network. Some file names and directory locations incorporate the alternative prefix in the filename. Such cases are identified as ALT_ in this documentation. The current base year network Alternative Prefix is “BS97V03”. All alternative-specific input data should be stored in its own directory, currently specified as \nymtc2\newnet4\.

3.1 Parameter File

The parameter file is a GISDK macro that contains the global specifications for the transit network processing procedures. The file may be named according to the user’s preferences but it must have an extension of “.rsc” so that it can be included in other GISDK procedures. We recommend that the file be located in the transit input data area and be named ALT_PARM.rsc. A sample parameter file, named “BS97V03PARM.RSC” is provided in Appendix 1 of this report.

Note that areas shaded in gray will be updated as the BPM develops to integrate these datasets with data used in other elements of the model.

Utility Directory Location

The file must begin with 2 comment lines that specify the directory location of utility GISDK resource files and the location of the GISDK toolbox. The example parameter file specifies these as:

```
//e:\tcw2\utils\  
//c:\progra~1\transcad4\gisdk\toolbox\
```

Macro Definition

On the third line, the parameter file defines the macro as follows:

```
Macro "LoadParm"
```

Section I: Alternative Name

This section defines parameter variables with a “Global” declaration and then assigns a value to the variable: The following (case sensitive) variables are defined:

- *AltPrefix*: Alternative name to be used as a prefix for all file names and report

Section II: Directory Locations

This section defines parameter variables with a “Global” declaration and then assigns a value to the variable: The following (case sensitive) variables are defined:

- *InputDirectory*: Location of all alternative-specific input data
- *WorkDirectory*: Location of all intermediate (scratch) files that can be deleted after runs are made
- *OutputDirectory*: Location of all permanent output files

- *ProgramDirectory*: Location of all Fortran and C programs
- *NJTPNRDir*: Location of New Jersey Transit Park-and-Ride connectors

Section III: Specialized Input File Names

- *InZone*: Filename for TransCAD Geographic File (.dbd) describing zone boundaries
- *TAZFile*: File name for ASCII file defining the zone boundaries
- *BarrierFile*: Filename for ASCII file defining barriers to walk access
- *ConTypeFile*: Filename for ASCII file defining the connector types
- *InCounty*: Filename for TransCAD Geographic File (.cdf or .dbd) defining county boundaries
- *EmpInFile*: Filename for ASCII file defining zone employment
- *RetEmpInFile*: File name for ASCII file defining zone retail employment
- *AreaTypeFile*: Filename for ASCII file defining zonal area type
- *ManhattanCBD*: Filename for TransCAD Geographic file defining the boundary of the Manhattan CBD for transit cordon reporting purposes

Section IV: Network-Building Parameters

- *AltLowMode*: Numeric constant that are used to limit network building to range of Mode_ID codes. AltLowMode defines the low end of the range and should generally be set to 1. Mode ID Codes for this parameter are defined as:
 - 01=Ferry
 - 02=Long Island Rail Road
 - 03=Metro-North Railroad
 - 04=New Jersey Transit Commuter Rail
 - 05=New York City Transit Subway
 - 06=PATH
 - 07=Newark City Subway
 - 08=Bus
- *AltHighMode*: Companion to AltLowMode used to define the highest ModeID to process
- *AltTimePeriod*: is a number constant used to limit the route building process to just those routes that exist in the specified time period. Time periods are defined as follows:
 - 0 = process all routes
 - 1 = process just routes with AM peak service (AM peak headway<999)
 - 2 = process just routes with midday service (midday headway<999)
 - 3 = process just routes with PM peak service (PM peak headway<999)
 - = process just routes with evening service (evening headway<999)
 - = process just routes with night service (night headway<999)
- *SkipCon*: a parameter used to control whether or not the network builder should invoke the transit connector program. Generally, this parameter is set to 0 which

means do not skip transit connectors (i.e., *build* connectors). If set to 1, the network builder will skip building transit connectors. This is done during debugging operations where the user wishes to see the resulting transit networks without incurring the time overhead associated with building connectors.

- *Lowupdatemode*: specifies the lowest Mode_ID that is included in the speed calculation procedures. Since fixed guideway speeds are specified on in the transit route file, this parameter should be set to the lowest bus mode id—800.
- *Hiupdatemode*: specifies the highest Mode_ID that is included in the speed calculation procedures. Since fixed guideway speeds are specified on in the transit route file, this parameter should be set to the lowest bus mode id—899.
- *Breakpoint*: specifies the distance at which buses are assumed to operate at highway speeds rather than at a lower “local” speeds. It is currently set to 3 miles. For stop-to-stop distances less than this distance, the maximum operating speed is assumed to be equal to “top speed” (defined below). For stop-to-stop distances greater than this distance, the maximum operating speed is assumed to be “maximum speed”. In both cases, the stop to stop travel time is a function of the maximum operating speed, the dwell time, and the time required to accelerate and decelerate.

- *PathWeight*: an array that specifies the relative value of in-vehicle time for each transit mode. This parameter is used for building weighted transit paths
- *NetType*: This parameter currently has only one value, “P”
- *qns_man_spd*: This parameter defines a special override speed for any bus links crossing the boundary between Queens and Manhattan
- *Nyc_arr* is an array that specifies the Census identifiers of counties that comprise the New York City core (currently New York County 36061)
- *Ny_arr* is an array that specifies the Census identifiers of counties that comprise the remainder of New York City
- *Li_arr* is an array that specifies the Census identifiers of counties that comprise the New York Counties outside of New York City
- *CT_arr* is an array that specifies the Census identifiers of counties that comprise the Connecticut Counties in the NYMTC model region
- *NJ_arr* is an array that specifies the Census identifiers of counties that comprise the New Jersey Counties in the NYMTC model region
- *Cnty_arr* is a master array that stores the area definitions above
- *Nyc_specs* is an array that stores the top speed (mph), acceleration rate (mphps), deceleration rate (mphps), dwell time, and maximum speed (mph) for the New York City Core. The use of top speed or maximum speed is governed by whether the stop-to-stop distance is less than or greater than or equal to “breakpoint” as discussed above.
- *Ny_specs* is an array that stores the top speed (mph), acceleration rate (mph per second), deceleration rate (mph per second), dwell time, and maximum speed (mph) for the remainder of New York City. The use of top speed or maximum speed is governed by whether the stop-to-stop distance is less than or greater than or equal to “breakpoint” as discussed above.

- *Li_specs* is an array that stores the top speed (mph), acceleration rate (mph per second), deceleration rate (mph per second), dwell time, and maximum speed (mph) for the New York Counties outside of New York City. The use of top speed or maximum speed is governed by whether the stop-to-stop distance is less than or greater than or equal to “breakpoint” as discussed above.
- *NJ_specs* is an array that stores the top speed (mph), acceleration rate (mph per second), deceleration rate (mph per second), dwell time, and maximum speed (mph) for the New Jersey Counties in the NYMTC modeling area. The use of top speed or maximum speed is governed by whether the stop-to-stop distance is less than or greater than or equal to “breakpoint” as discussed above.
- *CT_specs* is an array that stores the top speed (mph), acceleration rate (mph per second), deceleration rate (mph per second), dwell time, and maximum speed (mph) for the Connecticut Counties in the NYMTC modeling area. The use of top speed or maximum speed is governed by whether the stop-to-stop distance is less than or greater than or equal to “breakpoint” as discussed above.
- *Specs_arr* is the master array that combines the area-specific top speed, acceleration rates, deceleration rates, dwell time, and maximum speeds.
- *PrintFlags*: An array defining which routes to print in the transit reporting procedures.

Section V: Skimming and Assignment

- *Period*: Time period for processing, not set by user
- *ModeFile*: Filename for Mode definition file, not set by user
- *ModeXFFile*: Filename for Mode Transfer definition file, not set by user
- *FareMTXFile*: Filename for fare matrix file
- *Walkspeed*: walking speed in minutes per mile
- *WalkSpeed*: walking speed in miles per hour
- *TrnAMCapFactor*: Conversion factor to convert hourly transit capacity to AM period capacity
- *TrnMIDCapFactor*: Conversion factor to convert hourly transit capacity to Midday period capacity
- *TrnPMCapFactor*: Conversion factor to convert hourly transit capacity to PM period capacity
- *TrnEVECapFactor*: Conversion factor to convert hourly transit capacity to Evening period capacity
- *TrnNTCapFactor*: Conversion factor to convert hourly transit capacity to Night period capacity
- *CRAMCapFactor*: Conversion factor to convert hourly commuter rail capacity to AM period capacity
- *CRMIDCapFactor*: Conversion factor to convert hourly commuter rail capacity to Midday period capacity
- *CRPMCapFactor*: Conversion factor to convert hourly commuter rail capacity to PM period capacity
- *CREVECapFactor*: Conversion factor to convert hourly commuter rail capacity to Evening period capacity

- *CRNTCapFactor*: Conversion factor to convert hourly commuter rail capacity to Night period capacity
- *TrnAssnAlpha*. Capacity Constraint Alpha parameter
- *TrnAssnBeta*. Capacity Constraint Beta parameter
- *TrnAssnLimit*. Capacity Constraint multiplicative limit on impedance (as compared to original travel time).
- *TrnAssnFrac*. Capacity Constraint fractions for each iteration
- *Usemodechoicetable*. Flag =1 if using MTA mode choice results or 0=original survey defined trip table

// Section VI: Highway Network Information

Global

FFSkimInFile,ConSkimInFile,FFDistTab,FFTollTab,FFTTimeTab,ConDistTab,ConTimeTab,ConTollTab

Global ConPTZSkimInFile

// Section VII: Trip Tables

Global TripMatrixPrefix

// Section XX: DERIVED FILE NAMES -- NEED NOT MODIFY

Global

InAsciiLinkPrefix,InAsciiRtePrefix,InLinkNodes,InLinkLinks,InRouteRoute,InRouteLinks,InRouteStop,CentroidFile,TempDatabaseFilename,LinkDatabaseFilename,RouteSystemFilename

Global BufferFile,WalkPctFile,FFSkimFile,ConSkimFile,WalkAccFile,WalkAccPrn

Global

HwyAccFile,HwyAccPrn,EmpFile,RetEmpFile,AreaTypeRowFile,AreaTypeColFile

Global HwyAccRetFile,WalkAccRetFile

- **Non-Alternative Specific Filenames**
 - *TAZFile* specifies the file name for the file containing the TAZ edge records described in Chapter 3
 - *CentroidFile* specifies the file name for the file containing the centroid latitude and longitude
 - *BarrierFile* specifies the file name for the file containing the barrier line records described in Chapter 3
 - *ContypeFile* specifies the file name for the file describing definition of the various connector types
 - *InCounty* specifies the TransCAD database containing the county area definitions
- **Parameters for Calculation of Bus Travel Speeds in LOADTIME**
 - *Lowupdatemode* specifies the lowest Mode_ID that is included in the speed calculation procedures. Since fixed guideway speeds are specified on in the transit route file, this parameter should be set to the lowest bus mode id—800.

- *Hiupdatemode* specifies the highest Mode_ID that is included in the speed calculation procedures. Since fixed guideway speeds are specified on in the transit route file, this parameter should be set to the lowest bus mode id—899.
- *Breakpoint* specifies the distance at which buses are assumed to operate at highway speeds rather than at a lower “local” speeds. It is currently set to 3 miles. For stop-to-stop distances less than this distance, the maximum operating speed is assumed to be equal to “top speed” (defined below). For stop-to-stop distances greater than this distance, the maximum operating speed is assumed to be “maximum speed”. In both cases, the stop to stop travel time is a function of the maximum operating speed, the dwell time, and the time required to accelerate and decelerate.
- *Nyc_arr* is an array that specifies the Census identifiers of counties that comprise the New York City core (currently New York County 36061)
- *Ny_arr* is an array that specifies the Census identifiers of counties that comprise the remainder of New York City
- *Li_arr* is an array that specifies the Census identifiers of counties that comprise the New York Counties outside of New York City
- *CT_arr* is an array that specifies the Census identifiers of counties that comprise the Connecticut Counties in the NYMTC model region
- *NJ_arr* is an array that specifies the Census identifiers of counties that comprise the New Jersey Counties in the NYMTC model region
- *Cnty_arr* is a master array that stores the area definitions above
- *Nyc_specs* is an array that stores the top speed (mph), acceleration rate (mphps), deceleration rate (mphps), dwell time, and maximum speed (mph) for the New York City Core. The use of top speed or maximum speed is governed by whether the stop-to-stop distance is less than or greater than or equal to “breakpoint” as discussed above.
- *Ny_specs* is an array that stores the top speed (mph), acceleration rate (mph per second), deceleration rate (mph per second), dwell time, and maximum speed (mph) for the remainder of New York City. The use of top speed or maximum speed is governed by whether the stop-to-stop distance is less than or greater than or equal to “breakpoint” as discussed above.
- *Li_specs* is an array that stores the top speed (mph), acceleration rate (mph per second), deceleration rate (mph per second), dwell time, and maximum speed (mph) for the New York Counties outside of New York City. The use of top speed or maximum speed is governed by whether the stop-to-stop distance is less than or greater than or equal to “breakpoint” as discussed above.
- *NJ_specs* is an array that stores the top speed (mph), acceleration rate (mph per second), deceleration rate (mph per second), dwell time, and maximum speed (mph) for the New Jersey Counties in the NYMTC modeling area. The use of top speed or maximum speed is governed by whether the stop-to-stop distance is less than or greater than or equal to “breakpoint” as discussed above.
- *CT_specs* is an array that stores the top speed (mph), acceleration rate (mph per second), deceleration rate (mph per second), dwell time, and maximum speed (mph) for the Connecticut Counties in the NYMTC modeling area. The use of top

speed or maximum speed is governed by whether the stop-to-stop distance is less than or greater than or equal to “breakpoint” as discussed above.

- *Specs_arr* is the master array that combines the area-specific top speed, acceleration rates, deceleration rates, dwell time, and maximum speeds.

3.2 Fixed Guideway Schedule Data

The fixed guideway transit schedule data is used to represent the operating plans for fixed guideway services such as commuter rail, rapid transit and ferry operations. These services differ from bus transit services in that they tend to operate over a much more limited set of facilities than do buses but operate a much greater variety of different services. In particular, fixed guideway services tend to operate many different skip stop, express, and branch services. The fixed guideway schedule data provides a means for user preparation and coding of these different train patterns.

This fixed guideway schedule information is used by programs included in the model chain to develop the transit route files that are used for actual network building (described in Section 3.3) and ultimately to build the transit networks. The remainder of this section presents the structure of the fixed guideway schedule data.

Directory Location

Fixed Guideway input data must be stored in the input directory as defined in the parameter file (keyword InputDirectory):

Data Files and File Names

Fixed guideway schedule data is coded in a series of comma delimited ascii (.CSV) files that describe all commuter rail, rapid transit, LRT, and ferry operations in the New York Metropolitan area. The following files are defined (in the filenames below “ALT_” is to be replaced by the alternative prefix):

- Long Island Rail Road
 - ALT_LIRBAB.CSV: Babylon Branch
 - ALT_LIRFAR.CSV: Far Rockaway Branch
 - ALT_LIRHEM.CSV: Hempstead Branch
 - ALT_LIRLON.CSV: Long Beach Branch
 - ALT_LIRMON.CSV: Montauk Branch
 - ALT_LIROYS.CSV: Oyster Bay Branch
 - ALT_LIRPTJ.CSV: Port Jefferson Branch
 - ALT_LIRPTW.CSV: Port Washington Branch
 - ALT_LIRRON.CSV: Ronkonkoma Branch
 - ALT_LIRWHE.CSV: West Hempstead Branch

- Metro-North Railroad
 - ALT_MNRHAR.CSV: Harlem Line
 - ALT_MNRHUD.CSV: Hudson Line
 - ALT_MNRNHV.CSV: New Haven Line

- New Jersey Transit
 - ALT_NJTBRG.CSV: Bergen County Line/Main Line
 - ALT_NJTBOO.CSV: Boonton Branch
 - ALT_NJTMOR.CSV: Morris&Essex (Morristown, Gladstone, and Montclair)
 - ALT_NJTNEC.CSV: Northeast Corridor
 - ALT_NJTJNC.CSV: North Jersey Coast Line
 - ALT_NJT PAS.CSV: Pascack Valley
 - ALT_NJT RAR.CSV: Raritan Valley

- New York City Transit Subway
 - ALT_RTEM0519.CSV: 1 and 9 Trains
 - ALT_RTEM0523.CSV: 2 and 3 Trains
 - ALT_RTEM0545.CSV: 4 and 5 Trains
 - ALT_RTEM056X.CSV: 6 Train
 - ALT_RTEM057X.CSV: 7 Train
 - ALT_RTEM05GS.CSV: Grand Central Shuttle
 - ALT_RTEM05AC.CSV: A and C Train
 - ALT_RTEM05DX.CSV: D Train
 - ALT_RTEM05BX.CSV: B Train
 - ALT_RTEM05EX.CSV: E Train
 - ALT_RTEM05FX.CSV: F Train
 - ALT_RTEM05FS.CSV: Shuttle (to F in Brooklyn)
 - ALT_RTEM05GX.CSV: G Train
 - ALT_RTEM05HX.CSV: H Train (Extension to A in Rockaway)
 - ALT_RTEM05JX.CSV: J Train
 - ALT_RTEM05LX.CSV: L Train
 - ALT_RTEM05MX.CSV: M Train
 - ALT_RTEM05NR.CSV: N and R Trains
 - ALT_RTEM05QX.CSV: Q Train
 - ALT_RTEM05ZX.CSV: Z Train

- PATH
 - ALT_RTEM06XX.CSV: PATH

- New Jersey LRT
 - ALT_RTEM07XX.CSV: Newark City Subway/Hudson-Bergen LRT

An example of a fixed guideway transit coding file is shown below in Figure 3-1.

File Format

The file is saved as a excel comma separated format (.CSV) file

File Header Information

- **Mode:** The keyword MODE appears in A1. An integer representing the Mode appears in B1. Modes are defined as follows:
 - 1=Local Bus (all operators, **excluding** ferry shuttles)
 - 2=Limited Bus (NYCT Bus)
 - 3=Express Bus (all operators)
 - 4=Commuter Rail (LIRR, MNR, NJT)
 - 5=Subway (NYC Subway, PATH, Newark City Subway)
 - 7=Ferry/Tram/Ferry Shuttlebus
- **Link Update Code:** The keyword UPDATE appears in A2. An integer representing the Link Update Code appears in B2. Link Update Codes are not currently used in the model but can be coded as follows:
 - 1=Local Bus (all operators, **including** ferry shuttles)
 - 2=Limited Bus (NYCT Bus)
 - 3=Express Bus (all operators)
 - 4=Commuter Rail (LIRR, MNR, NJT)
 - 5=Subway (NYC Subway, PATH, Newark City Subway)
 - 6=Ferry/Tram
- **New Mode 1:** The keyword NEWMODE1 appears in A3. An integer representing the new mode code 1 appears in B3. This Field is not currently used in the model but can be coded as follows:
 - 1= NYC Local Bus
 - 2= NYC Limited Bus
 - 3= NYC Express Bus
 - 4= NYC Subway Local
 - 5= NYC Subway Express
 - 6=Westchester Local Bus (Bee Line)
 - 7=Westchester Express Bus (Bee Line)
 - 8=Huntington Area Rapid Transit Bus
 - 9=City of Long Beach Local Bus
 - 10=MTA Long Island Bus Local
 - 11=Suffolk County Local Bus
 - 12=Orange County Private Express Bus
 - 13=Westchester County Private Express Bus
 - 14=Transport of Rockland Local Bus
 - 15=Transport of Rockland Express Bus
 - 16=NJ Transit Local Bus (including Privates)
 - 17=NJ Transit Express Bus (including Privates)

- 18=PATH
- 19=Newark City Subway
- 20=Greater Bridgeport Local Bus
- 21=Norwalk Transit Local Bus
- 22=CT Transit New Haven Local Bus
- 23=Milford Transit Local Bus
- 24=Meriden Local Bus
- 25=Northeast Transportation Local Bus
- 26=Westport Transit Local Bus
- 27=Houstatonic Local Bus
- 28=CT Transit Stamford Local Bus
- 29=Roosevelt Island Tram
- 30=NYCDOT Ferry
- 31=NY Waterways Ferry
- 32=Ferry Shuttle Bus
- 33=MNR Local Commuter Rail
- 34=MNR Express Commuter Rail
- 35=LIRR Local Commuter Rail
- 36=LIRR Express Commuter Rail
- 37=NJT Local Commuter Rail
- 38=NJT Express Commuter Rail
- 39=Clarkstown Local Bus
- 40=Dutchess CTC Local Bus
- 41=Dutchess Express Bus
- 42=Dutchess Loop Local Bus
- 43=Village of Kirvas Joel Local Bus
- 44=Middletown Local Bus
- 45=Newburgh Local Bus
- 46=Poughkeepsie Local Bus
- 47=Putnam County Local Bus
- **New Mode 2:** The keyword NEWMODE2 appears in A4. An integer representing the new mode code 2 appears in B4. This Field is not currently used in the model but can be coded as follows:
 - 1=Local Route
 - 2=Express Route
- **Mode_ID:** The keyword MODE ID appears in A5. An integer representing the MODE_ID appears in B5. This Field is coded as follows:
 - 1 =Commuter Rail Service (LIRR, MNR, and NJT Rail)
 - 2 = NYC Local (\$1.50) Bus
 - 3 = NYC Express (\$3.00) Bus
 - 4 = PATH
 - 5 = Suburban Bus Charging Approximately \$.50
 - 6 = Suburban Bus Charging Approximately \$.75
 - 7 = Suburban Bus Charging Approximately \$1.00
 - 8 = Suburban Bus Charging Approximately \$1.25
 - 9 = Suburban Bus Charging Approximately \$1.50

- 10 = Suburban Bus Charging Approximately \$1.75
- 11 = Suburban Bus Charging Approximately \$2.00
- 12 = Suburban Bus Charging Approximately \$2.50
- 13 = Suburban Bus Charging Approximately \$3.00
- 14 = Suburban Bus Charging Approximately \$4.00
- 15 = Suburban Bus Charging Approximately \$5.00
- 16 = Suburban Bus Charging Approximately \$6.00
- 17 = Ferries
- 18 = Ferry Bus
- 19 = NYC Subway
- 950 = Reserved for Connector Links (Not Coded on Route Card)
- **Fare Table:** The keyword FARE TYPE appears in A6. An integer representing the Fare Table Number appears in B5. This Field is coded as follows:
 - 0 = Ferry
 - 1 = LIRR
 - 2 = MNR
 - 3 = NJT Rail
 - (Note: Code a “0” for all non-fare zone modes)

Route Header Information

Routes are defined for each unique train pattern--a combination of stops made, links used, and travel times that are the same for each physical train assigned to the pattern. The first train pattern appears in column F, the second train pattern appears in column G and so on until each unique train pattern is defined. Data is as follows:

- **Name:** The keyword NAME appears in A7. In F7, G7, H7... a UNIQUE train name is defined.
- **Direction:** The keyword DIRECTION appears in A8. In F8, G8, H8... a code indicating whether this train pattern operates from top to bottom in the spreadsheet (*Read Down*, coded as +) or whether it operates from bottom to top (*Read Up*, coded as -).
- **AM Headway:** The keyword AM HEADWAY appears in A9. In F9, G9, H8... the average headway for each pattern in the AM Peak period is defined. This headway is calculated as the number of minutes in the AM Peak Period (6AM to 10 AM=240 minutes) divided by the number of trains arriving in Manhattan during that period.
- **Midday Headway:** The keyword MD HEADWAY appears in A10. In F10, G10, H10... the average headway for each pattern in the Midday period is defined. This headway is calculated as the number of minutes in the Midday Period (10AM to 4 PM=360 minutes) divided by the number of trains arriving in Manhattan during that period.
- **PM Headway:** The keyword PM HEADWAY appears in A11. In F11, G11, H11 the average headway for each pattern in the PM Peak period is defined. This headway is calculated as the number of minutes in the PM Peak Period (4PM to 8 PM=240 minutes) divided by the number of trains arriving in Manhattan during that period.

- **Evening Headway:** The keyword EV HEADWAY appears in A12. In F12, G12, H12... the average headway for each pattern in the Evening period is defined. This headway is calculated as the number of minutes in the Evening Period (8PM to 1AM=300 minutes) divided by the number of trains arriving in Manhattan during that period.
- **Night Headway:** The keyword NT HEADWAY appears in A13. In F13, G13, H13... the average headway for each pattern in the Night period is defined. This headway is calculated as the number of minutes in the Night Period (1AM to 6AM=300 minutes) divided by the number of trains arriving in Manhattan during that period.
- **Fare:** The keyword FARE appears in A14. In F14, G14, H14... a number indicating the fare in dollars and cents (\$x.xx) appears. This value is coded as 0 for all zone fare modes (commuter rail and ferry).
- **AM Capacity:** The keyword AM CAPACITY appears in A15. In F15, G15, H15... the average hourly capacity (persons/hour) for each pattern in the AM Peak period is defined.
- **Midday Capacity:** The keyword MD CAPACITY appears in A16. In F16, G16, H16... the average hourly capacity (persons/hour) for each pattern in the Midday period is defined.
- **PM Capacity:** The keyword PM CAPACITY appears in A17. In F17, G17, H17... the average hourly capacity (persons/hour) for each pattern in the PM Peak period is defined.
- **Evening Capacity:** The keyword EV CAPACITY appears in A18. In F18, G18, H18... the average hourly capacity (persons/hour) for each pattern in the Evening period is defined.
- **Night Capacity:** The keyword NT CAPACITY appears in A19. In F18, G19, H19... the average hourly capacity (persons/hour) for each pattern in the Night period is defined.

Route Alignment/Stopping Patterns/Travel Times

The remainder of the file contains information on the alignment of each train pattern, the stops made by each train pattern and the station-to-station travel time.

Generic Link and Station Data

The left 5 columns define a complete set of all links and stops that occur along the line and are roughly analogous to the operation of a local train along the route. In cases where a file represents several branches (i.e., LIRR operation to Penn Station and Flatbush Avenue or MNR branches off of the New Haven Line), all branches are coded and special codes in the train patterns (X) are used to indicate where the branch coding is not applicable.

Each route begins with a LINK record indicating the Link ID in the base network (see the Base Network Geographic File) defining the starting segment of the route. After the first link record, a series of LINK records interspersed with

STOP records are used to define the alignment of the route (i.e., the series of links that make up route) and the locations of all stops along the way.

A LINK record is identified in column A with the keyword "LINK" in columns 4 through 7 of the field followed by the LINK ID in columns 9-19 of the field. Column 20 is used to indicate the direction the link is traversed. A "+" is used to indicate that the link is traversed in its natural topological direction (from its ANODE to its BNODE). A "-" is used to indicate that the train traverses the link in the opposite direction.

Stop records are used to define the occurrence of a stop somewhere on the preceding link. Stop records are distinguished from the other record types by the fact that they do not contain LINK keyword. The specific location of a stop at some point along the preceding link is identified in Column C with a number between 0.00 and 1.00 indicating the fraction of the distance along the link where the stop is located. For instance 0.00 indicates that the stop is located at the beginning of the link (as traversed by the route) and 1.00 indicates that the stop is located at the end of the link (again, as traversed by the route). Column D contains the PTZ ID (used in the station file) for each station and Column E contains the Station Name (used for information purposes only).

Pattern-Specific Stopping and Time Information

Column F contains the train stopping pattern information for the first train pattern. Pattern 2 is shown in column G, pattern 3 is shown in column H and so on. A number appears in each row where the train makes a station stop. The number represents the travel time to the next station. The last station is indicated with a time of "0". The train is assumed to traverse all links (and through all stations) between the first station stop and the last station stop unless an "X" appears in the field. This coding is used to indicate that the link and station coding is for a branch not used by this train.

Figure 3-1. Sample Transit Route File

MODE	4					
UPDATE	4					
NEWMODE1	35					
NEWMODE2	1					
MODE ID	1					
FARE TYPE	1					
NAME				West(1)	West(2)	West(3)
DIRECTION				+	+	+
AM HEADWAY				120	240	80
MD HEADWAY				60	999	999
PM HEADWAY				60	999	999
EV HEADWAY				100	999	999
NT HEADWAY				150	999	999
FARE				0	0	0
AM CAPACITY				720	360	1080
MD CAPACITY				1440	0	0
PM CAPACITY				1440	0	0
EV CAPACITY				864	0	0
NT CAPACITY				576	0	0
LINK 4405198 -	LIRR					X
		0	4615Far Rockaway	4	4	X
		1	4614Inwood	2	2	X
LINK 4405197 -	LIRR					X
		1	4613Lawrence	2	3	X
LINK 4405196 -	LIRR					X
		1	4612Cedarhurst	3	3	X
LINK 4405195 -	LIRR					
		1	4611Woodmere	2	2	2

3.3 Transit Route Files

The input transit route file contains all of the information necessary to construct a TransCAD 4.0 route system. As currently configured, a separate file is used for each mode_id as defined below. Three types of information are contained in each file:

- Route-specific information (e.g., route name, mode, headway)
- Route alignment (i.e., the links that the route traverses from the beginning to end of the run)
- Route stop locations

These different data elements are merged into one data file to simplify the task of reading and coding a transit network. Later processing steps check the integrity of the data and build three separate files that are actually used in the TransCAD *CreateRouteSystemFromTables* command.

Note: Transit Route Files for fixed guideway modes are created using the process described in Section 3.2. Like all route files, they may be also directly edited at the users discretion.

Directory Location and File Names

Transit route files must be stored in the input directory as defined in the parameter file (ALT_parm.rsc, keyword InputDirectory):

The transit route files are stored in a series of files with the following naming structure:

ALT_RTEMxx.ASC

Where:

- *ALT_* = the alternative name for the specific network
- *xx* = MODE_ID (8 in the case of all buses):
 - 01=Ferry
 - 02=Long Island Rail Road
 - 03=Metro-North Railroad
 - 04=New Jersey Transit Commuter Rail
 - 05=New York City Transit Subway
 - 06=PATH
 - 07=Newark City Subway
 - 08=Bus

For an alternative labeled “BS97”, the following input route file names would be defined:

BS97RTEM01.ASC (Ferry)
 BS97RTEM02.ASC (Long Island Rail Road)
 BS97RTEM03.ASC (Metro-North Railroad)
 BS97RTEM04.ASC (New Jersey Transit Commuter Rail)
 BS97RTEM05.ASC (New York City Transit Subway)
 BS97RTEM06.ASC (PATH)
 BS97RTEM07.ASC (Newark City Subway)
 BS97RTEM08.ASC (Bus)

Data File Structure

As shown in file excerpt presented in Figure 3-3, a route system file name is structured as an ordered list of information about each route. Each route begins with a "ROUTE" record showing the route name, mode information and headways. Following the ROUTE record are a series of link and stop records that define the base links that the route follows during its travels from the route origin to the route destination (LINK records) and the locations of individual stops (STOP records). The format of each type of record is defined below.

ROUTE Records

One ROUTE record appears as the first record for each transit route that is specified. ROUTE records are defined with "ROUTE" in columns 1 through 5. The record format for ROUTE Records is presented in Table 3-1.

LINK Records

Each route begins with a LINK record indicating the Link ID in the base network (see the Base Network Geographic File) defining the starting segment of the route. After the first link record, a series of LINK records interspersed with STOP records are used to define the alignment of the route (i.e., the series of links that make up route) and the locations of all stops along the way. A LINK record is identified with the keyword "LINK" in columns 4 through 7. The record format for LINK records is presented in Table 3-2.

STOP Records

Stop records are used to define the occurrence of a stop somewhere on the preceding link. Stop records are distinguished from the other record types by the fact that they do not contain either the ROUTE or LINK keywords. The specific location of a stop at some point along the preceding link is identified with a number between 0.00 and 1.00 indicating the fraction of the distance along the link where the stop is located. For instance 0.00 indicates that the stop is located at the beginning of the link (as traversed by the route) and 1.00 indicates that the stop is located at the end of the link (again, as traversed by the route). The full record format for STOP records is presented in Table 3-3.

Figure 3-1. Sample Transit Route File

1	2	3	4	5	6	7	8	9	10	11							
-----0-----0-----0-----0-----0-----0-----0-----0-----0-----0-----0-----0-----																	
ROUTE	7802622	NJ	Subway SB		5	5	3.00	6.00	3.00	12.00	999.00	1	19	1	7	0	0.00
LINK	6014358	+	Newark Sub														
			.00	4911	4813	-74171344	40780552	Franklin St	(NCS)								169262
			.97	4910	4847	-74173502	40773902	Heller Parkway	(NCS)								169263
LINK	6014357	+	Newark Sub														
			.99	4909	4847	-74177895	40766632	Davenport Av	(NCS)								169264
LINK	6014379	-	Newark Sub														
			.97	4908	4847	-74179697	40762510	Bloomfield Ave	(NCS)								169265
LINK	6014380	+	Newark Sub														
			.98	4907	4830	-74184707	40754549	Park Ave	(NCS)								169266
LINK	6014378	+	Newark Sub														
			.92	4906	4831	-74188443	40748656	Orange	(NCS)								169267
LINK	6014377	+	Newark Sub														
			.97	4905	4831	-74184816	40742753	Norfolk St	(NCS)								169268
LINK	6014376	+	Newark Sub														
LINK	6014375	+	Newark Sub														
			.05	4904	4831	-74184164	40740501	Warren St	(NCS)								169269
			1.00	4903	4831	-74180248	40738869	Washington St	(NCS)								169270
LINK	6014374	+	Newark Sub														
LINK	6014226	-	Newark Sub														
			.06	4902	4747	-74170730	40733590	Broad St.	(NCS)								169271
			.95	4901	4747	-74166094	40732394	Penn Station	(NCS)								169272
ROUTE	7802623	NJ	Subway NB		5	5	3.00	6.00	3.00	12.00	999.00	1	19	1	7	0	1.00
LINK	6014226	+	Newark Sub														
			.05	4901	4747	-74166094	40732394	Penn Station	(NCS)								169273
			.94	4902	4747	-74170730	40733590	Broad St.	(NCS)								169274
LINK	6014374	-	Newark Sub														
LINK	6014375	-	Newark Sub														
			.00	4903	4831	-74180248	40738869	Washington St	(NCS)								169275
			.95	4904	4831	-74184164	40740501	Warren St	(NCS)								169276
LINK	6014376	-	Newark Sub														
LINK	6014377	-	Newark Sub														
			.03	4905	4831	-74184816	40742753	Norfolk St	(NCS)								169277
LINK	6014378	-	Newark Sub														
			.08	4906	4831	-74188443	40748656	Orange	(NCS)								169278
LINK	6014380	-	Newark Sub														
			.02	4907	4830	-74184707	40754549	Park Ave	(NCS)								169279
LINK	6014379	+	Newark Sub														
			.03	4908	4847	-74179697	40762510	Bloomfield Ave	(NCS)								169280
LINK	6014357	-	Newark Sub														
			.01	4909	4847	-74177895	40766632	Davenport Av	(NCS)								169281
LINK	6014358	-	Newark Sub														
			.03	4910	4847	-74173502	40773902	Heller Parkway	(NCS)								169282
			1.00	4911	4813	-74171344	40780552	Franklin St	(NCS)								169283

Table 3-1. ROUTE Record Format

Columns	Variable Name	Type	Contents
1-5	--	--	"ROUTE" to identify record as a ROUTE record
7-16	Route ID	I*4	Unique Route ID (current convention is that ID is between 1,000,000 and 8,000,000 with first digit equal to MODE_ID (or 8 in the case of bus)
18-49	Route Name	C*32	User-specified route name
51-52	Mode	I*4	Mode (used for categorizing transit submodes) 1=Local Bus (all operators, excluding ferry shuttles) 2=Limited Bus (NYCT bus) 3=Express Bus (all operators) 4=Commuter Rail (LIRR, MNR, NJT) 5=Subway (NYC Subway, PATH, Newark City Subway) 7=Ferry/Tram/Ferry Shuttlebus
54-55	Link Update Code	I*4	Old Mode Code (Not Required) 1=Local Bus (all operators, including ferry shuttles) 2=Limited Bus (NYCT bus) 3=Express Bus (all operators) 4=Commuter Rail (LIRR, MNR, NJT) 5=Subway (NYC Subway, PATH, Newark City Subway) 6=Ferry/Tram
57-62	AM Headway	R*4	AM Peak Headway in minutes
64-69	Mid. Headway	R*4	Midday Headway in minutes
71-76	PM Headway	R*4	PM Peak Headway in minutes
78-83	Eve. Headway	R*4	Evening Headway in minutes
85-90	Night Headway	R*4	Overnight Headway in minutes
92-93	AM Route Flag	I*4	Flag=1 if operated in AM Peak (Not Required)

Continued

Table 3-1. ROUTE Record Format (continued)

Columns	Variable Name	Type	Contents
95-96	New Mode 1		<p>Mode Code identifying Operator (Not Required)</p> <p>1= NYC Local Bus 2= NYC Limited Bus 3= NYC Express Bus 4= NYC Subway Local 5= NYC Subway Express 6=Westchester Local Bus (Bee Line) 7=Westchester Express Bus (Bee Line) 8=Huntington Area Rapid Transit Bus 9=City of Long Beach Local Bus 10=MTA Long Island Bus Local 11=Suffolk County Local Bus 12=Orange County Private Express Bus 13=Westchester County Private Express Bus 14=Transport of Rockland Local Bus 15=Transport of Rockland Express Bus 16=NJ Transit Local Bus (including Privates) 17=NJ Transit Express Bus (including Privates) 18=PATH 19=Newark City Subway 20=Greater Bridgeport Local Bus 21=Norwalk Transit Local Bus 22=CT Transit New Haven Local Bus 23=Milford Transit Local Bus 24=Meriden Local Bus 25=Northeast Transportation Local Bus 26=Westport Transit Local Bus 27=Houstatonic Local Bus 28=CT Transit Stamford Local Bus 29=Roosevelt Island Tram 30=NYCDOT Ferry 31=NY Waterways Ferry 32=Ferry Shuttle Bus 33=MNR Local Commuter Rail 34=MNR Express Commuter Rail 35=LIRR Local Commuter Rail 36=LIRR Express Commuter Rail 37=NJT Local Commuter Rail 38=NJT Express Commuter Rail 39=Clarkstown Local Bus 40=Dutchess CTC Local Bus 41=Dutchess Express Bus 42=Dutchess Loop Local Bus 43=Village of Kirvas Joel Local Bus 44=Middletown Local Bus 45=Newburgh Local Bus 46=Poughkeepsie Local Bus 47=Putnam County Local Bus</p>

Continued

Table 3-1. ROUTE Record Format (continued)

Columns	Variable Name	Type	Contents
98-99	New Mode 2		New Mode Code identifying Local/Express Status (Not Required) 1=Local Route 2=Express Route
101-103	Mode ID	I*4	Mode ID for fare processing 1 = Commuter Rail (LIRR, MNR, and NJT Rail) 2 = NYC Local (\$1.50) Bus 3 = NYC Express (\$3.00) Bus 4 = PATH 5 = Suburban Bus Charging Approximately \$.50 6 = Suburban Bus Charging Approximately \$.75 7 = Suburban Bus Charging Approximately \$1.00 8 = Suburban Bus Charging Approximately \$1.25 9 = Suburban Bus Charging Approximately \$1.50 10 = Suburban Bus Charging Approximately \$1.75 11 = Suburban Bus Charging Approximately \$2.00 12 = Suburban Bus Charging Approximately \$2.50 13 = Suburban Bus Charging Approximately \$3.00 14 = Suburban Bus Charging Approximately \$4.00 15 = Suburban Bus Charging Approximately \$5.00 16 = Suburban Bus Charging Approximately \$6.00 17 = Ferry 18 = Ferry Bus 19 = NYC Subway 950 = Reserved for Connector Links (Not Coded on Route Card)
104-107	Fare Core	I*4	Fare Zone Matrix Table to Use 0 = Ferry 1 = LIRR 2 = MNR 3 = NJT Rail (Note: Code a "0" for all non-fare zone modes)
108-113	Fare	R*4	Initial Boarding Fare for non-fare zone modes in dollars (Note: Code a "0.00" for all zone-fare modes)
114-120	AM Capacity	R*4	AM Peak Period Capacity (persons per hour)
121-127	Mid Capacity	R*4	Midday Period Capacity (persons per hour)
128-134	PM Capacity	R*4	PM Peak Period Capacity (persons per hour)
135-141	Eve Capacity	R*4	Evening Period Capacity (persons per hour)
142-148	Night Capacity	R*4	Night Period Capacity (persons per hour)

Note: Suburban bus fare codes are used to represent ranges of fares for suburban carriers while minimizing the total number of individual categories and improve processing time. The fare codes are important in that they are used for computing transfer fare costs. Transfers among suburban carriers are assumed to be equal to the difference between the fares in cases where the transfer-to bus is more expensive than the transfer-from bus. If the transfer-to bus is less expensive then the transfer fare is free. The actual boarding fare for the initial boarding is coded as a separate field on the ROUTE record card as shown above.

Table 3-2. LINK Record Format

Columns	Variable Name	Type	Contents
4-7	--	'--	"LINK" to identify record as a LINK record
9-18	Link ID	I*4	LINK ID from Base Link Geographic File for beginning of route
20-20	Link Direction	C*1	"+" to indicate that link is traversed in same direction as the link topological direction "-" to indicate that link is traversed in the opposite direction as the link topological direction
22+	Comments		Not Currently Used

Table 3-3. STOP Record Format

Columns	Variable Name	Type	Contents
21-24	Link Loc	R*4	Location of stop as expressed as fraction of distance along the preceding link
25-30	Run Time	R*4	Running time to next stop (for rail and ferry modes only) Note: LOADTIME computes time for bus modes (800+)
31-35	PTZ ID	I*4	PTZ ID -- Identifies a Premium Transit Station (Ferry, Commuter Rail, Subway, PATH, or Newark City Subway. This value is automatically posted by the procedures used to create commuter rail schedules and are numbered between 4,001 and 4,999. The station data file must have a corresponding entry that defines the station in more detail.
36-40	PNR ID	I*4	PNR ID -- Identifies a the stop as a Park and Ride Location. This value should be "0" for all premium modes (Ferry, Ferry Bus, Commuter Rail, and Rail). It should also be "0" for bus stops where Park-and Ride is not available. If the stop is in close proximity to a Park-and-Ride coded in the BUSPNR file, then this field should be coded with the PNR ID from that file. This valued is automatically posted by the CHKBUS program.
41-45	Fare Zone ID	I*4	Fare Zone ID -- Fare Zone Assigned to Each Station Generally equal to PTZ for ferry or commuter rail stations Exceptions to the rule that rail fare zones = PTZ are as follows: 5000 (LIRR or NJT Penn Station & nearby bus) 6000 (NJT Hoboken, ferries and nearby Bus) 7000 (Long Island City, ferries and nearby bus) The assignment of a Fare Zone to non-zone fare modes is not critical except that it must be coded with a fare zones That exists in the fare zone matrix. The base year tables Are coded with the nearest PTZ. New coding can be coded Using a simpler rule (i.e., "7000" for all new stops).
46-55	Longitude	I*4	Longitude for stop (automatically posted by the programs that create the commuter rail inputs or by CHKBUS)
56-65	Latitude	I*4	Latitude for stop (automatically posted by the programs that create the commuter rail inputs or by CHKBUS))
67-90	Station Name	C*32	Station Name
91-100	OLDSTOPID	I*4	Not Required

3.3 Station/PTZ Connector File

The PTZ connector file defines the station-to-station connectors that are predefined by the user. Nearly all transfer opportunities are automatically defined by the TRAN_CON program. Station-to-station transfers, however, almost always involve a physical passageway or some other connection. To ensure that station-to-station transfers represent real transfer opportunities, the network processing procedures have been developed to ignore automatically-generated station-to-station connectors and substitute transfer links specified in the subway connector file.

Directory Location and File Name

Station Connector data is stored in the input directory as defined in the parameter file (keyword InputDirectory):

The subway connector file is named as follows:

ALT_PTZCON.ASC

Where:

ALT_ = the alternative name for the specific network

Data File Structure

The station connector file is coded by identifying the From-Station and the To-Station in terms of each station's "Premium Transit Zone" (PTZ). The PTZs are the same as coded on each stop record and are defined in the station data file (described later in this chapter).

Figure 3-4 presents a sample Station Connector File. As shown in this example, a typical station connector in which transfers can include in both directions must be coded as two records—one for each direction that the transfer can be made. A detailed description of the file format is presented in Table 3-4.

Figure 3-4 Sample Station/PTZ Connector File

1	2	3	4	5	6
4009	4438	0.17	3.00	2025.00	
4438	4009	0.17	3.00	2025.00	
4014	4099	0.08	1.00	10800.00	
4099	4014	0.08	1.00	10800.00	
4014	4157	0.02	1.00	7200.00	
4157	4014	0.02	1.00	7200.00	
4014	4185	0.18	5.00	7200.00	
4185	4014	0.18	5.00	7200.00	
4016	4953	0.02	1.00	7200.00	
4953	4016	0.02	1.00	7200.00	
4018	4936	0.00	0.25	33000.00	
4936	4018	0.00	0.25	33000.00	
4021	4938	0.00	0.25	33000.00	
4938	4021	0.00	0.25	33000.00	
4029	4200	0.03	3.50	1890.00	
4200	4029	0.03	3.50	1890.00	
4045	4126	0.03	1.00	5400.00	

Table 3-4 File Format for Station Connector File

Columns	Variable Name	Type	Contents
1-10	From PTZ ID	I*4	PTZ ID (e.g., 4001 – 4999) for the “From” Station
11-20	To PTZ ID	I*4	PTZ ID (e.g., 4001 – 4999) for the “To” Station
21-30	Distance	R*4	Distance in Miles
31-40	Time	R*4	Walking Time in Minutes
41-50	Capacity	R*4	Capacity in Persons per Hour

3.4 Station PNR Link Files

Zones that are connection to each station (i.e., non-bus) Park and Ride are defined in the PNR link files. PNR Links are defined in two files—a “.GEO” file for the link geography and a “.ASC” file for the link attributes.

Directory Location and File Name

Station PNR links files must be stored in the input directory as defined in the parameter file (keyword InputDirectory):

These files are named as follows:

ALT_PNR.GEO

ALT_PNR.ASC

Where:

ALT_ = the alternative name for the specific network

This file is automatically generated using the buildtrninputs.rsc (option 2) GISDK routine from information contained in the station database. As necessary, it may be edited by the analyst following the execution of buildtrninput.rsc.

Data File Structure

Figure 3-5 presents a sample PNR Link. Table 3-5 presents a detailed definition of this file.

Figure 3-5 Sample PNR Link File

Part A Link .GEO File

1	2	3	4	5	6	7	8
1,	10000319,	50004163,	1,2,	-73913545,	40782667,	-73940399,	40750239
2,	10000319,	50004163,	-1,2,	-73913545,	40782667,	-73940399,	40750239
3,	10000319,	50004259,	1,2,	-73913545,	40782667,	-73937256,	40748481
4,	10000319,	50004259,	-1,2,	-73913545,	40782667,	-73937256,	40748481
5,	10000319,	50004361,	1,2,	-73913545,	40782667,	-73911685,	40774595
6,	10000319,	50004361,	-1,2,	-73913545,	40782667,	-73911685,	40774595
7,	10000319,	50004945,	1,2,	-73913545,	40782667,	-73937256,	40748481
8,	10000319,	50004945,	-1,2,	-73913545,	40782667,	-73937256,	40748481
9,	10000320,	50004163,	1,2,	-73915715,	40780979,	-73940399,	40750239
10,	10000320,	50004163,	-1,2,	-73915715,	40780979,	-73940399,	40750239

Part B Link .ASC File

1	2	3	4	5	6	7	8
1	10000319	50004163	1	2.80	8.71	4.00	5.75 35
2	10000319	50004163	-1	2.80	8.71	4.00	5.75-35
3	10000319	50004259	1	2.99	9.59	4.00	5.75 35
4	10000319	50004259	-1	2.99	9.59	4.00	5.75-35

5	10000319	50004361	1	.92	2.46	2.00	2.25 35
6	10000319	50004361	-1	.92	2.46	2.00	2.25-35
7	10000319	50004945	1	.92	2.46	4.00	5.75 35
8	10000319	50004945	-1	.92	2.46	4.00	5.75-35
9	10000320	50004163	1	2.65	8.34	4.00	5.75 35
10	10000320	50004163	-1	2.65	8.34	4.00	5.75-35
	.						
	.						
	.						

Table 3-5. File Format for PNR Link (.ASC) File

Part A Link .GEO File

Columns	Variable Name	Type	Contents
1-10	PNR Link ID	I*4	PNR Link ID (e.g., 1 – 9999)
12-21	Zone	I*4	Zone number (either actual zone number or zone number + 10,000,000)
32-32	Station Endpoint	I*4	Generally the PTZ plus 50,000,000
34-36	Direction	I*4	1 if access to transit, -1 if egress from transit
38-38	Number of Points	I*4	Always=2
41-49	Zone Longitude	I*4	Zone Centroid Longitude in millionths of degrees
51-60	Zone Latitude	I*4	Zone Centroid Latitude in millionths of degrees
62-71	Station Longitude	I*4	Station Longitude in millionths of degrees
73-82	Station Latitude	I*4	Station Latitude in millionths of degrees

Part B Link .ASC File

Columns	Variable Name	Type	Contents
1-10	PNR Link ID	I*4	PNR Link ID (e.g., 1 – 9999)
11-20	Zone	I*4	Zone number (either actual zone number or zone number + 10,000,000)
21-30	Station Endpoint	I*4	Generally the PTZ plus 50,000,000
31-40	Direction	I*4	1 if access to transit, -1 if egress from transit
41-50	Distance	R*4	Distance in miles
51-60	Auto Time	R*4	Auto connector time in minutes
61-70	Walk Time	R*4	Walk time from parking to transit station in minutes
71-80	Cost	R*4	Parking Cost
81-83	Type	I*4	34=Commuter rail access -34=Commuter rail egress 35=Subway access -35=Subway egress 37=Ferry access -37=Ferry egress

3.5 Free Transfer File

The free transfer file contains a list of transfers that are critical to transit network connectivity. These “manual” transfer links will always be created, regardless of the TRAN_CON algorithm for building transfers

Directory Location and File Name

The Free Transfer File must be stored in the input directory as defined in the parameter file (ALT_parm.rsc, keyword InputDirectory):

This file is named

ALT_FREEXFER.ASC

Where:

ALT_ = the alternative name for the specific network

Data File Structure

Each record contains a mega-stop ID and a list of other mega-stop ID's to which this mega-stop should be connected. Given that mega-stops are predefined only for PTZs (premium transit zones) and PNRs this means that as a practical matter, transfers can only be coded for PTZs and PNRs. A sample free transfer file is shown in Figure 3-6. The file format is specified in Table 3-6.

Note: The free transfer file is used to define transfers that are automatically created by Tran_Con. As noted earlier station-to-station transfers are superceded by the subway transfer program. Accordingly, subway-to-subway transfers coded in the free transfer file will not be included in the built network. Station-to-station transfers must be coded in the station transfer file.

Figure 3-6. Sample Free Transfer File

1	2	3	4	5	6	7
427	1	417				
417	1	427				
378	2	101	74			
71	3	104	218	304		
104	3	218	71	304		
218	3	104	71	304		
178	2	70	217			
70	2	178	217			
304	3	104	218	71		
217	2	178	70			
105	1	305				
305	1	105				
373	2	106	306			
106	2	373	306			
306	2	373	106			
292	1	108				
108	1	292				
9	2	438	293			
437	1	182				
182	1	437				
438	2	293	9			
293	2	438	9			
439	2	110	370			
110	2	439	370			

Table 3-6 File Format for Free Transfer File

Columns	Variable Name	Type	Contents
1-10	Mega ID	I*4	Mega ID for stop being connected
11-20	Nxfers	I*4	Number of manual transfers to Mega ID
21-30	Transfer 1	I*4	2nd MegaID for manual transfer 1
31-40	Transfer 2	I*4	2nd MegaID for manual transfer 2
41-50	Transfer 3	I*4	2nd MegaID for manual transfer 3
.			
.			
.			

3.6 Bus PNR Data File

The PNR Data File contains supplemental information on bus park-and-rides that is used to identify available bus park-and-ride locations so that the TRAN_CON program can build drive-to-bus connector links.

Directory Location and File Names

The Bus PNR Data File must be stored in the input directory as defined in the parameter file (ALT_parm.rsc, keyword InputDirectory):

This file is named

ALT_BUSPNR.ASC

Where:

ALT_ = the alternative name for the specific network

Data File Structure

The bus pnr data file contains one record for each bus PNR location in the region. A sample bus PNR Data file is presented in Figure 3-7. A detailed description of the file contents is provided in Table 3-7.

Figure 3-7 Sample PNR Data File

1	2	3	4	5	6	7	8	9
5064	-74497409	40353780	3224	1	100	100	5	1
5023	-74894561	40642189	3320	1	100	100	5	1
5063	-74576883	40421067	3224	1	100	100	5	1
5062	-74459886	40482502	3216	1	100	100	5	1
5056	-74548028	40678439	3184	1	100	100	5	1
5055	-74535764	40887535	3178	1	100	100	5	1
5051	-74754913	41059104	3378	1	100	100	5	1
5054	-74550757	40908685	3178	1	100	100	5	1
5017	-74208961	39958016	3289	1	100	100	5	1
5008	-74227966	40154781	3249	1	100	100	5	1

Table 3-7. PNR Data File Format

Columns	Variable Name	Type	Contents
1-6	PNR ID	I*4	Bus PNR ID (e.g., 5001-7001)
7-13	PNR Sequence	I*4	Sequence number starting at 1501 (not currently used)
14-53	Name	C*40	PNR Name (not currently used)
54-64	Co Name	C*21	Short Name (not currently used)
65-144	Service Codes	10* C*8	10 Service Codes (not currently used)
145-155	PNR Longitude	I*4	Latitude in millionths of degrees
156-166	PNR Latitude	I*4	Longitude in millionths of degrees
167-177	RPNR Longitude	I*4	Latitude in millionths of degrees (not currently used)
168-188	RPNR Latitude	I*4	Longitude in millionths of degrees (not currently used)
189-195	NJT Zone	I*4	NJT Zone ID (not currently used)
196-202	TAZ	I*4	Zone ID where PNR is located (used for establishing the district in which the PNR belongs)
203-209	County	I*4	County Code (not currently used)
210-216	Mode of Transit	I*4	Mode of Transit at PNR Lot (Must be 1)
217-223	No of Spaces	I*4	Number of parking spaces at PNR
224-230	Parking Cost	I*4	Parking Cost
231-237	Walk time	I*4	Walk time between PNR and stop
238-244	Connect Flag	I*4	Connect Flag (Must be 1)

3.7 Centroid Location File

The centroid location file defines the latitude and longitude for each zone centroid and is used to establish the centroid location for walk and drive access connectors.

Directory Location and File Name

The Centroid Location File must be stored in the input directory as defined in the parameter file (ALT_parm.rsc, keyword InputDirectory):

This file is named

ALT_CENTROID.ASC

Where:

ALT_ = the alternative name for the specific network

Data File Structure

A sample file is shown in Figure 3-8 and the file format is described in Table 3-8.

Figure 3-8 Sample Centroid Location File

1	2	3	4	5	6	7	8	9	10
-----0-----0-----0-----0-----0-----0-----0-----0-----0-----0									
2441	-73885079	41919667							
2433	-73775828	41974511							
2437	-73651942	41969190							
2440	-73917288	41950713							
2434	-73542331	41944494							
2652	-73426735	41294457							
2624	-73391413	41407123							
2625	-73384287	41362897							
2623	-73406321	41377132							
2653	-73368030	41312844							
2521	-73401059	41179270							
2515	-73411409	41075157							
2518	-73448690	41220427							
2528	-73382467	41237783							
3287	-74223309	39923701							

Table 3-8 Centroid Location File Format

Columns	Variable Name	Type	Contents
1-10	Zone ID	I*4	Zone ID
11-20	Longitude	I*4	Centroid Longitude
21-30	Latitude	I*4	Centroid Latitude

3.8 Transit Base Link Geographic Files

The transit base link files contain the base links used by the transit routes described earlier in this chapter. They are organized according to transit mode for convenience in building test databases that involve some but not all transit modes. This capability may help in quickly developing test databases for rail modes, which have relatively few routes and links as compared to bus modes, but is not required for the overall operation of the procedures used to build the transit networks.

Directory Location and File Names

The Base Link Geographic Files must be stored in the input directory as defined in the parameter file (ALT_parm.rsc, keyword InputDirectory):

The file names used to store geographic data are as follows:

ALT_LNKMxx.GEO

Where:

- *ALT_* = the alternative name for the specific network
- *Xx* = MODE_ID (8 in the case of all buses):
 - 01=Ferry
 - 02=Long Island Rail Road
 - 03=Metro-North Railroad
 - 04=New Jersey Transit Commuter Rail
 - 05=New York City Transit Subway
 - 06=PATH
 - 07=Newark City Subway
 - 08=Bus

For an alternative labeled “BS97”, the following input link geography file names would be defined:

BS97LNKM01.GEO (Ferry)
 BS97LNKM02.GEO (Long Island Rail Road)
 BS97LNKM03.GEO (Metro-North Railroad)
 BS97LNKM04.GEO (New Jersey Transit Commuter Rail)
 BS97LNKM05.GEO (New York City Transit Subway)
 BS97LNKM06.GEO (PATH)
 BS97LNKM07.GEO (Newark City Subway)
 BS97LNKM08.GEO (Bus)

Data File Structure

The files are structured according to the format of a TransCAD geographic file and is a comma delimited file as compared to the fixed format file types described in earlier files. A sample transit base link file is shown in Figure 3-9. The file format is presented in Table 3-9.

The geographic file format forces the user to define three ID codes for each link:

1. **Link ID.** The Link ID must be a unique ID code for each link in the network. Although the Link Ids are arbitrary, they must be consistent with the Link Ids used to represent the transit route alignment in the route definition files described above.
2. **From Endpoint ID.** The “From” endpoint Ids are also arbitrary identification codes but the each endpoint ID (whether “From” or “To”) should be used to define one and only one point. Any links that connect to this endpoint must use the same endpoint ID and links that do not connect must use different Ids. From and To endpoint ids are used to define a single set of endpoints and must be internally consistent and consistent with one another.
3. **To Endpoint ID.** The “To” endpoint Ids have the same characteristics and restrictions as the “From” endpoint Ids.

Figure 3-9 Sample of a Transit Base Link File

```

      1          2          3          4          5          6          7
-----+-----0-----+-----0-----+-----0-----+-----0-----+-----0-----+-----0-----+-----0
4345256,0,4325095,4325096,2,-73.925546,40.751782,-73.924764,40.751760
4345257,0,4325097,4325095,4,-73.931322,40.751231,-73.929517,40.751666...
4345258,0,4325098,4325097,2,-73.931593,40.751139,-73.931322,40.751231
4345259,0,4325099,4325098,2,-73.931984,40.751002,-73.931593,40.751139
4345260,0,4325100,4325099,2,-73.932375,40.750910,-73.931984,40.751002
4345261,0,4325101,4325100,2,-73.933217,40.750566,-73.932375,40.750910
4345262,0,4325096,4325102,2,-73.924764,40.751760,-73.923982,40.751737
4345263,0,4325102,4325103,2,-73.923982,40.751737,-73.923500,40.751692
4345264,0,4325102,4325104,2,-73.923982,40.751737,-73.921455,40.751853
4345267,0,4325102,4325107,2,-73.923982,40.751737,-73.924914,40.751531
    
```

Table 3-9. File Format for Transit Base Link Geographic File

Field No.	Variable Name	Type	Contents
1	Link ID	I*4	Unique Link ID for the Record (relates to Transit Route File)
2	null	I*4	Contains "0"
3	Fr. Endpoint	I*4	Endpoint ID to be assigned to "From" Endpoint
4	To Endpoint	I*4	Endpoint ID to be assigned to "To" Endpoint

3.9 Transit Base Link Data Files

The Transit Base Link Data Files contain descriptive information for each transit base link previously identified in the base link geographic file.

Directory Location and File Names

The Base Link Geographic Files must be stored in the input directory as defined in the parameter file (ALT_parm.rsc, keyword InputDirectory):

As with the base link geographic files, the data files are structured with different file names for each transit mode. The general structure of these file names is as follows:

ALT_LNKMxx.DCB (binary dictionary file)

-and-

ALT_LNKMxx.BIN (the data)

Where:

- ALT_ = the alternative name for the specific network
- Xx = MODE_ID (8 in the case of all buses):
 - 01=Ferry
 - 02=Long Island Rail Road
 - 03=Metro-North Railroad
 - 04=New Jersey Transit Commuter Rail
 - 05=New York City Transit Subway
 - 06=PATH
 - 07=Newark City Subway
 - 08=Bus

For an alternative labeled “BS97”, the following input base link data file names would be defined:

BS97LNKM01.BIN (Ferry)
 BS97LNKM02.BIN (Long Island Rail Road)
 BS97LNKM03.BIN (Metro-North Railroad)
 BS97LNKM04.BIN (New Jersey Transit Commuter Rail)
 BS97LNKM05.BIN (New York City Transit Subway)
 BS97LNKM06.BIN (PATH)
 BS97LNKM07.BIN (Newark City Subway)
 BS97LNKM08.BIN (Bus)

BS97LNKM01.DCB (Ferry)

BS97LNKM02.DCB (Long Island Rail Road)
BS97LNKM03.DCB (Metro-North Railroad)
BS97LNKM04.DCB (New Jersey Transit Commuter Rail)
BS97LNKM05.DCB (New York City Transit Subway)
BS97LNKM06.DCB (PATH)
BS97LNKM07.DCB (Newark City Subway)
BS97LNKM08.DCB (Bus)

Data File Structure

The file is structured according to the format of a TransCAD binary file and is a comma delimited file as compared to the fixed format file types described in earlier files. The file format is presented in Table 3-10.

Table 3-10 File Format for Transit Base Link Data File

Field No.	Variable Name	Type	Contents
1	ID	I*4	Link ID (same ID as in Geographic File)
2	Length	R*4	Length (in miles)
3	Dir	I*1	Direction (0=2-way, 1=1-way in topological direction of link -1=1-way opposite of topological direction)
4	CLASS	C*4	Optional Highway Class Information
5	TYPE	I*4	Optional Type Information
6	NAME	C*30	Street Name or other identification
7	ROUTE	C*16	Optional description of route or service using link
8	EDIT_DATE	C*8	Date of edit
9	COMMENT	C*30	Comment
10	IVTT	R*8	In-vehicle time for link (NOT USED)
11	OLD_DIR	I*4	(NOT USED)
12	CNTYCODE	I*4	(NOT USED)
13	TRANSIT	I*4	(NOT USED)
14	COUNTY	C*24	(NOT USED)
15	FCLASS	R*8	(NOT USED)
16	SPD_AM	R*8	(NOT USED)
17	SPD_MID	R*8	(NOT USED)
18	SPD_PM	R*8	(NOT USED)
19	SPD_LTE	R*8	(NOT USED)
20	LBUS_AM	R*8	(NOT USED)
21	LBUS_MID	R*8	(NOT USED)
22	LBUS_PM	R*8	(NOT USED)

continued

Table 3-10 File Format for Transit Base Link Data File
(continued)

Field No.	Variable Name	Type	Contents
23	LBUS_LTE	R*8	(NOT USED)
24	XBUS_AM	R*8	(NOT USED)
25	XBUS_MID	R*8	(NOT USED)
26	XBUS_PM	R*8	(NOT USED)
27	XBUS_LTE	R*8	(NOT USED)
28	FromNID	I*4	(NOT USED)
29	ToNID	I*4	(NOT USED)
30	Direction	R*8	(NOT USED)
31	Distance	R*8	(NOT USED)
32	TT	R*8	(NOT USED)
33	CON_TYPE	I*2	(NOT USED)
34	XferMode	I*2	(NOT USED)
35	CT_DESCRIP	C*20	(NOT USED)
36	PREMIUM	I*4	(NOT USED)
37	NONPREM	I*4	(NOT USED)
38	DRIVE_AM	I*4	(NOT USED)
39	DRIVE_PM	I*4	(NOT USED)
40	MTASPEED	R*8	(NOT USED)
41	MTALBUSSPD	R*8	(NOT USED)
42	MTAXBUSSPD	R*8	(NOT USED)
43	TransitIVTT	R*8	(NOT USED)
44	LINK_MODE	I*4	(NOT USED)

3.10 Transit Station Definition Files

The station file is the repository for all station-specific data and is used for a variety of purposes within the procedures that build the transit networks. In particular, they are used for:

- Ensuring that the route system is built so that all given stations are coded at the same longitude and latitude, are given the same fare zone, and are given the same PTZ id.
- Building park-and-ride connectors that based on rules concerning driving distance and parking restrictions.

Directory Location and File Name

The Station File must be stored in the input directory as defined in the parameter file (ALT_parm.rsc, keyword InputDirectory):

This file is named

ALT_STATIONS.ASC

Where:

ALT_ = the alternative name for the specific network

Data File Structure

The file format is presented in Table 3-11.

Table 3-11 Station File Format

Field No.	Variable Name	Type	Contents
1-10	PTZ_ID	I*4	PTZ ID (e.g., 4001 - 4925)
11-20	Longitude	I*4	Longitude in millionths of degrees
21-30	Latitude	I*4	Latitude in millionths of degrees
31-37	NODEID	I*4	(NOT USED)
41-47	CSI_ID	C*4	(NOT USED)
48-48			Blank Space
49-54	Mode/service	C*6	Verbal description of mode used for identifying the mode service by the station: (Subway, Ferry, Rail, PATH, Tram, or Bus)
55-61			Blank
62-64	SERV_ID	I*2	(NOT USED)
65-67	Service ID2	I*2	(NOT USED)
68-72	STA_ID	I*2	(NOT USED)
73-73	?		Blank
74-103	Name	C*30	Station Name
104-112	Source	C*9	Source of data (for information only)
113-135	Schedule Name	C*23	Originally used for processing of schedule information (NOT USED)
136-140	Fare Area	I*4	(NOT USED)
141-145	FARE_ZONE	I*4	Fare Zone for Fare Processing
146-155	MEGA_ID	I*4	Mega Stop ID (sequential ID beginning with 1)
156-165	NSPACES	I*4	General number of parking spaces (for information only)
166-175	PARKCOST	I*4	General Parking Cost (for information only)
176-185	WALKTIME	I*4	General Walk Time (for information only)
186-190			Blank
191-195	TAZ	I*4	TAZ where parking lot is located
196-196			Blank
197-199	Station Access Limitations	C*3	Blank = inactive station RW = Rail/Ferry Walk Access only RKR = Rail/Ferry Walk or Kiss-Ride Access Only RPR = Rail/Ferry Walk/Kiss-Ride/Park-Ride Access CW = Commuter Rail Walk Access only CKR = Commuter Rail Walk or Kiss-Ride Access Only CPR = Commuter Rail Walk/Kiss-Ride/Park-Ride Access
200-200			Blank
201-201	Peak Only Flag	C*1	=P if peak access only, otherwise blank
202-206			Blank
207-214	Service 1	C*8	Code for first service available at station (See below for definitions)
215-222	Service 2	C*8	Code for second service available at station (See below for definitions)
223-230	Service 3	C*8	Code for third service available at station (See below for definitions)
231-238	Service 4	C*8	Code for fourth service available at station (See below for definitions)
239-246	Service 5	C*8	Code for fifth service available at station (See below for definitions)
247-254	Service 6	C*8	Code for sixth service available at station (See below for definitions)

255-262	Service 7	C*8	Code for seventh service available at station (See below for definitions)
263-270	Service 8	C*8	Code for eighth service available at station (See below for definitions)
271-278	Service 9	C*8	Code for ninth service available at station (See below for definitions)
279-286	Service 10	C*8	Code for tenth service available at station (See below for definitions)
287-290	Internal Walk Time	R*4	Internal walk time at transit station in minutes (NOT USED)
291-295	Feeder Bus Walk Time	R*4	Walk time from feeder bus to transit station in minutes (NOT USED)
296-300	K/R Walk Time	R*4	Kiss-Ride walk time in minutes (NOT USED)
301-305	P/R Walk Time	R*4	Park-Ride walk time from parking to Station in minutes
306-310	Unrestricted Parking Spaces	I*4	Number of unrestricted parking spaces
311-315	Restricted Parking Spaces	I*4	Number of restricted parking spaces
316-335	Restriction 1	C*20	City/Town/Village Restriction 1
336-355	Restriction 2	C*20	City/Town/Village Restriction 2
356-360	Unrestricted Parking Cost	I*4	Unrestricted Daily Parking Cost in cents
361-365	Restricted Parking Cost	I*4	Restricted Daily Parking Cost in cents
366-395	Notes	C*20	Notes (NOT USED)
396-402	Restricted Code 1	I*4	Code 1 indicating first set of zones that are allowed to use restricted parking CCMMM Where: CC=NYMTC County Code MMM=MCD or Village Code
403-409	Restricted Code 2	I*4	Code 2 indicating second set of zones that are allowed to use restricted parking (same coding as set 1)
410-415	County Code	I*4	NYMTC County Code (1 – 28)
			1 New York, NY
			2 Queens, NY
			3 Bronx, NY
			4 Kings, NY
			5 Richmond, NY
			6 Nassau, NY
			7 Suffolk, NY
			8 Westchester, NY
			9 Rockland, NY
			10 Putnam, NY
			11 Orange, NY
			12 Dutchess, NY
			13 Fairfield, CT
			14 Bergen, NJ
			15 Passaic, NJ
			16 Hudson, NJ
			17 Essex, NJ
			18 Union, NJ
			19 Morris, NJ
			20 Somerset, NJ
			21 Middlesex, NJ

22	Monmouth, NJ
23	Ocean, NJ
24	Hunterdon, NJ
25	Warren, NJ
26	Sussex, NJ
27	New Haven, CT
28	Mercer, NJ

Service Codes:

FERRIES

FATL Atlantic Highlands
 FBAT Brooklyn Army Terminal
 FCOLG Colgate Ferry Terminal
 FHARB Harborside
 FHIGH Highlands (Seastreak)
 FHOB Hoboken
 FHUNT Long Island City
 FLGA LaGuardia
 FLIB Liberty Harbor
 FLINC Lincoln Harbor
 FPLIB Port Liberte
 FROOS Roosevelt Island Tram
 FSHOOK Sandy Hook/Highlands Fast Ferry
 FSI Staten Island Ferry
 FWEEH Weehawken

LONG ISLAND RAIL ROAD

LBABY Babylon Branch
 LFROCK Far Rockaway Branch
 LGPT Greenport (beyond end of electrification at Ronkonkoma)
 LHEMP Hempstead Branch
 LLBCH Long Beach Branch
 LLONGB Long Beach Branch (applied to City Terminal Zone Stations)
 LMONT Montauk Branch
 LOYST Oyster Bay Branch
 LPJEF1 Port Jefferson Branch (beyond end of electrification at Port
 Jefferson)
 LPJEF2 Port Jefferson Branch (electrified territory)
 LPWASH Port Washington
 LROCK Key Mainline Stations
 LRONK Ronkonkoma
 LWHEMP West Hempstead

METRO-NORTH RAILROAD

MHAR1 Harlem Line (Brewster North and out to Dover Plains)
 MHAR2 Harlem Line (Brewster North and in)
 MHUD1 Hudson Line (Croton Harmon and out to Poughkeepsie)
 MHUD2 Hudson Line (Croton Harmon and in)
 MNHAV1 New Haven Line
 MNHAV2 New Canaan Branch
 MNHAV3 Danbury Branch
 MNHAV4 Waterbury Branch

NEW JERSEY TRANSIT

NBERG Bergen County Line

NBOON Boonton Line
NCS Newark City Subway
NGLAD Gladstone Branch
NMAIN Main Line
NMDXG Midtown Direct - Gladstone
NMDXM Midtown Direct - Morris & Essex
NMONT Montclair Branch
NMORR Morris & Essex
NNEC Northeast Corridor
NNJC1 North Jersey Coast Line (Diesel service: Bay Head to Newark)
NNJC2 North Jersey Coast Line (Electric service: Long Branch to New
York)
NPASC Pascack Valley Line
NPJER Port Jervis Line
NRAR1 Raritan Valley Line (High Bridge to Newark)
NRAR2 Raritan Valley Line (Raritan to Newark)

SUBWAY AND PATH

PATH	PATH
S1	1
S2	2
S3	3
S4	4
S5	5
S6	6
S7	7
S9	9
SA	A
SB	B
SBROOK	Brooklyn CBD
SC	C
SD	D
SE	E
SF	F
SG	G
SJ	J
SL	L
SM	M
SN	N
SQ	Q
SR	R
SS	S
SSIR	Staten Island Rapid Transit
STSQ	Times Square
SZ	Z

3.11 TAZ Locality File

The station locality file contains locality information about each Traffic Analysis Zone for use by the park-and-ride (PNR) generation program. Locality information is compared to parking restrictions at each station to determine which zones can be connected to each station.

Directory Location and File Name

The TAZ Locality File must be stored in the input directory as defined in the parameter file (ALT_parm.rsc, keyword InputDirectory):

This file is named

ALT_CTAZLOCAL.PRN

Where:

ALT_ = the alternative name for the specific network

Data File Structure

The TAZ Locality File contains one record for each TAZ. An excerpt is shown in Figure 3-10. The file structure is presented in Table 3-12. Locality Codes are presented in Table 3-13.

Figure 3-10 Sample of a TAZ Locality File

	1	2	3	4	5	6	7
1660	6	4		-73612479	40749874		
1661	6	4		-73570940	40768736		
1662	6	4		-73592778	40755668		
1663	6	4		-73580366	40759598		
1664	6	4		-73565156	40759765		
1665	6	3		-73720981	40723015		
1666	6	3	124	-73711760	40720733		
1667	6	3	124	-73697449	40724793		
1668	6	3	124	-73699350	40717588		
1669	6	3	125	-73685647	40725722		
1670	6	3		-73709634	40712483		
1671	6	3		-73719435	40702617		
1672	6	3		-73703192	40701016		
1673	6	3		-73699962	40699762		
1674	6	3		-73682855	40711727		
1675	6	3		-73671037	40712074		
1676	6	3		-73685573	40698462		

Table 3-12. File Format for TAZ Locality File

Columns	Variable Name	Type	Contents
1-8	TAZ	I*4	TAZ Number
9-17	County Code	I*4	NYMTC County Code (Defined below)
18-26	MCD Code	I*4	MCD Code (Defined below)
27-35	Village Code	I*4	Village Code (Defined below)
36-46	Longitude	I*4	Longitude in millionths of degrees
47-57	Latitude	I*4	Latitude in millionths of degrees

Table 3-13. Locality Codes for Use in TAZ Locality File

County	County Code	Seq FIPS	Name
Nassau	6	1 3605956000	Oyster Bay town
Nassau	6	2 36059291 13	Glen Cove city
Nassau	6	3 3605934000	Hempstead town
Nassau	6	4 3605953000	North Hempstead town
Nassau	6	5 3605943335	Long Beach city
Nassau	6	101 3629113	Glen Cove city
Nassau	6	102 3647405	Mill Neck village
Nassau	6	103 3618597	Cove Neck village
Nassau	6	104 3641432	Lattingtown village
Nassau	6	105 3613618	Centre Island village
Nassau	6	106 3627485	Freeport village
Nassau	6	107 3628178	Garden City village
Nassau	6	108 3613233	Cedarhurst village
Nassau	6	109 3626352	Flower Hill village
Nassau	6	110 3653022	North Hills village
Nassau	6	111 3663792	Roslyn Estates village
Nassau	6	112 3644787	Malverne village
Nassau	6	113 3643874	Lynbrook village
Nassau	6	114 3622876	East Rockway village
Nassau	6	115 3634297	Hewlett Bay Park village
Nassau	6	116 3634308	Hewlett Harbor village
Nassau	6	117 3643335	Long Beach City
Nassau	6	118 3641553	Lawrence village
Nassau	6	119 3602968	Atlantic Beach village
Nassau	6	120 3634319	Hewlett Neck village
Nassau	6	121 3682986	Woodsburg village
Nassau	6	122 3640937	Lake Success village
Nassau	6	123 3676705	Valley Stream village
Nassau	6	124 3626264	Floral Park village
Nassau	6	125 3650397	New Hyde Park village
Nassau	6	126 3669023	South Floral Park village
Nassau	6	127 3605639	Bellerose village
Nassau	6	128 3671267	Stewart Manor village
Nassau	6	129 3639694	Kings Point village
Nassau	6	130 3630169	Great Neck village
Nassau	6	131 3665035	Sands Point village
Nassau	6	132 3639309	Kensington village
Nassau	6	133 3630213	Great Neck Plaza village
Nassau	6	134 3664232	Russell Gardens village
Nassau	6	135 3664430	Saddle Rock village

Table 3-13. Locality Codes for Use in TAZ Locality File (continued)

County	County Code	Seq FIPS	Name
Nassau	6	136 3630191	Great Neck Estates village
Nassau	6	137 3658497	Plandome Manor village
Nassau	6	138 3658475	Plandome village
Nassau	6	139 3649231	Munsey Park village
Nassau	6	140 3673605	Thomaston village
Nassau	6	141 3658486	Plandome Heights village
Nassau	6	142 3659531	Port Washington North village
Nassau	6	143 3604803	Baxter Estates village
Nassau	6	144 3645106	Manorhaven village
Nassau	6	145 3637847	Island Park village
Nassau	6	146 3654705	Old Westbury village
Nassau	6	147 3679444	Westbury village
Nassau	6	148 3649330	Muttontown village
Nassau	6	149 3682117	Williston Park village
Nassau	6	150 3623217	East Williston village
Nassau	6	151 3633139	Hempstead village
Nassau	6	152 3663264	Rockville Center village
Nassau	6	153 3647636	Mineola village
Nassau	6	154 3646107	MAtinecock village
Nassau	6	155 3676331	Upper Brookville village
Nassau	6	156 3654562	Old Brookville village
Nassau	6	157 3610132	Brookville village
Nassau	6	158 3663803	Roslyn Harbor village
Nassau	6	159 3622260	East Hills village
Nassau	6	160 3663770	Roslyn village
Nassau	6	161 3666047	Sea Cliff village
Nassau	6	162 3645997	Massapequa Park village
Nassau	6	163 3625384	Farmingdale village
Nassau	6	164 3641487	Laurel Hollow village
Nassau	6	165 3656011	Oyster Bay Cove village
Nassau	6	166 3605034	Bayville village
Suffolk	7	1 3610337000	Huntington town
Suffolk	7	2 3610304000	Babylon town
Suffolk	7	3 3610369463	Southold town
Suffolk	7	4 3610368473	Southampton town
Suffolk	7	5 3610310000	Brookhaven town
Suffolk	7	6 3610368000	Smithtown town
Suffolk	7	7 3610361984	Riverhead town
Suffolk	7	8 3610366839	Shelter Island town

Table 3-13. Locality Codes for Use in TAZ Locality File (continued)

County	County Code	Seq FIPS	Name
Suffolk	7	93610338000	Islip town
Suffolk	7	103610359106	Poospatuck reservation
Suffolk	7	113610367059	Shinnecock reservation
Suffolk	7	123610322194	East Hampton town
Suffolk	7	1013602737	Ashroken village
Suffolk	7	1023637022	Huntington Bay village
Suffolk	7	1033643005	Lloyd Harbor village
Suffolk	7	1043602044	Amityville village
Suffolk	7	1053668462	Southampton village
Suffolk	7	1063630576	Greenport village
Suffolk	7	1073653396	Northport village
Suffolk	7	1083651286	Nissequoque village
Suffolk	7	1093632963	Head of the Harbor village
Suffolk	7	1103608323	Brightwaters village
Suffolk	7	1113642554	Lindenhurst village
Suffolk	7	1123603408	Babylon village
Suffolk	7	1133664881	Saltaire village
Suffolk	7	1143654430	Ocean Beach village
Suffolk	7	1153637840	Islandia village
Suffolk	7	1163677519	Village of the Branch village
Suffolk	7	1173656660	Patchogue village
Suffolk	7	1183605771	Bellport village
Suffolk	7	1193640530	Lake Grove village
Suffolk	7	1203660422	Quoque village
Suffolk	7	1213680181	Westhampton Beach village
Suffolk	7	1223680186	unknown
Suffolk	7	1233654617	Old Field village
Suffolk	7	1243605672	Belle Terre village
Suffolk	7	1253659355	Port Jefferson village
Suffolk	7	1263659157	Poquott village
Suffolk	7	1273667191	Shoreham village
Suffolk	7	1283652188	North Haven village
Suffolk	7	1293622183	East Hampton village
Suffolk	7	1303664485	Sag Harbor village
Suffolk	7	1313620379	Dering Harbor village
Westchester	8	13611950617	New Rochelle city
Westchester	8	23611949121	Mount Vernon city
Westchester	8	33611957012	Pelham town

Table 3-13. Locality Codes for Use in TAZ Locality File (continued)

County	County Code	Seq FIPS	Name
Westchester	8	4 3611918410	Cortlandt town
Westchester	8	5 3611956979	Peekskill city
Westchester	8	6 3611942136	Lewisboro town
Westchester	8	7 3611905320	Bedford town
Westchester	8	8 3611955541	Ossining town
Westchester	8	9 3611950078	New Castle town
Westchester	8	10 3611968308	Somers town
Westchester	8	11 3611949011	Mount Pleasant town
Westchester	8	12 3611964309	Rye city
Westchester	8	13 3611951693	North Castle town
Westchester	8	14 3611953517	North Salem town
Westchester	8	15 3611964320	Rye town
Westchester	8	16 3611959685	Round Ridge town
Westchester	8	17 3611930367	Greenburgh town
Westchester	8	18 3611981677	White Plains city
Westchester	8	19 3611932413	Harrison town
Westchester	8	20 3611965442	Scarsdale town
Westchester	8	21 3611921820	Eastchester town
Westchester	8	22 3611984000	Yonkers city
Westchester	8	23 3611944842	Mamaroneck town
Westchester	8	24 3611984077	Yorktown town
Westchester	8	25 3611948895	Mount Kisco town
Westchester	8	101 3650617	New Rochelle city
Westchester	8	102 3649121	Mount Vernon city
Westchester	8	103 3657023	Pelham Manor village
Westchester	8	104 3656979	Peekskill City
Westchester	8	105 3610341	Buchanan village
Westchester	8	106 3655530	Ossining village
Westchester	8	107 3608103	Briarcliff Manor village
Westchester	8	108 3664309	Rye city
Westchester	8	109 3659223	Port Chester village
Westchester	8	110 3658728	Pleasantville village
Westchester	8	111 3637803	Irvington village
Westchester	8	112 3681677	White Plains city
Westchester	8	113 3632402	Harrison village
Westchester	8	114 3665431	Scarsdale village
Westchester	8	115 3664325	Rye Brook village
Westchester	8	116 3620698	Dobbs Ferry village
Westchester	8	117 3632710	Hastings-on-Hudson village

Table 3-13. Locality Codes for Use in TAZ Locality File (continued)

County	County Code	Seq FIPS	Name
Westchester	8	118 3684000	Yonkers city
Westchester	8	119 3675583	Tuckahoe village
Westchester	8	120 3608532	Bronxville village
Westchester	8	121 3657001	Pelham village
Westchester	8	122 3602506	Ardsley village
Westchester	8	123 3653671	Sleepy Hollow village
Westchester	8	124 3673176	Tarrytown village
Westchester	8	125 3624295	Elmsford village
Westchester	8	126 3644831	Mamaroneck village
Westchester	8	127 3641333	Larchmont village
Westchester	8	128 3619213	Croton-on-Hudson village
Westchester	8	129 3648890	Mount Kisco village
Rockland	9	1 3608715968	Clarkstown town
Rockland	9	2 3608755211	Orangetown town
Rockland	9	3 3608760510	Ramapo town
Rockland	9	4 3608771674	Stony Point town
Rockland	9	5 3608732765	Haverstraw town
Rockland	9	101 3676386	Upper Nyack village
Rockland	9	102 3654100	Nyack village
Rockland	9	103 3669441	South Nyack village
Rockland	9	104 3657749	Piermont village
Rockland	9	105 3679174	Wesley Hills village
Rockland	9	106 3650353	New Hempstead village
Rockland	9	107 3667708	Sloatsburg village
Rockland	9	108 3634660	Hillburn village
Rockland	9	109 3670420	Spring Valley village
Rockland	9	110 3615400	Chestnut Ridge village
Rockland	9	111 3648090	Montebello village
Rockland	9	112 3600408	Airmont cdp
Rockland	9	113 3671894	Suffern village
Rockland	9	114 3638934	unknown
Rockland	9	115 3650705	New Square village
Rockland	9	116 3658992	Pamona village
Rockland	9	117 3680203	West Haverstraw village
Rockland	9	118 3632754	Haverstraw village
Rockland	9	119 3629872	Grand View-on-Hudson village
Putnam	10	1 3607957584	Philipstown town
Putnam	10	2 3607960147	Putnam Valley town
Putnam	10	3 3607912529	Carmel town

Table 3-13. Locality Codes for Use in TAZ Locality File (continued)

County	County Code	Seq FIPS	Name
Putnam	10	4 3607968924	Southeast town
Putnam	10	5 3607939331	Kent town
Putnam	10	6 3607956748	Patterson town
Putnam	10	101 3616936	Cold Spring village
Putnam	10	102 3649781	Nelsonville village
Putnam	10	103 3608070	Brewster village
Orange	11	1 3607129553	Goshen town
Orange	11	2 3607178839	Wawayanda town
Orange	11	3 3607177992	Wallkill town
Orange	11	4 3607147042	Middletown city
Orange	11	5 3607178366	Warwick town
Orange	11	6 3607119961	Deerpark town
Orange	11	7 3607130631	Greenville town
Orange	11	8 3607148857	Mount Hope town
Orange	11	9 3607159388	Port Jervis city
Orange	11	10 3607147713	Minisink town
Orange	11	11 3607107003	Blooming Grove town
Orange	11	12 3607118300	Cornwall town
Orange	11	13 3607131907	Hamptonburgh town
Orange	11	14 3607134550	Highlands town
Orange	11	15 3607147999	Monroe town
Orange	11	16 3607175781	Tuxedo town
Orange	11	17 3607115308	Chester town
Orange	11	18 3607182755	Woodbury town
Orange	11	19 3607150848	New Windsor town
Orange	11	20 3607148153	Montgomery town
Orange	11	21 3607118916	Crawford town
Orange	11	22 3607150045	Newburgh town
Orange	11	23 3607150034	Newburgh city
Orange	11	101 3647042	Middletown city
Orange	11	102 3659388	Port Jervis city
Orange	11	103 3676210	Unionville village
Orange	11	104 3655673	Otisville village
Orange	11	105 3678465	Washingtonville village
Orange	11	106 3618333	Cornwall on Hudson village
Orange	11	107 3647988	Monroe village
Orange	11	108 3675803	Tuxedo Park village
Orange	11	109 3629542	Goshen village

Table 3-13. Locality Codes for Use in TAZ Locality File (continued)

County	County Code	Seq FIPS	Name
Orange	11	110 3678355	Warwick village
Orange	11	111 3626319	Florida village
Orange	11	112 3630752	Greenwood Lake village
Orange	11	113 3615297	Chester village
Orange	11	114 3639853	Kirvas Joel village
Orange	11	115 3632325	Harriman village
Orange	11	116 3634495	Highland Falls village
Orange	11	117 3677849	Walden village
Orange	11	118 3646162	Maybrook village
Orange	11	119 3648142	Montgomery village
Orange	11	120 3650034	Newburgh city
Dutchess	12	1 3602747207	Milan town
Dutchess	12	2 3602761357	Rhinebeck town
Dutchess	12	3 3602758156	Pine Plains town
Dutchess	12	4 3602751891	North East town
Dutchess	12	5 3602770662	Stanford town
Dutchess	12	6 3602759641	Poughkeepsie city
Dutchess	12	7 3602759652	Poughkeepsie town
Dutchess	12	8 3602778157	Wappinger town
Dutchess	12	9 3602725978	Fishkill town
Dutchess	12	10 3602737209	Hyde Park town
Dutchess	12	11 3602705100	Beacon city
Dutchess	12	12 3602720819	Dover town
Dutchess	12	13 3602778388	Washington town
Dutchess	12	14 3602776166	Union Vale town
Dutchess	12	15 3602740299	La Grange town
Dutchess	12	16 3602705452	Beekman town
Dutchess	12	17 3602756825	Pawling town
Dutchess	12	18 3602721996	East Fishkill town
Dutchess	12	19 3602716408	Clinton town
Dutchess	12	20 3602758695	Pleasant Valley town
Dutchess	12	21 3602701693	Amenia town
Dutchess	12	22 3602760928	Red Hook town
Dutchess	12	101 3647361	Millerton village
Dutchess	12	102 3659641	Poughkeepsie city
Dutchess	12	103 3678168	Wappinger Falls village
Dutchess	12	104 3605100	Beacon city
Dutchess	12	105 3625967	Fishkill village
Dutchess	12	106 3661346	Rhinebeck village
Dutchess	12	107 3647273	Millbrook village
Dutchess	12	108 3656814	Pawling village
Dutchess	12	109 3674023	Tivoli village
Dutchess	12	110 3660895	Red Hook village

3.12 Mode Definition File

The mode definition file defines the characteristics of each transit mode and is used by the transit network-builder and path-builder to define modes for purpose of determining fare and establishing path-building weights.

Directory Location and File Name

The Mode Definition File is a Dbase Format File and must be stored in the input directory as defined in the parameter file (ALT_parm.rsc, keyword InputDirectory):

This file is named

MODES40.DBF

Data File Structure

The mode definition file is a Dbase file and an example is shown in Figure 3-11. The structure of the mode definition file is presented in Table 3-13. The mode file contains one record for each defined mode in the model system and the Mode_ID must match the Mode_ID coded in the route header for each transit line.

Figure 3-11 Sample Mode Definition File
(part 1)

MODE_NAME	MODE_ID	TYPE	IMP_FIELD	FARE_TYPE	FARE_FIELD	FARE_CORE	HEADWAY	FARE
Commuter Rail	1	T	Transit_IVTT		2		15.00	1.50
NYC Bus 1.50	2	T	Transit_IVTT		1		15.00	1.50
NYC Bus 3.00	3	T	Transit_IVTT		1		15.00	1.50
PATH	4	T	Transit_IVTT		1		15.00	1.50
Suburban .50	5	T	Transit_IVTT		1		15.00	0.50
Suburban .75	6	T	Transit_IVTT		1		15.00	0.75
Suburban 1.00	7	T	Transit_IVTT		1		15.00	1.00
Suburban 1.25	8	T	Transit_IVTT		1		15.00	1.25
Suburban 1.50	9	T	Transit_IVTT		1		15.00	1.50
Suburban 1.75	10	T	Transit_IVTT		1		15.00	1.75
Suburban 2.00	11	T	Transit_IVTT		1		15.00	2.00
Suburban 2.50	12	T	Transit_IVTT		1		15.00	2.50
Suburban 3.00	13	T	Transit_IVTT		1		15.00	3.00
Suburban 4.00	14	T	Transit_IVTT		1		15.00	4.00
Suburban 5.00	15	T	Transit_IVTT		1		15.00	5.00
Suburban 6.00	16	T	Transit_IVTT		1		15.00	6.00
Connector	950	W						
Ferry	17	T	Transit_IVTT		2		15.00	1.50
Ferry Bus	18	T	Transit_IVTT		2		15.00	1.50
NYC Subway 1.50	19	T	Transit_IVTT		1		15.00	1.50

Figure 3-11 Sample Mode Definition File
(part 2)

XFARE	XFER_TIME	XFER_TIM2	IN_V_W	WALK_W	WAIT_W	XFER_W	DWELL_W	WAIT_WGHT	COMMENTS
0.00	4.70	4.70						0.50	
0.00	14.70	4.70						2.00	
0.00	14.70	4.70						2.00	
0.00	4.70	4.70						2.00	
0.00	14.70	4.70						2.00	
0.00	14.70	4.70						2.00	
0.00	14.70	4.70						2.00	
0.00	14.70	4.70						2.00	
0.00	14.70	4.70						2.00	
0.00	14.70	4.70						2.00	
0.00	14.70	4.70						2.00	
0.00	14.70	4.70						2.00	
0.00	14.70	4.70						2.00	
0.00	14.70	4.70						2.00	
0.00	14.70	4.70						2.00	
0.00	14.70	4.70						2.00	
0.00	4.70	4.70						2.00	
0.00	4.70	4.70						2.00	

Table 3-13. File Format for Mode Definition File

Field #	Variable Name	Type	Contents
---------	---------------	------	----------

1	MODE_NAME	Character 22	Name for this mode
2	MODE_ID	Numeric 8	MODE_ID as used in the route header for each transit line
3	TYPE	Character 22	Code indicating mode type "T" for transit or "W" for access/egress/transfer link
4	IMP_FIELD	Character 16	Not currently used
5	FARE_TYPE	Numeric 8.0	Coded as "1" for flat fare type or "2" for zone fare type
6	FARE_FIELD	Character 22	Table name in Fare Matrix File, Not currently used
7	FARE_CORE	Character 8	Table number in Fare Matrix File, Not currently used
8	HEADWAY	Numeric 10.2	Default mode headway, Not currently used
9	FARE	Numeric 10.2	Default fare for this mode in dollars (overridden by route fare unless not coded on route record)
10	XFARE	Numeric 10.2	Default transfer fare for this mode, (overridden by transfer fare in mode transfer table)
11	XFER_TIME	Numeric 10.2	Transfer time (in minutes) applied to commuter rail paths
12	XFER_TIM2	Numeric 10.2	Transfer time (in minutes) applied to other transit paths
13	IN_V_W	Character 10	Not currently used
14	WALK_W	Character 10	Not currently used
15	WAIT_W	Character 10	Not currently used
16	XFER_W	Character 10	Not currently used
17	DWELL_W	Character 10	Not currently used
18	WAIT_WGHT	Numeric 10.2	Wait time weight for this mode
19	Comments	Character 22	Comments (not used in model)
20	MIN_WAIT	Numeric 10.2	Minimum waiting time for this mode, in minutes
21	MAX_WAIT	Numeric 10.2	Maximum waiting time for this mode, in minutes

3.13 Mode Transfer Definition File

The mode transfer definition file defines the characteristics of each mode-to-mode transfer for purposes of defining transit paths.

Directory Location and File Name

The Mode Transfer File is a Dbase Format File and must be stored in the input directory as defined in the parameter file (ALT_parm.rsc, keyword InputDirectory):

This file is named

MODEXF.DBF

Data File Structure

The mode transfer file is a Dbase file and an example is shown in Figure 3-12. The structure of the mode transfer file is presented in Table 3-14. The file contains one record for each Mode_ID to Mode_ID combination.

Figure 3-12 Sample Mode Transfer File

FROM	TO	STOP	COST	FARE	
1	1	1	3	0.00	
1	1	2	3	0.80	
1	1	3	3	0.80	
1	1	4	3	0.80	
1	1	5	3	0.80	
1	1	6	3	0.80	
1	1	7	3	0.80	
1	1	8	3	0.80	
1	1	9	3	0.80	
1	1	10	3	0.80	
1	1	11	3	0.80	
1	1	12	3	0.80	
1	1	13	3	0.80	
1	1	14	3	0.80	
1	1	15	3	0.80	
1	1	16	3	0.80	
2	2	1	3	0.80	
2	2	2	3	0.00	0.00
2	2	3	3	0.00	1.50
2	2	4	3	0.00	
2	2	5	3	0.00	
2	2	6	3	0.00	
2	2	7	3	0.00	
2	2	8	3	0.00	
2	2	9	3	0.00	
2	2	10	3	0.00	
2	2	11	3	0.00	
2	2	12	3	0.00	
2	2	13	3	0.00	
2	2	14	3	0.00	
2	2	15	3	0.00	
2	2	16	3	0.00	
3	3	1	3	0.80	
3	3	2	3	0.00	0.00
3	3	3	3	0.00	0.00
3	3	4	3	0.00	

Table 3-14. File Format for Mode Transfer File

Field #	Variable Name	Type	Contents
1	FROM	Numeric 9	“From” MODE_ID
2	TO	Numeric 9	“To” MODE_ID
3	STOP	Numeric 8	Stop ID that this record applies to (leave as missing value)
4	COST	Numeric 9.2	Not Used
5	FARE	Character 9	Fare in dollars for this transfer combination

3.14 Fare Matrix File

The fare matrix file defines zone-based fares.

Directory Location and File Name

The Mode Definition File is a TransCAD Matrix File and must be stored in the input directory as defined in the parameter file (ALT_parm.rsc, keyword InputDirectory):

This file is named

FARE_MTX.MTX

Data File Structure

This file is a TransCAD matrix file organized with each row and column representing each “from” fare zone and each “to” fare zone, respectively. Each cell contains the desired fare in dollars. The file contains one table for each mode requiring a zone fare. The tables are as follows:

1. Ferry
2. Long Island Rail Road
3. Metro-North Railroad
4. New Jersey Transit commuter rail

The existing fare matrix file contains additional tables for various bus modes that were required for earlier versions of TransCAD but are not used with the current implementation.

3.15 Transit Market Segment Files

The segment file defines market segments that are used by the MTA RTFM mode split model but are not used or required in the NYMTC Model.

Directory Location and File Name

The Mode Definition File is a Dbase Format File must be stored in the input directory as defined in the parameter file (ALT_parm.rsc, keyword InputDirectory):

This file is named

ALT_SEGMENT.ASC

Where:

ALT_ = the alternative name for the specific network

Data File Structure

3.16 Manhattan Sidewalk Network

The Manhattan sidewalk network defines the CBD walk network to provide complete access from each zone to all nearby transit services.

Directory Location and File Name

The sidewalk network is a TransCAD geographic import file (.GEO) and must be stored in the input directory as defined in the parameter file (keyword InputDirectory):

This file is named

ALT_SIDEWALK.GEO

Where:

ALT_ = the alternative name for the specific network

Data File Structure

This file is stored in comma delimited format with the following fields:

Table 3-14. File Format for TAZ Locality File

Field	Variable Name	Type	Contents
1	ID	I*4	Sidewalk link ID number
2	Direction	I*4	Generally 0 to indicate walking is possible in both directions
3	A-Node	I*4	Node ID for beginning of link
4	B-Node	I*4	Node ID for ending of link
5	Ncoords	I*4	Number of coordinates
6+	Coordinates	R*8	Longitudes and Latitudes

4. Alternative-Independent Input Data

4.1 TAZ Edge File

The TAZ edge file describes each zone in the modeling region and is used by the TRAN_CON program (in conjunction with transit stop locations) to identify the stops that are within walking distance of each zone.

Directory Location and File Name

The TAZ Edge File name and location is directly specified in the parameter file (ALT_parm.rsc, keyword TAZFile):

Data File Structure

A portion of the NYMTC TAZ Edge File is presented in Figure 3-8. Table 3-8 presents the file record format.

The TAZ edge file contains one record for each line segment that comprise the zone boundary. In this case, a line segment is generally defined as that section of a zone boundary that borders a single zone. For example, in the NYMTC zone system, zone 1 borders zones 2, 7, 145, and the Hudson River (no zone). Therefore, four zone segments are defined. The source of this information is the original TAZ area geographic file. TransCAD can be used to convert the area database into a series of line segments using the command “Tools>Utilities>Line/Area Conversion>Create Lines”. These line segments can be converted to ASCII format using the export tools command. A simple program can be created to associated data representing the zone numbers (left side and right side) to create the appropriate edge file records.

Figure 3-8. Sample TAZ Edge File

1	2	3	4	5	6	7	8	9	10
0	0	0	0	0	0	0	0	0	0
1	2	-74014555	40711196	-74018138	40711718				
1	6	-74016833	40705454	-74016856	40705174	-74016856	40705075	-74017159	
1	16	-74017724	40704666	-74017658	40704831	-74018611	40704669	-74018654	40704817
1	11	-74014555	40711196	-74014709	40710735	-74014928	40710159	-74015103	40709748
2	2	-74013174	40719443	-74012568	40719408				
2	3	-74012568	40719408	-74012965	40717730	-74013010	40717516		
2	8	-74013010	40717516	-74013208	40716841	-74013230	40716775	-74013406	40716002
2	2	-74013892	40713977	-74014555	40711196				
2	2	-74014555	40711196	-74018138	40711718				
2	9	-74018138	40711718	-74017925	40712374	-74017060	40712273	-74016665	40713557
2	22	-74027877	40684782	-74027805	40686379	-74021807	40692243	-74021217	40693345
3	4	-74013505	40690968	-74013054	40690160	-74013008	40690802	-74013505	40690968
4	14	-74014498	40705020	-74014786	40703505	-74014767	40703028	-74014703	40702813

Table 3-8 TAZ Edge File Format

Columns	Variable Name	Type	Contents
1-10	Zone ID	I*4	Zone ID for this edge line segment
11-20	NPTS	I*4	Number of points in this edge record
21-30	Longitude1	I*4	Longitude for point 1 in millionths of degrees
31-40	Latitude1	I*4	Latitude for point 1 in millionths of degrees
41-50	Longitude2	I*4	Longitude for point 2 in millionths of degrees
51-60	Latitude2	I*4	Latitude for point 2 in millionths of degrees
61-70	Longitude3	I*4	Longitude for point 3 in millionths of degrees
71-80	Latitude3	I*4	Latitude for point 3 in millionths of degrees
.			
.			
.			
	LongitudeNPTS	I*4	Longitude for point NPTS in millionths of degrees
	LatitudeNPTS	I*4	Latitude for point NPTS in millionths of degrees

Table 3-10 Barrier Line File Format

Columns	Variable Name	Type	Contents
1-10	Zone ID	I*4	Zone ID for this barrier line segment
11-20	NPTS	I*4	Number of points in this barrier line record
21-30	Longitude1	I*4	Longitude for point 1 in millionths of degrees
31-40	Latitude1	I*4	Latitude for point 1 in millionths of degrees
41-50	Longitude2	I*4	Longitude for point 2 in millionths of degrees
51-60	Latitude2	I*4	Latitude for point 2 in millionths of degrees
61-70	Longitude3	I*4	Longitude for point 3 in millionths of degrees
71-80	Latitude3	I*4	Latitude for point 3 in millionths of degrees
.			
.			
.			
	LongitudeNPTS	I*4	Longitude for point NPTS in millionths of degrees
	LatitudeNPTS	I*4	Latitude for point NPTS in millionths of degrees

4.3 Connector Type Definition

The Connector Type Definition File contains information to be appended to the connector links developed by TRAN_CON as part of the process that creates the transit base link database. The NYMTC Connector Type Definitions are shown in Figure 3-10 and the file format is described in Table 3-12. It should be noted that at this point the flag definitions are NOT used in subsequent processing.

Figure 3-10 Sample Connector Type Definition File

	1	2	3	4	5	6	7
1Premium walk access				1			1
2Non-premium walk access					1		1
3Drive access						1	
4Transfer link				1	1	1	1
12Prem. & non-prem. access				1	1		1
999Dummy link				1	1	1	1
-1Premium walk egress				1		1	
-2Non-premium walk egress					1	1	
-12Prem. & non-prem. egress				1	1	1	
-3Drive egress							1

Table 3-11 Centroid Location File Format

Columns	Variable Name	Type	Contents
1-8	Con Type	I*4	Connector Type Code
9-28	Description	C*20	Connector Type Description
34-41	Premium Flag	I*4	Flag to indicate inclusion in Premium Paths (NOT USED)
42-49	Non Prem Flag	I*4	Flag to indicate inclusion in Non-premium Paths (NOT USED)
50-57	AM Drive Flag	I*4	Flag to indicate inclusion in AM Drive Paths (NOT USED)
58-65	PM Drive Flag	I*4	Flag to indicate inclusion in PM Drive Paths (NOT USED)

4.4 County Definition File

The county definition file is a TransCAD geographic file containing boundaries and related data for each county in the modeling area. The counties that make up the modeling region are presented in Table 3-12.

Directory Location and File Name

The TAZ Edge File name and location is directly specified in the parameter file (ALT_parm.rsc, keyword EmpInFile):

Data File Structure

The county file was extracted from the high-definition county file included on the Caliper data CD. A field was added to this database to hold the county code presented in Table 3-12. The final data structure is provided in Appendix 2 to this report.

Table 3-12 Counties Included in New York Modeling Region

County Code	County Name	State	Region	Zone Range		Additional Zone Range	
				Beginning	End	Beginning	End
1	New York	NY	1	1	318		
2	Queens	NY	1	319	752		
3	Bronx	NY	1	753	1,025		
4	Kings	NY	1	1,026	1,538		
5	Richmond	NY	1	1,539	1,622		
6	Nassau	NY	2	1,623	1,860		
7	Suffolk	NY	2	1,861	2,094	3,585	3,586
8	Westchester	NY	3	2,095	2,263		
9	Rockland	NY	3	2,264	2,301		
10	Putnam	NY	3	2,302	2,315		
11	Orange	NY	3	2,316	2,381		
12	Dutchess	NY	3	2,382	2,447		
13	Fairfield	CT	4	2,448	2,660		
14	Bergen	NJ	5	2,661	2,730		

15	Passaic	NJ	5	2,731	2,746
16	Hudson	NJ	5	2,747	2,904
17	Essex	NJ	5	2,905	3,122
18	Union	NJ	5	3,123	3,143
19	Morris	NJ	5	3,144	3,182
20	Somerset	NJ	5	3,183	3,203
21	Middlesex	NJ	5	3,204	3,228
22	Monmouth	NJ	5	3,229	3,281
23	Ocean	NJ	5	3,282	3,314
24	Hunterdon	NJ	5	3,315	3,340
25	Warren	NJ	5	3,341	3,363
26	Sussex	NJ	5	3,364	3,387
27	New Haven	CT	4	3,388	3,571
28	Mercer	NJ	5.00	3,572.00	3,584.00

4.5 Employment File

4.6 Area Type Definition File

4.7 NJT Park-Ride Connectors

4.8 Free Flow Highway Skims

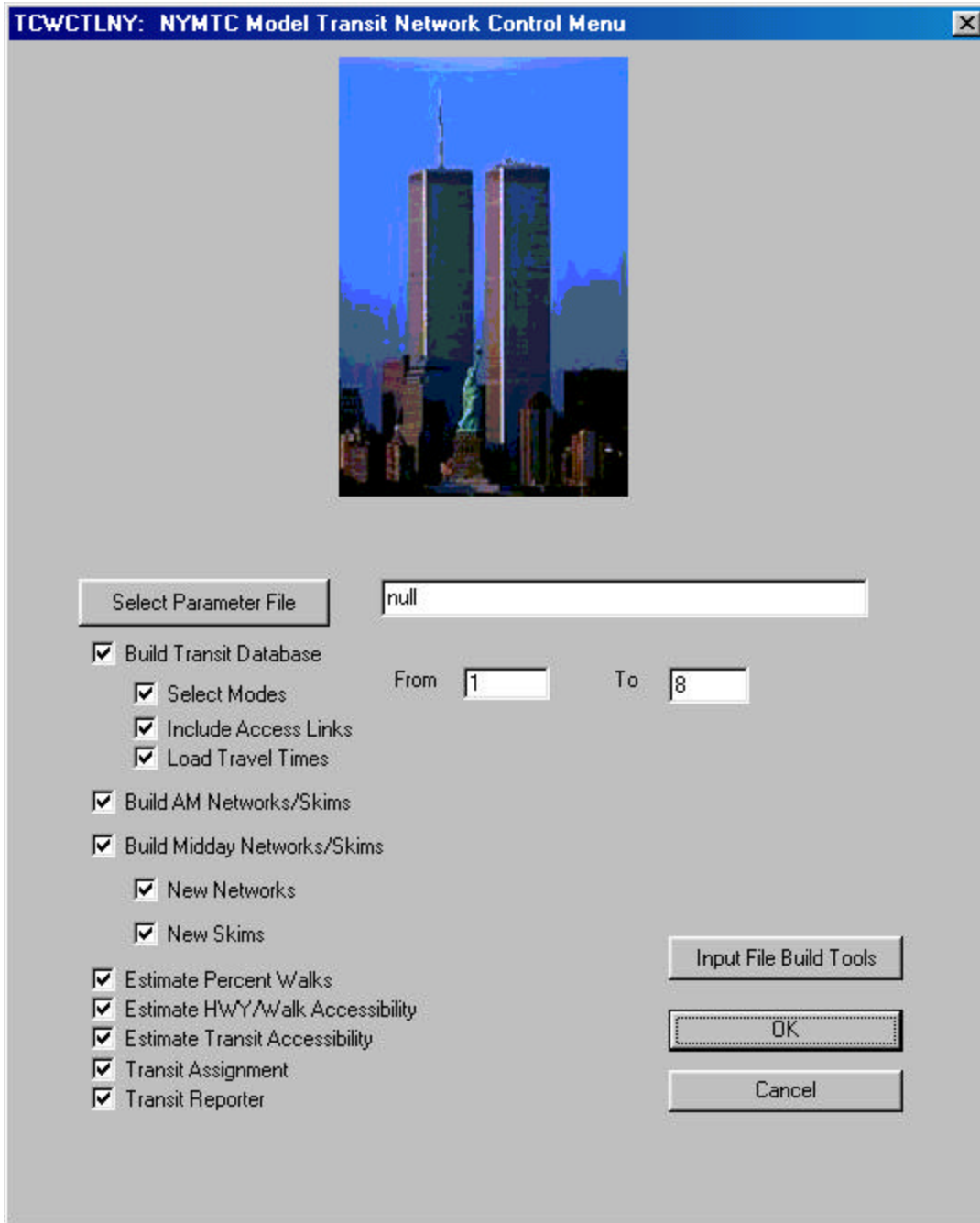
4.9 Congested Highway Skims

4.10 Congested Highway Skims (Zone-to-PTZ)

4. BPM Transit Network Processing

All transit network processing functions of the NYMTC Best Practices Model are accessed through the program TCWCTLNY.EXE. This program is designed as an temporary external interface that controls all elements of the transit processing until the final BPM user interface is developed. When this program is run, the dialog window shown in Figure 5-1 appears.

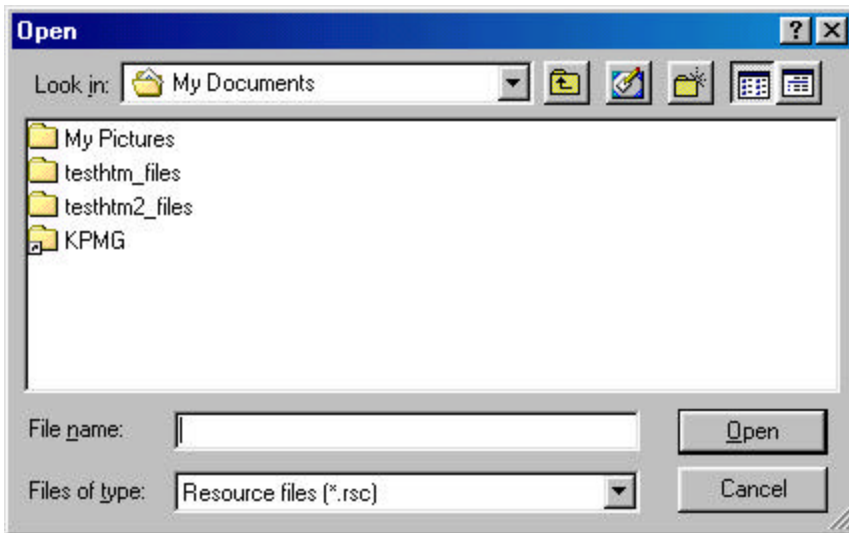
Figure 5-1 TCWCTL Main Menu



Key operations are described below.

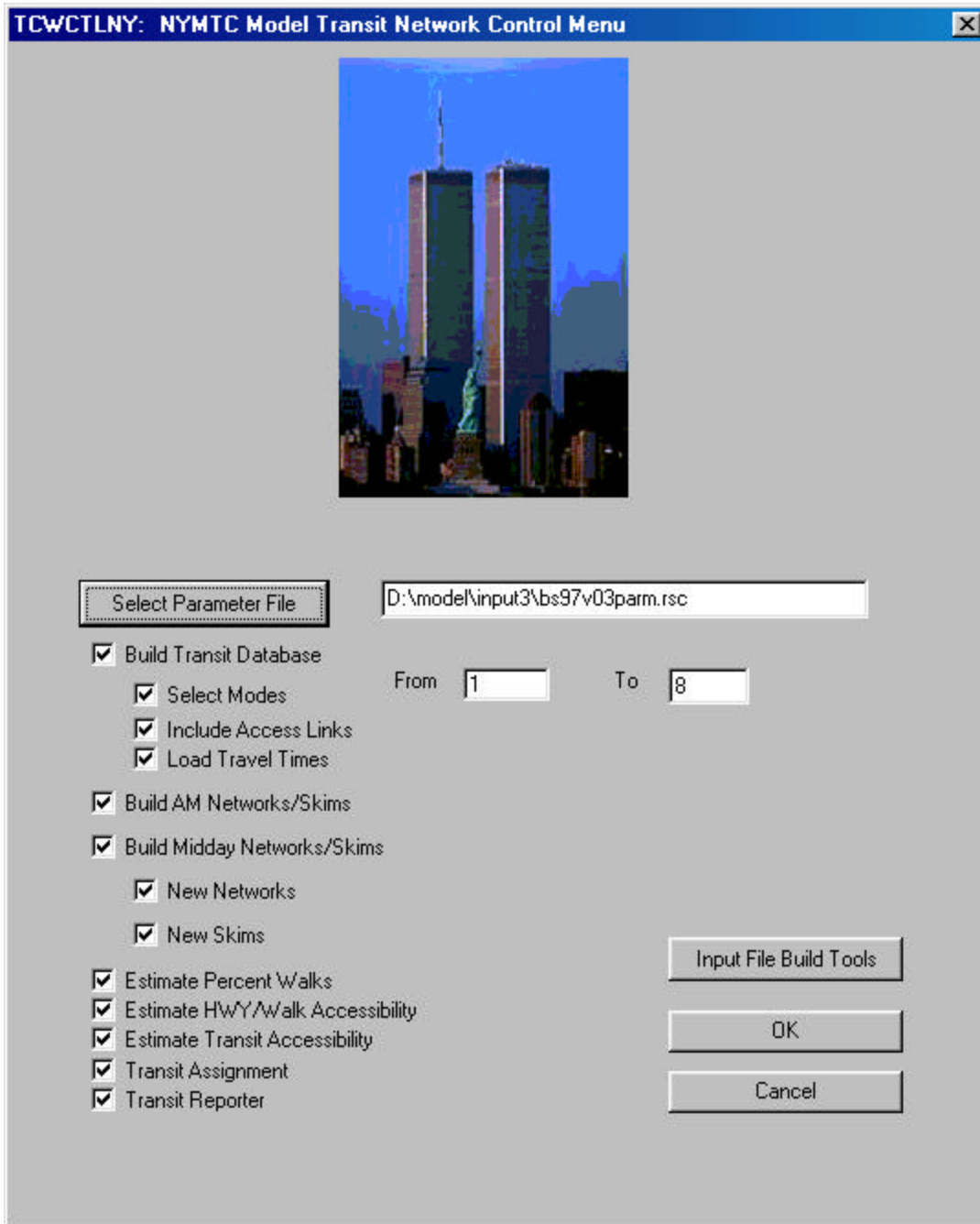
Specification of Parameter File

The parameter file that defines the alternative being run, input file names, and other parameters must be defined prior to performing any operation in TCWCTLNY. This is done by clicking on the “Select Parameter File” Button. When is is done, the following dialog window appears:



This window is the standard windows dialog for specifying an existing file name. Simply locate the correct parameter file (e.g., \model\input3\bs97v03parm.rsc) and click OK. When this is done, the parameter file name appears in the main menu as shown in Figure 5-2.

Figure 5-2 TCWCTLNY Main Menu After Selection of Parameter File



Input File Build Tools

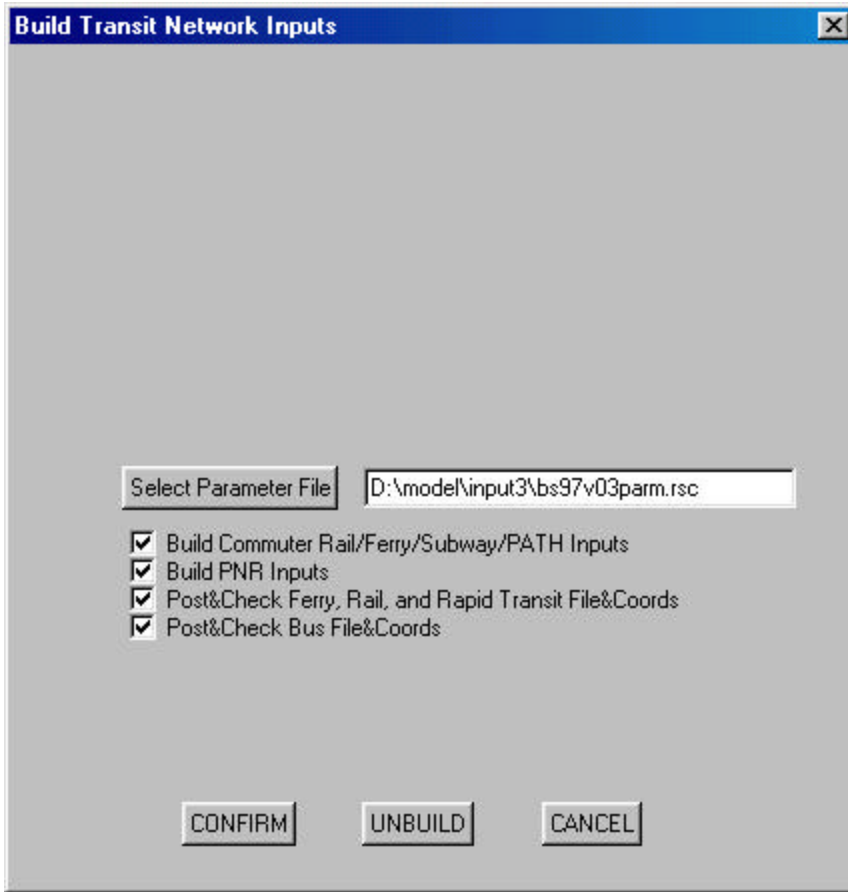
The Input File Build Tools button from the main menu, opens TransCAD and displays the dialog box shown in Figure 5-3

Tools Used for Creating Transit Network Files

The four check boxes define the potential actions that can be used to build and check the input data files:

- Build Commuter Rail/Ferry/Subway/PATH Inputs. This option generates route files from the route schedule inputs described in Section 3-2.
- Build PNR Inputs. This option generates park-and-ride links given the parking characteristics coded in the station database.
- Post and Check Ferry, Rail, and Rapid Transit File&Coordinates. This option checks the fixed guideway route and link files for internal consistency and posts correct coordinates for all stops
- Post and Check Bus File&Coords. This option checks the bus route and link files for internal consistency and posts correct coordinates for all stops.

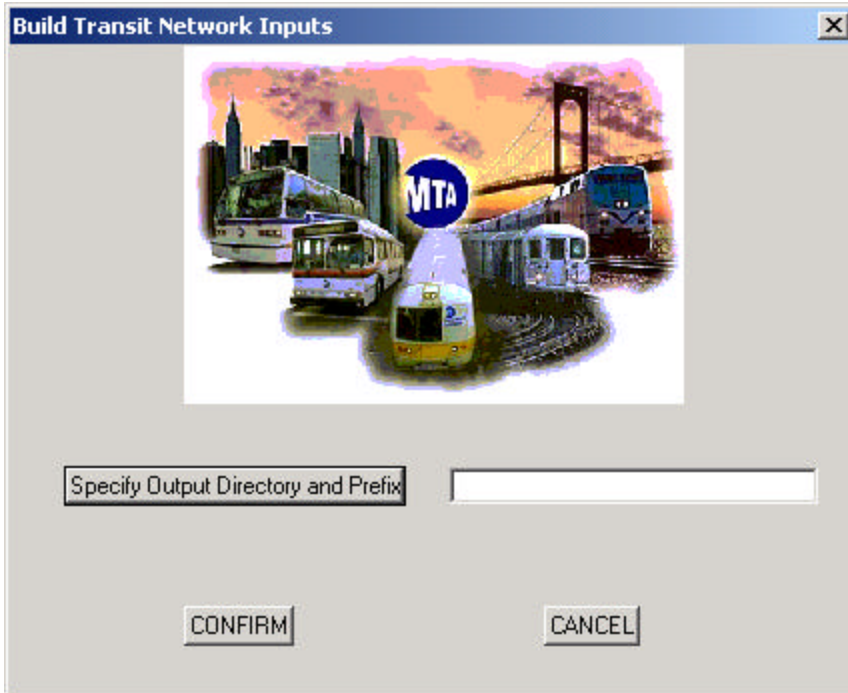
These options are run when the “Confirm” button is clicked.



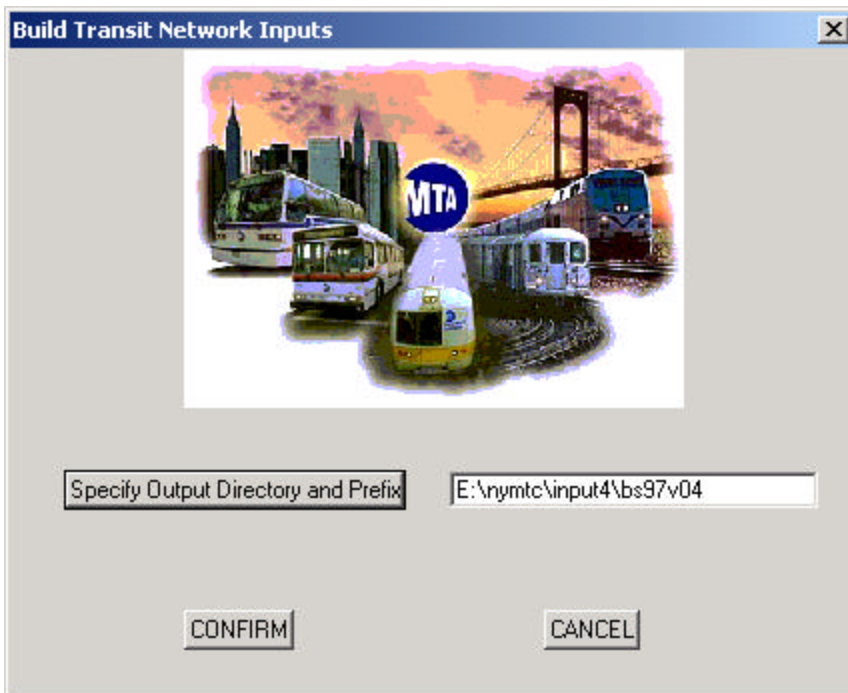
Tools for Unbuilding a Transit Network

The **unbuild** button is used to convert a “built” TransCAD network into the underlying route and link files. Unlike the other actions on the input file tools menu, this action operates on a TransCAD transit database that has already been created (first group on the main menu) and subsequently manually edited. For this reason, this action is invoked separately from the other items on this menu and occurs when the “Unbuild” button is clicked.

When **Unbuild** is selected, the following dialog window appears:



The dialog asks the user to specify an output directory and prefix for the new route and link files. The specify output directory and prefix button assist the user in locating or creating an appropriate output directory. The user, must type a new prefix (e.g., BS97V04). When this is done, the dialog displays the directory and prefix as shown below.



When “Confirm” is clicked, the program will create a new series of route and link files that represent the data coded in the route database specified in the parameter file. Only the route and link input files are created. The Unbuild function does **NOT** create the fixed guideway schedule data (.CSV) files, the station file, or any other related data. These files are typically external to TransCAD and can be copied as desired to create a new alternative scenario.

BPM Transit Processing Steps

The main menu is organized as a series of primary steps and secondary options that are active only if the related primary step is specified. The following steps are defined:

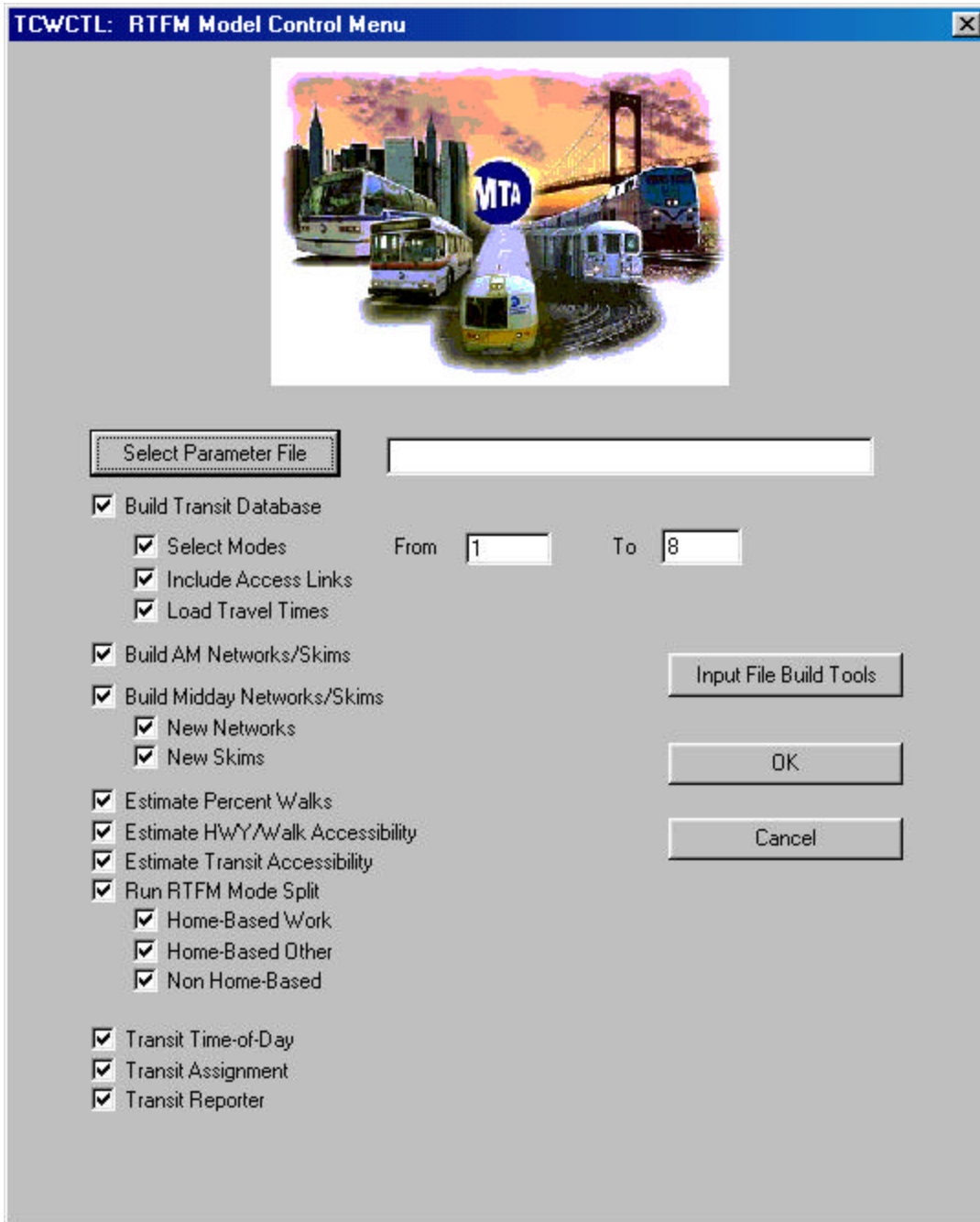
1. Build Transit Database
2. Build AM Networks/Skims // Build Midday Networks/Skims
3. Estimate Percent Walks
4. Estimate HWY/Walk Accessibility
5. Transit Assignment
6. Transit Reporter

To run the model, these steps are left checked and the user should click on the “OK” button.

5. MTA Transit Network and Model Processing

All transit network processing functions of the MTA Regional Transit Forecasting Model are accessed through the program TCWCTL.EXE. This program is designed as an external interface that controls all elements of the RTFM. When this program is run, the dialog window shown in Figure 5-1 appears.

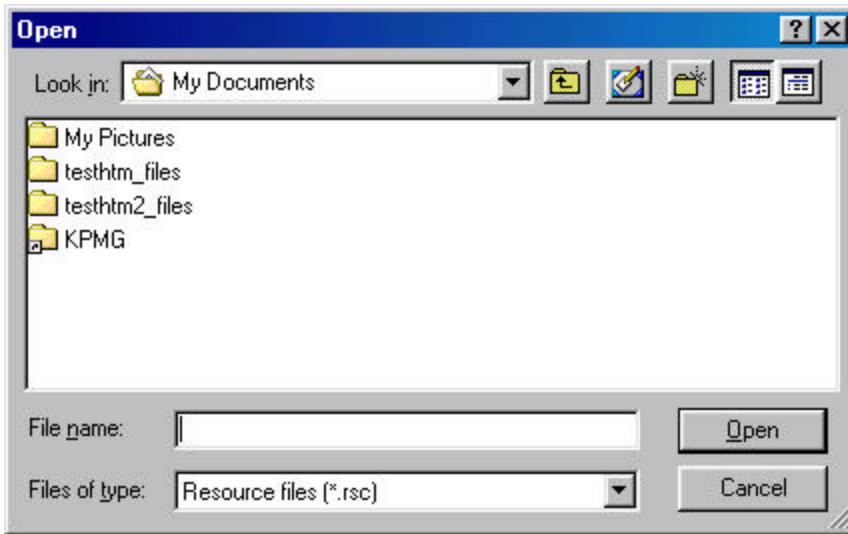
Figure 5-1 TCWCTL Main Menu



Key operations are described below.

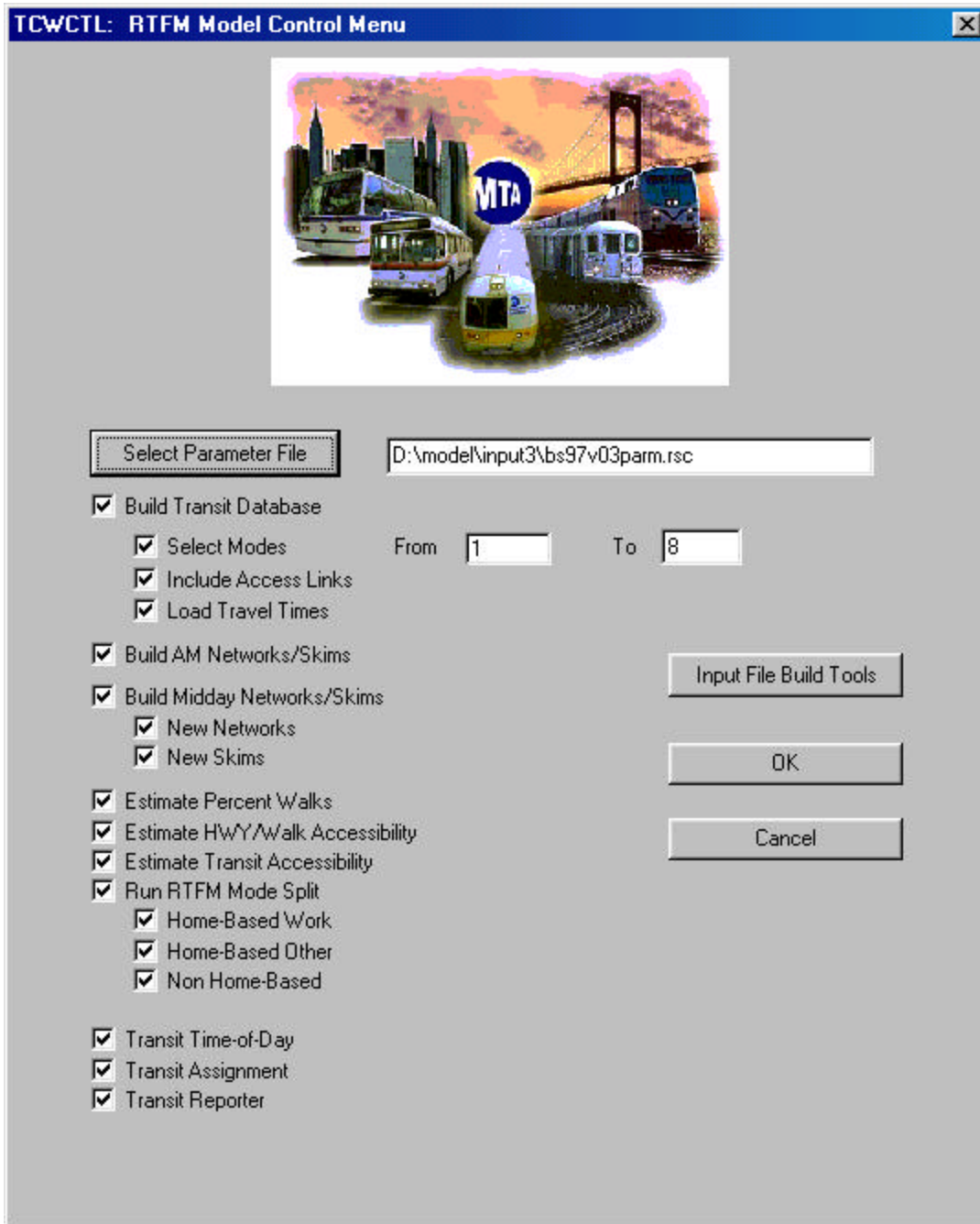
Specification of Parameter File

The parameter file that defines the alternative being run, input file names, and other parameters must be defined prior to performing any operation in TCWCTL. This is done by clicking on the “Select Parameter File” Button. When is is done, the following dialog window appears:



This window is the standard windows dialog for specifying an existing file name. Simply locate the correct parameter file (e.g., \model\input3\bs97v03parm.rsc) and click OK. When this is done, the parameter file name appears in the main menu as shown in Figure 5-2.

Figure 5-2 TCWCTL Main Menu After Selection of Parameter File



Input File Build Tools

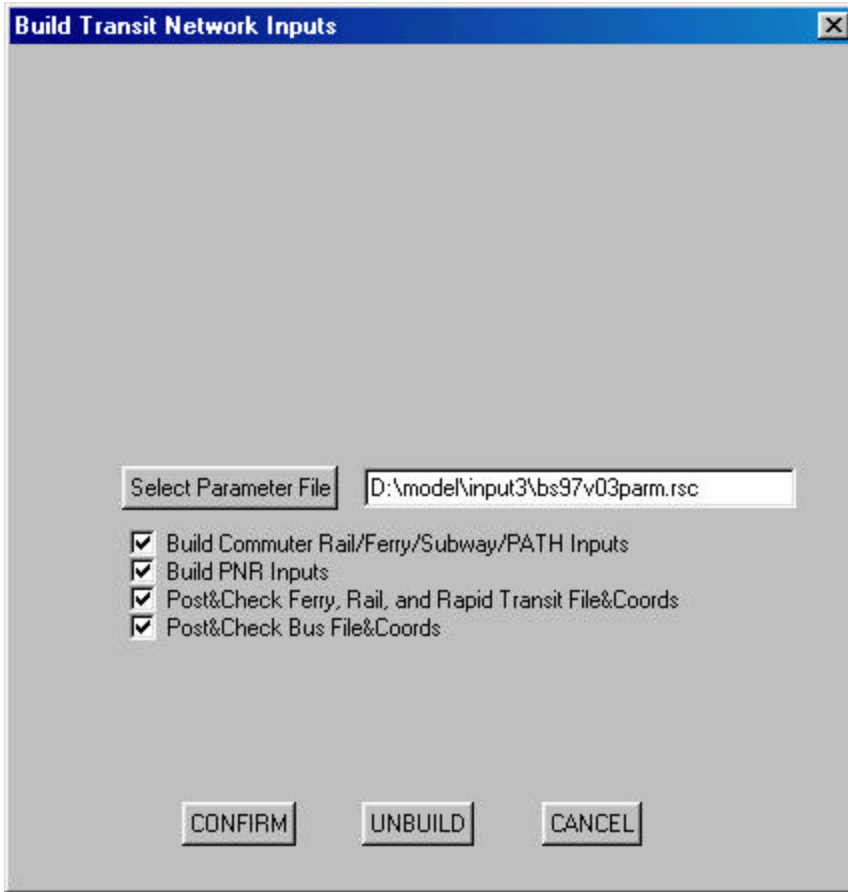
The Input File Build Tools button from the main menu, opens TransCAD and displays the dialog box shown in Figure

Tools Used for Creating Transit Network Files

5-1. The four check boxes define the potential actions that can be used to build and check the input data files:

- Build Commuter Rail/Ferry/Subway/PATH Inputs. This option generates route files from the route schedule inputs described in Section 3-2.
- Build PNR Inputs. This option generates park-and-ride links given the parking characteristics coded in the station database.
- Post and Check Ferry, Rail, and Rapid Transit File&Coordinates. This option checks the fixed guideway route and link files for internal consistency and posts correct coordinates for all stops
- Post and Check Bus File&Coords. This option checks the bus route and link files for internal consistency and posts correct coordinates for all stops.

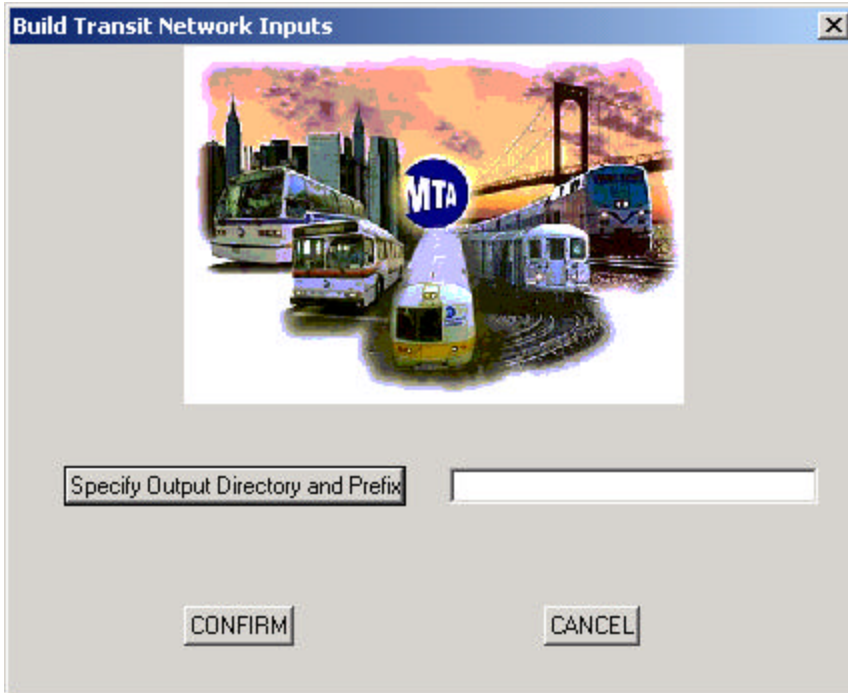
These options are run when the “Confirm” button is clicked.



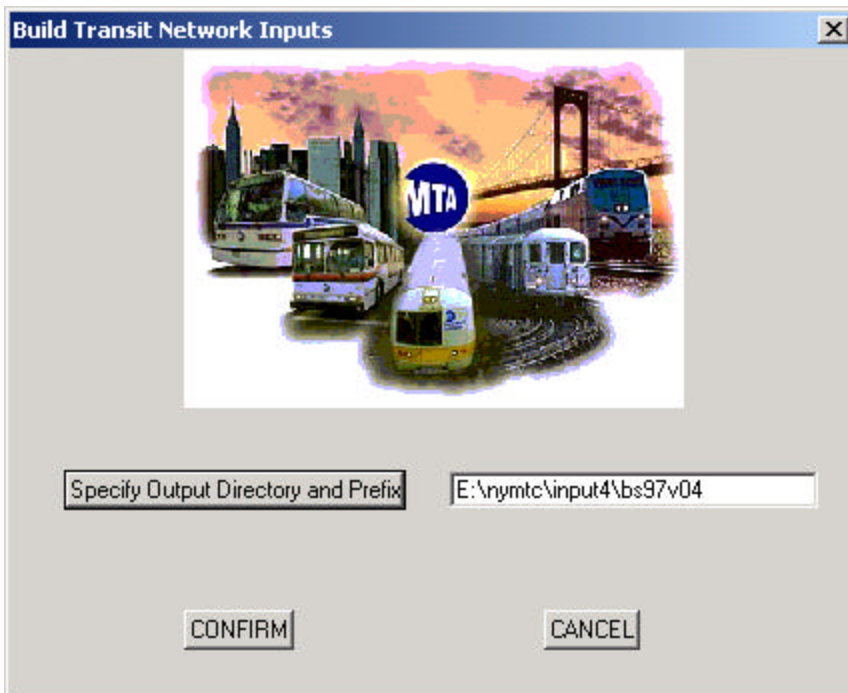
Tools for Unbuilding a Transit Network

The **unbuild** button is used to convert a “built” TransCAD network into the underlying route and link files. Unlike the other actions on the input file tools menu, this action operates on a TransCAD transit database that has already been created (first group on the main menu) and subsequently manually edited. For this reason, this action is invoked separately from the other items on this menu and occurs when the “Unbuild” button is clicked.

When **Unbuild** is selected, the following dialog window appears:



The dialog asks the user to specify an output directory and prefix for the new route and link files. The specify output directory and prefix button assist the user in locating or creating an appropriate output directory. The user, must type a new prefix (e.g., BS97V04). When this is done, the dialog displays the directory and prefix as shown below.



When “Confirm” is clicked, the program will create a new series of route and link files that represent the data coded in the route database specified in the parameter file. Only the route and link input files are created. The Unbuild function does **NOT** create the fixed guideway schedule data (.CSV) files, the station file, or any other related data. These files are typically external to TransCAD and can be copied as desired to create a new alternative scenario.

RTFM Model Steps

The main menu is organized as a series of primary steps and secondary options that are active only if the related primary step is specified. The following steps are defined:

7. Build Transit Database
8. Build AM Networks/Skims // Build Midday Networks/Skims
9. Estimate Percent Walks
10. Estimate HWY/Walk Accessibility
11. Run RTFM Mode Split
12. Transit Time-Of-Day
13. Transit Assignment
14. Transit Reporter

To run the model, these steps are left checked and the user should click on the “OK” button.

6. Transit Path Tracing

NYMTC transit databases are built using the GISDK procedure RUNBUILD3.RSC. This program reads the input files described in Chapter 3 (possibly created by the preprocessor tools described in Chapter 4) to build the complete transit database. Once RUNBUILD3 is invoked, TransCAD will build transit networks without further user intervention. This chapter describes the RUNBUILD3 and its subsidiary procedures.

5.1 RUNBUILD3

RUNBUILD3 is a GISDK resource that serves as a shell for specifying program parameters and running all of the subsidiary programs and macros the build a transit database in TransCAD. RUNBUILD3 performs the following operation:

1. Specify program parameters
2. Build TransCAD Route System Import Files (BUILD)
3. Create Megastops (MEGASTOP)
4. Create Transit Connectors (TRAN_CON)
5. Post Megastops on Stop Record (POSTMEGA)
6. Build TransCAD Base Link Import Files (BUILDB)
7. Import Base Links and Route Systems (BUILDTNET)
8. Load Transit Travel Times (LOADTIME)

Each element of RUNBUILD3 is discussed in the sections that follow.

5.1.1 Specify RUNBUILD3 Program Parameters

RUNBUILD3.RSC begins with a section in which program parameters are identified that are used throughout the remainder of the processing for building the transit database. The code that shows the specification of parameters is provided in Figure 5-1. Each parameter is described below.

- **Alternative Information**

- *AltPrefix* is the character string used to describe the scenario that is represented within the database. The prefix is used as the beginning of all filenames associated with the transit database. File prefixes should be kept as short as possible while still meaningfully and uniquely describing the scenario. Information in the prefix should include the year and nature of the scenario and will generally be between 4 and 8 characters long.
- *AltLowMode* and *AltHighMode* are numeric constants that are used to limit RUNBUILD3 to processing a range of Mode_ID codes. Generally AltLowMode should be set to 1 and AltHighMode should be set to 8 unless RUNBUILD3 is being used to build just part of the network for debugging or special visualization. AltLowMode and AltHighMode relate to the ModeID codes which are defined as follows:

- 01=Ferry
 - 02=Long Island Rail Road
 - 03=Metro-North Railroad
 - 04=New Jersey Transit Commuter Rail
 - 05=New York City Transit Subway
 - 06=PATH
 - 07=Newark City Subway
 - 08=Bus
- *AltTimePeriod* is a number constant used to limit the route building process to just those routes that exist in the specified time period. Time periods are defined as follows:
 - 0 = process all routes
 - 1 = process just routes with AM peak service (AM peak headway<999)
 - 2 = process just routes with midday service (midday headway<999)
 - 3 = process just routes with PM peak service (PM peak headway<999)
 - 4 = process just routes with evening service (evening headway<999)
 - 5 = process just routes with night service (night headway<999)
- **Skip Connectors Option**
 - *SkipCon* is a parameter used to control whether or not RUNBUILD3 invokes the transit connector program. Generally, this parameter is set to 0 which means do not skip transit connectors (i.e., *build* connectors). If set to 1, RUNBUILD3 will skip building transit connectors. This is done during debugging operations where the user wishes to see the resulting transit networks without incurring the time overhead associated with building connectors.
- **Directories**
 - *InputDirectory* describes the path in which all alternative- (scenario-) specific input files (i.e., route files, link files, etc.) are located
 - *Work Directory* describe the path in which all intermediate temporary or scratch files are created. Generally any files that remain in the work directory after the execution of this procedure may be deleted.
 - *Output Directory* describes the path in which all permanent output files associated with the alternative (scenario) are created
 - *KPMG Program Directory* describes the path in which all transit processing programs created by KPMG are located. For RUNBUILD3, KPMG programs include BUILD.EXE, BUILDB.EXE, and POSTMEGA.EXE. This directory also contains the original Fortran source code for these programs.
 - *PB Program Directory* describes the path in which all transit processing programs created by Parsons Brinckerhoff are located. For RUNBUILD3, PB programs include MEGASTOP.EXE and TRAN_CON.EXE. This directory also contains the original Fortran source code for these programs.

- **Non-Alternative Specific Filenames**
 - *TAZFile* specifies the file name for the file containing the TAZ edge records described in Chapter 3
 - *CentroidFile* specifies the file name for the file containing the centroid latitude and longitude
 - *BarrierFile* specifies the file name for the file containing the barrier line records described in Chapter 3
 - *ContypeFile* specifies the file name for the file describing definition of the various connector types
 - *InCounty* specifies the TransCAD database containing the county area definitions

- **Parameters for Calculation of Bus Travel Speeds in LOADTIME**
 - *Lowupdatemode* specifies the lowest Mode_ID that is included in the speed calculation procedures. Since fixed guideway speeds are specified on in the transit route file, this parameter should be set to the lowest bus mode id—800.
 - *Hiupdatemode* specifies the highest Mode_ID that is included in the speed calculation procedures. Since fixed guideway speeds are specified on in the transit route file, this parameter should be set to the lowest bus mode id—899.
 - *Breakpoint* specifies the distance at which buses are assumed to operate at highway speeds rather than at a lower “local” speeds. It is currently set to 3 miles. For stop-to-stop distances less than this distance, the maximum operating speed is assumed to be equal to “top speed” (defined below). For stop-to-stop distances greater than this distance, the maximum operating speed is assumed to be “maximum speed”. In both cases, the stop to stop travel time is a function of the maximum operating speed, the dwell time, and the time required to accelerate and decelerate.
 - *Nyc_arr* is an array that specifies the Census identifiers of counties that comprise the New York City core (currently New York County 36061)
 - *Ny_arr* is an array that specifies the Census identifiers of counties that comprise the remainder of New York City
 - *Li_arr* is an array that specifies the Census identifiers of counties that comprise the New York Counties outside of New York City
 - *CT_arr* is an array that specifies the Census identifiers of counties that comprise the Connecticut Counties in the NYMTC model region
 - *NJ_arr* is an array that specifies the Census identifiers of counties that comprise the New Jersey Counties in the NYMTC model region
 - *Cnty_arr* is a master array that stores the area definitions above
 - *Nyc_specs* is an array that stores the top speed (mph), acceleration rate (mphps), deceleration rate (mphps), dwell time, and maximum speed (mph) for the New York City Core. The use of top speed or maximum speed is governed by whether the stop-to-stop distance is less than or greater than or equal to “breakpoint” as discussed above.
 - *Ny_specs* is an array that stores the top speed (mph), acceleration rate (mph per second), deceleration rate (mph per second), dwell time, and maximum speed

(mph) for the remainder of New York City. The use of top speed or maximum speed is governed by whether the stop-to-stop distance is less than or greater than or equal to “breakpoint” as discussed above.

- *Li_specs* is an array that stores the top speed (mph), acceleration rate (mph per second), deceleration rate (mph per second), dwell time, and maximum speed (mph) for the New York Counties outside of New York City. The use of top speed or maximum speed is governed by whether the stop-to-stop distance is less than or greater than or equal to “breakpoint” as discussed above.
- *NJ_specs* is an array that stores the top speed (mph), acceleration rate (mph per second), deceleration rate (mph per second), dwell time, and maximum speed (mph) for the New Jersey Counties in the NYMTC modeling area. The use of top speed or maximum speed is governed by whether the stop-to-stop distance is less than or greater than or equal to “breakpoint” as discussed above.
- *CT_specs* is an array that stores the top speed (mph), acceleration rate (mph per second), deceleration rate (mph per second), dwell time, and maximum speed (mph) for the Connecticut Counties in the NYMTC modeling area. The use of top speed or maximum speed is governed by whether the stop-to-stop distance is less than or greater than or equal to “breakpoint” as discussed above.
- *Specs_arr* is the master array that combines the area-specific top speed, acceleration rates, deceleration rates, dwell time, and maximum speeds.

Figure 5-1 Sample Code Showing RUNBUILD3 Parameter Specification

```
// Alternative Information
    AltPrefix="BS97v01"
    AltLowMode=1
    AltHighMode=8
    AltTimePeriod=0
// Set Skip Con to 1 if you wish to skip generating
connectors
    SkipCon=0

// Process Input File Directory
    InputDirectory="e:\\nymtc\\inputs\\"

// Working File Directory
    WorkDirectory="e:\\nymtc\\work\\"

// Process Output File Directory
    OutputDirectory="e:\\nymtc\\newnet\\"

// KPMG Program Directory
    KPMGDir="e:\\tcwutils\\kpmgprog\\"

// PB Program Directory
    PBDir="e:\\tcwutils\\pbprogram\\"

// TAZ file
    TAZFile="e:\\nymtc\\zones\\taz_edge\\taz_edge.asc"
// Centroid file
CentroidFile="e:\\nymtc\\zones\\centroid\\centroid.asc"
// Barrier file
    BarrierFile="e:\\nymtc\\trn\\barriers\\barrier.asc"
// ConType file
    ConTypeFile="e:\\nymtc\\trn\\conlinks\\contype.asc"
// parameters for loadtime
    InCounty="e:\\nymtc\\nymtcco\\county_h.cdf"
    lowupdatemode=800
    hiupdatemode=899
    BREAKPOINT=3

//county codes for the various areas
    nyc_arr={36061} //the manhattan
    ny_arr={36005,36081,36047,36085} //the other boroughs

li_arr={36087,36071,36119,36079,36027,36059,36103} //the
suburban counties
```

```

nj_arr={34037,34041,34031,34027,34019,34021,34035,34039,340
13,34003,34017,
        34023,34025,34029} //the new jerseyans
ct_arr={9001,9005,9009} //the connecticites

dim cnty_arr[5]
cnty_arr[1]=nyc_arr
cnty_arr[2]=ny_arr
cnty_arr[3]=li_arr
cnty_arr[4]=nj_arr
cnty_arr[5]=ct_arr

//topspeed, acc, dec, dwelltime, higher speed for longer
stops*****
nyc_specs={7,1.0,1.0,0,35}
ny_specs={10,1.2,1.2,0,60}
li_specs={15,1.4,1.4,0,55}
nj_specs={20,1.6,1.6,0,55}
ct_specs={16,1.1,1.1,20,55}

dim specs_arr[5]
specs_arr[1]=nyc_specs
specs_arr[2]=ny_specs
specs_arr[3]=li_specs
specs_arr[4]=nj_specs
specs_arr[5]=ct_specs

```

5.1.2 Build TransCAD Route System Import Files (BUILD)

Build is a Fortran program that creates the transit route system files in the format required by TransCAD 3.61 GISDK function "CreateRouteSystemFromTables." These files are as follows:

1. *Route File.* This file contains route header information and contains the information found in the ROUTE records in the input route files. This data includes the Route ID, Mode, Mode ID, and Headway information. The file is formatted as a TransCAD Binary (.BIN) file and the associated data dictionary (.DCB) file. The format of the Route File is presented in Table 3-15.
2. *Link File.* This file contains information on the links that make up the alignment of the route and contains the ROUTE_ID from the applicable (i.e., prior) ROUTE record and LINK_ID and direction of travel data from the LINK record. The file is formatted as a TransCAD Binary (.BIN) file and the associated data dictionary (.DCB) file. The format of the Route File is presented in Table 3-16.

3. *Stop File.* This file contains the stop information and includes one record for each STOP record included in the input route files. The information on this record includes the active Route ID, and Link ID from the input route file. It also includes the computed Pass Count (a number representing whether the stop is on the first, second, etc. time the route traverses the particular link). BUILD copies selected information from the stop and station records and creates space for megastop and running time information to be posted by later processes. The file is formatted as a TransCAD Binary (.BIN) file and the associated data dictionary (.DCB) file. Since the binary file created by BUILD is just a temporary file until it is updated by POSTMEGA (see below), BUILD opens the binary file with the extension of “.TMP.” The final binary file is created by POSTMEGA. The format of the Route File is presented in Table 3-17.

BUILD also creates the two input files required by the MEGA STOP program:

1. *Stop Location File.* This file is sorted in longitude/latitude/route id/mile post order and provides the information shown in Table 3-18.
2. *Route Stop File.* This file provides information on the stops associated with each route as shown in Table 3-19.

Table 3-15 Data Dictionary for Working Route File For TransCAD Import

Field No.	Variable Name	Type	Contents
1	Route ID	I*4	Unique Route ID
2	Route Name	C*32	User-specified route name (from Route Record)
3	Mode	I*4	Mode (from Route Record, see Table 3-1)
4	Link Update Code	I*4	(From Route Record, Not Currently Used)
5	AM Headway	R*8	AM Peak Headway in minutes (from Route Record)
6	Mid. Headway	R*8	Midday Headway in minutes (from Route Record)
7	PM Headway	R*8	PM Peak Headway in minutes (from Route Record)
8	Eve. Headway	R*8	Evening Headway in minutes (from Route Record)
9	Night Headway	R*8	Overnight Headway in minutes (from Route Record)
10	AM Route Flag	I*4	Flag=1 if operated in AM Peak (Not Currently Used)
11	New Mode 1	I*4	Mode Code (Not Currently Used) (from Route Record, see Table 3-1)
12	New Mode 2	I*4	Mode Code (Not Currently Used) (from Route Record, see Table 3-1)
13	Mode ID	I*4	Mode ID for fare processing (from Route Record, see Table 3-1)

Table 3-16 Data Dictionary for Working Link File For TransCAD Import

Columns	Variable Name	Type	Contents
1	Route ID	I*4	Route ID for the Route (from previous Route Record)
2	Link ID	I*4	LINK ID from Base Link Geographic File for beginning of route (from Link Record)
3	Link Direction	C*1	"+" to indicate that link is traversed in same direction as the link topological direction "-" to indicate that link is traversed in the opposite direction as the link topological direction (from link record)

Table 3-17 Data Dictionary for Working Stop File For TransCAD Import

Columns	Variable Name	Type	Contents
Variables Assigned in BUILD			
1	Stop ID	I*4	Stop ID assigned sequentially by BUILD
2	Route ID	I*4	Route ID for the Route
3	Link ID	I*4	LINK ID for the Link where the stop is located
4	Pass Count	I*4	Pass number (I.e., is stop on first time LINK ID is traversed, the second time, etc.) calculated by BUILD
5	Milepost	R*8	This value is set to 0.0, it is calculated by TransCAD
6	Longitude	I*4	Longitude for stop (from stop record)
7	Latitude	I*4	Latitude for stop (from stop record)
8	Old ID	I*4	Old ID (not used, always=0)
9	Node ID	I*4	Node ID (not used, always=0)
10	Stop Label	C*24	Stop Label (not used, always blank)
11	Station Name	C*24	Station Name (from stop record)
12	Station Code	I*4	Station Code (not used)
13	PTZ ID	I*4	Station PTZ (if any) (from stop record)
14	Location	I*4	Location (not used, always=0)
15	Old PTZ ID	I*4	Old PTZ Location (not used, always=0)
16	Farezone	I*4	Farezone for stop (from stop record unless PTZ then from station database)
Variables Assigned in PostMega			
17	Mega ID	I*4	Mega Stop ID
18	Rep SID	I*4	Representative Stop ID
19	Long Mega	I*4	Longitude for Mega Stop
20	Lat Mega	I*4	Latitude for Mega Stop
21	Rep Long	I*4	Longitude for Representative Stop ID
22	Rep Lat	I*4	Latitude for Representative Stop ID
23	Merge ID	I*4	Assigned value used to assure network connectivity
Additional Variables Assigned in BUILD			
24	Run Time	I*4	Running time from route record (from stop record)
25	PNR ID	I*4	PNR ID (from stop record)
26	Old Stop ID	I*4	Old Stop ID (not used, from Stop Record)
Variables Assigned in LOADTIME			
24	Bus IVTT	R*8	Local Bus In -vehicle time
24	XBus IVTT	R*8	Express Bus In -vehicle time
24	Rail IVTT	R*8	Rail (Subway, PATH, Nwk Subway) In -vehicle time
24	CR IVTT	R*8	Commuter Rail In -vehicle time
24	Ferry IVTT	R*8	Ferry In -vehicle time

Table 3-18. Stop Location File Passed to Mega Stop Program

Columns	Variable Name	Type	Contents
1-10	Stop ID	I*4	Stop ID
11-20	Longitude	I*4	Longitude in millionths of degrees
21-30	Latitude	I*4	Latitude in millionths of degrees
31-40	Mode	I*4	Mode for route serving this stop 1=Local Bus (all operators, excluding ferry shuttles) 2=Limited Bus (NYCT bus) 3=Express Bus (all operators) 4=Commuter Rail (LIRR, MNR, NJT) 5=Subway (NYC Subway, PATH, Newark City Subway) 7=Ferry/Tram/Ferry Shuttlebus
41-50	PTZ ID	I*4	PTZ ID (if any)
51-60	PNR ID	I*4	PNR ID (if any)
61-70	Route ID	I*4	Route ID for route serving this stop
71-84	Milepost	F14.8	Milepost for this route

Table 3-19. Route Stop File Passed to Mega Stop Program

Columns	Variable Name	Type	Contents
1-10	Route ID	I*4	Route ID
11-20	Mode	I*4	Mode for this route 1=Local Bus (all operators, excluding ferry shuttles) 2=Limited Bus (NYCT bus) 3=Express Bus (all operators) 4=Commuter Rail (LIRR, MNR, NJT) 5=Subway (NYC Subway, PATH, Newark City Subway) 7=Ferry/Tram/Ferry Shuttlebus
21-30	Stop ID	I*4	Stop ID

This program is automatically invoked by RUNBUILD3.RSC and should not require user intervention to operate properly.

BUILD performs the following operations:

- *Read Parameters.* The following parameters are passed from RUNBUILD3 to BUILD:
 1. *Fileprefix.* This is the Altprefix from RUNBUILD3 and is used as a prefix for all file names.
 2. *InputDirectory.* This is the InputDirectory from RUNBUILD3 and contains the location of all user-created input files.
 3. *FileWkDirectory.* This is the WorkDirectory from RUNBUILD3 and contains the location of all working (output) files created by RUNBUILD3.
 4. *StartMode.* This is AltLowMode from RUNBUILD3 and contains the first mode to process for building the transit network. It is used to both speed execution times and to allow building of portions of the transit network as an aid for visualization and network error-checking
 5. *EndMode.* This is the AltHiMode from RUNBUILD 3 and contains the last mode to process for building the transit network.
 6. *TimePeriod.* This is AltTimePeriod from RUNBUILD3 and contains the time period to be built. If coded as “0”, then all time periods are included.

- *Build File Names.* The actual file names are built from the above parameters as follows:
 1. *Input Route Data.* FileInDirectory+FilePrefix+RTE
 2. *Input Station Data.* FileInDirectory+FilePrefix+STATIONS.ASC
 3. *Output Route File.* FileWkDirectory+FilePrefix+RTER.BIN/DCB
 4. *Output Route Links.* FileWkDirectory+FilePrefix+RTEL.BIN/DCB
 5. *Output Route Stops.* FileWkDirectory+FilePrefix+RTES.BIN/DCB
 6. *Output Stop Location File (to MEGASTOP).*
FileWkDirectory+FilePrefix+STPLOC.ASC
 7. *Output Route-Stop File (to MEGASTOP).*
FileWkDirectory+FilePrefix+RTSTPASC

- *Create Dictionary files.* Build creates the dictionary files (.DCB) for all binary (.BIN) files identified above

- *Create TransCAD BIN Files.* BUILD reads the input route files and converts them into the TransCAD Binary format required for “CreateRouteSystemfromTables” command. The source of each data element is provided in Tables 3-15, 3-16, and 3-17.

- *Create Mega Stop Interface Files.* BUILD creates the appropriate input data files for the MEGASTOP program as defined in Tables 3-18 and 3-19.

5.1.3 Create Megastops (MEGASTOP)

The MEGASTOP program is used to identify collections of TransCAD stops that are in close proximity to one-another and to provide a single stop identification code (MEGA ID) that is used to define a single physical stop. This program and related documentation were developed by Parsons Brinckerhoff and are included (for the moment) in their technical documentation on transit networks.

5.1.4 Create Transit Connectors (TRAN_CON)

The TranCON program is used to create a series of links in the base layer that represent:

1. Opportunities to walk from zone centroids to Mega Stops
2. Opportunities to drive from zone centroids to PTZs and PNR nodes
3. Opportunities to transfer between megastops (particularly adjacent megastops representing PTZs and bus nodes).

This program and related documentation were developed by Parsons Brinckerhoff and are included (for the moment) in their technical documentation on transit networks.

5.1.5 Post Megastops on Stop Record (POSTMEGA)

POSTMEGA is a Fortran program that updates the working stop file that was created in BUILD for eventual use by the by TransCAD 3.61 GISDK function "CreateRouteSystemFromTables." This update posts the results of the mega stop identification process (MEGASTOP) to the stop file.

POSTMEGA reads the following input files:

1. *Stop File.* POSTMEGA reads the temporary stop file created by BUILD. The file format is as described in Table 3-17.

is intended e files are as follows:

Route File. This file contains route header information and contains the information found in the ROUTE records in the input route files. This data includes the Route ID,

5.1.6 Build TransCAD Base Link Import Files (BUILDB)

5.1.7 Import Base Links and Route Systems (BUILDTNET)

5.1.8 Load Transit Travel Times (LOADTIME)

6. Procedures to Build Transit Skims

6. Appendices

Appendix 1 Sample Parameter File

```

1           2           3           4           5           6           7           8           9           10          11
-----0-----0-----0-----0-----0-----0-----0-----0-----0-----0-----0-----0-----

//e:\tcw2\utils\
//c:\progra-1\transcad4\gisdk\toolbox\
Macro "LoadParm"
// Section I: Alternative Name
Global AltPrefix

// Section II: Directory Locations
Global InputDirectory,WorkDirectory,OutputDirectory,ProgramDirectory,NJTPNRDir

// Section III: Specialized Input File Names
Global TAZFile,BarrierFile,ConTypeFile,ModeFile,InCounty,InZone,EmpInFile,RetEmpInFile,AreaTypeFile
Global ManhattanCBD

// Section IV: Network-Building Parameters
Global AltLowMode,
AltHighMode,AltTimePeriod,SkipCon,lowupdatemode,hiupdatemode,BREAKPOINT,PathWeight,cnty_arr,specs_arr,NetType,qns_man_s
pd
Global PrintFlags

// Section V: Skimming and Assignment
Global Period,TransitNetFilename,ModeXFFile,FareMTXFile,walkspeed,WalkSpeed
Global TrnAMCapFactor,TrnMIDCapFactor,TrnPMCcapFactor,TrnEVECapFactor,TrnNTCapFactor
Global CRAMCapFactor,CRMIDCapFactor,CRPMCcapFactor,CREVECapFactor,CRNTCapFactor
Global TrnAssnAlpha,TrnAssnBeta,TrnAssnLimit,TrnAssnFrac
Global usemodechoicetable

// Section VI: Highway Network Information
Global FFSkimInFile,ConSkimInFile,FFDistTab,FFTollTab,FFTTimeTab,ConDistTab,ConTimeTab,ConTollTab
Global ConPTZSkimInFile

// Section VII: Trip Tables
Global TripMatrixPrefix

// Section XX: DERIVED FILE NAMES -- NEED NOT MODIFY
Global
InAsciiLinkPrefix,InAsciiRtePrefix,InLinkNodes,InLinkLinks,InRouteRoute,InRouteLinks,InRouteStop,CentroidFile,TempDatab
aseFilename,LinkDatabaseFilename,RouteSystemFilename
Global BufferFile,WalkPctFile,FFSkimFile,ConSkimFile,WalkAccFile,WalkAccPrn
Global HwyAccFile,HwyAccPrn,EmpFile,RetEmpFile,AreaTypeRowFile,AreaTypeColFile
Global HwyAccRetFile,WalkAccRetFile

//-----
// Section I: Alternative Name

// Alternative Information
AltPrefix="BS97v03"

//-----
// Section II: Directory Locations

// Process Input File Directory
InputDirectory="e:\nymtc2\input3\\"

// Working File Directory
WorkDirectory="e:\nymtc2\work\\"

// Process Output File Directory
OutputDirectory="e:\nymtc2\newnet4\\"

// Program Directory
ProgramDirectory="e:\tcw2\programs\\"

// NJT PNR Connector Directory
NJTPNRDir="e:\nymtc2\njtpnr\\"

//-----
// Section III: Specialized Input File Names

// TAZ TransCAD Geographic File
InZone="e:\nymtc2\zones\taz\bpz91209.dbd"
// TAZ Edge File
TAZFile="e:\nymtc2\zones\taz_edge\taz_edge.asc"
// Barrier file
BarrierFile="e:\nymtc2\trn\barriers\barrier.asc"
// ConType file
ConTypeFile="e:\nymtc2\trn\conlinks\contype.asc"
// Mode File
ModeFile=InputDirectory+"modes40.dbf"
// County Definition File
InCounty="e:\nymtc2\nymtcco\county_h.cdf"
// Employment File
EmpInFile="e:\nymtc2\zones\taz\se_data\allind96.ASC"
// Retail Employment File

```

```

RetEmpInFile="e:\nymtc2\zones\taz\se_data\retemp96.ASC"
// Area Type Definition File
AreaTypeFile="e:\nymtc2\zones\taz\areatype\ATYPE.ASC"
// Manhattan CBD Definition
ManhattanCBD="e:\nymtc2\zones\taz\manhattan.dbd"

//-----
// Section IV: Network-Building Parameters

// Alternative Building Information
AltLowMode=1
AltHighMode=8
AltTimePeriod=0
// Set Skip Con to 1 if you wish to skip generating connectors
SkipCon=0

// parameters for loadtime
lowupdatemode=2
hiupdatemode=18
BREAKPOINT=0.30

Dim PathWeight[3]
// PathWeight [1][1] Path1, bus
// PathWeight [1][2] Path1, ltd bus
// PathWeight [1][3] Path1, x bus
// PathWeight [1][4] Path1, cr
// PathWeight [1][5] Path1, subway
// PathWeight [1][6] Path1, ??
// PathWeight [1][7] Path1, ferry

// bus path
PathWeight[1]={0.8, 0.8, 0.8, 1.0, 1.0, 0.0, 1.0}
// subway/ferry path
PathWeight[2]={1.4, 1.4, 1.4, 1.0, 1.0, 0.0, 1.0}
// PathWeight[2]={1.2, 1.2, 1.2, 1.0, 1.0, 0.0, 0.5}
// rail path
orig PathWeight[3]={1.5, 1.5, 1.5, 0.8, 1.5, 0.0, 1.0}
PathWeight[3]={1.2, 1.5, 1.5, 0.8, 1.5, 0.0, 1.0}

//county codes for the various areas
nyc_arr={36061}//the manhattan
ny_arr={36005,36081,36047,36085}//the other boroughs
li_arr={36087,36071,36119,36079,36027,36059,36103}//the suburban counties
nj_arr={34037,34041,34031,34027,34019,34021,34035,34039,34013,34003,34017,
34023,34025,34029}//the new jerseyans
ct_arr={9001,9005,9009}//the connecticities

dim cnty_arr[5]
cnty_arr[1]=nyc_arr
cnty_arr[2]=ny_arr
cnty_arr[3]=li_arr
cnty_arr[4]=nj_arr
cnty_arr[5]=ct_arr

//topspeed, acc, dec, dwelltime, higher speed for longer stops*****
// nyc_specs={12,1.0,1.0,20,35}
// ny_specs={10,1.2,1.2,0,60}
nyc_specs={10,1.0,1.0,20,18}
ny_specs={10,1.2,1.2,0,18}
li_specs={15,1.4,1.4,0,55}
nj_specs={20,1.6,1.6,0,55}
ct_specs={16,1.1,1.1,20,55}
// special override for queens to manhattan bus speeds
qns_man_spd=13.
qns_man_dly=10.

dim specs_arr[5]
specs_arr[1]=nyc_specs
specs_arr[2]=ny_specs
specs_arr[3]=li_specs
specs_arr[4]=nj_specs
specs_arr[5]=ct_specs

//
// Network type "P"=PB/Trancon, "K"=KPMG/Makecon
//
NetType="P"

// Which Modes to print detail
PrintFlags={1,1,1,1,1,1,1}

//-----
// Section V: Skimming and Assignment

TransitNetFilename=OutputDirectory+AltPrefix+"tnet"
ModeFile=InputDirectory+"modes40.dbf"
ModeXFile=InputDirectory+"modexf.dbf"
FareMTXFile=InputDirectory+"Fare_mtx.mtx"
// walkspeed is minutes per mile for network and is a number
walkspeed=30.
// WalkSpeed is mph for walking for walk accessibility and is a string
WalkSpeed="2."

TrnAMCapFactor=1/0.425
TrnMIDCapFactor=0
TrnPMCapFactor=0

```

```

TrnEVECapFactor=0
TrnNFCapFactor=0

CRAMCapFactor=1/0.487
CRMIDCapFactor=0
CRPMCcapFactor=0
CREVECapFactor=0

CRNTCapFactor=0

// Transit Assignment Parameters
TrnAssnAlpha=0.15
TrnAssnBeta=3.5
TrnAssnLimit=9.9
TrnAssnFrac={0.2,0.1,0.1,0.1,0.1,0.1,0.1,0.1,0.1,0.1,0.1,0.1}
usemodechoicetable=1

//-----
// Section VI: Highway Network Information
FFSkimInFile="e:\nymtc2\hwynet\run27freeflowskims.mtx"
ConSkimInFile="e:\nymtc2\hwynet\run27congestedskims.mtx"
ConPTZSkimInFile="e:\NYMTC2\hwypztz\run27cong(taz2ptz).TAB"

FFDistTab="Distance (Skim)"
FFTollTab="TOLL (Skim)"
FFTTimeTab="Free-Flow Time (Skim)"
ConDistTab="Length (Skim)"
ConTollTab="TOLL (Skim)"
ConTimeTab="CongestedTime (Skim)"

//-----
// Section VII: Trip Tables
TripMatrixPrefix="e:\nymtc2\triptabs\bs90"

// Section XX: DERIVED FILE NAMES -- NEED NOT MODIFY

// BUILD FILE NAMES
// Input ASCII Prefix for link files
InAsciiLinkPrefix=InputDirectory+AltPrefix+"LNK"
// Input ASCII Prefix for route files
InAsciiRtePrefix=InputDirectory+AltPrefix+"RTE"
// Bin files from Fortran to TransCad
/// Nodes
InLinkNodes=WorkDirectory+AltPrefix+"basenode"
/// Links
InLinkLinks=WorkDirectory+AltPrefix+"baselink"
/// Transit Routes
InRouteRoute=OutputDirectory+AltPrefix+"rter.bin"
/// Transit Route Link List
InRouteLinks=OutputDirectory+AltPrefix+"rtel.bin"
/// Transit Route Stop List
InRouteStop=OutputDirectory+AltPrefix+"rtes.bin"
// Centroid file
CentroidFile=WorkDirectory+AltPrefix+"centroid.asc"
/// Temporary Data File
TempDatabaseFilename=WorkDirectory+"temp.dbd"
// BufferFile
BufferFile=WorkDirectory+AltPrefix+"buff"
//

// Output File Names
// Base Links
LinkDatabaseFilename=OutputDirectory+AltPrefix+"_lnk.dbd"
// Route System
RouteSystemFilename=OutputDirectory+AltPrefix+"_rte"
// Walk Percent File
WalkPctFile=OutputDirectory+AltPrefix+"walkpct"

FFSkimFile=OutputDirectory+AltPrefix+"HwyFFSkim"
ConSkimFile=OutputDirectory+AltPrefix+"HwyConSkim"

WalkAccFile=OutputDirectory+AltPrefix+"WalkAccess.ten"
WalkAccRetFile=OutputDirectory+AltPrefix+"WalkRetAccess.ten"
WalkAccPrn=OutputDirectory+AltPrefix+"WalkAccess.prn"
HwyAccFile=OutputDirectory+AltPrefix+"HwyAccess.ten"
HwyAccRetFile=OutputDirectory+AltPrefix+"HwyRetAccess.ten"
HwyAccPrn=OutputDirectory+AltPrefix+"HwyAccess.prn"
EmpFile=OutputDirectory+AltPrefix+"Employment.tab"
RetEmpFile=OutputDirectory+AltPrefix+"RetailEmp.tab"
AreaTypeRowFile=OutputDirectory+AltPrefix+"atyper.tab"
AreaTypeColFile=OutputDirectory+AltPrefix+"atypec.tab"

EndMacro

```

Appendix 2 County File Data Structure

#	Field Name	Data Type	Width	Dec	Index
1	ID	Integer	12		0 Yes
2	AREA	Real	10		2
3	DATA	Integer	12		0
4	FIPSCODE	Integer	12		0
5	Name	String	34		0
6	CountyCode	Real	12		0
7	Population	Integer	10		0
8	% in Sample	Real	10		2
9	Families	Integer	10		0
10	Households	Integer	10		0
11	Male	Integer	10		0
12	Female	Integer	10		0
13	White	Integer	10		0
14	Black	Integer	10		0
15	[Am.Indian/Esk./Aleut]	Integer	10		0
16	[Asian/Pacific Islander]	Integer	10		0
17	[Other Race]	Integer	10		0
18	[White (spec.)]	Integer	10		0
19	[Black (spec.)]	Integer	10		0
20	[American Indian]	Integer	10		0
21	Eskimo	Integer	10		0
22	Aleut	Integer	10		0
23	Chinese	Integer	10		0
24	Filipino	Integer	10		0
25	Japanese	Integer	10		0
26	[Asian Indian]	Integer	10		0
27	Korean	Integer	10		0
28	Vietnamese	Integer	10		0
29	Cambodian	Integer	10		0
30	Hmong	Integer	10		0
31	Laotian	Integer	10		0
32	Thai	Integer	10		0
33	[Other Asian]	Integer	10		0
34	Hawaiian	Integer	10		0
35	Samoan	Integer	10		0
36	Tongan	Integer	10		0
37	[Other Polynesian]	Integer	10		0
38	Guamarian	Integer	10		0
39	[Other Micronesian]	Integer	10		0
40	Melanesian	Integer	10		0
41	[Other Pacific Islander]	Integer	10		0
42	[Other race (spec.)]	Integer	10		0
43	[Hispanic Origin]	Integer	10		0
44	[Not Hisp.]	Integer	10		0
45	Mexican	Integer	10		0

46 Puerto Rican	Integer	10	0
47 Cuban	Integer	10	0
48 Dominican	Integer	10	0
49 Guatemalan	Integer	10	0
50 Honduran	Integer	10	0
51 Nicaraguan	Integer	10	0
52 Panamanian	Integer	10	0
53 Salvadoran	Integer	10	0
54 [Other Cent. American]	Integer	10	0
55 Colombian	Integer	10	0
56 Ecuadorian	Integer	10	0
57 Peruvian	Integer	10	0
58 [Other S.American]	Integer	10	0
59 [Other Hispanic]	Integer	10	0
60 [Age <1]	Integer	10	0
61 [Age 1-2]	Integer	10	0
62 [Age 3-4]	Integer	10	0
63 [Age 5]	Integer	10	0
64 [Age 6]	Integer	10	0
65 [Age 7-9]	Integer	10	0
66 [Age 10-11]	Integer	10	0
67 [Age 12-13]	Integer	10	0
68 [Age 14]	Integer	10	0
69 [Age 15]	Integer	10	0
70 [Age 16]	Integer	10	0
71 [Age 17]	Integer	10	0
72 [Age 18]	Integer	10	0
73 [Age 19]	Integer	10	0
74 [Age 20]	Integer	10	0
75 [Age 21]	Integer	10	0
76 [Age 22-24]	Integer	10	0
77 [Age 25-29]	Integer	10	0
78 [Age 30-34]	Integer	10	0
79 [Age 35-39]	Integer	10	0
80 [Age 40-44]	Integer	10	0
81 [Age 45-49]	Integer	10	0
82 [Age 50-54]	Integer	10	0
83 [Age 55-59]	Integer	10	0
84 [Age 60-61]	Integer	10	0
85 [Age 62-64]	Integer	10	0
86 [Age 65-69]	Integer	10	0
87 [Age 70-74]	Integer	10	0
88 [Age 75-79]	Integer	10	0
89 [Age 80-84]	Integer	10	0
90 [Age 85+]	Integer	10	0
91 [1 Person HH]	Integer	10	0
92 [2 Person HH]	Integer	10	0
93 [3 Person HH]	Integer	10	0

94 [4 Person HH]	Integer	10	0
95 [5 Person HH]	Integer	10	0
96 [6 Person HH]	Integer	10	0
97 [7+ Person HH]	Integer	10	0
98 [HH Marr. w/Ch. <18]	Integer	10	0
99 [HH Marr. w/o Ch. <18]	Integer	10	0
100 [HH M w/Ch. <18]	Integer	10	0
101 [HH M w/o Ch. <18]	Integer	10	0
102 [HH F w/Ch. <18]	Integer	10	0
103 [HH F w/o Ch. <18]	Integer	10	0
104 [HH Nonfamily]	Integer	10	0
105 [Fam Marr. w/Ch. <18]	Integer	10	0
106 [Fam Marr. w/o Ch. <18]	Integer	10	0
107 [Fam M w/Ch. <18]	Integer	10	0
108 [Fam M w/o Ch. <18]	Integer	10	0
109 [Fam F w/Ch. <18]	Integer	10	0
110 [Fam F w/o Ch. <18]	Integer	10	0
111 [Male: Single]	Integer	10	0
112 [Male: Married]	Integer	10	0
113 [Male: Separated]	Integer	10	0
114 [Male: Other]	Integer	10	0
115 [Male: Widowed]	Integer	10	0
116 [Male: Divorced]	Integer	10	0
117 [Female: Single]	Integer	10	0
118 [Female: Married]	Integer	10	0
119 [Female: Separated]	Integer	10	0
120 [Female: Other]	Integer	10	0
121 [Female: Widowed]	Integer	10	0
122 [Female: Divorced]	Integer	10	0
123 [Lang: English]	Integer	10	0
124 [Lang: German]	Integer	10	0
125 [Lang: Yiddish]	Integer	10	0
126 [Lang: Oth. W.German]	Integer	10	0
127 [Lang: Scandinavian]	Integer	10	0
128 [Lang: Greek]	Integer	10	0
129 [Lang: Indic]	Integer	10	0
130 [Lang: Italian]	Integer	10	0
131 [Lang: French]	Integer	10	0
132 [Lang: Portugese]	Integer	10	0
133 [Lang: Spanish]	Integer	10	0
134 [Lang: Polish]	Integer	10	0
135 [Lang: Russian]	Integer	10	0
136 [Lang: S. Slavic]	Integer	10	0
137 [Lang: Other Slavic]	Integer	10	0
138 [Lang: Other IndoEuro.]	Integer	10	0
139 [Lang: Arabic]	Integer	10	0
140 [Lang: Tagalog]	Integer	10	0
141 [Lang: Chinese]	Integer	10	0

142[Lang: Hungarian]	Integer	10	0
143[Lang: Japanese]	Integer	10	0
144[Lang: Mon-Khmer]	Integer	10	0
145[Lang: Korean]	Integer	10	0
146[Lang: Native N. Amer.]	Integer	10	0
147[Lang: Vietnamese]	Integer	10	0
148[Lang: Other]	Integer	10	0
149[Ancestry: Arab]	Integer	10	0
150[Ancestry: Austrian]	Integer	10	0
151[Ancestry: Belgian]	Integer	10	0
152[Ancestry: Canadian]	Integer	10	0
153[Ancestry: Czech]	Integer	10	0
154[Ancestry: Danish]	Integer	10	0
155[Ancestry: Dutch]	Integer	10	0
156[Ancestry: English]	Integer	10	0
157[Ancestry: Finnish]	Integer	10	0
158[Ancestry: French]	Integer	10	0
159[Ancestry: Fr. Canadian]	Integer	10	0
160[Ancestry: German]	Integer	10	0
161[Ancestry: Greek]	Integer	10	0
162[Ancestry: Hungarian]	Integer	10	0
163[Ancestry: Irish]	Integer	10	0
164[Ancestry: Italian]	Integer	10	0
165[Ancestry: Lithuanian]	Integer	10	0
166[Ancestry: Norwegian]	Integer	10	0
167[Ancestry: Polish]	Integer	10	0
168[Ancestry: Portuguese]	Integer	10	0
169[Ancestry: Romanian]	Integer	10	0
170[Ancestry: Russian]	Integer	10	0
171[Ancestry: Scotch-Irish]	Integer	10	0
172[Ancestry: Scottish]	Integer	10	0
173[Ancestry: Slovak]	Integer	10	0
174[Ancestry: African]	Integer	10	0
175[Ancestry: Swedish]	Integer	10	0
176[Ancestry: Swiss]	Integer	10	0
177[Ancestry: Ukrainian]	Integer	10	0
178[Ancestry: U.S.]	Integer	10	0
179[Ancestry: Welsh]	Integer	10	0
180[Ancestry: West Indian]	Integer	10	0
181[Ancestry: Yugoslavian]	Integer	10	0
182[Ancestry: Hispanic]	Integer	10	0
183[Ancestry: Other]	Integer	10	0
184[Ancestry: Not reported]	Integer	10	0
185[F: 15-24 Never Married]	Integer	10	0
186[F: 25-34 Never Married]	Integer	10	0
187[F: 35-44 Never Married]	Integer	10	0
188[F: 45+ Never Married]	Integer	10	0
189[F: 15-24 Ever Married]	Integer	10	0

190[F: 25-34 Ever Married]	Integer	10	0
191[F: 35-44 Ever Married]	Integer	10	0
192[F: 45+ Ever Married]	Integer	10	0
193[Mode: Drove Alone]	Integer	10	0
194[Mode: Carpool]	Integer	10	0
195[Mode: Bus]	Integer	10	0
196[Mode: Streetcar]	Integer	10	0
197[Mode: Subway]	Integer	10	0
198[Mode: Railroad]	Integer	10	0
199[Mode: Ferry]	Integer	10	0
200[Mode: Taxi]	Integer	10	0
201[Mode: Motorcycle]	Integer	10	0
202[Mode: Bicycle]	Integer	10	0
203[Mode: Walk]	Integer	10	0
204[Mode: Other Means]	Integer	10	0
205[Mode: Work at Home]	Integer	10	0
206[TT 2WK: <5 Min.]	Integer	10	0
207[TT 2WK: 5-9 Min.]	Integer	10	0
208[TT 2WK: 10-14 Min.]	Integer	10	0
209[TT 2WK: 15-19 Min.]	Integer	10	0
210[TT 2WK: 20-24 Min.]	Integer	10	0
211[TT 2WK: 25-29 Min.]	Integer	10	0
212[TT 2WK: 30-34 Min.]	Integer	10	0
213[TT 2WK: 35-39 Min.]	Integer	10	0
214[TT 2WK: 40-44 Min.]	Integer	10	0
215[TT 2WK: 45-59 Min.]	Integer	10	0
216[TT 2WK: 60-89 Min.]	Integer	10	0
217[TT 2WK: 90+ Min.]	Integer	10	0
218[TT 2WK: Wk at Home]	Integer	10	0
219[Left 4WK: 12am-4:59]	Integer	10	0
220[Left 4WK: 5am-5:29]	Integer	10	0
221[Left 4WK: 5:30am-5:59]	Integer	10	0
222[Left 4WK: 6am-6:29]	Integer	10	0
223[Left 4WK: 6:30am-6:59]	Integer	10	0
224[Left 4WK: 7am-7:29]	Integer	10	0
225[Left 4WK: 7:30am-7:59]	Integer	10	0
226[Left 4WK: 8am-8:29]	Integer	10	0
227[Left 4WK: 8:30am-8:59]	Integer	10	0
228[Left 4WK: 9am-9:59]	Integer	10	0
229[Left 4WK: 10am-10:59]	Integer	10	0
230[Left 4WK: 11am-11:59]	Integer	10	0
231[Left 4WK: 12pm-3:59]	Integer	10	0
232[Left 4WK: 4pm-11:59]	Integer	10	0
233[Left 4WK: Wk at Home]	Integer	10	0
234[Sch: Pub. Preprimary]	Integer	10	0
235[Sch: Priv. Preprimary]	Integer	10	0
236[Sch: Pub. Elem/High]	Integer	10	0
237[Sch: Priv. Elem/High]	Integer	10	0

238 [Sch: Pub. College]	Integer	10	0
239 [Sch: Priv. College]	Integer	10	0
240 [Sch: Not Enrolled]	Integer	10	0
241 [Edu. 25+: <9th grade]	Integer	10	0
242 [Edu. 25+: 9-12 n/grad]	Integer	10	0
243 [Edu. 25+: HS Grad]	Integer	10	0
244 [Edu. 25+: Coll. n/grad]	Integer	10	0
245 [Edu. 25+: Associate]	Integer	10	0
246 [Edu. 25+: Bachelor's]	Integer	10	0
247 [Edu. 25+: Grad/Prof.]	Integer	10	0
248 [Wht 25+: <9th grade]	Integer	10	0
249 [Wht 25+: 9-12 n/grad]	Integer	10	0
250 [Wht 25+: HS Grad]	Integer	10	0
251 [Wht 25+: Coll. n/grad]	Integer	10	0
252 [Wht 25+: Associate]	Integer	10	0
253 [Wht 25+: Bachelor's]	Integer	10	0
254 [Wht 25+: Grad/Prof.]	Integer	10	0
255 [Blk 25+: <9th grade]	Integer	10	0
256 [Blk 25+: 9-12 n/grad]	Integer	10	0
257 [Blk 25+: HS Grad]	Integer	10	0
258 [Blk 25+: Coll. n/grad]	Integer	10	0
259 [Blk 25+: Associate]	Integer	10	0
260 [Blk 25+: Bachelor's]	Integer	10	0
261 [Blk 25+: Grad/Prof.]	Integer	10	0
262 [A. I. 25+: <9th grade]	Integer	10	0
263 [A. I. 25+: 9-12 n/grad]	Integer	10	0
264 [A. I. 25+: HS Grad]	Integer	10	0
265 [A. I. 25+: Coll. n/grad]	Integer	10	0
266 [A. I. 25+: Associate]	Integer	10	0
267 [A. I. 25+: Bachelor's]	Integer	10	0
268 [A. I. 25+: Grad/Prof.]	Integer	10	0
269 [As. 25+: <9th grade]	Integer	10	0
270 [As. 25+: 9-12 n/grad]	Integer	10	0
271 [As. 25+: HS Grad]	Integer	10	0
272 [As. 25+: Coll. n/grad]	Integer	10	0
273 [As. 25+: Associate]	Integer	10	0
274 [As. 25+: Bachelor's]	Integer	10	0
275 [As. 25+: Grad/Prof.]	Integer	10	0
276 [Oth. 25+: <9th grade]	Integer	10	0
277 [Oth. 25+: 9-12 n/grad]	Integer	10	0
278 [Oth. 25+: HS Grad]	Integer	10	0
279 [Oth. 25+: Coll. n/grad]	Integer	10	0
280 [Oth. 25+: Associate]	Integer	10	0
281 [Oth. 25+: Bachelor's]	Integer	10	0
282 [Oth. 25+: Grad/Prof.]	Integer	10	0
283 [Milit. >9/80 only <2 yrs]	Integer	10	0
284 [Milit. >9/80 only 2+ yrs]	Integer	10	0
285 [Milit. 5/75-8/80 only]	Integer	10	0

286[Milit. 5/75+]	Integer	10	0
287[Milit. Viet.-Korea-WWII]	Integer	10	0
288[Milit. Viet.+Korea-WWII]	Integer	10	0
289[Milit. Viet.+Korea+WWII]	Integer	10	0
290[Milit. 2/55-7/64 only]	Integer	10	0
291[Milit. Korea-Viet.-WWII]	Integer	10	0
292[Milit. Korea+WWII-Viet.]	Integer	10	0
293[Milit. WWII-Korea-Viet.]	Integer	10	0
294[Milit. WWI]	Integer	10	0
295[Milit. Other Service]	Integer	10	0
296[Emp. 16+ M Armed Fcs.]	Integer	10	0
297[Emp. 16+ M Civ Emp.]	Integer	10	0
298[Emp. 16+ M Civ n/Emp.]	Integer	10	0
299[Emp. 16+ M n/in Wk. Fc.]	Integer	10	0
300[Emp. 16+ F Armed Fcs.]	Integer	10	0
301[Emp. 16+ F Civ Emp.]	Integer	10	0
302[Emp. 16+ F Civ n/Emp.]	Integer	10	0
303[Emp. 16+ F n/in Wk. Fc.]	Integer	10	0
304[Wht 16+ M Armed Fcs.]	Integer	10	0
305[Wht 16+ M Civ Wht]	Integer	10	0
306[Wht 16+ M Civ U/E Wht]	Integer	10	0
307[Wht 16+ M n/in Wk. Fc.]	Integer	10	0
308[Wht 16+ F Armed Fcs.]	Integer	10	0
309[Wht 16+ F Civ Wht]	Integer	10	0
310[Wht 16+ F Civ U/E Wht]	Integer	10	0
311[Wht 16+ F n/in Wk. Fc.]	Integer	10	0
312[Blk 16+ M Armed Fcs.]	Integer	10	0
313[Blk 16+ M Civ Blk]	Integer	10	0
314[Blk 16+ M Civ U/E Blk]	Integer	10	0
315[Blk 16+ M n/in Wk. Fc.]	Integer	10	0
316[Blk 16+ F Armed Fcs.]	Integer	10	0
317[Blk 16+ F Civ Blk]	Integer	10	0
318[Blk 16+ F Civ U/E Blk]	Integer	10	0
319[Blk 16+ F n/in Wk. Fc.]	Integer	10	0
320[A. I. 16+ M Armed Fcs.]	Integer	10	0
321[A. I. 16+ M Civ A. I.]	Integer	10	0
322[A. I. 16+ M Civ U/E A. I]	Integer	10	0
323[A. I. 16+ M n/in Wk. Fc.]	Integer	10	0
324[A. I. 16+ F Armed Fcs.]	Integer	10	0
325[A. I. 16+ F Civ A. I.]	Integer	10	0
326[A. I. 16+ F Civ U/E A. I]	Integer	10	0
327[A. I. 16+ F n/in Wk. Fc.]	Integer	10	0
328[As. 16+ M Armed Fcs.]	Integer	10	0
329[As. 16+ M Civ As.]	Integer	10	0
330[As. 16+ M Civ U/E As.]	Integer	10	0
331[As. 16+ M n/in Wk. Fc.]	Integer	10	0
332[As. 16+ F Armed Fcs.]	Integer	10	0
333[As. 16+ F Civ As.]	Integer	10	0

334 [As. 16+ F Civ U/E As.]	Integer	10	0
335 [As. 16+ F n/in Wk. Fc.]	Integer	10	0
336 [Oth. 16+ M Armed Fcs.]	Integer	10	0
337 [Oth. 16+ M Civ Oth.]	Integer	10	0
338 [Oth. 16+ M Civ U/E Oth.]	Integer	10	0
339 [Oth. 16+ M n/in Wk. Fc.]	Integer	10	0
340 [Oth. 16+ F Armed Fcs.]	Integer	10	0
341 [Oth. 16+ F Civ Oth.]	Integer	10	0
342 [Oth. 16+ F Civ U/E Oth.]	Integer	10	0
343 [Oth. 16+ F n/in Wk. Fc.]	Integer	10	0
344 [Ind: Agric. Forest. Fish]	Integer	10	0
345 [Ind: Mining]	Integer	10	0
346 [Ind: Construction]	Integer	10	0
347 [Ind: Manuf. n/Durable]	Integer	10	0
348 [Ind: Manuf. Durable]	Integer	10	0
349 [Ind: Transportation]	Integer	10	0
350 [Ind: Comm./Pub. Utils.]	Integer	10	0
351 [Ind: Wholesale Trade]	Integer	10	0
352 [Ind: Retail Trade]	Integer	10	0
353 [Ind: Finance Ins. R.Est.]	Integer	10	0
354 [Ind: Bus. & Repair Serv.]	Integer	10	0
355 [Ind: Personal Services]	Integer	10	0
356 [Ind: Entertain./Rec.]	Integer	10	0
357 [Ind: Health Services]	Integer	10	0
358 [Ind: Educ. Services]	Integer	10	0
359 [Ind: Other Prof. Serv.]	Integer	10	0
360 [Ind: Public Admin.]	Integer	10	0
361 [Occ: Exec/Adm/Manag]	Integer	10	0
362 [Occ: Prof. Specialty]	Integer	10	0
363 [Occ: Technicians]	Integer	10	0
364 [Occ: Sales]	Integer	10	0
365 [Occ: Admin. Support]	Integer	10	0
366 [Occ: Priv. Household]	Integer	10	0
367 [Occ: Protective Serv.]	Integer	10	0
368 [Occ: General Service]	Integer	10	0
369 [Occ: Farm./Forest./Fish]	Integer	10	0
370 [Occ: Prec. Prod/Repair]	Integer	10	0
371 [Occ: Machine]	Integer	10	0
372 [Occ: Transportation]	Integer	10	0
373 [Occ: Misc. Labor]	Integer	10	0
374 [Class of Wk: Prv. Profit]	Integer	10	0
375 [Class of Wk: Prv. n/Prof]	Integer	10	0
376 [Class of Wk: Loc. Gov't]	Integer	10	0
377 [Class of Wk: St. Gov't]	Integer	10	0
378 [Class of Wk: Fed. Gov't]	Integer	10	0
379 [Class of Wk: Self-empl.]	Integer	10	0
380 [Class of Wk: n/Pd. Fam]	Integer	10	0
381 [HH Inc: <\$5K]	Integer	10	0

382 [HH Inc: \$5K-9999]	Integer	10	0
383 [HH Inc: \$10K-12499]	Integer	10	0
384 [HH Inc: \$12500-14999]	Integer	10	0
385 [HH Inc: \$15K-17499]	Integer	10	0
386 [HH Inc: \$17500-19999]	Integer	10	0
387 [HH Inc: \$20K-22499]	Integer	10	0
388 [HH Inc: \$22500-24999]	Integer	10	0
389 [HH Inc: \$25K-27499]	Integer	10	0
390 [HH Inc: \$27500-29999]	Integer	10	0
391 [HH Inc: \$30K-32499]	Integer	10	0
392 [HH Inc: \$32500-34999]	Integer	10	0
393 [HH Inc: \$35K-37499]	Integer	10	0
394 [HH Inc: \$37500-39999]	Integer	10	0
395 [HH Inc: \$40K-42499]	Integer	10	0
396 [HH Inc: \$42500-44999]	Integer	10	0
397 [HH Inc: \$45K-47499]	Integer	10	0
398 [HH Inc: \$47500-49999]	Integer	10	0
399 [HH Inc: \$50K-54999]	Integer	10	0
400 [HH Inc: \$55K-59999]	Integer	10	0
401 [HH Inc: \$60K-74999]	Integer	10	0
402 [HH Inc: \$75K-99999]	Integer	10	0
403 [HH Inc: \$100K-124999]	Integer	10	0
404 [HH Inc: \$125K-149999]	Integer	10	0
405 [HH Inc: \$150K+]	Integer	10	0
406 [HH Median Income]	Integer	10	0
407 [HH Inc Wht: <\$5K]	Integer	10	0
408 [HH Inc W: \$5K-9999]	Integer	10	0
409 [HH Inc W: \$10K-14999]	Integer	10	0
410 [HH Inc W: \$15K-24999]	Integer	10	0
411 [HH Inc W: \$25K-34999]	Integer	10	0
412 [HH Inc W: \$35K-49999]	Integer	10	0
413 [HH Inc W: \$50K-74999]	Integer	10	0
414 [HH Inc W: \$75K-99999]	Integer	10	0
415 [HH Inc W: \$100K+]	Integer	10	0
416 [HH Inc B: <\$5K]	Integer	10	0
417 [HH Inc B: \$5K-9999]	Integer	10	0
418 [HH Inc B: \$10K-14999]	Integer	10	0
419 [HH Inc B: \$15K-24999]	Integer	10	0
420 [HH Inc B: \$25K-34999]	Integer	10	0
421 [HH Inc B: \$35K-49999]	Integer	10	0
422 [HH Inc B: \$50K-74999]	Integer	10	0
423 [HH Inc B: \$75K-99999]	Integer	10	0
424 [HH Inc B: \$100K+]	Integer	10	0
425 [HH Inc A. I.: <\$5K]	Integer	10	0
426 [HH Inc A. I.: \$5K-9999]	Integer	10	0
427 [HH Inc A. I.: \$10K-14999]	Integer	10	0
428 [HH Inc A. I.: \$15K-24999]	Integer	10	0
429 [HH Inc A. I.: \$25K-34999]	Integer	10	0

430 [HH Inc A. I.: \$35K-49999	Integer	10	0
431 [HH Inc A. I.: \$50K-74999	Integer	10	0
432 [HH Inc A. I.: \$75K-99999	Integer	10	0
433 [HH Inc A. I.: \$100K+]	Integer	10	0
434 [HH Inc As.: <\$5K]	Integer	10	0
435 [HH Inc As.: \$5K-9999]	Integer	10	0
436 [HH Inc As.: \$10K-14999]	Integer	10	0
437 [HH Inc As.: \$15K-24999]	Integer	10	0
438 [HH Inc As.: \$25K-34999]	Integer	10	0
439 [HH Inc As.: \$35K-49999]	Integer	10	0
440 [HH Inc As.: \$50K-74999]	Integer	10	0
441 [HH Inc As.: \$75K-99999]	Integer	10	0
442 [HH Inc As.: \$100K+]	Integer	10	0
443 [HH Inc Oth: <\$5K]	Integer	10	0
444 [HH Inc Oth: \$5K-9999]	Integer	10	0
445 [HH Inc Oth: \$10K-14999]	Integer	10	0
446 [HH Inc Oth: \$15K-24999]	Integer	10	0
447 [HH Inc Oth: \$25K-34999]	Integer	10	0
448 [HH Inc Oth: \$35K-49999]	Integer	10	0
449 [HH Inc Oth: \$50K-74999]	Integer	10	0
450 [HH Inc Oth: \$75K-99999]	Integer	10	0
451 [HH Inc Oth: \$100K+]	Integer	10	0
452 [HH Inc Hs <\$5K]	Integer	10	0
453 [HH Inc Hs \$5K-9999]	Integer	10	0
454 [HH Inc Hs \$10K-14999]	Integer	10	0
455 [HH Inc Hs \$15K-24999]	Integer	10	0
456 [HH Inc Hs \$25K-34999]	Integer	10	0
457 [HH Inc Hs \$35K-49999]	Integer	10	0
458 [HH Inc Hs \$50K-74999]	Integer	10	0
459 [HH Inc Hs \$75K-99999]	Integer	10	0
460 [HH Inc Hs \$100K+]	Integer	10	0
461 [HH With Earnings]	Integer	10	0
462 [HH w/o Earnings]	Integer	10	0
463 [HH w/Wage or Sal. Inc]	Integer	10	0
464 [HH no Wage or Sal. Inc]	Integer	10	0
465 [HH w/Nonfarm S-E Inc]	Integer	10	0
466 [HH no Nonfarm S-E. Inc]	Integer	10	0
467 [HH w/Farm S-E Inc]	Integer	10	0
468 [HH no Farm S-E Inc]	Integer	10	0
469 [HH w/Int Divid Rent Inc]	Integer	10	0
470 [HH no Int Div Rent Inc]	Integer	10	0
471 [HH w/Soc. Security Inc]	Integer	10	0
472 [HH no Soc. Security Inc]	Integer	10	0
473 [HH w/Public Assist. Inc]	Integer	10	0
474 [HH no Public Assist. Inc]	Integer	10	0
475 [HH w/Retirement Inc]	Integer	10	0
476 [HH no Retirement Inc]	Integer	10	0
477 [HH w/Oth. Type Inc]	Integer	10	0

478[HH no Oth. Type Inc]	Integer	10	0
479[HH Aggr. Wage/Sal. Inc]	Real	15	0
480[HH Aggr. n/Farm S-E Inc]	Real	15	0
481[HH Aggr. Farm S-E Inc]	Real	15	0
482[HH Aggr. Int/Div/Rent In	Real	15	0
483[HH Aggr. Soc. Sec. Inc]	Real	15	0
484[HH Aggr. Pub Assist Inc]	Real	15	0
485[HH Aggr. Retiremt. Inc]	Real	15	0
486[HH Aggr. Oth Type Inc]	Real	15	0
487[P In HH w/PA Inc <15 yrs	Integer	10	0
488[P In HH w/PA Inc 15-64]	Integer	10	0
489[P In HH w/PA Inc 65+]	Integer	10	0
490[Fam: No Workers]	Integer	10	0
491[Fam: 1 Worker]	Integer	10	0
492[Fam: 2 Workers]	Integer	10	0
493[Fam: 3+ Workers]	Integer	10	0
494[Per Capita Income]	Integer	10	0
495[Per Capita Inc: White]	Integer	10	0
496[Per Capita Inc: Black]	Integer	10	0
497[Per Capita Inc: Am. Ind.	Integer	10	0
498[Per Capita Inc: Asian]	Integer	10	0
499[Per Capita Inc: Other]	Integer	10	0
500[Per Capita Inc: Hisp.]	Integer	10	0
501[P Bel. Pov. Lev.: <5]	Integer	10	0
502[P Bel. Pov. Lev.: 5]	Integer	10	0
503[P Bel. Pov. Lev.: 6-11]	Integer	10	0
504[P Bel. Pov. Lev.: 12-17]	Integer	10	0
505[P Bel. Pov. Lev.: 18-24]	Integer	10	0
506[P Bel. Pov. Lev.: 25-34]	Integer	10	0
507[P Bel. Pov. Lev.: 35-44]	Integer	10	0
508[P Bel. Pov. Lev.: 45-54]	Integer	10	0
509[P Bel. Pov. Lev.: 55-59]	Integer	10	0
510[P Bel. Pov. Lev.: 60-64]	Integer	10	0
511[P Bel. Pov. Lev.: 65-74]	Integer	10	0
512[P Bel. Pov. Lev.: 75+]	Integer	10	0
513[Housing Units (HU)]	Integer	10	0
514[100% Count of HU]	Integer	10	0
515[HU Occupied]	Integer	10	0
516[HU Vacant]	Integer	10	0
517[HU Inside Urban Area]	Integer	10	0
518[HU Outside Urb. Area]	Integer	10	0
519[HU Rural Farm]	Integer	10	0
520[HU Rural Nonfarm]	Integer	10	0
521[HU Owner Occupied]	Integer	10	0
522[HU Renter Occupied]	Integer	10	0
523[HU Occ: White]	Integer	10	0
524[HU Occ: Black]	Integer	10	0
525[HU Occ: A. I. Esk ...]	Integer	10	0

526 [HU Occ: Asian/P. Isle]	Integer	10	0
527 [HU Occ: Other]	Integer	10	0
528 [HU Own. Occ: White]	Integer	10	0
529 [HU Own. Occ: Black]	Integer	10	0
530 [HU Own. Occ: A. I.]	Integer	10	0
531 [HU Own. Occ: Asian]	Integer	10	0
532 [HU Own. Occ: Other]	Integer	10	0
533 [HU Rent Occ: White]	Integer	10	0
534 [HU Rent Occ: Black]	Integer	10	0
535 [HU Rent Occ: A. I.]	Integer	10	0
536 [HU Rent Occ: Asian]	Integer	10	0
537 [HU Rent Occ: Other]	Integer	10	0
538 [HU w/1 Room]	Integer	10	0
539 [HU w/2 Rooms]	Integer	10	0
540 [HU w/3 Rooms]	Integer	10	0
541 [HU w/4 Rooms]	Integer	10	0
542 [HU w/5 Rooms]	Integer	10	0
543 [HU w/6 Rooms]	Integer	10	0
544 [HU w/7 Rooms]	Integer	10	0
545 [HU w/8 Rooms]	Integer	10	0
546 [HU w/9+ Rooms]	Integer	10	0
547 [HU: Rent Occ 1 Det. HU]	Integer	10	0
548 [HU: Rent Occ 1 Att. HU]	Integer	10	0
549 [HU: Rent Occ 2 HU]	Integer	10	0
550 [HU: Rent Occ 3-4 HU]	Integer	10	0
551 [HU: Rent Occ 5-9 HU]	Integer	10	0
552 [HU: Rent Occ 10-19 HU]	Integer	10	0
553 [HU: Rent Occ 20-49 HU]	Integer	10	0
554 [HU: Rent Occ 50+ HU]	Integer	10	0
555 [HU: Rent Occ Mob. Hm.]	Integer	10	0
556 [HU: Rent Occ Other]	Integer	10	0
557 [HU Water: System/Co.]	Integer	10	0
558 [HU Water: Drilled Well]	Integer	10	0
559 [HU Water: Dug Well]	Integer	10	0
560 [HU Water: Other Source]	Integer	10	0
561 [HU Sewage: Public]	Integer	10	0
562 [HU Sewage: Tank]	Integer	10	0
563 [HU Sewage: Other]	Integer	10	0
564 [HU Built: 1989-3/90]	Integer	10	0
565 [HU Built: 1985-1988]	Integer	10	0
566 [HU Built: 1980-1984]	Integer	10	0
567 [HU Built: 1970-1979]	Integer	10	0
568 [HU Built: 1960-1969]	Integer	10	0
569 [HU Built: 1950-1959]	Integer	10	0
570 [HU Built: 1940-1949]	Integer	10	0
571 [HU Built: 1939 or Earlie]	Integer	10	0
572 [HU Median Yr. Built]	Integer	10	0
573 [HU Heating: Utility Gas]	Integer	10	0

574 [HU Heating: Tank Gas]	Integer	10	0
575 [HU Heating: Electricity]	Integer	10	0
576 [HU Heating: Fuel Oil]	Integer	10	0
577 [HU Heating: Coal/Coke]	Integer	10	0
578 [HU Heating: Wood]	Integer	10	0
579 [HU Heating: Solar]	Integer	10	0
580 [HU Heating: Other Fuel]	Integer	10	0
581 [HU Heating: No Fuel]	Integer	10	0
582 [HU w/No Bedroom]	Integer	10	0
583 [HU w/1 Bedroom]	Integer	10	0
584 [HU w/2 Bedroom]	Integer	10	0
585 [HU w/3 Bedroom]	Integer	10	0
586 [HU w/4 Bedroom]	Integer	10	0
587 [HU w/5+ Bedroom]	Integer	10	0
588 [HU Own Occ: w/Phone]	Integer	10	0
589 [HU Own Occ: w/o Phone]	Integer	10	0
590 [HU Rent Occ: w/Phone]	Integer	10	0
591 [HU Rent Occ: w/o Phone]	Integer	10	0
592 [HU Own Occ: Aggr. Veh.]	Integer	10	0
593 [HU Rent Occ: Aggr. Veh.]	Integer	10	0
594 [HU w/Complete Kitchen]	Integer	10	0
595 [HU w/o Complete Kitchen]	Integer	10	0
596 [HU Rent Occ: Med. Rent]	Integer	10	0
597 [HU Rent Occ: Aggr. Rent]	Real	10	0
598 [HU Rent Occ: n/Utils Inc]	Integer	10	0
599 [HU Rent Occ: Utils Incl.]	Integer	10	0
600 [Med. Rent as % HH Inc]	Real	10	2
601 [HU Mortg Med Own Costs]	Integer	10	0
602 [HU n/Mtg Md Own Costs]	Integer	10	0
603 [HU Mortg Agr Own Costs]	Real	15	0
604 [HU n/Mtg Agr Own Costs]	Real	15	0
605 [HU Mtg Own Costs: %inc]	Real	10	2
606 [HU n/Mtg Own Cost: %inc]	Real	10	2
607 [HU Own Occ: Med Value]	Integer	10	0
608 [HU Mtg O Occ: Agr Value]	Real	15	0
609 [HU n/Mtg O Occ: Agr Val.]	Real	15	0
610 [HH Agr Inc: O Occ w/Mtg]	Real	15	0
611 [HH Agr Inc: O Occ n/Mtg]	Real	15	0
612 [HH Agr Inc: Renter Occ]	Real	15	0
613 [HU: Complete Plumbing]	Integer	10	0
614 [HU: Incomplete Plumbing]	Integer	10	0