Southeast Regional Planning Model 7.0
Activity-Based Model Users Guide

Prepared for:

Florida Department of Transportation District 4
3400 West Commercial Boulevard
Fort Lauderdale, FL 33309

Miami-Dade County Metropolitan Planning Organization
111 NW First Street, Suite 910
Miami, FL 33128

Broward County Metropolitan Planning Organization
100 West Cypress Creek Road, Suite 850
Fort Lauderdale, FL 33309

Palm Beach County Metropolitan Planning Organization
2300 North Jog Road
West Palm Beach, FL 33411

Prepared by:

Parsons Brinckerhoff, Inc.
400 SW 6th Ave, Suite 802
Portland, OR 97204

The Corradino Group, Inc.
200 S. 5th Street, Suite 300N
Louisville, KY 40202

March 2015

Table of Contents

CHAPTER 1 Development of the SERPM7 Synthetic Population 7
1.1 Introduction 7
1.2 Inputs 12
1.3 Running PopSyn II 14
1.4 Outputs 14
1.5 Validation 14
1.6 FDOT Setup 15
Appendix A: Creating a PopSyn II Setup 16
Appendix B: PopSyn II Properties 17
CHAPTER 2 CT-RAMP Activity Based Modeling 20
Overview 20
Computing Environment 25
System Design 27
Setup and Configuration 30
Input Files 34
Running the Model 87
Output Files 90
Running the Population Synthesizer 119
Integration with Summit for User Benefits Analysis 120
Utility Expression Calculator Files 121
Model Logging/Trace Results 122
CHAPTER 3 SERPM ABM Reporting Database 124
Overview 124
Reporting ETL Program 124
Database 124
Reports 140
Report Packaging 142
Running the Reporting System 142
CHAPTER 4 Survey and Activity Model Visualization System 146
4.1 Overview 146
4.2 Visualization Dashboard 148
4.3 Startup 149
4.4 User Interface 150

Time Use 164
Tree Map 167
4.5 Installation/Setup 169
4.6 References 172

Table of Tables

Table 1: MAZ Data File – maz_data.csv 36
Table 2: Population Synthesizer Household data – hhFile.csv 39
Table 3: Population Synthesizer Person data – personFile.csv 40
Table 4: Transit Walk Access –walkacc 43
Table 5: Transit Drive Access TAPS - access.prp 43
Table 6: Transit TAP PTYPE –tap.ptype 43
Table 7: MGRA Walk Distance –mgrataz.walk 43
Table 8: Zone Terminal Time –zone.term 43
Table 9: Occupation Coding – pecas_occ_occcen1.csv 44
Table 10: Activity Coding – activity_code_indcen.csv 44
Table 11: Transponder Ownership Model Data – tc_avgtt.csv 44
Table 12: Network database – SERPM_NETS.mdb 44
Table 13: All streets network - Full-SERPM_Navteq_NAD83.net 44
Table 14: MAZ All Streets Network Connectors – connectors.txt 44
Table 15: Time-of-day factor file –TOD.dbf 44
Table 16: Enplanements – ENPLANE.DBF 45
Table 17: Airport and truck trip generation rates - airport_truck_rates.dbf 45
Table 18: External-Internal Data - {DATADIR}\EIdata.dbf 45
Table 19: Vfactors file - MVFACTORS.{year}{alt} 46
Table 20: MTURNDEF.{Year}{ALT} - MTURNDEF file 46
Table 21: TSPEED.DBF – Transit speed dwell time definitions 48
Table 22: TSYSD.PTS – Transit System File (standard PT file) 48
Table 23: TFARES.FAR– Transit System Fares (standard PT file) 48
Table 24: daily_loaded.net – Loaded highway network (new fields only) 48
Table 25: X_AMHSKIMS.mat – AM peak period congested skims 49
Table 26: X_OFHSKIMS.mat – Off peak period congested skims 50
Table 27: X_PMHSKIMS.mat – PM peak period congested skims 50
Table 28: Visitor Business Frequency Distribution – visitor_businessFrequency.csv 51
Table 29: Visitor Personal Frequency Distribution – visitor_personalFrequency.csv 51
Table 30: Visitor Party Size Distribution – visitor_partySize.csv 52
Table 31: Visitor Auto Availability – visitor_autoAvailable.csv 52
Table 32: Visitor Income Distribution – visitor_income.csv 52
Table 33: Visitor Tour Time of Day Distribution – visitor_tourTOD.csv 52
Table 34: Visitor Stop Frequency Distribution – visitor_stopFrequency.csv 53
Table 35: Visitor Stop Purpose Distribution – visitor_stopPurpose.csv 53
Table 36: Visitor Outbound Stop Duration Distribution – visitor_outboundStopDuration.csv 54
Table 37: Visitor Inbound Stop Duration Distribution – visitor_inboundStopDuration.csv 54

Table 38: Temporary Static Inputs – visitor_inboundStopDuration.csv 55
Table 39: CT-RAMP Properties File 55
Table 40: UEC Files 84
Table 41: Output Files 90
Table 42: ASCII Transit Line File (standard PT file) 92
Table 43: EE Trip Table by time period - EETAB-PRD.{ALT}{Year}.MAT 94
Table 44: Dynamic Area Type - zdata3.dbf 95
Table 45: Airport and Truck P's and A's (vehicles) - Air_truck_panda.dbf 95
Table 46: External-internal P's and A's - EI_Auto_Panda.dbf (vehicles) 95
Table 47: TAP to Transit Node Connectors - TAP_LINKS.DBF 95
Table 48: TAP Node Data - TAP_Node.dbf 96
Table 49: Turn penalties - TURNS_{Year}.PEN 96
Table 50: Non-ABM Trip Tables - nonABM_<<time period>>_OD.mat 97
Table 51: Daily loaded highway network - DAILY_LOADED.net (added fields) 97
Table 52: Loaded highway network - <<time period>>_HLOAD}.NET (added fields) 97
Table 53: Loaded transit links (premium), TLINK_<<time period>>_<<ACCESSM>>_<<path name>>.DBF (primary fields) 98
Table 54: Loaded transit links (local only), TLINK_<<time period>>_Local.DBF, (primary fields) 98
Table 55: Stop to stop loads (premium paths), FIRST_LAST_<<time period>>_<<ACCESSM>>_<<pathname>>.DBF 98
Table 56: Stop to stop loads (local-only paths), FIRST_LAST_<<time period>>_Local.DBF 99
Table 57: Alternative highway net with congested times and speeds - up.net 99
Table 58: TAP to TAP Local Only Transit Skim Matrices 99
Table 59: TAP to TAP Premium Transit Skim Matrices 100
Table 60: TAZ Highway Skim Matrices 100
Table 61: Accessibilities – accessibilities.csv 101
Table 62: Auto Ownership Output –aoresults.csv 103
Table 63: Work and School Location Output – wslocresults_<<iteration>>.csv 103
Table 64: Household Data Output – householddata_<<iteration>>.csv 104
Table 65: Person Data Output – personData_<<iteration>>.csv 105
Table 66: Individual Tour Data Output – indivtourdata_<<iteration>>.csv 106
Table 67: Joint Tour Data Output – jointTourData_<<iteration>>.csv 108
Table 68: Individual Trip Data Output – indivTripData_<<iteration>>.csv 110
Table 69: Joint Trip Data Output – jointTripData_<<iteration>>.csv 112
Table 70: Visitor Tour Data – visitorTours.csv 115
Table 71: Visitor Trip Data – visitorTrips.csv 117
Table 72: TAZ Demand (Resident + Visitor) Matrices – TAZ_Demand_<<time period>>.mat 118
Table 73: TAP Demand (Resident + Visitor) Matrices – TAP_Demand_<<time period>>.mat 119

Table 74 – SERPM Reporting Database Tables 125
Table 75 – SERPM Reporting Database Fields 127
Table 76 - Startup Settings 149
Table 77- XML Special Characters 150
Table 78 – User Interface Tabs 151
Table 79 - Table Signature Fields 155
Table 80 – Database Input Files 171
Table 81 - Database Tables 171

Table of Figures

Figure 1: SERPM ABM Model System 21
Figure 2: MAZ – TAP Transit Accessibility 22
Figure 3: Basic Model Design and Linkage between Sub-Models 25
Figure 4: Disaggregate Travel Demand Model Software Components 27
Figure 5: System Setup 28
Figure 6: JPPF Framework 29
Figure 7: SERPM 7 Application Manager 33
Figure 8: Household Trace Results 123
Figure 9 - Model Visualization System 147
Figure 10 - Model Visualization Dashboard 148
Figure 11 – User Interface Layout 151
Figure 12 – Database Selection 152
Figure 13 – Saved Query Selection 153
Figure 14 – Query Builder 154
Figure 15 – Query Pane 155
Figure 16 – Tables Tab 156
Figure 17 – Bar Chart and Map – Bar Chart 158
Figure 18 – Bar Chart and Map – Map (Zones) 159
Figure 19 - Bubble Map 161
Figure 20 – Day Pattern 162
Figure 21 – Time Use 164
Figure 22 – Radar Charts 165
Figure 23 – Bar Chart (Series, Stacked, 100%) 166
Figure 24 – Line Chart 167
Figure 25 – Tree Map 168

CHAPTER 1 Development of the SERPM7 Synthetic Population

1.1 Introduction

The purpose of this memo is to describe the development of the SERPM7 (Southeast Regional Planning Model Version 7) synthetic population that was used for the new activity-based model (ABM). The SERPM7 synthetic population was developed using the SANDAG PopSyn II population synthesizer. For more information about PopSyn II, please refer to *PopSyn II Memo* written by SANDAG. This memo describes how to setup PopSyn II, including the required SQL Server database, what the required inputs are, how to run PopSyn II, and the validation that was done to ensure the results are reasonable.

PopSyn II Setup

The PopSyn II software requires Java and Microsoft SQL Server 2008 R2. The setup is done in four steps: database, spatial database, ACS database, and software.

Database Setup

The steps to setup the database are:

1) Setup a SQL Server instance
    a. for example: W-AMPDX-D-TFS05
    b. Create a "popSyn" login with SQL authentication for JDBC access. Ensure the user has bulk load privileges as well since PopSyn bulk inserts household records.
2) Create four separate databases with Windows authentication:
    a. popSyn – main pop syn database
    b. data_cafe – spatial database; optional census 2000 pums data tables
    c. acs – acs data database
    d. census – optional census 2000 data database
3) Create some tables and stored procedures
    a. Run **getxref.sql** to add function to data_cafe and popsyn databases to spatially look up Census shape using the TAZ and micro-zone shape centroid.
    b. Run **create_hh_discretized.sql** to create hh_discretized and allocated table in popSyn database
    c. Run **create_version_table.sql** to create popSyn version (run history) table
    d. Run **proc_get_subzones.sql** to create the get subzone function
    e. Load **proc_insert_household_allocated.sql** to define the inserts the synthesized household records into the database using bulk insert procedure
    f. Load **proc_insert_household_discretized.sql** to define the inserts the synthesized household records into the database using bulk insert procedure
4) Create ODBC database connections in your software of choice in order to load some required CSV tables into the database. For example, in R using the RODBC library, these are:
    a. db_dc = odbcDriverConnect("Driver={SQL Server}; Server=W-AMPDX-D-TFS05; Database=data_cafe; Integrated Security=SSPI")
    b. db_ps = odbcDriverConnect("Driver={SQL Server}; Server=W-AMPDX-D-TFS05; Database=popSyn; Integrated Security=SSPI")

c. db_acs = odbcDriverConnect("Driver={SQL Server}; Server=W-AMPDX-D-TFS05; Database=acs; Integrated Security=SSPI")
d. db_cen = odbcDriverConnect("Driver={SQL Server}; Server=W-AMPDX-D-TFS05; Database=census; Integrated Security=SSPI")

- The **serpm-popsyn.R** script contains the R code to create and load the data tables.
-

5) Load the following default/definition tables to the databases.

Database	Table	Description
popSyn	dwelling_type.csv	dwelling type codes
popSyn	employment_type.csv	employment type codes
popSyn	grade_type.csv	grade type codes
popSyn	person_type.csv	person type codes
popSyn	race_type.csv	race type codes
popSyn	relationship_type.csv	relationship type codes
popSyn	source.csv	PUMS source codes
popSyn	target_category.csv	control target category type codes
popSyn	target_lu_version.csv	target land use scenario version codes
popSyn	unit_type.csv	unit type codes
popSyn	work_hour_type.csv	work type codes
popSyn	occupation_type.csv	occupation type codes
data_cafe	geography_type.csv	geography (layer) type codes

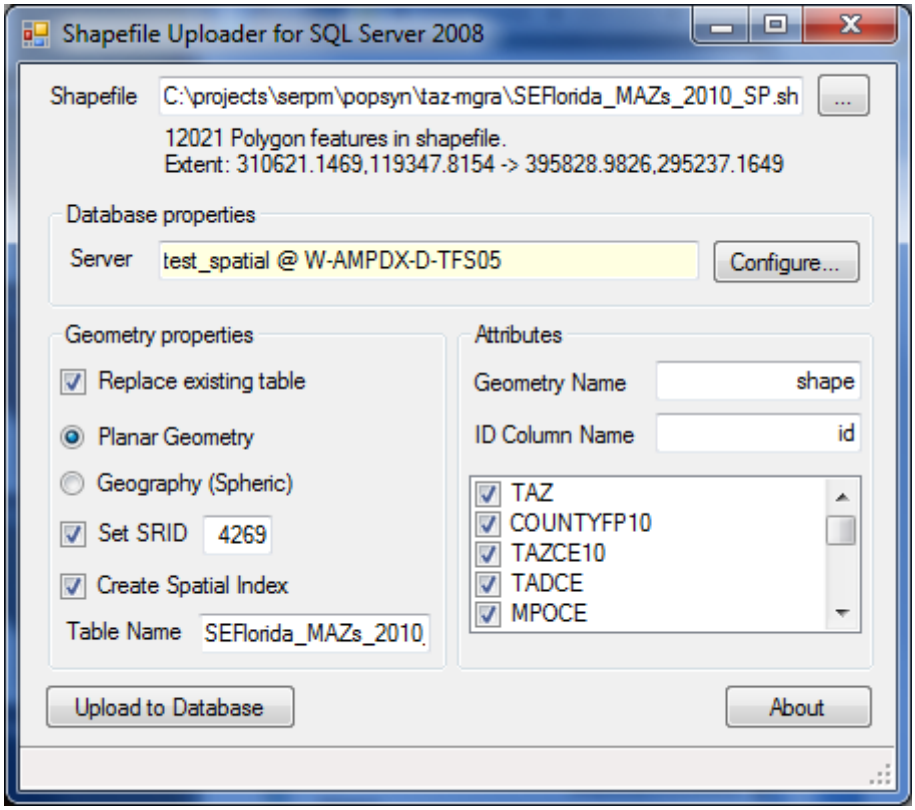
-

6) Run **updateConstraints.sql** to define the primary keys for many of the tables just loaded.
7) Load targets.csv file using the ODBC connection as well. This table includes all the control targets at the TAZ level, as well as total number of households and group quarters population at the micro-zone level. This table was created with the **create_targets_table.R** script based on the TAZ and micro-zone data discussed in the Inputs section later. See the inputs section for more details.

Spatial Database Setup

All spatial data is stored in the database using SQL Server Spatial, which is included with SQL Server 2008 R2. Note that SQL Server Spatial stores both projected data as geometry data type and spherical (or lat/long data) as geography type. PopSyn II assumes geometry data so input spatial data may need to be re-projected. The steps to setup the spatial database are:

1) Download the following 5 year ACS (2005 – 2009) geography shapefiles. Note that the 5 year ACS uses Census 2000 geography.
   a. 2000 PUMAs for Florida - http://www.census.gov/geo/www/cob/bdy_files.html
   b. 2000 blocks for Broward, Miami Dade, Palm Beach counties - http://arcdata.esri.com/data/tiger2000/tiger_download.cfm
   c. 2000 block groups for Broward, Miami Dade, Palm Beach counties - http://arcdata.esri.com/data/tiger2000/tiger_download.cfm



   d. 2000 tracts for Broward, Miami Dade, Palm Beach counties - http://arcdata.esri.com/data/tiger2000/tiger_download.cfm

2) Download Shape2SQL (http://www.sharpgis.net/page/shape2sql.aspx) in order to load shapefiles into SQL Server, which will be done later.

3) Run **create_geography_zone.sql** to create the geography zone table with no data. This table will hold all the GIS layers in the database using SQL Spatial geometry/geography data tables.
4) Re-project all shapefiles as listed below from lat/long to State Plane NAD 1983 Florida West. This can be done in ArcGIS desktop.
5) ArcGIS Dissolve the PUMA shapefile since it has multiple features with the same PUMA5 value.
6) Run Shape2Sql.exe with the following settings to load the shapefiles into the database:
   a. Planar geometry.
   b. For State Plane NAD 83 geometry, select SRID 4269, which is NAD83. The SQL Server SRID is similar to the shapefile projection (prj) file.
   c. Uncheck the ID field to import since Shape2Sql.exe wants to create a unique ID field as well.
   d. Accept the default table names in order to use the script below.
   e. Load the shapefiles into the database:
      i. p512_d00 - pumas

ii. tgr12011blk00 – blocks x3 counties
iii. tgr12086blk00
iv. tgr12099blk00
v. tgr12011grp00 – block groups x3 counties
vi. tgr12086grp00
vii. tgr12099grp00
viii. tgr12011trt00 – tracts x3 counties
ix. tgr12086trt00
x. tgr12099trt00
xi. SEFlorida_MAZs_2010 – micro-zones
xii. SERPM7TAZ_GCS_WGS84_11152011 – TAZs
xiii. TAZ_Centroids_SP – TAZ centroids

f. Run **load_geography_zone.sql** to merge all the shapes into a new table – geography_zone (with fields zone, geo_type_id, alias, shape, centroid)

7) Create spatial indexes on the geography_zone shape and centroid fields in order to speed up spatial operations. This is done with the **create_spatial_indexes.sql** script.
8) Shift the centroid for TAZ 1901, 3533, and 3535 since it is outside the PUMAs using the **shiftZoneCentroids.sql** script.

ACS Database Setup

The next step to setup the database is to load all the Census ACS tables. This is done by:

1) Load ACS PUMS data
   a. Download the 2005-2009 5 year ACS household csv data file for Florida - http://www2.census.gov/acs2005_2009_5yr/pums/csv_hfl.zip
   b. Load the csv file into the acs database using the ODBC connection as the following table: v_pums_hh_sd_2009e5
   c. Download the 2005-2009 5 year ACS person csv data file for Florida – http://www2.census.gov/acs2005_2009_5yr/pums/csv_pfl.zip
   d. Load the csv file into the acs database using the ODBC connection as the following table: v_pums_pp_sd_2009e5
2) Create a number of ACS summary file tables
   a. Download a zip file of all ACS data for Florida by block group - http://www2.census.gov/acs2005_2009_5yr/summaryfile/2005-2009_ACSSF_By_State_All_Tables
   b. Use the **acs_processing.R** script to process the zip and create a number of user-specified summary file csv tables.
      i. Copy Sequence_Number_and_Table_Number_Lookup.csv to the same folder as the zip file since it contains the location of all the summary files in the zip folder.
      ii. Specify at the top of the script, the state code ("fl") and tables numbers. The following tables are currently export by the script: "B01001","B03002","B06012","B08202","B09016","B11001","B11005",

"B11007","B11010","B11011","B11012","B11016","B14002","B17001","B19001","B23001","B23007","B23022","C24010"

iii. The script outputs a csv file for each table as well as two geography tables: acs_2009_5yr_ca_geo.csv (all geographies) and acs_geos_sd_ctblkgrp.csv (which is the same table but only for tracts and block groups).

iv. Load all export tables into the acs database using the ODBC connection.

3) Execute procedure **proc_acs_validation.sql** to create the ACS validation summaries with the following settings:
   a. county fips 1-5 = 11,86,99,0,0
   b. sumlevel ct (census tract) = 140
   c. sumlevel blkgp (block group) = 150
4) Execute procedure **proc_generate_hh_pp.sql** to create the popSyn household and person tables from the ACS PUMS tables

Software Setup

The PopSyn II software is a collection of Java JAR files, XML files, and database stored procedures. The setup of the software is as follows:

1) Create a PopSyn II setup for the user (see Appendix A for how to do this)
2) Edit the dataSource-config.xml to refer to the popSyn and data_cafe databases:
   a. <property name="jdbcUrl" value="jdbc:sqlserver://W-AMPDX-D-TFS05:1433;databaseName=popSyn;integratedSecurity=false;" />
3) Edit the properties file as below. There are a number of other settings in the property file as described in Appendix B.
   a. version.workDir=\\\\w-ampdx-d-tfs05\\projects\\sandag_popsyn\\ #example project working folder that is also accessible to the SQL Server database for bulk loading; make sure to include "\\" at the end
   b. version.target.year=2007 #2007is the "year" for ACS 2005-2009 data
   c. version.target.geography.type= TAZ_11 #to specify control targets at the TAZ level
   d. version.balance.geography.type= TAZ_11 #to specify TAZ as the target balancing level
   e. version.allocation.geography.type=MGRA_10 #to specify the allocation of results at the micro-zone level
   f. version.hhAlAtt = TOTAL_HOUSEHOLDS #allocation to micro-zone household weight target field
   g. version.gqAlAtt = TOTAL_GQ #allocation to micro-zone group quarters weight target field
   h. version.description=Scenario Name #scenario name as "SERPM base year run"
   i. version.balanceSelectionFactor 0.2 #Multiplied by the target number of households in the TAZ to define the number of households to randomly draw from the PUMA to use for balancing

j. version.pumasToDrawFrom = 3400’3501’3502’etc. #The list of PUMAs to draw from when a qualifying household is not found in the balancing step. Use the following query to determine what PUMAs intersect the TAZs:

```
SELECT DISTINCT pzone FROM popsyn.dbo.getxref(69,13)
```

1.2 Inputs

In order to run PopSyn II, TAZ and micro-zone data is required. The TAZ and micro-zone geography is defined as following:

1) TAZ data is geography_type = 13, which is connected to label “TAZ_11” as defined in org.sandag.popsyn.enums.GeographyType.java
2) Micro-zone data is geography_type = 9, which is connected to label “MGRA_10” as defined in org.sandag.popsyn.enums.GeographyType.java

The TAZ and micro-zone data is specified in the targets table. The TAZ targets are defined in control-target-config.xml and are numbered based on org.sandag.popsyn.enums.ControlCategoryType.java:

TAZ Field	Description	H/P/GQ	Target ID
HHSIZE_1	household size 1 person	H	10
HHSIZE_2	household size 2 person	H	13
HHSIZE_3	household size 3 person	H	16
HHSIZE_4PLUS	household size 4+ person	H	19
INCOME_25K	household income <$25k	H	249
INCOME_50K	household income $25k-$50k	H	252
INCOME_75K	household income $50k-$75k	H	255
INCOME_100K	household income $75k-$100k	H	55
INCOME_100KPLUS	household income $100k+	H	261
WORKERS_0	household workers 0	H	22
WORKERS_1	household workers 1	H	25
WORKERS_2	household workers 2	H	28
WORKERS_3PLUS	household workers 3+	H	31
SINGLE_FAMILY	household is single family	H	64
MULTI_UNIT	household is multi-family dwelling	H	67
MOBILE_HOME	household is mobile home	H	70
AGE0TO17	persons age 0 – 17	P	139
AGE18TO24	persons age 18 - 24	P	142
AGE25TO34	persons age 25 - 34	P	145
AGE35TO49	persons age 35 - 49	P	148
AGE50TO64	persons age 50 - 64	P	151
AGE65TO79	persons age 65 - 79	P	154
AGE80PLUS	persons age 80+	P	157
CHILDREN_0	no children in household	H	4

CHILDREN_1PLUS	1+ child in household	H	7
HISPANIC	persons of hispanic race	P	115
WHITE	persons of white race alone	P	118
BLACK	persons of black race alone	P	121
NINDIAN	persons of American Indian or Alaska Native race	P	124
ASIAN	persons of Asian race alone	P	127
ISLANDER	persons of Native Hawaiian and Other Pacific Islander alone race	P	130
RACEOTHER	persons of other race alone	P	133
MIXED	persons of two or more races	P	136
MALE	persons of male gender	P	163
FEMALE	persons of female gender	P	160
GQ_INST	institutionalized group quarters population	GQ	34
GQ_COLL	college group quarters population	GQ	37
GQ_MIL	military group quarters population	GQ	40
GQ_OTHER	non-institutionalized group quarters population	GQ	43
TOTAL_HOUSEHOLDS	total households	H	109
TOTAL_GQ	total persons in group quarters	GQ	112

The micro-zone control target data is also defined in the targets table and has the same labeling and numbering as the TAZ data, but a different geography_type.

Micro-zone Field	**Description**	**H/P/GQ**	**Target ID**
TOTAL_HOUSEHOLDS	total households	H	109
TOTAL_GQ	total persons in group quarters	GQ	112

The TAZ and micro-zone target data was reformatted into the following table and then loaded into the database with the **create_targets_table.R** script and then the **serpm-popsyn-target-loader.R** script as described earlier.

Target Field	**Description**	**Example**
CATEGORY	Target ID (category number) (see above)	4
GEO_TYPE_ID	Geography type (13=TAZ, 9=micro_zone)	9
GEO_ZONE	Zone ID	1
YEAR	Target year (set to 2007 for now)	2007
LU_REVISION_NO	Land use revision (scenario) number (set to 1 for now)	1
LU_MAJOR_VERSION	Land use revision (scenario) number sub major version number (set to 1 for now)	1
LU_MINOR_VERSION	Land use revision (scenario) number sub major sub minor version number (set to 0 for now)	0
VALUE	Target value	4.567

1.3 Running PopSyn II

Run PopSyn II with the runPopSynII.bat file which calls Java with the following command:

java -cp "popsyn2-1.0.jar;jars;." org.sandag.popsyn.PopulationSynthesizer "popsyn"*

1.4 Outputs

The key output of PopSyn II is the **household_allocated** table in the popSyn database. This table needs to be joined to the households and persons table in order to populate the synthesized households and persons with attributes. The **createSynHHPerFiles.sql** script is run to create the hhFile and personFile tables that are then saved to CSV for input to the ABM. After creating hhFile and personFile, get the tables with the popSyn ODBC connection and then save them to CSV file. The **serpm-popsyn-savefiles.R** script contains the R code to create the hhFile and personFile CSV files.

Beyond the synthesized population, the following summaries are created:

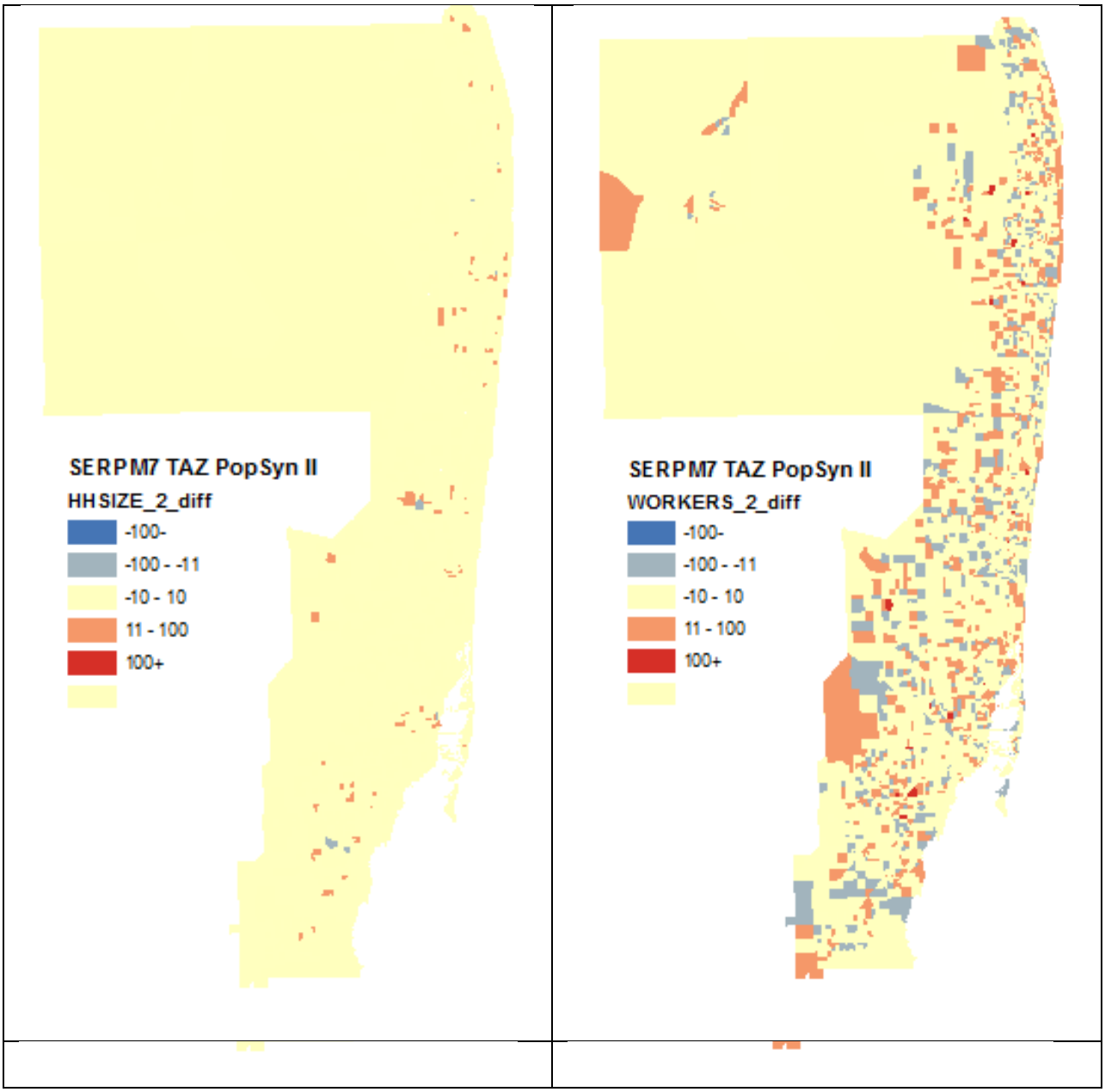
1) BalanceDetails_v<version>.csv – Results of the balancing by TAZ and control. Each row is a TAZ, control pair with the target value, synthesized PUMS value, and convergence indicator (ConStat field).
2) NonConvergedZone_n<num_nonconverged_zones>_v<version>.csv – The number of zones with non-converged controls by control.
3) validationStatsByPuma_v<version>.csv – The difference between the synthesized and ACS targets by PUMA. Each PUMA is a row and each validation metric is a column (and is identified by number).
4) validationStats_v<version>.csv – Each validation metric is a row with the observed (ACS) and estimated (PopSyn II) total, difference, standard deviation, minimum difference and maximum difference.

1.5 Validation

PopSyn II was run for SERPM for 2010 and a series of validation measures were prepared. This run includes all of the controls specified in the targets file above except for the race controls since the data for these are still being developed. The validation measures below were generated from the BalanceDetails_v<version>.csv file and compare the input TAZ level controls to the synthesized TAZ totals. These include both household and person level controls. The **serpm-popsyn-validation.R** script is used to generate the plot and the file used to generate the maps. The plot below plots the weighted (by total target households) mean percent difference as a dot and the weighted standard deviation percent difference as a vertical line by TAZ.

TC

In addition to the regional totals, a series of maps were prepared to spatially validate results by TAZ. These validation maps were prepared for number of two person households, two worker households, three person households, and persons age 65–79. These maps are below. The red TAZs are zones where PopSyn II is over the target, where as the blue zones are where PopSyn II is under the target.



1.6 FDOT Setup

The PopSyn II application is installed in C:\popsyn on the Windows Server 2008 system DOTSD4HQMODEL2.d4b.dot.state.fl.us. The SQL Server database backups are located on the database server in \\DOTSD4HQSQL\public\PB\popsyn\databases. Java 1.6.0_31 was installed to run PopSyn II.

SQL Server 2008 client tools are installed on the application server to provide access to the SQL Server instance DOTSD4HQSQL\DOTSD4PLEMO which listens on port 57467. The application server firewall was configured to allow "remote debugging"—in this case, that stands for "remote querying" from the application server. SQL Server is set to mixed mode authentication in support of the login popsyn.

Appendix A: Creating a PopSyn II Setup

This appendix discusses the steps to build the PopSyn II programs, as well as prepare a package for running it on a user machine.

1) Build popsyn2-1.0.jar with Eclipse
    a. Install Maven for Eclipse plugin
    b. Edit eclipse.ini to refer to JDK for –vm & installed JDKs under references
    c. Get settings.xml maven file from SANDAG and add it to the .m2 maven repository to connect to the SANDAG Maven repository in order to get dependencies.
    d. Checkout popsyn II project from SANDAG svn server
    e. Delete the Eclipse project created by checkout but keep the files
    f. Do File + Import + Existing Maven Project and select kept files
    g. Setup Eclipse compiler options: Eclipse + Project + Properties + Java Compiler + Uncheck "Use Compliance" and set to 1.6 for @override to work
    h. WriterDAO.java has the location of the SQL Server bulk load temp file hard coded to a SANDAG server. This was changed to be the version.workDir property instead.
    i. Create new income controls (0-25, 25-50, 50-75,75-100,100+)
        i. Edit control-target-config.xml income beans to refer to new income breaks
        ii. Run **addNewIncomeControls.sql** on the PopSyn db to create additional controls in the database
        iii. Add new control categories to org.sandag.popsyn.enums.ControlCategoryType
    j. Run maven build and check “skip tests” to create tags\v1.0\target\popsyn2-1.0.jar
    k. Rename the jar to zip and remove the following files so they can be read on-the-fly from the project folder:
        i. popsyn.properties
        ii. log4j.xml
        iii. applicationContext.xml
        iv. control-target-config.xml
        v. dataSource-config.xml
    l. Rename the zip file back to jar
    m. Copy the jar to a working folder
2) Copy over all the additional Java jar files from the maven repository to a jars folder in the working folder
    a. activation-1.1.jar
    b. aopalliance-1.0.jar
    c. c3p0-0.9.1.2.jar
    d. common-base-1.0.jar
    e. common-utils-1.3.jar
    f. commons-logging-1.1.1.jar
    g. ibatis-sqlmap-2.3.4.726.jar
    h. jms-1.1.jar
    i. jmxri-1.2.1.jar
    j. jmxtools-1.2.1.jar

k. junit-4.7.jar
l. log4j-1.2.15.jar
m. mail-1.4.jar
n. spring-aop-3.0.0.RELEASE.jar
o. spring-asm-3.0.0.RELEASE.jar
p. spring-beans-3.0.0.RELEASE.jar
q. spring-context-3.0.0.RELEASE.jar
r. spring-core-3.0.0.RELEASE.jar
s. spring-expression-3.0.0.RELEASE.jar
t. spring-jdbc-3.0.0.RELEASE.jar
u. spring-orm-3.0.0.RELEASE.jar
v. spring-tx-3.0.0.RELEASE.jar
w. sql-jdbc-4-3.0.jar
x. transcad-matrix-4.8.jar

3) Copy over all xml config files and put them in the working folder
   a. log4j.xml
   b. dataSource-config.xml
   c. sqlmap-config-popsyn.xml
   d. sqlmap-config-datacafe.xml
   e. control-target-config.xml
   f. applicationContext.xml
4) Create an org\sandag\popsyn\io\ibatis\sqlmap folder and copy over the xml sql map files
   a. Household.xml
   b. Target.xml
   c. TargetGrowthFactor.xml
   d. Validator.xml
   e. VersionManager.xml

Appendix B: PopSyn II Properties

This appendix discusses the steps to build the PopSyn II programs, as well as prepare a package for running it on a user machine.

Property	Description	Likely to Change	Example
jdbc.driverClassName	database connection specification	no	com.microsoft.sqlserver.jdbc.SQLServerDriver
jdbc.user	database username	no	popsyn
jdbc.password	database password	no	p0ps1n
jdbc.pool.initialPoolSize	database connection pool information	no	1
jdbc.pool.minPoolSize	database connection pool information	no	1
jdbc.pool.maxPoolSize	database connection pool information	no	5

jdbc.pool.acquireIncrement	database connection pool information	no	2
jdbc.pool.maxConnectionAge	database connection pool information	no	3600
jdbc.pool.maxIdleTime ExcessConnections	database connection pool information	no	120
jdbc.pool.idleConnection TestPeriod	database connection pool information	no	60
version.maxLoop	balancing iterations	no	8
version.convergence.criteria	convergence criteria	no	0.05
version.description	logging label	yes	Development Version
version.target.year	year of targets	yes	2007
version.source.id	PUMS data source id (4= 2007 ACS)	no	4
version.valCount	number of validation measures	no	86
version.targetMethod	method for targets	no	regular
version.regenerateTargets	regenerate ACS targets (i.e. run proc_regenerate_acs_targets.sql)	no	FALSE
version.selectZones	specify select zones to trace (1900-1910 or 1900'1905'1908 or NA for none)	yes	NA
version.workDir	project working directory	yes	\\\\w-ampdx-d-tfs05\\sandag_popsyn\\
version.target.lu.major.revision.no	target land use major revision (scenario) number	no	1
version.target.lu.minor.revision.no	target land use minor revision (scenario) number	no	0
version.target.geography.type	geography level for targets	no	TAZ_11
version.allocation.geography.type	geography level for allocation	no	MGRA_10
version.balance.geography.type	geography level for balancing	no	TAZ_11
version.small.value	small number	no	-999999999
version.large.value	large number	no	999999999
version.pumasToDrawFrom	PUMAs that intersect TAZs	yes	3501’3601’3602
version.numberOfExternalTazs	number of external tazs; used for indexing into TAZ geography table; can set to 0 if no external TAZs in database	no	0
version.hhAlAtt	allocation of TAZ households to micro-zones based on this micro-zone weight field	no	TOTAL_HOUSEHOLDS
version.gqAlAtt	allocation of TAZ group quarters to micro-zones based on this micro-zone weight field	no	TOTAL_GQ

version.useInitWeight	use PUMS weight for sampling (or set to 1 for false)	no	TRUE
version.balanceSelectionFactor	balance selection factor	no	0.2
version.balanceAllPumaSamples	balance all PUMA samples	no	FALSE
version.fixRandomSeed	use a fixed random seed	no	TRUE
version.randomSeed	random seed	no	1000
version.acsYear	use ACS PUMS records from what years; set to 0 for all years	no	0

CHAPTER 2 CT-RAMP Activity Based Modeling

Overview

This document describes how to setup and run the Southeast Florida Regional Planning Model, version 7.0 (SERPM 7.0). It also describes all inputs and outputs. This model has been developed to ensure that the regional transportation planning process can rely on forecasting tools that will be adequate for new socioeconomic environments and emerging planning challenges. It is equally suitable for conventional highway projects, transit projects, and various policy studies such as highway pricing and managed lane analysis.

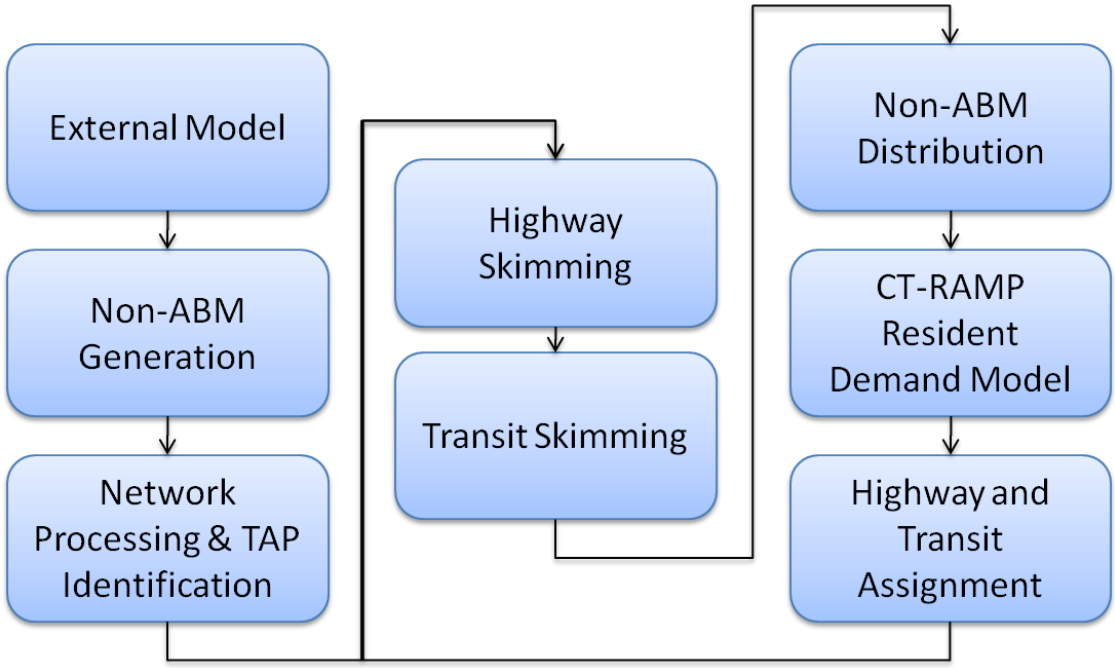
The SE Florida model is based on the CT-RAMP (Coordinated Travel Regional Activity-Based Modeling Platform) family of Activity-Based Models (ABM). The CT-RAMP framework, which is fully described in the model specification, adheres to the following basic principles:

- The CT-RAMP design corresponds to the most advanced principles of modeling individual travel choices with maximum behavioral realism. In particular, it addresses both household-level and person-level travel choices including intra-household interactions between household members.
- Operates at a detailed temporal (half-hourly) level, and considers congestion and pricing effects on time-of-day and peak spreading.
- Reflects and responds to detailed demographic information, including household structure, aging, changes in wealth, and other key attributes.
- Is implemented in the Common Modeling Framework (CMF), an open-source library developed by Parsons Brinckerhoff (PB) specifically for implementing advanced travel demand models.
- Offers sensitivity to demographic and socio-economic changes observed or expected in the dynamic SE Florida metropolitan region. This is ensured by the enhanced and flexible population synthesis procedures as well as by the fine level of model segmentation. In particular, the SERPM ABM incorporates different household, family, and housing types including a detail analysis of different household compositions in their relation to activity-travel patterns.
- Accounts for the full set of travel modes. Our experience with previously developed ABMs has shown that mode choice is one of the least transferable model components, because each region has a specific mix of modes developed in the context of the regional urban conditions.
- Integrates with other model components. The CT-RAMP model is one component (person travel) and is integrated with other components such as the existing SERPM truck model.
- Provides detailed inputs to traffic micro-simulation software. The CT-RAMP models operates at a half-hour time scale, which can provide detailed inputs to traffic micro-simulation software for engineering-level analysis of corridor and intersection design.

SERPM 7.0 was developed by transferring the model developed for the San Diego Association of Governments (SANDAG). As described below, the model system is primarily implemented with

three software technologies: Cube, Java, and SQL Server. Some of the model components are implemented in Java using the CMF libraries, while other, aggregate model components are implemented in Cube Voyager script, along with network skimming and assignment procedures. The Population Synthesizer is implemented with Java and SQL Server. The overall model is run from within the Cube Application Manager interface. Figure 1 shows the overall model system. The model starts by generating non-ABM trips for external, airport, truck and hotel visitors travel. Next the model completes a series of network processing routines to prepare the network for skimming and assignment. Included in the network processing step is the identification of transit access point (TAP) nodes, which are used as transit virtual zones centroids for assignment. The highway and transit networks are then skimmed to produce network level-of-service matrices for use in later steps of the model. Using the new skim matrices, the non-ABM trips are distributed and CT-RAMP is run. CT-RAMP is run from within Cube and runs a series of Cube scripts and Java programs via the command line. CT-RAMP outputs microsimulated lists of trips, as well as aggregate trip demand matrices for assignment in Cube. The demand matrices are assigned to the networks in the final step. New level-of-service matrices are generated in the final step and then feedback into the model for additional iterations of the model.

Figure 1: SERPM ABM Model System



Treatment of Space

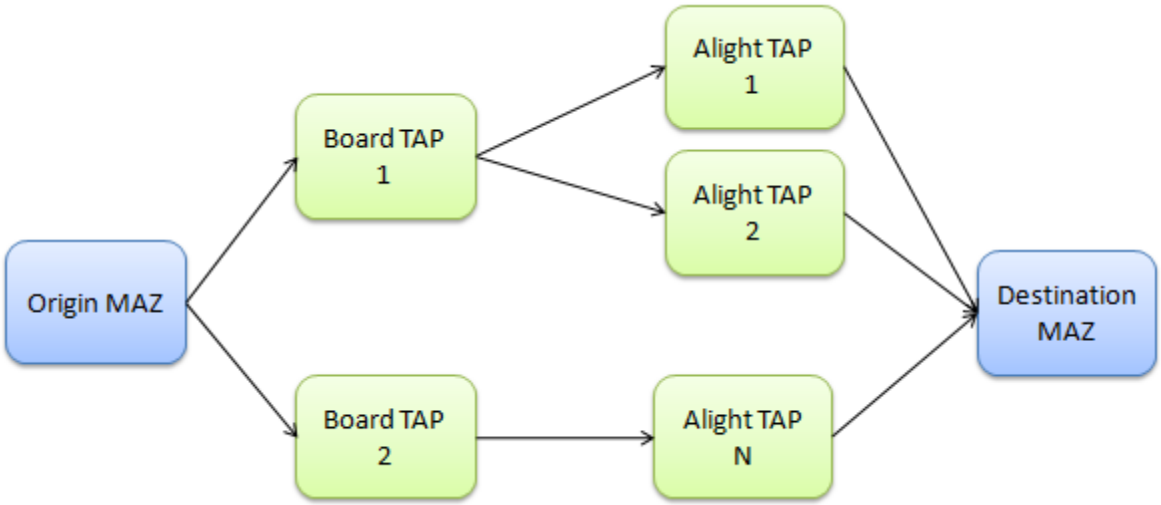
Activity-based and tour-based models can exploit more explicit geographic and location information, but the advantages of additional spatial detail must be balanced against the additional efforts required to develop zone and associated network information at this level of detail, as well as against the increases in model runtime associated primarily with path-building and assignment.

The use of a spatially disaggregate zone system helps ensure appropriate model sensitivity. Use of large zones may produce aggregation biases, especially in destination choice, where the use of aggregate data can lead to illogical parameter estimates due to reduced variation in estimation data,

and in mode choice, where modal access may be distorted. Smaller zones help minimize these effects, and can also support more detailed network assignments. Strategies to address the modal access limitations of large zones through the use of transit sub-zonal procedures are discussed in the transit network section of this document.

The SERPM ABM operates on micro-zones (MAZs).The current SE Florida MAZ system design consists of approximately 12,000 zones. To avoid computational burden, SERPM 7.0 relies on a 4,200 Transportation Analysis Zone (TAZ) system for highway skims and assignment, but performs transit calculations at the more detailed MAZ level. This is accomplished by generalizing transit stops into pseudo-TAZs called Transit Access Points (TAPs), and relying on Cube to generate TAP-TAP skims such as in-vehicle time, first wait, transfer wait, and fare. All access and egress calculations and ultimate Origin MAZ – Boarding TAP – Alighting TAP- Destination MAZ path are computed within custom-built software, and rely upon detailed geographic information regarding MAZ-TAP distances and accessibilities. A graphical depiction of the MAZ – TAP transit calculations is given in Figure 2.

Figure 2: MAZ – TAP Transit Accessibility



General Model Design

The general design of the SERPM 7.0 CT-RAMP implementation is shown in Figure 3 below. The following outline describes the basic sequence of sub-models and associated travel choices. For a more detailed specification of the model design, please refer to the model specification.

1. ***Synthetic population:***
   - 1.1. Zonal distributions of population by controlled variables
   - 1.2. Household residential location choice (allocation to zones)
2. ***Long term level:***
   - 2.1. Usual location for each mandatory activity for each relevant household member (workplace/university/school)
3. ***Mobility Level:***

3.1. Free Parking Eligibility (determines whether workers pay to park if workplace is an MGRA with parking cost)

3.2. Household car ownership

3.3. Transponder ownership for use of toll lanes

4. ***Daily pattern/schedule level:***

4.1. Daily pattern type for each household member (main activity combination, at home versus on tour) with a linkage of choices across various person categories, and an indicator at the household level determining whether one or more joint tours is made by 2 or more household members

4.2. Individual mandatory activities/tours for each household member (note that locations of mandatory tours have already been determined in long-term choice model)

4.2.1. Frequency of mandatory tours

4.2.2. Mandatory tour time of day (departure/arrival time combination)

4.3. Joint travel tours (conditional upon the available time window left for each person after the scheduling of mandatory activities)

4.3.1. Joint tour frequency

4.3.2. Travel party composition (adults, children, mixed)

4.3.3. Person participation in each joint tour

4.3.4. Primary destination for each joint tour

4.3.5. Joint tour time of day (departure/arrival time combination)

4.4. Individual non-mandatory activities/tours (conditional upon the available time window left for each person after the scheduling of mandatory and joint activities)

4.4.1. Person frequency of non-mandatory tours

4.4.2. Non-mandatory tour primary destination

4.4.3. Non-mandatory tour departure/arrival time

4.5. Individual at-work subtours (conditional upon the available time window within the work tour duration)

4.5.1. Person frequency of at-work sub-tours

4.5.2. Primary destination for each at-work sub-tour

4.5.3. At-work sub-tour departure/arrival time

5. ***Tour level***

5.1. Tour mode

5.2. Frequency of secondary stops (and their purpose)

5.3. Location of secondary stops

5.4. Departure time for secondary stops

6. *Trip level*

6.1. Trip mode choice conditional upon the tour mode

6.2. Auto trip parking location choice

6.3. Trip assignment

Choices that relate to the entire household or a group of household members and assume explicit modeling of intra-household interactions (sub-models 3.2, 4.1, 4.3.1, 4.3.2) are shadowed in Figure 3. The other models are assumed to be individual-based.

The model system uses synthetic household population as a base input (sub-model 1). It is followed by long-term choices that relate to the usual workplace/university/school for each worker and student (sub-model 2.1). Medium-term mobility choices relate to free parking eligibility for workers in parking constrained areas (sub-model 3.1), household car ownership (sub-model 3.2), and transponder ownership (sub-model 3.3). The daily activity pattern type of each household member (model 4.1) is the first travel-related sub-model in the modeling hierarchy. This model classifies daily patterns by three types: 1) mandatory (that includes at least one out-of-home mandatory activity), 2) non-mandatory (that includes at least one out-of-home non-mandatory activity, but does not include out-of-home mandatory activities), and 3) home (that does not include any out-of-home activity and travel). However, the pattern type sub-model leaves open the frequency of tours for mandatory and non-mandatory purposes (maintenance, discretionary) since these sub-models are applied later in the model sequence.

The pattern choice set contains a non-travel option in which the person can be engaged in in-home activity only (purposely or because of being sick) or can be out of town. In the model system application, a person who chooses a non- travel pattern is not considered further in the modeling stream. Daily pattern-type choices of the household members are linked in such a way that decisions made by some members are reflected in the decisions made by the other members.

The next set of sub-models (4.2.1-4.2.2) defines the frequency and time-of-day for each mandatory tour. The scheduling of mandatory activities is generally considered a higher priority decision than any decision regarding non-mandatory activities for either the same person or for the other household members. As the result of the mandatory activity scheduling, “residual time windows” are calculated for each person and their overlaps across household members are estimated. Time window overlaps, which are left in the daily schedule after the mandatory commitment of the household members has been made, constitute the potential for joint activity and travel.

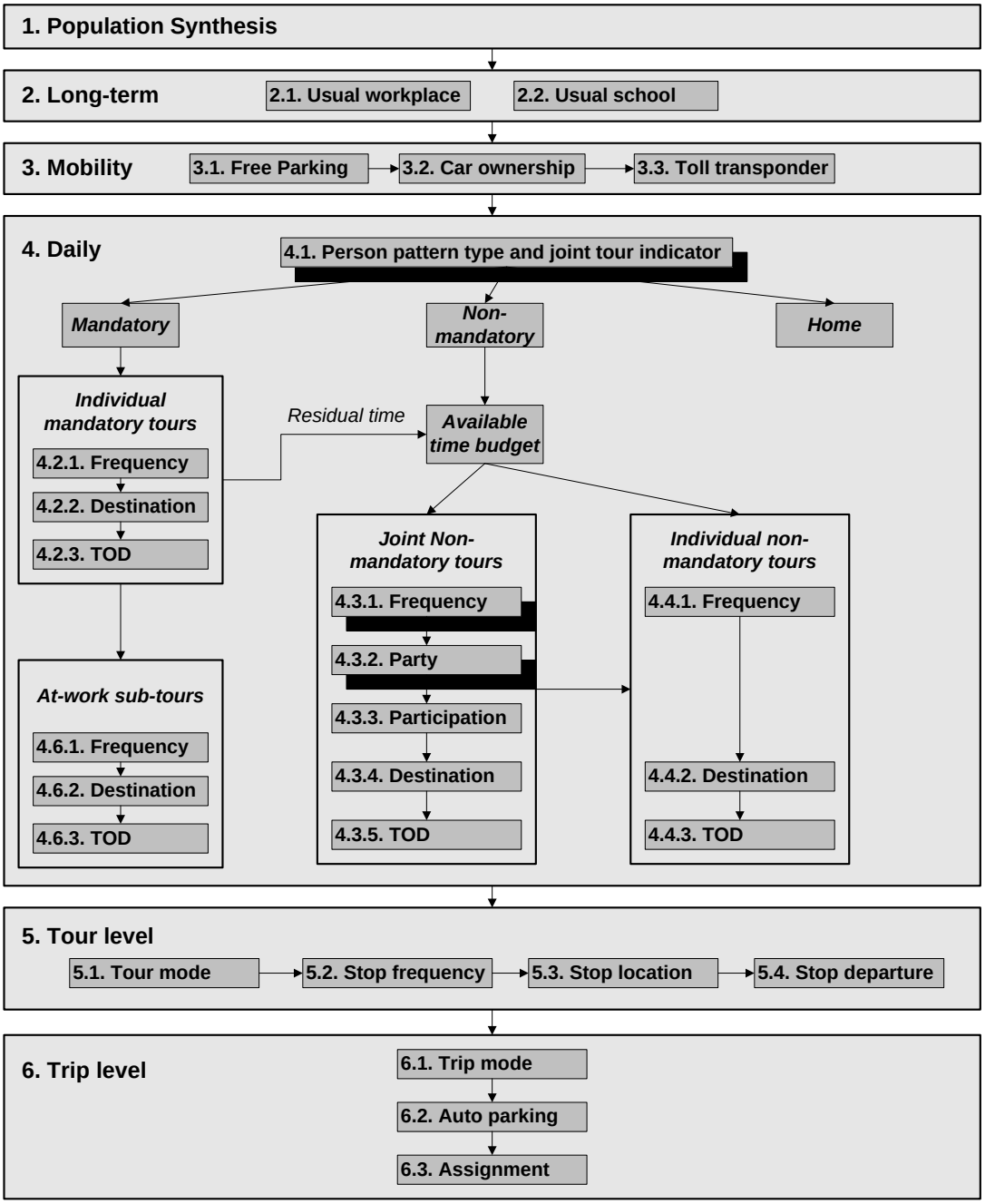
At-work sub-tours are modeled next, taking into account the time-window constraints imposed by their parent work tours (sub-models 4.5.1-4.5.3).

The next major model component relates to joint household travel. This component produces a number of joint tours by travel purpose for the entire household (4.3.1), travel party composition in terms of adults and children (4.3.2), and then defines the participation of each household member in each joint household tour (4.3.3). Model 4.3.1 only generates joint tours if the CDAP model 4.1 generates a joint tour indicator for the household. It is followed by choice of destination (4.3.4) and time-of-day (4.3.5).

The next stage relates to non-mandatory tours--shopping, escort, other maintenance, visit, eating out and other discretionary tours. Non-mandatory tours are generated by individuals (4.4.1). Their destination and time of day are chosen next (4.4.2 and 4.4.3).

The next set of sub-models relate to the tour-level details on mode (5.1), exact number of intermediate stops on each half-tour (5.2), stop location (5.3), and stop departure time (5.4). It is followed by the last set of sub-models that add details for each trip including trip mode details (6.1) and parking location for auto trips (6.2). The trips are then assigned to highway and transit networks depending on trip mode (6.3).

Figure 3: Basic Model Design and Linkage between Sub-Models



Computing Environment

The hardware and software used to execute the entire modeling system is described in this section. FDOT currently has the model configured to run on two machines, which together represent one computing cluster. It is important to note that the computing configuration is highly configurable and flexible and the required computing power required for a specific analysis could vary dramatically. The model can be configured to run on one machine or multiple machines.

Hardware

- Operating system: Microsoft Windows Server 2008 R2 Enterprise, 64-bit
- Processors: Two Intel Xeon X5690 @ 3.47 GHz 6 core
- Memory (RAM): 144 GB
- Name: dotsd4hqmodel1.d4b.dot.state.fl.us; dotsd4hqmodel2.d4b.dot.state.fl.us

Software

The following software is required to execute the SERPM 7.0.

Cube 6.1+

The travel model currently uses Version 6.1 of Cube. Cube is used to build network skims, manipulate networks, manipulate matrices, perform assignments, and run non-ABM models. The Java software, as discussed below, needs to access matrix data in Cube format. In order to do this, the new 64bit Cube matrix I/O DLL is used. The details of setting up and configuring the software are presented in the Setup and Configuration section.

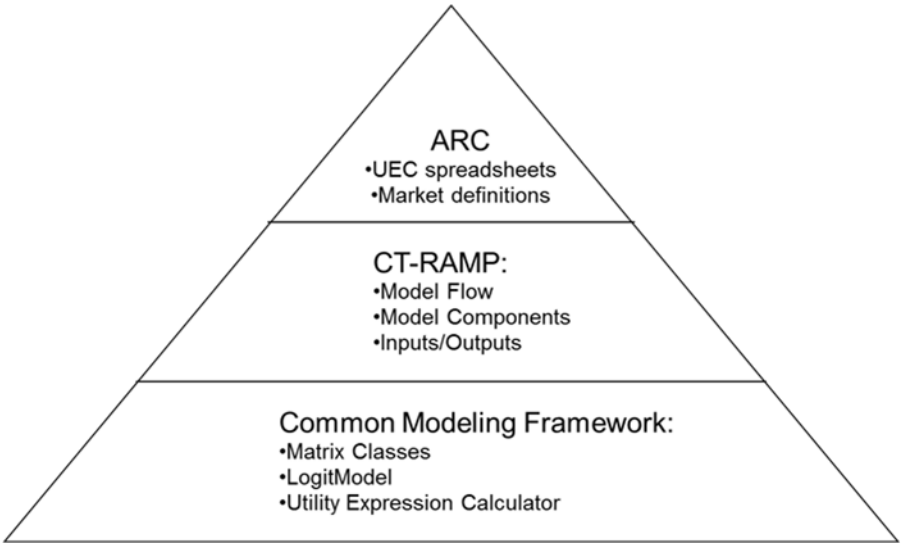
Java and CT-RAMP

The modeling system requires 64bit Java for CT-RAMP. As shown in Figure 4, the CT-RAMP software for the microsimulation components of the model relies on the Common Modeling Framework (CMF), a collection of Java libraries specifically designed for the implementation of disaggregate travel demand models. The FDOT ABM utilizes the CT-RAMP Java package, which contains model logic, choice model structure, and model flow, while utility equations and model inputs and outputs specific to SERPM are contained in Utility Expression Calculator (UEC) files. These Excel-based files open up the models so the parameters, input filenames, etc can be easily accessed which helps prevent errors and makes the model equations more accessible.

Microsoft Excel (not required but helpful)

The CT-RAMP UEC files are best edited with Microsoft Excel. It is helpful to have Excel installed so that the spreadsheets can be opened, though it is not essential for running the model system.

Figure 4: Disaggregate Travel Demand Model Software Components



System Design

The CT-RAMP model software uses distributed computing, which allows the models to be distributed across multiple threads and multiple computers to improve runtimes. The model currently uses two machines, but the software can be configured to utilize fewer or more machines. The modeling system is implemented with Cube and Java, with Cube Cluster and Java Parallel Processing Framework (JPPF) for distributed/threaded computing respectively. JPPF is an open-source Java package. The system design, which is summarized in Figure 5 below, consists of the following basic setup:

1) A main computer which:
    a. runs the main Cube work, including some of the Cube Cluster processes
    b. runs the main CT-RAMP Java process (SERPM Tour Based Model)
    c. runs the Household Data Manager Java process
    d. runs the Matrix Data Manager Java process
    e. runs the JPPF Driver process which manages communicating with the JPPF nodes
    f. optionally runs a JPPF node process which listens for tasks from the JPPF driver
2) One additional node computer which listens for tasks from the JPPF driver

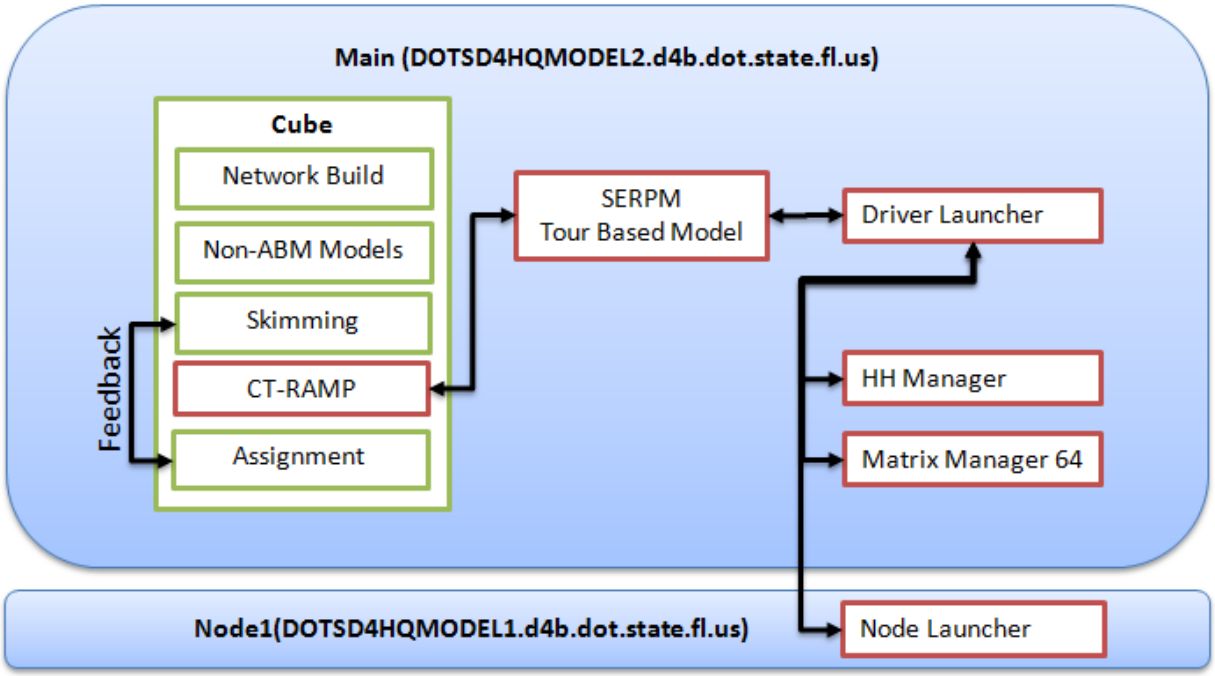
Household Data Manager and Matrix Data Manager

Before executing a model run, the travel model requires a Matrix Manager and a Household Manager to be created. The Matrix Manager resides on the same machine as the main CT-RAMP client process. Although the Matrix Manager can be instantiated on any machine, the machine must have a Cube installation. This is important because the software utilizes the Cube matrix I/O DLL. The Matrix Manager is tasked with managing all of the skim matrices used by the CT-RAMP choice models. When a skim is needed, a request is made to the Matrix Manager, which then reads the required skim from disk and stores it in memory. Once in memory, each matrix is available to any other JPPF node process that may need it.

The Household Manager is tasked with storing the simulated households, as well the persons, tours, and trips in those households. The Household Manager provides the JPPF nodes with information

regarding the households for which the JPPF nodes are applying choice models and stores the resulting information computed by the JPPF nodes. To help keep run time down, the synthetic population is read from disk and stored in memory at the beginning of the application and then continuously updated as choice models are completed and iterations are performed. When the last iteration is complete, the necessary information is written to disk. Both the Household Manager and Matrix Manager have substantial memory footprints that depend on the forecasted population size and the amount network skim matrices.

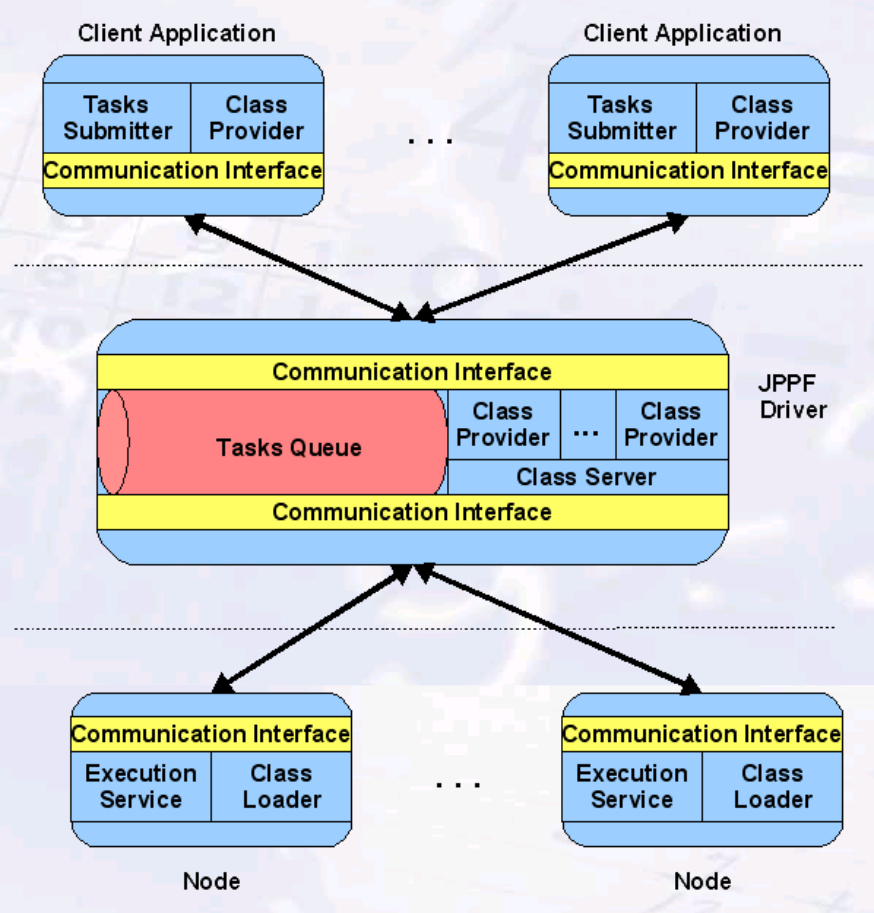
Figure 5: System Setup



Distributed Computing

CT-RAMP threads the application of ABM models into groups of households. Given that CT-RAMP is implemented in Java, it uses JPPF to manage the distribution of tasks. As illustrated in Figure 6 below, the JPPF framework consists of 3 main parts: a driver, a set of one or more nodes, and a client. The client is in this case CT-RAMP. The nodes are also additional separate processes, typically one per computer. The driver is a separate server process that is run on one of the cluster machines. The driver is a facilitator that receives tasks from the client application, sends them to node processes, receives results from nodes, and returns those back to the client.

Figure 6: JPPF Framework



Node processes receive tasks of calculations, perform those calculations, and return results. Nodes are configured through a properties file to communicate with the driver process upon their start-up. A typical configuration might be to set memory equal to 32 GB and threads equal to 12 (for a 12 core machine). The majority of parallel computations in the CT-RAMP implementation occur through tasks executed in parallel on nodes.

The driver process uses logic contained in the JPPF framework to balance computational loads across Java Virtual Machines running on the nodes in the cluster. The driver receives tasks from the client application and submits them in bundles to the nodes. The driver also retrieves class files from the client application and passes those to the nodes, as needed by the nodes. Additional nodes can therefore be added by simply editing two properties files and running a Java command.

The client application, which is called by the main Cube model script and configured through a properties file, communicates with the driver as described above. The client application is responsible for creating task objects that can be run in parallel and submitting those to the driver. The tasks are submitted to the driver and the driver assembles the X number of tasks into bundles and submits them to nodes that have notified the driver that they are part of the cluster. As the nodes complete the tasks, the driver receives their results and submits new bundles, while balancing the submission of bundles to keep the nodes uniformly busy.

Setup and Configuration

This section provides details on setting up the travel model to run on a cluster of computers, including descriptions of the necessary configuration files.

Step 1: Create the Project Folder

SERPM 7 is delivered as a zip file containing four directories and the serpm7.cat Cube application manager file. The four directories are:

1) ctramp – CT-RAMP code, scripts, and configuration files
    a. application – Java code and Cube scripts for CT-RAMP
    b. config – JPPF and CT-RAMP logging settings
    c. popsyn – helper files to run PopSyn to create the HH and Person input files
    d. reports_sql – SQL scripts for the reporting module
    e. uec – UEC model specification files
2) cube – Cube application manager programs and scripts
3) input – All inputs for the model run
    a. IN-2010R – working scenario inputs
        i. Ctramp – CT-RAMP inputs
            1. Visitor – additional visitor model inputs
        ii. FromS67 – SERPM 6.7 inputs
        iii. Transit – Transit inputs
    b. seed_net_skm – seed "warm start" network and skims for feedback iteration one
    c. TAnalyst – truck matrix
    d. TAZshape – TAZ shapefile
    e. MAZshape = MAZ shapefile
4) Installers – Program installers for setting up on a new machine
5) output – All outputs for a model run
    a. OUT-2010R – working scenario outputs

After unzipping the file, copy the contents to C:\serpm7 to setup a project folder. It is best if this folder is shared with "Everyone", since Cube will need read/write access to it. Additional user restrictions may cause problems.

Step 2: Configure the Machines

In order for Cube to run CT-RAMP and for CT-RAMP to read/write Cube matrices, some configuration of the modeling machines is required. The following must be done to configure each machine:

1) Add the CubeVoyager installation folder (C:\Program Files (x86)\Citilabs\CubeVoyager) to the Windows system PATH variable
2) Add c:\serpm7\ctramp\application to the Windows system PATH variable

3) Ensure 64bit Java is installed and is invoked on the command line by calling “java”. This can be checked with "java -version" on the command line and should return “64-Bit server”.
4) Ensure Cube 64bit matrix I/O DLLs are installed by running the VoyagerFileAPIInstaller.msi installer in the installers folder. Finally, run the vcredist_x64.exe installer to install additional Microsoft DLLs required by the Cube 64bit matrix I/O DLLs.
5) Add the mleg.exe from the SERPM 7 template Cube folder to the Cube installer folder (C:\Program Files (x86)\Citilabs\Cube).
6) If needed, copy the LCISRegs.exe from the CubeVoyager installation folder into the Tpplus installation folder. This may be required in order to register the correct installation of Cube with the license in order to use the Cube matrix I/O DLL.
7) Create a mapped network drive (Y:) to the c:\serpm7 project directory on the main machine. This will be used as the project folder across the multiple machines.

Step 3: Configure the JPPF Services

After creating the project folder and copying over the template files, the JPPF services need to be configured in order to run the model in a distributed fashion. If the model is not run across a cluster, then only the following file in the config folder needs to be edited:

1) jppf-client.properties – JPPF client properties file for running the model on 1 machine
    a. jppf.local.execution.threads = 20 #number of processing threads

If the model is to be run in distributed model, then the following properties files in the config folder need to be edited:

2) jppf-clientDistributed.properties – JPPF client properties file for running the model in a distributed fashion
    a. driver1.jppf.server.host = 156.75.49.78 #IP of main computer
    b. jppf.local.execution.enabled = true #use processors on the main machine as well
    c. jppf.local.execution.threads = 12 #number of processing threads

3) jppf-driver.properties - JPPF driver properties file for managing distributed processing
    a. jppf.server.host = 156.75.49.78 #IP of main computer
    b. jppf.management.host = 156.75.49.78 #IP of main computer

4) jppf-node1.properties - JPPF worker node properties file.
    a. jppf.server.host = 156.75.49.78 #IP of main computer
    b. jppf.management.host = 156.75.49.78 #IP of main computer
    c. processing.threads = 12 #number of computing cores on node
    d. other.jvm.options = -Xms24000m -Xmx24000m #memory on node

Step 4: Configure the Run Component Files

The final CT-RAMP setup step is to configure the run component files. These files are located in the ctramp folder and are described below. The following setting may need to be adjusted:

1) serpm_abm.properties – Java properties file. There are a number of settings in this file that need to be revised depending on the model run. See the CT-RAMP Properties File section later in this document for more information. The only settings that are specific to setting up a new model run are below:
    a. RunModel.MatrixServerAddress=156.75.49.78 #IP of main computer
    b. RunModel.HouseholdServerAddress= 156.75.49.78 #IP of main computer
2) runCreateAccessFiles.cmd – Runs the following Cube scripts to create additional inputs for CT-RAMP:
    a. save_maz_dbf.s – converts the MAZ data CSV file into DBF form for use by Cube
    b. abm_access_files.s – calculates a number of CT-RAMP input files:
        i. tap.ptype – TAP file
        ii. walkacc - MAZ to TAP potential alternatives file
        iii. mgrataz – MAZ to MAZ potential alternatives file
        iv. access.prp – MAZ to TAP drive access potential alternatives file
    c. maz_densities.s – Calculates intersection, total employment, retail employment, household, and population density measures at the MAZ level for input into CT-RAMP.
3) runSERPMAbm.cmd – Runs CT-RAMP resident and visitor ABM models
    a. This script is currently set to run CT-RAMP on 1 machine. To run it in distributed mode, comment out (i.e. rem) the following line:

        java -server -Xms100000m -Xmx100000m SERPMTourBasedModel …

        and un-comment out (i.e. remove the rem) the following line:

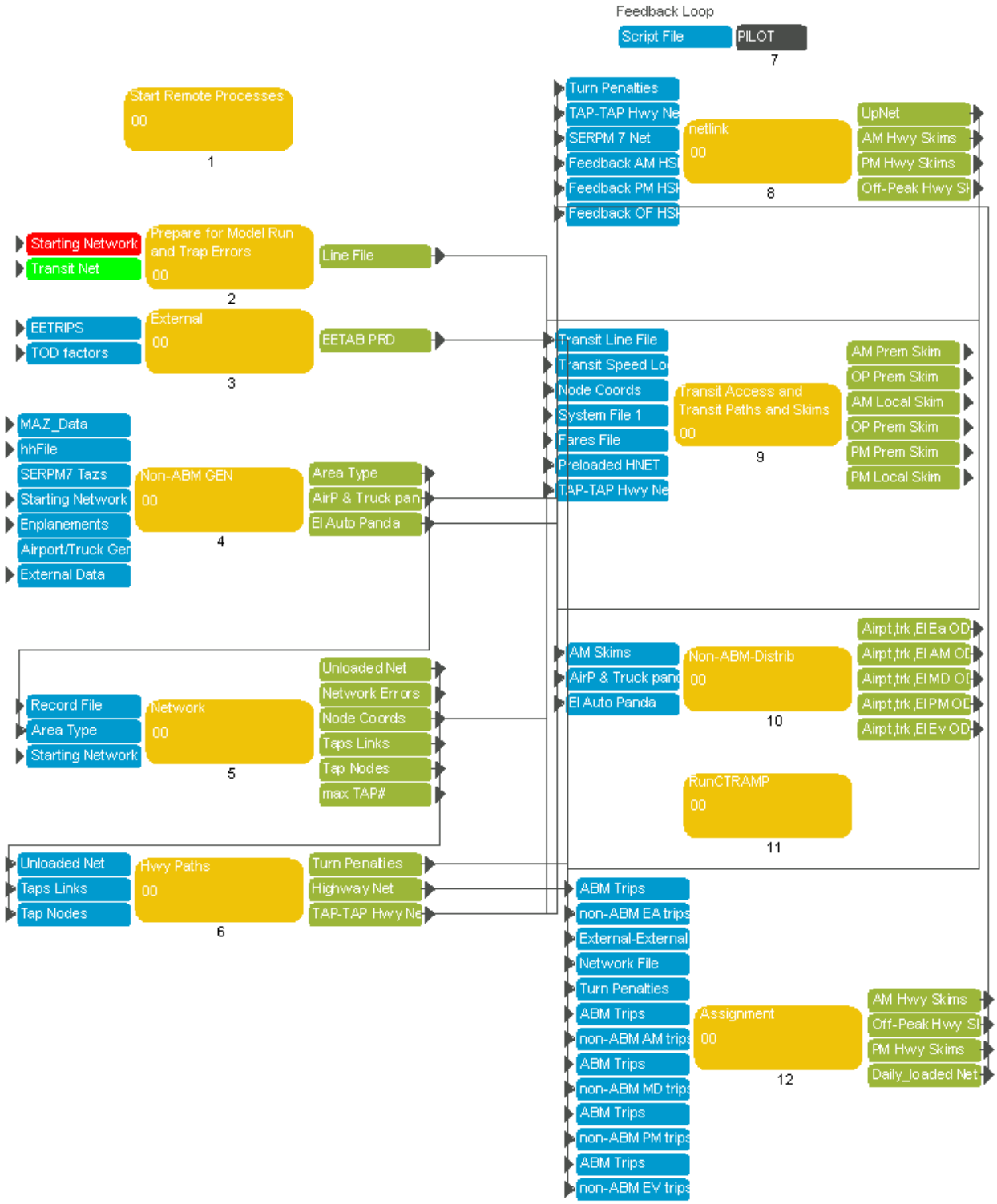
        rem java -server -Xms80000m -Xmx80000m SERPMTourBasedModel …

4) runCreateDemandMatrices.cmd – Run the Java program to create a series of Cube demand matrices based on the microsimulated trip lists and then run the Cube merge_*.s scripts to create the final assignment matrices for Cube.
    a. VOYAGER="C:\Program Files (x86)\Citilabs\CubeVoyager\Voyager.exe" #Voyager

Step 5: Configure SERPM 7 in Application Manager

Open the serpm7.cat application manager file in Cube to ensure the file references are setup correctly. The catalog file should be setup to assume C:\serpm7 as the root location. If not, then Cube will ask the user if it is ok to update the dependent scripts to the new file location. The application manager flow chart is illustrated in Figure 7 below.

Figure 7: SERPM 7 Application Manager



Step 6: Create Seed Skims

The final step is to create the seed skims (X_AMHSKIMS.mat, X_OFHSKIMS.mat, X_PMHSKIMS.mat). This can be done via Cube, by double clicking on the "netlink" procedure step (step 8 in the diagram

above) and running the "HIGHWAY" procedure (step 0), located below the main application, under "Build Highway Skims". Running this step should produce the X_**** skims in the /input/seed_net_skm folder.

Input Files

The table below contains brief descriptions of the input files required to execute SERPM 7. See the Input Files Data Dictionary section for details on each input file.

File name	Purpose	File type
maz_data.csv	Land use data	CSV
hhFile.csv	Synthetic households	CSV
personFile.csv	Synthetic persons	CSV
pecas_occ_occcen1.csv	PECAS occupation categories mapping to Census SOC codes	CSV
activity_code_indcen.csv	Census industry code mapping	CSV
tc_avgtt.csv	Transponder ownership model TAZ attributes	CSV
zone.term	TAZ terminal times	Text file – space delimited
SERPM_NETS.mdb	Highway network {Input Highway Net} and transit network {TLines} Cube personal geodatabase networks	Access database
Full-SERPM_Navteq_NAD83.net	All streets network	Net
connectors.txt	MAZ connectors	CSV
wkacc	TAPs within walk distance from MAZs. **(will be calculated by the abm_access_files.s script)**	Text file – space delimited
access.prp	TAPs within drive distance from TAZs. **(will be calculated by the abm_access_files.s script)**	Text file – space delimited
tap.ptype	TAP attribute table. **(will be calculated by the abm_access_files.s script)**	Text file – space delimited
mgrataz.walk	MAZs within walk distance from MAZs. **(will be calculated by the abm_access_files.s script)**	Text file – space delimited
EETRIPS.MAT	Daily external-external vehicle trip table, defined outside the model.	Cube matrix
TOD.DBF	Time-of-day and directional factors for EE,	DBF

File name	Purpose	File type
	Truck, Airport and EI trips	
Enplane.dbf	Daily enplanements for MIA, FLL, and PBI, by TAZ (3 records).	DBF
airport_truck_rates.dbf	Airport and truck trip generation rates	DBF
EIdata.dbf	External station data (like FSUTMS Zdata4)	DBF
MVFACTORS	FSUTMS Vfactors file	Text
MTURNDEF	Turn penalties and control file	Text
transit\TSPEED.DBF	Dwell time assumptions by USERA4 group, which is used to calculate transit running times	DBF
TSYSD.PTS	Public Transport (PT) system file	Text
TFARES.FAR	Transit system fare definitions	Text
daily_loaded.net	Loaded highway network file. On the first feedback iteration it is a “seed” from an earlier run to provide initial link times for the transit network. In subsequent iterations, it contains times calculated by the method of successive averages (MSA) from earlier iterations.	Net
X_AMHSKIMS.mat	AM period congested highway skims. On the first feedback iteration it is a “seed” from an earlier run to provide initial link times for the transit network. In subsequent iterations, the skims are based on MSA times from earlier iterations.	Cube matrix
X_OFHSKIMS.mat	Off-peak period congested highway skims. On the first feedback iteration it is a “seed” from an earlier run to provide initial link times for the transit network. In subsequent iterations, the skims are based on MSA times from earlier iterations.	Cube matrix
X_PMHSKIMS.mat	PM period congested highway skims. On the first feedback iteration it is a “seed” from an earlier run to provide initial link times for the transit network. In subsequent iterations, the skims are based on MSA times from earlier iterations.	Cube matrix
visitor_businessFrequency.csv	Visitor model tour frequency distribution for business travelers: visitor.business.tour.file	CSV

File name	Purpose	File type
visitor_personalFrequency.csv	Visitor model tour frequency distribution for personal travelers: visitor.personal.tour.file	CSV
visitor_partySize.csv	Visitor model party size distribution: visitor.partySize.file	CSV
visitor_autoAvailable.csv	Visitor model auto availability distribution: visitor.autoAvailable.file	CSV
visitor_income.csv	Visitor model income distribution: visitor.income.file	CSV
visitor_tourTOD.csv	Visitor model tour time-of-day distribution: visitor.tour.tod.file	CSV
visitor_stopFrequency.csv	Visitor model stop frequency distribution: visitor.stop.frequency.file	CSV
visitor_stopPurpose.csv	Visitor model stop purpose distribution: visitor.stop.purpose.file	CSV
visitor_outboundStopDuration.csv	Visitor model time-of-day offsets for outbound stops: visitor.stop.outbound.duration.file	CSV
visitor_inboundStopDuration.csv	Visitor model time-of-day offsets for inbound stops: visitor.stop.inbound.duration.file	CSV

Input Files Data Dictionary

Table 1: MAZ Data File – maz_data.csv

Column Name	Description
mgra	MAZ
TAZ	TAZ
hh	total number of households
pop	total population
emp_self	Self-employed
emp_ag	Agriculture
emp_const_non_bldg_prod	Construction Non-Building production (including mining)
emp_const_non_bldg_office	Construction Non-Building office support (including mining)
emp_utilities_prod	Utilities production
emp_utilities_office	Utilities office support
emp_const_bldg_prod	Construction of Buildings production

Column Name	Description
emp_const_bldg_office	Construction of Buildings office support
emp_mfg_prod	Manufacturing production
emp_mfg_office	Manufacturing office support
emp_whsle_whs	Wholesale and Warehousing
emp_trans	Transportation Activity
emp_retail	Retail Activity
emp_prof_bus_svcs	Professional and Business Services
emp_prof_bus_svcs_bldg_maint	Professional and Business Services (Building Maintenance)
emp_pvt_ed_k12	Private Education K-12
emp_pvt_ed_post_k12_oth	Private Education Post-Secondary (Post K-12) and Other
emp_health	Health Services
emp_personal_svcs_office	Personal Services Office Based
emp_amusement	Amusement Services
emp_hotel	Hotels and Motels
emp_restaurant_bar	Restaurants and Bars
emp_personal_svcs_retail	Personal Services Retail Based
emp_religious	Religious Activity
emp_pvt_hh	Private Households
emp_state_local_gov_ent	State and Local Government Enterprises Activity
emp_scrap_other	Scrap other
emp_fed_non_mil	Federal Non-Military Activity
emp_fed_mil	Federal Military Activity
emp_state_local_gov_blue	State and Local Government Non-Education Activity production
emp_state_local_gov_white	State and Local Government Non-Education Activity office support
emp_public_ed	Public Education K-12 and other
emp_own_occ_dwell_mgmt	Owner-Occupied Dwellings Management and Maintenance Activity
emp_fed_gov_accts	Federal Government Accounts
emp_st_lcl_gov_accts	State and Local Government Accounts
emp_cap_accts	Capital Accounts

Column Name	Description
emp_total	Total employment
EnrollGradeKto8	Grade School K-8 enrollment
EnrollGrade9to12	Grade School 9-12 enrollment
collegeEnroll	Major College enrollment
otherCollegeEnroll	Other College enrollment
AdultSchEnrl	Adult School enrollment
ech_dist	Elementary school district
hch_dist	High school district
parkarea	1: Trips with destinations in this MAZ may choose to park in a different MAZ, parking charges apply (downtown) 2: Trips with destinations in parkarea 1 may choose to park in this MAZ, parking charges might apply (quarter mile buffer around downtown) 3: Only trips with destinations in this MAZ may park here, parking charges apply (outside downtown paid parking, only show cost no capacity issue) 4: Only trips with destinations in this MAZ may park here, parking charges do not apply (outside downtown, free parking)
hstallsoth	Number of stalls allowing hourly parking for trips with destinations in other MAZs
hstallssam	Number of stalls allowing hourly parking for trips with destinations in the same MAZ
hparkcost	Average cost of parking for one hour in hourly stalls in this MAZ, dollars
numfreehrs	Number of hours of free parking allowed before parking charges begin in hourly stalls
dstallsoth	Stalls allowing daily parking for trips with destinations in other MAZs
dstallssam	Stalls allowing daily parking for trips with destinations in the same MAZ
dparkcost	Average cost of parking for one day in daily stalls, dollars
mstallsoth	Stalls allowing monthly parking for trips with destinations in other MAZs
mstallssam	Stalls allowing monthly parking for trips with destinations in the same MAZ

Column Name	Description
mparkcost	Average cost of parking for one day in monthly stalls, amortized over 22 workdays, dollars
TotInt	Total intersections (calculated by maz_densities.s script)
DUDen	Dwelling unit density (calculated by maz_densities.s script)
EmpDen	Employment density (calculated by maz_densities.s script)
PopDen	Population density (calculated by maz_densities.s script)
RetEmpDen	Retail employment density (calculated by maz_densities.s script)
TotIntBin	Total intersection bin (calculated by maz_densities.s script)
EmpDenBin	Employment density bin (calculated by maz_densities.s script)
DuDenBin	Dwelling unit density bin (calculated by maz_densities.s script)
POINT_X	Centroid X coordinate
POINT_Y	Centroid Y coordinate
ACRES	MAZ acres
HotelRoomTotal	Total hotel rooms
mall_flag	Shopping mall flag
beachAcres	Beach acres

Table 2: Population Synthesizer Household data – hhFile.csv

Column Name	Description
HHID	Unique Household ID
household_serial_no	Household serial number
TAZ	TAZ of household
MGRA	MAZ of household
HINCCAT1	Household income category
HINC	Household income
HWORKERS	Number of workers in household
VEH	Number of vehicles in household
PERSONS	Number of persons in household
HHT	Household/family type: **0.** Not in universe (vacant or GQ)

Column Name	Description
	1. Family household:married-couple 2. Family household:male householder,no wife present 3. Family household:female householder,no husband present 4. Nonfamily household:male householder, living alone 5. Nonfamily household:male householder, not living alone 6. Nonfamily household:female householder, living alone 7. Nonfamily household:female householder, not living alone
BLDGSZ	Building size - Number of Units in Structure & Quality: 1. Mobile home or trailer 2. One-family house detached 3. One-family house attached 4. 2 Apartments 5. 3-4 Apartments 6. 5-9 Apartments 7. 10-19 Apartments 8. 20-49 Apartments 9. 50 or more apartments 10. Boat, RV, van, etc.
UNITTYPE	Household unit type (3): 0. Household 1. Non-Institutional Group Quarters 2. Institutional Group Quarters

Table 3: Population Synthesizer Person data – personFile.csv

Column Name	Description
HHID	Household ID
PERID	Person ID
household_serial_no	Household serial number
PNUM	Person Number
AGE	Age of person
SEX	Gender of person 1 = Male, 2 = Female

Column Name	Description
MILTARY	Military service: **0.** Not in universe (Under 17 years) **1.** Yes,now on active duty **2.** Yes,on active duty in the past,but not now **3.** No,training for reserves or National Guard only **4.** No active duty service
PEMPLOY	Person employment status: **1.** full time **2.** part time **3.** unemployed **4.** not in universe, under 16 yrs old
PSTUDENT	Person student status: **1.** preschool, kindergarten, grade 1-12 **2.** college/graduate student **3.** not in universe, under 3yrs old
PTYPE	Person type: **1.** full time worker **2.** part time worker **3.** university student **4.** non-working adult **5.** retiree **6.** driving age school children **7.** pre-driving age school children **8.** pre-school children

Column Name	Description
EDUC	Educational Attainment: **0.** Not in universe (Under 3 years) **1.** No schooling completed **2.** Nursery school to 4th grade **3.** 5th grade or 6th grade **4.** 7th grade or 8th grade **5.** 9th grade **6.** 10th grade **7.** 11th grade **8.** 12th grade,no diploma **9.** High school graduate **10.** Some college,but less than 1 year **11.** One or more years of college,no degree **12.** Associate degree **13.** Bachelor 's degree **14.** Master 's degree **15.** Professional degree **16.** Doctorate degree
GRADE	School enrollment: grade level attending **0.** Not in universe (Under 3 years) **1.** Nursery school,preschool **2.** Kindergarten **3.** Grade 1 to grade 4 **4.** Grade 5 to grade 8 **5.** Grade 9 to grade 12 **6.** College undergraduate **7.** Graduate or professional school
OCCCEN1	Occupation **0 .**Not in universe (Under 16 years or LAST- WRK =2) **1..997.** Legal census occupation code
INDCEN	Industry **0.** Not in universe (Under 16 years or LAST- WRK =2) **1..997.** Legal census 2000 industry code
WKW	Weeks worked in 1999 **0.** Not in universe (Under 16 years) **1..52.** 1 to 52 weeks
WKHP	Hours worked in 1999 **0.** Not in universe (Under 16 years) **1..99.** 1 to 99 hours worked per week

Table 4: Transit Walk Access –walkacc

Column Name	Description
MAZ	MAZ
TAP	TAP
DISTANCE	Distance

Table 5: Transit Drive Access TAPS - access.prp

Column Name	Description
Origin TAZ	Origin TAZ (and count of TAPs)
Destination TAP	TAP
Distance	Distance
Time	Time

Table 6: Transit TAP PTYPE –tap.ptype

Column Name	Description
TAP	TAP
LOT ID	Lot ID
PARKING TYPE	Parking type (1 formal, 2 other parking, 3 other light rail, 4/5 high survey)
TAZ	TAZ
CAPACITY	Lot capacity
DIST_LOT_TAP	Distance from lot to TAP

Table 7: MGRA Walk Distance –mgrataz.walk

Column Name	Description
OMAZ	Origin MAZ
DMAZ	Destination MAZ
Distance	Distance

Table 8: Zone Terminal Time –zone.term

Column Name	Description
Zone	TAZ number
Terminal time	Terminal time (3, 4, 5, 7, 10 minutes)

Table 9: Occupation Coding – pecas_occ_occcen1.csv

Column Name	Description
Occcen1	Census occupation code
Pecas_occ	PECAS occupation code

Table 10: Activity Coding – activity_code_indcen.csv

Column Name	Description
Indcen	Census industry code
Activity_code	CT-RAMP activity code

Table 11: Transponder Ownership Model Data – tc_avgtt.csv

Column Name	Description
TAZ	TAZ
AVGTTS	Average travel time savings
DIST	Distance
PCTDETOUR	Percent detour

Table 12: Network database – SERPM_NETS.mdb

ESRI personal geodatabase with the SERPM 7 highway and transit network.

Table 13: All streets network - Full-SERPM_Navteq_NAD83.net

Cube network file contains the all-streets network. This file is used for generating the MAZ to MAZ and MAZ to TAP impedances.

Table 14: MAZ All Streets Network Connectors – connectors.txt

Column Name	Description
A	A node
B	B node
LENGTH	Link length

Table 15: Time-of-day factor file –TOD.dbf

Field	Description
IPRD	Record Number
PNAM	Variable Name
EE	Fraction of trips that are EE

Field	Description
TRUCK	Fraction of truck trips for the time period, fraction P-A for the period
AIRPORT	Fraction of airport trips for the time period, fraction P-A for the period
EI	Fraction of EI trips for the time period, fraction P-A for the period

Table 16: Enplanements – ENPLANE.DBF

Field	Description
TAZ	Zone number
ENPLANE	Daily Enplanements (FAA)
NAME	Airport Name

Table 17: Airport and truck trip generation rates - airport_truck_rates.dbf

Field	Description
PNO	Purpose number (1=airport, 2=4tire, 3=SU truck, 4=combination)
PNAM	Purpose name
CO	County number (1=PB, 2=BO, 3-MD)
AT	Area type
KEY	(County, Area Type, Purpose) codes
INDUS	Industrial employment trip rate
COMM	Commercial employment trip rate
SVC	Service employment trip rate
TOTE	Total employment trip rate
HH	Household trip rate
HM	Hotel/Motel rooms
ENPLANE	Daily Enplanements (FAA)

Table 18: External-Internal Data - {DATADIR}\EIdata.dbf

Field	Description
TAZ	TAZ number
TYPE	NCHRP 716 station type
DESCRIPT	NCHRP 716 station type name
ROAD	Road name

Field	Description
TERMT	External terminal time (not used)
PROD	Daily productions (vehicles)
PPCT4TIRE	Daily production percent 4tire trucks
PPCTSU	Daily production percent single-unit trucks
PPCTCOMB	Daily production percent combination trucks
ATTR	Daily attractions (vehicles)
APCT4TIRE	Daily attraction percent 4tire trucks
APCTSU	Daily attraction percent single-unit trucks
APCTCOMB	Daily attraction percent combination trucks
TAZ65	SERPM 65 TAZ number (not used)

Table 19: Vfactors file - MVFACTORS.{year}{alt}

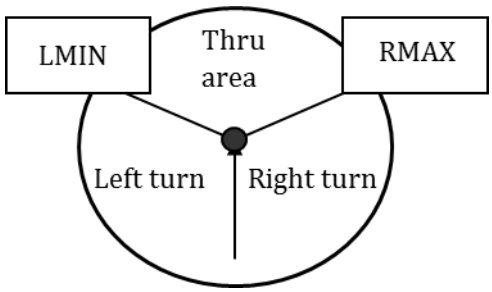
Variable	Description
FTC2	2-digit facility type
UROADF	UROAD factor (LOS C)/(LOS E)
CONFAC24H	Fraction of 24 volume in the peak hour
BPR LOS	BPR capacity restraint coefficient
BPR EXP	BPR capacity restraint exponent
CONFACAMP	Highest of 3 AM peak hours
CONFACPMP	Highest of 3 PM peak hours
CONFACOFP	Highest of 18 off peak hours

Table 20: MTURNDEF.{Year}{ALT} - MTURNDEF file

MTURNDEF contains all data needed to construct a TCARDS file. Any existing TCARDS file must be copied into MTURNDEF file, as MKTURN2 program will overwrite TCARDS file. MTURNDEF is a fixed-format ASCII file, which can contain six types of records, depending on the code entered in column 1 of each record:

- C – Means a comment record. These records are read and output to the report file, but are otherwise ignored.
- "blank" – These records contain entries in the turn penalty definition table, as described later.
- T – These records are standard Tranplan turn data, and should be copied from your original TCARDS file.

- R – This is the specification of the maximum included angle for a right-turn (RMAX).



- L – This is the specification of the minimum included angle for a left-turn (LMIN).
- F – Factors to be applied to penalties at specified intersection nodes. The penalty obtained from the turn penalty definition table will be factored by this amount at the specified nodes.

Records with "R" in column 1, specify the maximum included angle for a right-turn. There should be only one of these records. The format is:

- cc: 1 – R
- cc: 2-5 – Maximum angle in degrees.
- cc: 6 - 80 – Optional description

Records with "L" in column 1, specify the minimum included angle for a left-turn. There should be only one of these records. The format is:

- cc: 1 – L
- cc: 2-5 – Minimum angle in degrees.
- cc: 6 - 80 – Optional description

Records with "T" in column 1 are traditional turn penalty/prohibitor records. These records contain a T in column 1, from node, through node, to node and penalty in hundredths of a minute.

Records with "F" in column 1 are factors. The penalty obtained from the penalty table is multiplied by this factor for all movements passing through the specified intersection. The format is:

- cc: 1 – F
- cc: 2-6 – Intersection node number (I5).
- cc: 7 - 12 – Factor (F6.0).

Penalty table records have a blank in column 1. Only facility type pairs (from – to) with non-zero penalties need to be entered, as the penalty table is initialized with zeros by the program. The format is:

- cc: 1 – blank
- cc: 6 – from facility type (one digit)
- cc: 11 – to facility type (one digit)
- cc: 24-25 – left turn penalty in hundredths of minutes
- cc: 29-30 – through penalty in hundredths of minutes
- cc: 34-35 – right turn penalty in hundredths of minutes
- cc: 36-80 – an optional description

Table 21: TSPEED.DBF – Transit speed dwell time definitions

Variable	Description
USERA4	USERA4 Code in transit line records (geodatabase)
PK_DWELL	Peak period dwell time per stop in minutes
OP_DWELL	Off-peak period dwell time per stop in minutes
PK_TIMEFAC	Peak travel time factor
OP_TIMEFAC	Offpeak travel time factor
GRP_NAME	USERA4 route group name

Table 22: TSYSD.PTS – Transit System File (standard PT file)

Record Name	Description
Mode Number	Mode number and name
Operator Number	Operator number and name
WAITCRVDEF	Wait curve definitions for standard and zero wait time routes

Table 23: TFARES.FAR– Transit System Fares (standard PT file)

Record Name	Description
Faresystem Number	Fare structure ID
NAME	Structure name
LONGNAME	Structure longname
STRUCTURE	Structure type
SAME	Fare for transfers between like structures
IBOARDFARE	Initial boarding structure
FAREFROMS	Transfer fares

Table 24: daily_loaded.net – Loaded highway network (new fields only)

Record Name	Description
AM_CONGTIME	AM period congested time
AM_VCLOSE	AM period volume/(LOS E capacity)
AM_V_1	AM period load in vehicles
OF_CONGTIME	Off peak period congested time
OF_VCLOSE	Off peak period volume/(LOS E capacity)

Record Name	Description
MD_V_1	Midday period load in vehicles
PM_CONGTIME	PM period congested time
PM_VCLOSE	PM period volume/(LOS E capacity)
PM_V_1	PM period load in vehicles
EA_V_1	Early period load in vehicles
EV_V_1	Evening period load in vehicles
DAY_V_1	Daily load in vehicles
AM_V	AM period load in PCE's
MD_V	Midday period load in PCE's
PM_V	PM period load in PCE's
EA_V	Early period load in PCE's
EV_V	Evening period load in PCE's

Table 25: X_AMHSKIMS.mat – AM peak period congested skims

Record Name	Description
AM_GP_TIME	Congested times on general purpose lane generalized paths
AM_GP_FFTIME	Freeflow times on general purpose lane generalized paths
AM_GP_DIST	Distance on general purpose lane generalized paths
AM_DAT_TIME	Congested times on drive alone toll generalized paths
AM_DAT_FFTIME	Freeflow times on drive alone toll generalized paths
AM_DAT_DIST	Distance on drive alone toll generalized paths
AM_DAT_TOLLCOST	Toll cost $ on drive alone toll generalized paths
AM_DAT_TOLLDIST	Distance on toll links on drive alone toll generalized paths
AM_S2NH_TIME	Congested times on SR2+ non-toll HOV generalized paths
AM_S2NH_FFTIME	Freeflow times on SR2+ non-toll HOV generalized paths
AM_S2NH_DIST	Distance on SR2+ non-toll HOV generalized paths
AM_S2NH_HOVDIST	HOV lane distance on SR2+ non-toll HOV generalized paths
AM_S2TH_TIME	Congested times on SR2+ toll HOV generalized paths
AM_S2TH_FFTIME	Freeflow times on SR2+ toll HOV generalized paths

Record Name	Description
AM_S2TH_DIST	Distance on SR2+ toll HOV generalized paths
AM_S2TH_TOLLCOST	Toll cost $ on SR2+ toll HOV generalized paths
AM_S2TH_TOLLDIST	Distance on toll links on SR2+ toll HOV generalized paths
AM_S2TH_HOVDIST	HOV lane distance on SR2+ toll HOV generalized paths

Table 26: X_OFHSKIMS.mat – Off peak period congested skims

Record Name	Description
OF_GP_TIME	Congested times on general purpose lane generalized paths
OF_GP_FFTIME	Freeflow times on general purpose lane generalized paths
OF_GP_DIST	Distance on general purpose lane generalized paths
OF_DAT_TIME	Congested times on drive alone toll generalized paths
OF_DAT_FFTIME	Freeflow times on drive alone toll generalized paths
OF_DAT_DIST	Distance on drive alone toll generalized paths
OF_DAT_TOLLCOST	Toll cost $ on drive alone toll generalized paths
OF_DAT_TOLLDIST	Distance on toll links on drive alone toll generalized paths
OF_S2NH_TIME	Congested times on SR2+ non-toll HOV generalized paths
OF_S2NH_FFTIME	Freeflow times on SR2+ non-toll HOV generalized paths
OF_S2NH_DIST	Distance on SR2+ non-toll HOV generalized paths
OF_S2NH_HOVDIST	HOV lane distance on SR2+ non-toll HOV generalized paths
OF_S2TH_TIME	Congested times on SR2+ toll HOV generalized paths
OF_S2TH_FFTIME	Freeflow times on SR2+ toll HOV generalized paths
OF_S2TH_DIST	Distance on SR2+ toll HOV generalized paths
OF_S2TH_TOLLCOST	Toll cost $ on SR2+ toll HOV generalized paths
OF_S2TH_TOLLDIST	Distance on toll links on SR2+ toll HOV generalized paths
OF_S2TH_HOVDIST	HOV lane distance on SR2+ toll HOV generalized paths

Table 27: X_PMHSKIMS.mat – PM peak period congested skims

Record Name	Description
PM_GP_TIME	Congested times on general purpose lane generalized paths
PM_GP_FFTIME	Freeflow times on general purpose lane generalized paths

Record Name	Description
PM_GP_DIST	Distance on general purpose lane generalized paths
PM_DAT_TIME	Congested times on drive alone toll generalized paths
PM_DAT_FFTIME	Freeflow times on drive alone toll generalized paths
PM_DAT_DIST	Distance on drive alone toll generalized paths
PM_DAT_TOLLCOST	Toll cost $ on drive alone toll generalized paths
PM_DAT_TOLLDIST	Distance on toll links on drive alone toll generalized paths
PM_S2NH_TIME	Congested times on SR2+ non-toll HOV generalized paths
PM_S2NH_FFTIME	Freeflow times on SR2+ non-toll HOV generalized paths
PM_S2NH_DIST	Distance on SR2+ non-toll HOV generalized paths
PM_S2NH_HOVDIST	HOV lane distance on SR2+ non-toll HOV generalized paths
PM_S2TH_TIME	Congested times on SR2+ toll HOV generalized paths
PM_S2TH_FFTIME	Freeflow times on SR2+ toll HOV generalized paths
PM_S2TH_DIST	Distance on SR2+ toll HOV generalized paths
PM_S2TH_TOLLCOST	Toll cost $ on SR2+ toll HOV generalized paths
PM_S2TH_TOLLDIST	Distance on toll links on SR2+ toll HOV generalized paths
PM_S2TH_HOVDIST	HOV lane distance on SR2+ toll HOV generalized paths

Table 28: Visitor Business Frequency Distribution – visitor_businessFrequency.csv

Column Name	Description
WorkTours	Number of work tours (0, 1, 2+)
RecreationTours	Number of recreation tours (0, 1, 2+)
DiningTours	Number of dining tours (0, 1+)
TotalTours	Total number of tours
Percent	Distribution of tours by purpose frequency

Table 29: Visitor Personal Frequency Distribution – visitor_personalFrequency.csv

Column Name	Description
WorkTours	Number of work tours (0, 1+)
RecreationTours	Number of recreation tours (0, 1, 2+)
DiningTours	Number of dining tours (0, 1, 2+)

TotalTours	Total number of tours
Percent	Distribution of tours by purpose frequency

Table 30: Visitor Party Size Distribution – visitor_partySize.csv

Column Name	Description
Party Size	Party Size (1 through 10+)
work	Distribution of visitor work tours by party size
recreate	Distribution of visitor recreation tours by party size
dining	Distribution of visitor dining tours by party size

Table 31: Visitor Auto Availability – visitor_autoAvailable.csv

Column Name	Description
autoAvailable	Is auto available for visitor tour?
Work	Percentage of visitor work tours where auto is available
recreate	Percentage of visitor recreation tours where auto is available
dining	Percentage of visitor dining out tours where auto is available

Table 32: Visitor Income Distribution – visitor_income.csv

Column Name	Description
Income	Visitor income: 0 = < $30K 1 = $30K-$60K 2 = $60K-100K 3 = $100K-$150K 4 = $150K+
Work	Distribution of visitor work tours by income
recreate	Distribution of visitor recreation tours by income
dining	Distribution of visitor dining tours by income

Table 33: Visitor Tour Time of Day Distribution – visitor_tourTOD.csv

Column Name	Description
Purpose	Visitor tour purpose: 0 = Work 1 = Recreation 2 = Dining Out

EntryPeriod	Entry Period: 1=Before 5:00AM, 2=5:00AM-5:30AM, 3 through 39 is every half hour time slots 40 = After 12:00AM
ReturnPeriod	Return Period: 1=Before 5:00AM, 2=5:00AM-5:30AM, 3 through 39 is every half hour time slots 40 = After 12:00AM
Percent	Distribution of visitor tours by time of day

Table 34: Visitor Stop Frequency Distribution – visitor_stopFrequency.csv

Column Name	Description
Purpose	Visitor tour purpose: 0 = Work 1 = Recreation 2 = Dining Out
DurationLo	Lower bound of tour duration
DurationHi	Upper bound of tour duration
Outbound	Number of stops on the outbound (0, 1+)
Inbound	Number of stops on the inbound (0, 1, 2, 3+)
Percent	Distribution of tours by purpose, duration, number of outbound/inbound stops

Table 35: Visitor Stop Purpose Distribution – visitor_stopPurpose.csv

Column Name	Description
Purpose	Visitor tour purpose: 0 = Work 1 = Recreation 2 = Dining Out
Inbound	Boolean for whether stop is inbound (0=No, 1=Yes)
StopNum	Stop number on tour
Multiple	Boolean for whether there are multiple stops on tour (0=No, 1=Yes)
StopPurp0	Distribution of Work stops
StopPurp1	Distribution of School stops
StopPurp2	Distribution of Cargo stops

Table 36: Visitor Outbound Stop Duration Distribution – visitor_outboundStopDuration.csv

Column Name	Description
RemainingLow	Lower bound of remaining half hour periods after last scheduled trip: 1=Before 5:00AM, 2=5:00AM-5:30AM, 3 through 39 is every half hour time slots 40 = After 12:00AM
RemainingHigh	Upper bound of remaining half hour periods after last scheduled trip: 1=Before 5:00AM, 2=5:00AM-5:30AM, 3 through 39 is every half hour time slots 40 = After 12:00AM
Stop	Stop number on tour
1	
2	
3	
4	
5	
6	
7	
8	
9	
10	
11	

Table 37: Visitor Inbound Stop Duration Distribution – visitor_inboundStopDuration.csv

Column Name	Description
RemainingLow	Lower bound of remaining half hour periods after last scheduled trip: 1=Before 5:00AM, 2=5:00AM-5:30AM, 3 through 39 is every half hour time slots 40 = After 12:00AM
RemainingHigh	Upper bound of remaining half hour periods after last scheduled trip: 1=Before 5:00AM, 2=5:00AM-5:30AM, 3 through 39 is every half hour time slots 40 = After 12:00AM
Stop	Stop number on tour

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

Table 38: Temporary Static Inputs – visitor_inboundStopDuration.csv

Column Name	Description
TAZ	Zone
AVGTTS	Average Travel Time
DIST	Distance
PCTDETOUR	Percentage Detour

CT-RAMP Properties File

The CT-RAMP resident demand model software is controlled by a Java properties file. The table below identifies, describes, and provides an example for each of the variables expected to be in the properties file. After initially configuring the travel model, only a handful of these properties will be modified for a typical scenario analysis. The primary use for many of the properties is to facilitate software execution when calibrating the travel model and/or to locate and fix errors. Comments preceded with a pound (#) sign are ignored when reading in the properties file.

Table 39: CT-RAMP Properties File

Property	Data Type	Example Value	Purpose
Trace	Boolean	false	True or False whether to trace zones
Trace.otaz	Integer	1638	Specify which origin taz to trace
Trace.dtaz	Integer	2447	Specify which destination taz to trace

Property	Data Type	Example Value	Purpose
Seek	Boolean	false	True or False whether to seek households
Process.Debug.HHs.Only	Boolean	false	True of False whether to debug households
Debug.Trace.HouseholdIdList	String	566425	Specify which household IDs to trace
run.this.household.only	String	566425	Specify that this household ID only will be run through the model
Project.Directory	String	%project.folder%	File locations specified in the properties file as well as the UEC DataSheet pages are expressed as relative to this location. %project.folder% is set to C:/serpm7/ctramp.
alt	String	R	Model run alternative used for skim matrix file names
Year	String	2010	Model run year used for skim matrix file names
Model.Random.Seed	Integer	1	Starting value for model random seed number (added to household IDs to create unique random number for each household)
RunModel.MatrixServerAddress	String	156.75.49.78	matrix server address
RunModel.MatrixServerPort	Integer	1191	matrix server port number
RunModel.HouseholdServerAddress	String	156.75.49.78	household server address
RunModel.HouseholdServerPort	Integer	1117	household server port number
generic.path	String	%project.folder%/input/	Inputs folder
scenario.path	String	%project.folder%/	scenario folder
skims.path	String	%project.folder%/output/	Skim matrix files folder
uec.path	String	%project.folder%/uec/	UEC folder

Property	Data Type	Example Value	Purpose
acc.uec.file	String	%project.folder%/uec/Accessibilities.xls	Accessibilities.xls location
acc.data.page	Integer	0	Accessibilities data page
acc.transit.offpeak.page	Integer	1	Accessibilities offpeak page
acc.transit.peak.page	Integer	2	Accessibilities peak page
acc.transit.walkaccess.page	Integer	3	Accessibilities walk access page
acc.transit.driveaccess.page	Integer	4	Accessibilities drive access page
acc.sov.offpeak.page	Integer	5	Accessibilities SOV offpeak
acc.sov.peak.page	Integer	6	Accessibilities SOV peak
acc.hov.offpeak.page	Integer	7	Accessibilities HOV offpeak
acc.hov.peak.page	Integer	8	Accessibilities HOV peak
acc.nonmotorized.page	Integer	9	Accessibilities non-motorized
acc.constants.page	Integer	10	Accessibilities constants
acc.sizeTerm.page	Integer	11	Accessibilities size terms
acc.schoolSizeTerm.page	Integer	12	Accessibilities school size terms
acc.workerSizeTerm.page	Integer	13	Accessibilities worker size terms
acc.dcUtility.uec.file	String	%project.folder%/uec/Accessibilities_DC.xls	Accessibilities_DC.xls location
acc.dcUtility.data.page	Integer	0	DC Accessibilities data page
acc.dcUtility.page	Integer	1	DC Accessibilities utility page
accessibility.alts.file	String	Acc_alts.csv	Accessibilities alternatives
acc.output.file	String	input/accessibilities.csv	Accessibilities.csv location
acc.read.input.file	Boolean	false	Read the accessibilities as input instead of running the module
acc.mandatory.uec.file	String	%project.folder%/uec/MandatoryAccess.xls	MandatoryAccess.xls location

Property	Data Type	Example Value	Purpose
acc.mandatory.data.page	Integer	0	Mandatory Access data page
acc.mandatory.auto.page	Integer	1	Mandatory Access auto page
acc.mandatory.autoLogsum.page	Integer	2	Mandatory Access autoLogsum page
acc.mandatory.bestWalkTransit.page	Integer	3	Mandatory Access best Walk Transit page
acc.mandatory.bestDriveTransit.page	Integer	4	Mandatory Access best Drive Transit page
acc.mandatory.transitLogsum.page	Integer	5	Mandatory Access transit logsum page
PopulationSynthesizer.InputToCTRAMP.HouseholdFile	String	input/households.csv	location of popsyn households file
PopulationSynthesizer.InputToCTRAMP.PersonFile	String	input/persons.csv	location of popsyn persons file
PopulationSynthesizer.OccupCodes	String	input/pecas_occ_occsoc.csv	location of popsyn pecas_occ_occsoc.csv file
PopulationSynthesizer.IndustryCodes	String	input/activity_code_indcen.csv	location of popsyn activity_code_indcen.csv file
mgra.wlkacc.taps.and.distance.file	String	input/wkacc	location of walk access and distance file
taz.driveaccess.taps.file	String	input/accessam.prp	location of drive access file
tap.ptype.file	String	input/tap.ptype	location of ptype file
mgra.walkdistance.file	String	input/mgrataz.walk	location of MAZ walk distance file
taz.parkingtype.file	String	input/zone.park	location of parking type file
taz.terminal.time.file	String	input/zone.term	location of zonal terminal time file
mgra.socec.file	String	maz_data.csv	location of MAZ land use file
RunModel.Clear.MatrixMgr.At.Start	Boolean	FALSE	Clear matrix manager at the start of run?
Results.WriteDataToFiles	Boolean	true	write data to files?

Property	Data Type	Example Value	Purpose
Results.HouseholdDataFile	String	output/householdData.csv	output name of household data file
Results.PersonDataFile	String	output/personData.csv	output name of person data file
Results.IndivTourDataFile	String	output/indivTourData.csv	output name of individual tour data file
Results.JointTourDataFile	String	output/jointTourData.csv	output name of joint tour data file
Results.IndivTripDataFile	String	output/indivTripData.csv	output name of individual trip data file
Results.JointTripDataFile	String	output/jointTripData.csv	output name of joint trip data file
Results.CBDFile	String	output/cbdParking.csv	output name of CBD parking file
Results.PNRFile	String	output/pnrParking.csv	output name of PNR parking file
Results.WriteDataToDatabase	Boolean	false	write data to a database?
Results.HouseholdTable	String	household_data	output name of household data file in database
Results.PersonTable	String	person_data	output name of person data file in database
Results.IndivTourTable	String	indiv_tour_data	output name of individual tour data file in database
Results.JointTourTable	String	joint_tour_data	output name of joint tour data file in database
Results.IndivTripTable	String	indiv_trip_data	output name of individual trip data file in database
Results.JointTripTable	String	joint_trip_data	output name of joint trip data file in database
Results.AutoTripMatrix	String	output/autoTrips	output name of auto trip matrix
Results.TranTripMatrix	String	output/tranTrips	output name of transit trip matrix
Results.NMotTripMatrix	String	output/nmotTrips	output name of non-motorized trip matrix

Property	Data Type	Example Value	Purpose
Results.OthrTripMatrix	String	output/othrTrips	output name of other modes trip matrix
Results.PNRFile	String	output/PNRByTAP_Vehicles.csv	output name of PNR by TAP Vehicle Trip file
Results.CBDFile	String	output/CBDByMGRA_Vehicles.csv	output name of CBD by MGRA Vehicle Trip file
TourModeChoice.Save.UtilsAndProbs	Boolean	true	Save utilities and probabilities in tour mode choice output files
distributed.task.packet.size	Integer	200	JPPF distributed task packet size
RunModel.RestartWithHhServer	String	none	model can be restarted with certain files already generated…..values include 'none' (run whole model), 'uwsl', 'ao', 'stf'
RunModel.PreAutoOwnership	Boolean	true	True or False whether to run this model component
RunModel.UsualWorkAndSchoolLocationChoice	Boolean	true	True or False whether to run this model component
RunModel.AutoOwnership	Boolean	true	True or False whether to run this model component
RunModel.TransponderChoice	Boolean	true	True or False whether to run this model component
RunModel.FreeParking	Boolean	true	True or False whether to run this model component
RunModel.CoordinatedDailyActivityPattern	Boolean	true	True or False whether to run this model component
RunModel.IndividualMandatoryTourFrequency	Boolean	true	True or False whether to run this model component
RunModel.MandatoryTourModeChoice	Boolean	true	True or False whether to run this model component
RunModel.MandatoryTourDepartureTimeAndDuration	Boolean	true	True or False whether to run this model component

Property	Data Type	Example Value	Purpose
RunModel.JointTourFrequency	Boolean	true	True or False whether to run this model component
RunModel.JointTourLocationChoice	Boolean	true	True or False whether to run this model component
RunModel.JointTourDepartureTimeAndDuration	Boolean	true	True or False whether to run this model component
RunModel.JointTourModeChoice	Boolean	true	True or False whether to run this model component
RunModel.IndividualNonMandatoryTourFrequency	Boolean	true	True or False whether to run this model component
RunModel.IndividualNonMandatoryTourLocationChoice	Boolean	true	True or False whether to run this model component
RunModel.IndividualNonMandatoryTourDepartureTimeAndDuration	Boolean	true	True or False whether to run this model component
RunModel.IndividualNonMandatoryTourModeChoice	Boolean	true	True or False whether to run this model component
RunModel.AtWorkSubTourFrequency	Boolean	true	True or False whether to run this model component
RunModel.AtWorkSubTourLocationChoice	Boolean	true	True or False whether to run this model component
RunModel.AtWorkSubTourDepartureTimeAndDuration	Boolean	true	True or False whether to run this model component
RunModel.AtWorkSubTourModeChoice	Boolean	true	True or False whether to run this model component
RunModel.StopFrequency	Boolean	true	True or False whether to run this model component
RunModel.StopLocation	Boolean	true	True or False whether to run this model component
uwsl.use.new.soa	Boolean	false	true or false whether to use new soa for the work/school DC model
nmdc.use.new.soa	Boolean	false	true or false whether to use new soa for the non-mandatory DC models

Property	Data Type	Example Value	Purpose
slc.use.new.soa	Boolean	false	true or false whether to use new soa for the stop location chocie models
Results.AutoOwnership	String	output/aoResults.csv	auto ownership output file name and location
read.pre.ao.results	Boolean	FALSE	read in the old pre-auto ownership results file
read.pre.ao.filename	String	output/aoResults_pre.csv	pre auto ownership output file nAme and location
Results.UsualWorkAndSchoolLocationChoice	String	output/wsLocResults.csv	usual work and school location output file name and location
read.uwsl.results	Boolean	FALSE	Read in the old uwsl results?
read.uwsl.filename	String	output/wsLocResults_1.csv	old uwsl result file name and location to read in
uwsl.run.workLocChoice	Boolean	true	True or False whether to run this model component
uwsl.run.schoolLocChoice	Boolean	true	True or False whether to run this model component
uwsl.write.results	Boolean	true	True of False whether to write out usual work and school location results
ao.uec.file	String	AutoOwnership.xls	File name of auto ownership UEC
ao.data.page	Integer	0	Auto ownership UEC data page
ao.model.page	Integer	1	Auto ownership UEC utility page
uwsl.dc.uec.file	String	TourDestinationChoice.xls	File Name of Tour Destination Choice UEC
uwsl.dc2.uec.file	String	TourDestinationChoice2.xls	File Name of Tour Destination Choice 2 UEC
uwsl.soa.uec.file	String	DestinationChoiceAlternativeSample.xls	File Name of Destination Choice Alternative Sample UEC

Property	Data Type	Example Value	Purpose
uwsl.soa.alts.file	String	DestinationChoiceAlternatives.csv	File name of the alternatives (MGRAs) available to the destination choice models (part of the model design; this should not be changed)
uwsl.work.soa.SampleSize	Integer	30	Sample size of Work Destination Choice
uwsl.school.soa.SampleSize	Integer	30	Sample size of School Destination Choice
work.soa.uec.file	String	TourDcSoaDistance.xls	File Name of Tour Distance DC SOA UEC for Work Purpose, includes TAZ Size in the expressions
work.soa.uec.data	Integer	0	Work Tour Distance SOA UEC data page
work.soa.uec.model	Integer	1	Work Tour Distance SOA UEC utility page
univ.soa.uec.file	String	TourDcSoaDistanceNoSchoolSize.xls	File Name of Tour Distance DC SOA UEC for School Purpose; school purposes do not include TAZ Size in the expressions so that the utilities can be stored as exponentiated distance utility matrices for university, and then multiplied by the university segment size terms
univ.soa.uec.data	Integer	0	University Tour Distance SOA UEC data page
univ.soa.uec.model	Integer	1	University Tour Distance SOA UEC utility page

Property	Data Type	Example Value	Purpose
hs.soa.uec.file	String	TourDcSoaDistanceNoSchoolSize.xls	File Name of Tour Distance DC SOA UEC for School Purpose; school purposes do not include TAZ Size in the expressions so that the utilities can be stored as exponentiated distance utility matrices for high school, and then multiplied by the high school segment size terms
hs.soa.uec.data	Integer	0	High School Tour Distance SOA UEC data page
hs.soa.uec.model	Integer	2	High School Tour Distance SOA UEC utility page
gs.soa.uec.file	String	TourDcSoaDistanceNoSchoolSize.xls	File Name of Tour Distance DC SOA UEC for School Purpose; school purposes do not include TAZ Size in the expressions so that the utilities can be stored as exponentiated distance utility matrices for grade school, and then multiplied by the grade school segment size terms
gs.soa.uec.data	Integer	0	Grade School Tour Distance SOA UEC data page
gs.soa.uec.model	Integer	3	Grade School Tour Distance SOA UEC utility page
ps.soa.uec.file	String	TourDcSoaDistanceNoSchoolSize.xls	File Name of Tour Distance DC SOA UEC for School Purpose; school purposes do not include TAZ Size in the expressions so that the utilities can be stored as exponentiated distance utility matrices for preschool, and then multiplied by the preschool segment size terms

Property	Data Type	Example Value	Purpose
ps.soa.uec.data	Integer	0	Preschool Tour Distance SOA UEC data page
ps.soa.uec.model	Integer	4	Prescehool Tour Distance SOA UEC utility page
UsualWorkLocationChoice.ShadowPrice.Input.File	String	input/ShadowPricingOutput_work_39.csv	File Name Work Location shadow price
UsualSchoolLocationChoice.ShadowPrice.Input.File	String	input/ShadowPricingOutput_school_19.csv	File Name School Location shadow price
uwsl.ShadowPricing.Work.MaximumIterations	Integer	1	maximum number of iterations for work shadow price
uwsl.ShadowPricing.School.MaximumIterations	Integer	1	maximum number of iterations for school shadow price
uwsl.ShadowPricing.OutputFile	String	output/ShadowPricingOutput.csv	output file name for shadow price
tc.choice.avgtts.file	String	input/tc_avgtt.csv	File name of average travel times for transponder ownership
tc.uec.file	String	TransponderOwnership.xls	File name of transponder ownership UEC
tc.data.page	Integer	0	Transponder ownership UEC data page
tc.model.page	Integer	1	Transponder ownership UEC utility page
fp.uec.file	String	ParkingProvision.xls	File name of parking provision UEC
fp.data.page	Integer	0	Parking Provision UEC data page
fp.model.page	Integer	1	Parking Provision UEC utility page
cdap.uec.file	String	CoordinatedDailyActivityPattern.xls	File name of CDAP UEC
cdap.data.page	Integer	0	CDAP UEC data page

Property	Data Type	Example Value	Purpose
cdap.one.person.page	Integer	1	CDAP UEC utility for one person page
cdap.two.person.page	Integer	2	CDAP UEC utility for 2 persons page
cdap.three.person.page	Integer	3	CDAP UEC utility for 3 persons page
cdap.all.person.page	Integer	4	CDAP UEC utility for All member interation page
cdap.joint.page	Integer	5	CDAP UEC utility for joint tours page
imtf.uec.file	String	MandatoryTourFrequency.xls	File name of Mandatory tour frequency UEC
imtf.data.page	Integer	0	Mandatory tour frequency UEC data page
imtf.model.page	Integer	1	mandatory tour frequency UEC utility page
nonSchool.soa.uec.file	String	TourDcSoaDistance.xls	File Name of Tour Distance DC SOA UEC for Non Work/School Purposes, includes TAZ Size in the expressions
escort.soa.uec.data	Integer	0	Escort Tour Distance SOA UEC data page
escort.soa.uec.model	Integer	2	Escort Tour Distance SOA UEC utility page
other.nonman.soa.uec.data	Integer	0	Other Non-mandatory Tour Distance SOA UEC data page
other.nonman.soa.uec.model	Integer	3	Other Non-mandatory Tour Distance SOA UEC utility page
atwork.soa.uec.data	Integer	0	At-Work Sub-Tour Distance SOA UEC data page
atwork.soa.uec.model	Integer	4	At-Work Sub-Tour Distance SOA UEC utility page

Property	Data Type	Example Value	Purpose
soa.taz.dist.alts.file	String	SoaTazDistAlts.csv	File name of Sample of Alternatives of TAZs
nmdc.dist.alts.file	String	NonMandatoryTlcAlternatives.csv	File name of non-mandatory tour alternatives
nmdc.soa.alts.file	String	DestinationChoiceAlternatives.csv	File name of the alternatives (MGRAs) available to the destination choice models (part of the model design; this should not be changed)
nmdc.soa.SampleSize	Integer	30	Sample size of non-mandatory Destination choice
nmdc.uec.file2	String	TourDestinationChoice2.xls	File Name of Tour Destination Choice 2 UEC
nmdc.uec.file	String	TourDestinationChoice.xls	File Name of Tour Destination Choice UEC
nmdc.data.page	Integer	0	Non-mandatory Tour DC UEC data page
nmdc.escort.model.page	Integer	7	Escort Tour Destination Choice UEC utility page
nmdc.shop.model.page	Integer	8	Shop Tour Destination Choice UEC utility page
nmdc.maint.model.page	Integer	9	Maintenance Tour Destination Choice UEC utility page
nmdc.eat.model.page	Integer	10	Eating Out Tour Destination Choice UEC utility page
nmdc.visit.model.page	Integer	11	Visiting Tour Destination Choice UEC utility page
nmdc.discr.model.page	Integer	12	Discretionary Tour Destination Choice UEC utility page
nmdc.atwork.model.page	Integer	13	At-Work Sub-Tour Destination Choice UEC utility page

Property	Data Type	Example Value	Purpose
nmdc.soa.uec.file	String	DestinationChoiceAlternativeSample.xls	File Name of Destination Choice Alternative Sample UEC
nmdc.soa.data.page	Integer	0	Non-mandatory TOUR SOA UEC data page
nmdc.soa.escort.model.page	Integer	6	Escort TOUR SOA UEC utility page
nmdc.soa.shop.model.page	Integer	7	Shop TOUR SOA UEC utility page
nmdc.soa.maint.model.page	Integer	7	Maintenance TOUR SOA UEC utility page
nmdc.soa.eat.model.page	Integer	7	Eating Out TOUR SOA UEC utility page
nmdc.soa.visit.model.page	Integer	7	Visiting TOUR SOA UEC utility page
nmdc.soa.discr.model.page	Integer	7	Discretionary TOUR SOA UEC utility page
nmdc.soa.atwork.model.page	Integer	8	At-Work Sub-Tour SOA UEC utility page
tourModeChoice.uec.file	String	TourModeChoice.xls	File name of Tour Mode choice UEC
tourModeChoice.maint.model.page	Integer	4	Maintenance Tour Mode Choice UEC utility page
tourModeChoice.discr.model.page	Integer	5	Discretionary Tour Mode Choice UEC utility page
tourModeChoice.atwork.model.page	Integer	6	At-Work Sub-Tour Mode Choice UEC utility page
departTime.uec.file	String	TourDepartureAndDuration.xls	File name of Tour TOD Choice UEC
departTime.data.page	Integer	0	Tour TOD Choice UEC data page
departTime.work.page	Integer	1	Work Tour TOD Choice UEC utility page

Property	Data Type	Example Value	Purpose
departTime.univ.page	Integer	2	University Tour TOD Choice UEC utility page
departTime.school.page	Integer	3	School Tour TOD Choice UEC utility page
departTime.escort.page	Integer	4	Escort Tour TOD Choice UEC utility page
departTime.shop.page	Integer	5	Shop Tour TOD Choice UEC utility page
departTime.maint.page	Integer	6	Maintenance Tour TOD Choice UEC utility page
departTime.eat.page	Integer	7	Eating Out Tour TOD Choice UEC utility page
departTime.visit.page	Integer	8	Visiting Tour TOD Choice UEC utility page
departTime.discr.page	Integer	9	Discretionary Tour TOD Choice UEC utility page
departTime.atwork.page	Integer	10	At-Work Sub-Tour TOD Choice UEC utility page
departTime.alts.file	String	DepartureTimeAndDurationAlternatives.csv	File name of Departure time and duration alternatives
jtfcp.uec.file	String	JointTourFrequency.xls	File name of Joint Tour Frequency UEC
jtfcp.alternatives.file	String	JointAlternatives.csv	File name of joint tour alternatives by purpose and party composition combinations
jtfcp.data.page	Integer	0	Joint Tour Frequency UEC data page
jtfcp.freq.comp.page	Integer	1	Joint Tour Frequency UEC utility composition page
jtfcp.participate.page	Integer	2	Joint Tour Frequency UEC utility participation page
inmtf.uec.file	String	NonMandatoryIndividualTourFrequency.xls	File name of Individual non-mandatory tour frequency UEC

Property	Data Type	Example Value	Purpose
inmtf.FrequencyExtension.ProbabilityFile	String	IndividualNonMandatoryTourFrequencyExtensionProbabilities_p1.csv	File name of Individual non-mandatory tour frequency extension probabilities
IndividualNonMandatoryTourFrequency.AlternativesList.InputFile	String	IndividualNonMandatoryTourFrequencyAlternatives.csv	File name of individual non-mandatory tour frequency alternatives (combinations)
inmtf.data.page	Integer	0	Individual Non-mandatory tour frequency UEC data page
inmtf.perstype1.page	Integer	1	Individual Non-mandatory tour frequency UEC utility for Full time workers page
inmtf.perstype2.page	Integer	2	Individual Non-mandatory tour frequency UEC utility for Part time workers page
inmtf.perstype3.page	Integer	3	Individual Non-mandatory tour frequency UEC utility for University students page
inmtf.perstype4.page	Integer	4	Individual Non-mandatory tour frequency UEC utility for Non-workers page
inmtf.perstype5.page	Integer	5	Individual Non-mandatory tour frequency UEC utility for Retirees page
inmtf.perstype6.page	Integer	6	Individual Non-mandatory tour frequency UEC utility for Driving students page
inmtf.perstype7.page	Integer	7	Individual Non-mandatory tour frequency UEC utility for Pre-driving students page
inmtf.perstype8.page	Integer	8	Individual Non-mandatory tour frequency UEC utility for Preschool students page
awtf.uec.file		AtWorkSubtourFrequency.xls	File name of at-work sub-tour frequency UEC
awtf.data.page	Integer	0	At-Work Sub-Tour Frequency UEC Data page

Property	Data Type	Example Value	Purpose
awtf.model.page	Integer	1	At-Work Sub-Tour Frequency UEC Utility page
stf.uec.file	String	StopFrequency.xls	File name of Stop Frequency UEC
stf.purposeLookup.proportions	String	StopPurposeLookupProportions.csv	File name of Stop Purpose Lookup proportions
stf.data.page	Integer	0	Stop Frequency UEC data page
stf.work.page	Integer	1	Stop Frequency for Work Tour UEC utility page
stf.univ.page	Integer	2	Stop Frequency for University Tour UEC utility page
stf.school.page	Integer	3	Stop Frequency for School Tour UEC utility page
stf.escort.page	Integer	4	Stop Frequency for Escort Tour UEC utility page
stf.shop.page	Integer	5	Stop Frequency for Shop Tour UEC utility page
stf.maint.page	Integer	6	Stop Frequency for Maintenance Tour UEC utility page
stf.eat.page	Integer	7	Stop Frequency for Eating Out Tour UEC utility page
stf.visit.page	Integer	8	Stop Frequency for Visiting Tour UEC utility page
stf.discr.page	Integer	9	Stop Frequency for Discretionary Tour UEC utility page
stf.subtour.page	Integer	10	Stop Frequency for At-Work Sub-Tour UEC utility page
slc.uec.file	String	StopLocationChoice.xls	File Name of Stop Location Choice UEC
slc.uec.data.page	Integer	0	Stop Location Choice UEC data page

Property	Data Type	Example Value	Purpose
slc.mandatory.uec.model.page	Integer	1	Stop Location Choice for Mandatory Tours UEC utility page
slc.maintenance.uec.model.page	Integer	2	Stop Location Choice for Maintenance Tours UEC utility page
slc.discretionary.uec.model.page	Integer	3	Stop Location Choice for Discretionary Tours UEC utility page
slc.alts.file	String	SlcAlternatives.csv	File name of stop location choice alternatives
slc.soa.alts.file	String	DestinationChoiceAlternatives.csv	File name of the alternatives (MGRAs) available to the destination choice models (part of the model design; this should not be changed)
auto.slc.soa.distance.uec.file	String	SlcSoaDistanceUtility.xls	File name of Stop Location Sample of Alternatives Choice UEC for tourmodes other than walk or bike - for transit, availability of stop for transit is set in java code
auto.slc.soa.distance.data.page	Integer	0	Stop Location SOA Choice UEC data page
auto.slc.soa.distance.model.page	Integer	1	Stop Location SOA Choice UEC utility page
slc.soa.size.uec.file	String	SlcSoaSize.xls	File Name of Stop Location Choice Size Terms UEC
slc.soa.size.uec.data.page	Integer	0	Stop Location Choice Size terms UEC data page
slc.soa.size.uec.model.page	Integer	1	Stop Location Choice Size terms UEC utility page
stop.depart.arrive.proportions	String	StopDepartArriveProportions.csv	File name of Stop Location Time of Day proportions
tripModeChoice.uec.file	String	TripModeChoice.xls	File name of Trip mode choice UEC

Property	Data Type	Example Value	Purpose
plc.uec.file	String	ParkLocationChoice.xls	File name of Parking Location Choice UEC
plc.uec.data.page	Integer	0	Parking Location Choice UEC data page
plc.uec.model.page	Integer	1	Parking Location Choice UEC utility page
plc.alts.corresp.file	String	ParkLocationAlts.csv	File name of parking location alternatives (MAZs)
plc.alts.file	String	ParkLocationSampleAlts.csv	File name of parking location sample of alternatives
mgra.avg.cost.output.file	String	mgraParkingCost.csv	File name of average parking costs by MAZ
mgra.avg.cost.trace.zone	Integer	1814	MAZ parking cost trace zone
mgra.max.parking.distance	Integer	0.75	Max parking walk distance
mgra.avg.cost.dist.coeff.work	Float	-8.6	Parking location model coefficient for walking distance to destination for Work purpose
mgra.avg.cost.dist.coeff.other	Float	-4.9	Parking location model coefficient for walking distance to destination for other purposes
park.cost.reimb.mean	Float	-0.05	Parking location model mean parking cost reimbursement
park.cost.reimb.std.dev	Float	0.54	Parking location model standard deviation for parking cost reimbursement
utility.bestTransitPath.uec.file	String	BestTransitPathUtility.xls	File name of best transit path UEC
utility.bestTransitPath.data.page	Integer	0	Best Transit Path UEC data page
utility.bestTransitPath.tapToTap.ea.page	Integer	1	Best Transit Path UEC for TAP to TAP Early AM utility page

Property	Data Type	Example Value	Purpose
utility.bestTransitPath.tapToTap.am.page	Integer	2	Best Transit Path UEC for TAP to TAP AM utility page
utility.bestTransitPath.tapToTap.md.page	Integer	3	Best Transit Path UEC for TAP to TAP MD utility page
utility.bestTransitPath.tapToTap.pm.page	Integer	4	Best Transit Path UEC for TAP to TAP PM utility page
utility.bestTransitPath.tapToTap.ev.page	Integer	5	Best Transit Path UEC for TAP to TAP Evening utility page
utility.bestTransitPath.walkAccess.page	Integer	6	Best Transit Path UEC for Walk Access to Transit utility page
utility.bestTransitPath.driveAccess.page	Integer	7	Best Transit Path UEC for Drive Access to Transit utility page
utility.bestTransitPath.walkEgress.page	Integer	8	Best Transit Path UEC for Walk Egress to Transit utility page
utility.bestTransitPath.driveEgress.page	Integer	9	Best Transit Path UEC for Drive Egress to Transit utility page
skims.auto.uec.file	String	AutoSkims.xls	File name of Auto Skims UEC
skims.auto.data.page	Integer	0	Auto Skims data page
skims.auto.ea.page	Integer	1	Auto skims Early AM utility page
skims.auto.am.page	Integer	2	Auto skims AM utility page
skims.auto.md.page	Integer	3	Auto skims MD utility page
skims.auto.pm.page	Integer	4	Auto skims PM utility page
skims.auto.ev.page	Integer	5	Auto skims Evening utility page
taz.distance.uec.file	String	tazDistance.xls	File name of TAZ Distance UEC

Property	Data Type	Example Value	Purpose
taz.distance.data.page	Integer	0	TAZ Distance UEC data page
taz.od.distance.ea.page	Integer	1	TAZ Distance UEC Early AM utility page
taz.od.distance.am.page	Integer	2	TAZ Distance UEC AM utility page
taz.od.distance.md.page	Integer	3	TAZ Distance UEC MD utility page
taz.od.distance.pm.page	Integer	4	TAZ Distance UEC PM utility page
taz.od.distance.ev.page	Integer	5	TAZ Distance UEC Evening utility page
skim.walk.transit.walk.uec.file	String	WalkTransitWalkSkims.xls	File name of Walk Transit Walk Skims UEC
skim.walk.transit.walk.data.page	Integer	0	Walk Transit Walk Skims UEC data page
skim.walk.local.walk.ea.page	Integer	1	Walk Local Walk Early AM Skims UEC utility page
skim.walk.local.walk.am.page	Integer	3	Walk Local Walk AM Skims UEC utility page
skim.walk.local.walk.md.page	Integer	5	Walk Local Walk MD Skims UEC utility page
skim.walk.local.walk.pm.page	Integer	7	Walk Local Walk PM Skims UEC utility page
skim.walk.local.walk.ev.page	Integer	9	Walk Local Walk Evening Skims UEC utility page
skim.walk.premium.walk.ea.page	Integer	2	Walk Premium Walk Early AM Skims UEC utility page
skim.walk.premium.walk.am.page	Integer	4	Walk Premium Walk AM Skims UEC utility page
skim.walk.premium.walk.md.page	Integer	6	Walk Premium Walk MD Skims UEC utility page
skim.walk.premium.walk.pm.page	Integer	8	Walk Premium Walk PM Skims UEC utility page

Property	Data Type	Example Value	Purpose
skim.walk.premium.walk.ev.page	Integer	10	Walk Premium Walk Evening Skims UEC utility page
skim.walk.transit.drive.uec.file	String	WalkTransitDriveSkims.xls	File name of Walk Transit Drive Skims UEC
skim.walk.transit.drive.data.page	Integer	0	Walk Transit Drive Skims UEC data page
skim.walk.local.drive.ea.page	Integer	1	Walk Local Drive Early AM Skims UEC utility page
skim.walk.local.drive.am.page	Integer	3	Walk Local Drive AM Skims UEC utility page
skim.walk.local.drive.md.page	Integer	5	Walk Local Drive MD Skims UEC utility page
skim.walk.local.drive.pm.page	Integer	7	Walk Local Drive PM Skims UEC utility page
skim.walk.local.drive.ev.page	Integer	9	Walk Local Drive Evening Skims UEC utility page
skim.walk.premium.drive.ea.page	Integer	2	Walk Premium Drive Early AM Skims UEC utility page
skim.walk.premium.drive.am.page	Integer	4	Walk Premium Drive AM Skims UEC utility page
skim.walk.premium.drive.md.page	Integer	6	Walk Premium Drive MD Skims UEC utility page
skim.walk.premium.drive.pm.page	Integer	8	Walk Premium Drive PM Skims UEC utility page
skim.walk.premium.drive.ev.page	Integer	10	Walk Premium Drive Evening Skims UEC utility page
skim.drive.transit.walk.uec.file	String	DriveTransitWalkSkims.xls	File name of Drive Transit Walk Skims UEC
skim.drive.transit.walk.data.page	Integer	0	Drive Transit Walk Skims UEC data page
skim.drive.local.walk.ea.page	Integer	1	Drive Local Walk Early AM Skims UEC utility page
skim.drive.local.walk.am.page	Integer	3	Drive Local Walk AM Skims UEC utility page

Property	Data Type	Example Value	Purpose
skim.drive.local.walk.md.page	Integer	5	Drive Local Walk MD Skims UEC utility page
skim.drive.local.walk.pm.page	Integer	7	Drive Local Walk PM Skims UEC utility page
skim.drive.local.walk.ev.page	Integer	9	Drive Local Walk Evening Skims UEC utility page
skim.drive.premium.walk.ea.page	Integer	2	Drive Premium Walk Early AM Skims UEC utility page
skim.drive.premium.walk.am.page	Integer	4	Drive Premium Walk AM Skims UEC utility page
skim.drive.premium.walk.md.page	Integer	6	Drive Premium Walk MD Skims UEC utility page
skim.drive.premium.walk.pm.page	Integer	8	Drive Premium Walk PM Skims UEC utility page
skim.drive.premium.walk.ev.page	Integer	10	Drive Premium Walk Evening Skims UEC utility page
visitor.seek	Boolean	false	True or False to seek Visitor Model
visitor.trace	Integer	58	Specify what zone to trace for Visitor model
visitor.hotel.occupancyRate	Float	0.7	specify Hotel occupancy rate
visitor.household.occupancyRate	Float	0.018	Specify household/residential occupancy rate
visitor.hotel.businessPercent	Float	0.3	Specify proportion of hotel visitors on business trip
visitor.household.businessPercent	Float	0.04	Specify proportion of household/residential visitors on business trip
visitor.business.tour.file	String	input/visitor_businessFrequency.csv	File location/name of visitors on business trip tour frequency proportions

Property	Data Type	Example Value	Purpose
visitor.personal.tour.file	String	input/visitor_personalFrequency.csv	File location/name of visitors on personal trip tour frequency proportions
visitor.partySize.file	String	input/visitor_partySize.csv	File location/name of visitor proportion of trips by tour purpose and party size
visitor.autoAvailable.file	String	input/visitor_autoAvailable.csv	File location/name of visitor's auto availability by tour purpose
visitor.income.file	String	input/visitor_income.csv	File location/name of visitor's income proportion by tour purpose
visitor.dc.soa.uec.file	String	VisitorDestinationChoiceSample.xls	File name of Visitor Destination choice sample UEC
visitor.dc.soa.data.page	Integer	0	Visitor destination choice Sample of Alternatives UEC data page
visitor.dc.soa.work.page	Integer	1	Visitor destination choice Sample of Alternatives UEC Work Tour utility page
visitor.dc.soa.recreate.page	Integer	2	Visitor destination choice Sample of Alternatives UEC Recreation Tour utility page
visitor.dc.soa.dining.page	Integer	3	Visitor destination choice Sample of Alternatives UEC Dining Tour utility page
visitor.dc.soa.size.page	Integer	4	Visitor destination choice Sample of Alternatives UEC Size Terms utility page
visitor.dc.soa.alts.file	String	SoaTazDistAlts.csv	File name of Sample of Alternatives of TAZs
visitor.dc.uec.file	String	VisitorDestinationChoice.xls	File name of Visitor Destination Choice UEC
visitor.dc.data.page	Integer	0	Visitor Destination Choice UEC data page

Property	Data Type	Example Value	Purpose
visitor.dc.work.page	Integer	1	Visitor Destination Choice UEC Work Tour utility page
visitor.dc.recreate.page	Integer	2	Visitor Destination Choice UEC Recreation Tour utility page
visitor.dc.dining.page	Integer	3	Visitor Destination Choice UEC Dining Tour utility page
visitor.tour.tod.file	String	input/visitor_tourTOD.csv	File location/name of Visitor Tour TOD proportions
visitor.mc.uec.file	String	VisitorTourModeChoice.xls	File name of Visitor Tour Mode choice UEC
visitor.mc.data.page	Integer	0	Visitor Tour Mode Choice UEC data page
visitor.mc.model.page	Integer	1	Visitor Tour Mode Choice UEC utility page
visitor.stop.frequency.file	String	input/visitor_stopFrequency.csv	File location/name of Visitor Stop Frequency proportions
visitor.stop.purpose.file	String	input/visitor_stopPurpose.csv	File location/name of Visitor Stop Purpose proportions
visitor.stop.outbound.duration.file	String	input/visitor_outboundStopDuration.csv	File location/name of Visitor Outbound Stop Duration proportions
visitor.stop.inbound.duration.file	String	input/visitor_inboundStopDuration.csv	File location/name of Visitor Inbound Stop Duration proportions
visitor.slc.soa.uec.file	String	VisitorStopLocationChoiceSample.xls	File name of Visitor Stop Location Choice Sample of Alternatives UEC
visitor.slc.soa.data.page	Integer	0	Visitor Stop Location Choice SOA UEC data page
visitor.slc.soa.model.page	Integer	1	Visitor Stop Location Choice SOA UEC utility page
visitor.slc.uec.file	String	VisitorStopLocationChoice.xls	File name of Visitor Stop Location Choice UEC
visitor.slc.data.page	Integer	0	Visitor Stop Location Choice UEC data page

Property	Data Type	Example Value	Purpose
visitor.slc.model.page	Integer	1	Visitor Stop Location Choice UEC utility page
visitor.trip.mc.uec.file	String	VisitorTripModeChoice.xls	File name of Visitor Trip Mode Choice UEC
visitor.trip.mc.data.page	Integer	0	Visitor Trip Mode Choice UEC data page
visitor.trip.mc.model.page	Integer	1	Visitor Trip Mode Choice UEC utility page
visitor.tour.output.file	String	output/visitorTours.csv	File location/name of visitor tour output file
visitor.trip.output.file	String	output/visitorTrips.csv	File location/name of visitor trip output file
visitor.results.autoTripMatrix	String	output/autoVisitorTrips	File location/name of visitor auto trips output file
visitor.results.nMotTripMatrix	String	output/nmotVisitorTrips	File location/name of visitor non-motorized trips output file
visitor.results.tranTripMatrix	String	output/tranVisitorTrips	File location/name of visitor transit trips output file
visitor.results.othrTripMatrix	String	output/othrVisitorTrips	File location/name of visitor other modes trips output file
visitor.uec.file	String	VisitorSize.xls	File name for visitor size UEC file (for visitor estimation only)
visitor.uec.data.page	Integer	0	Visitor UEC data page (for visitor estimation only)
visitor.uec.sizeTerms.page	Integer	1	Visitor UEC size terms page (for visitor estimation only)
summit.output.directory	String	output/	File location for Summit output
summit.purpose.Work	Integer	1	Specify code for Work Purpose
summit.purpose.University	Integer	2	Specify code for University Purpose

Property	Data Type	Example Value	Purpose
summit.purpose.School	Integer	3	Specify code for School Purpose
summit.purpose.Escort	Integer	4	Specify code for Escort Purpose
summit.purpose.Shop	Integer	4	Specify code for Shop Purpose
summit.purpose.Maintenance	Integer	4	Specify code for Maintenance Purpose
summit.purpose.EatingOut	Integer	5	Specify code for Eating out Purpose
summit.purpose.Visiting	Integer	5	Specify code for Visiting Purpose
summit.purpose.Discretionary	Integer	5	Specify code for Discretionary Purpose
summit.purpose.WorkBased	Integer	6	Specify code for At-Work Sub Tour Purpose
summit.filename.1	String	Work	Specify file name for Work Purpose
summit.filename.2	String	University	Specify file name for University Purpose
summit.filename.3	String	School	Specify file name for School Purpose
summit.filename.4	String	Maintenance	Specify file name for Maintenance Purpose
summit.filename.5	String	Discretionary	Specify file name for Discretionary Purpose
summit.filename.6	String	Workbased	Specify file name for At-Work Sub tour Purpose
summit.ivt.file.1	Float	-0.016	Specify in-vehicle time coefficient for Work Purpose
summit.ivt.file.2	Float	-0.016	Specify in-vehicle time coefficient for University Purpose

Property	Data Type	Example Value	Purpose
summit.ivt.file.3	Float	-0.01	Specify in-vehicle time coefficient for School Purpose
summit.ivt.file.4	Float	-0.017	Specify in-vehicle time coefficient for Maintenance Purpose
summit.ivt.file.5	Float	-0.015	Specify in-vehicle time coefficient for Discretionary Purpose
summit.ivt.file.6	Float	-0.032	Specify in-vehicle time coefficient for At-work Sub tour Purpose
summit.modes	Integer	26	Specify number of modes in the model
summit.mode.array	String	0,0,0,0,0,0,0,0,0, 0, 1, 1, 1, 1, 1, 2, 2, 2, 2, 2, 2, 2, 2, 2, 0	Specify mode array; 0 for auto modes/school bus mode, 1 for walk transit, 2 for drive transit modes
summit.upperEA	Integer	3	Specify upper limit code for Early AM time period
summit.upperAM	Integer	9	Specify upper limit code for AM time period
summit.upperMD	Integer	22	Specify upper limit code for Midday time period
summit.upperPM	Integer	29	Specify upper limit code for PM time period
occ3plus.purpose.Work	Float	3.5	Specify 3+ occupancy multiplier for trip table creation for Work Purpose
occ3plus.purpose.University	Float	3.5	Specify 3+ occupancy multiplier for trip table creation for University Purpose
occ3plus.purpose.School	Float	3.5	Specify 3+ occupancy multiplier for trip table creation for School Purpose

Property	Data Type	Example Value	Purpose
occ3plus.purpose.Escort	Float	3.5	Specify 3+ occupancy multiplier for trip table creation for Escort Purpose
occ3plus.purpose.Shop	Float	3.5	Specify 3+ occupancy multiplier for trip table creation for Shop Purpose
occ3plus.purpose.Maintenance	Float	3.5	Specify 3+ occupancy multiplier for trip table creation for Maintenance Purpose
occ3plus.purpose.EatingOut	Float	3.5	Specify 3+ occupancy multiplier for trip table creation for Eating Out Purpose
occ3plus.purpose.Visiting	Float	3.5	Specify 3+ occupancy multiplier for trip table creation for Visiting Purpose
occ3plus.purpose.Discretionary	Float	3.5	Specify 3+ occupancy multiplier for trip table creation for Discretionary Purpose
occ3plus.purpose.WorkBased	Float	3.5	Specify 3+ occupancy multiplier for trip table creation for At-Work Sub Tour Purpose
workSchoolSegments.definitions	String	workSchoolSegments.definitions	Work school segment definitions file
HouseholdManager.MinValueOfTime	Float	1.0	Distributed person VOT settings
HouseholdManager.MaxValueOfTime	Float	50	Distributed person VOT settings
HouseholdManager.MeanValueOfTime.Values	String	6.01, 8.81, 10.44, 12.86	Distributed person VOT settings
HouseholdManager.MeanValueOfTime.Income.Limits	String	30000, 60000, 100000	Distributed person VOT settings

Property	Data Type	Example Value	Purpose
HouseholdManager.Mean.ValueOfTime.Multiplier.Mu	Float	0.684	Distributed person VOT settings
HouseholdManager.ValueOfTime.Lognormal.Sigma	Float	0.87	Distributed person VOT settings
HouseholdManager.HH.ValueOfTime.Multiplier.Under18	Float	0.66667	Distributed person VOT settings
HouseholdManager.MinValueOfTime	Float	1	Distributed person VOT settings
District.Definitions	String	V7Zone_District_County.csv	TAZ district scheme for reporting

UEC Files

The following UEC files are stored in the ctramp/uec folder and are required to run the model:

Table 40: UEC Files

File	Purpose
Acc_alts.csv	Accessibilities alternatives
Accessibilities.xls	Accessibilities model
Accessibilities_DC.xls	Accessibilities destination choice model
AtWorkSubtourFrequency.xls	At work sub tour frequencies model
AutoOwnership.xls	Auto ownership model
AutoSkims.xls	Auto skims lookup UEC
BestTransitPathUtility.xls	Best transit path utility model
CoordinatedDailyActivityPattern.xls	CDAP model
DepartureTimeAndDurationAlternatives.csv	Departure time and duration alternatives
DestinationChoiceAlternatives.csv	Destination choice alternatives
DestinationChoiceAlternativeSample.xls	Destination choice alternatives sample model
DriveTransitWalkSkims.xls	Drive transit walk skims lookup UEC

File	Purpose
DriveTransitWalkUtility.xls	Drive transit walk utility model (used for estimation)
IndividualNonMandatoryTourFrequencyAlternatives.csv	Individual non mandatory tour frequency alternatives
IndividualNonMandatoryTourFrequencyExtensionProbabilities_p1.csv	Individual non mandatory tour frequency extension probabilities lookup file
JointAlternatives.csv	Joint tour alternatives
JointTourFrequency.xls	Joint tour frequency model
MandatoryAccess.xls	Mandatory accessibilities model
MandatoryTourFrequency.xls	Mandatory tour frequency model
NonMandatoryIndividualTourFrequency.xls	Non mandatory individual tour frequency model
NonMandatoryTlcAlternatives.csv	Non mandatory TLC alternatives
ParkingProvision.xls	Free parking model
ParkLocationAlts.csv	Parking location alternatives
ParkLocationChoice.xls	Parking location choice model
ParkLocationSampleAlts.csv	Parking location alternatives
SlcAlternatives.csv	Stop location choice alternatives
SlcSoaDistanceUtility.xls	Stop location choice sample of alternatives distance utility model
SlcSoaSize.xls	Stop location choice sample of alternatives size terms
SoaTazDistAlts.csv	Sample of alternatives TAZ distance alternatives
StopDepartArriveProportions.csv	Stop departure and arrival proportions lookup file
StopFrequency.xls	Stop frequency model
StopLocationChoice.xls	Stop location choice model
StopPurposeLookupProportions.csv	Stop purpose lookup proportions lookup file
TazDistance.xls	TAZ distance lookup UEC

File	Purpose
TourDcSoaDistance.xls	Tour destination choice sample of alternatives distance model
TourDcSoaDistanceAlts.csv	Tour destination choice sample of alternatives distance alternatives for work purposes
TourDcSoaDistanceNoSchoolSize.xls	Tour destination choice sample of alternatives distance for school purposes
TourDepartureAndDuration.xls	Tour departure and duration model
TourDestinationChoice.xls	Tour destination choice model
TourModeChoice.xls	Tour mode choice model
TransponderOwnership.xls	Transponder ownership model
TripModeChoice.xls	Trip mode choice model
WalkTransitDriveSkims.xls	Walk transit drive skim lookup UEC
WalkTransitDriveUtility.xls	Walk transit drive utility model (used for estimation)
WalkTransitWalkSkims.xls	Walk transit walk skim lookup UEC
WalkTransitWalkUtility.xls	Walk transit walk utility model (used for estimation)
VisitorDestinationChoiceSample.xls	Visitor Destination choice sample model
VisitorDestinationChoice.xls	Visitor Destination Choice model
VisitorTourModeChoice.xls	Visitor Tour Mode choice model
VisitorStopLocationChoiceSample.xls	Visitor Stop Location Choice Sample of Alternatives model
VisitorStopLocationChoice.xls	Visitor Stop Location Choice model
VisitorTripModeChoice.xls	Visitor Trip Mode Choice model
VisitorSize.xls	Visitor size term model (for visitor estimation only)

Running the Model

This section provides instructions for executing the travel model.

Run the Entire Model via Cube

The entire model stream can be run from Cube. This includes running the non-ABM trip models, the network build procedures, the highway and transit skimming procedures, the CT-RAMP ABM demand model, and the final assignments.

To run the entire model, do the following:

1) If running the model in distributed mode:
    a. Ensure the JPPF service settings and Run Component files are configured for distributed mode, as described in the Setup and Configuration section
    b. Ensure the JPPF driver is started in the ORPIL00A.s Cube script by removing the rem statement at the beginning of the call (see below)
    c. Remote desktop to the machine with the JPPF worker node
    d. Double click runNode1.cmd in ctramp/config folder to start the remote JPPF process
2) Open serpm7.cat in Cube
3) Click the Run button to run the entire Cube application, which will run CT-RAMP via the command line.
    a. The CT-RAMP command line calls are in the ORPIL00A.s script.
    b. The number of overall model feedback iterations is set in the 01PIL00I.s script and is currently set to 3. The model averages the link times via method of successive averages (MSA).

Run only CT-RAMP Components

The CT-RAMP model components can be run without Cube if the highway and transit skims have already been created. This method of running the model is useful for calibrating certain components of the model. It can be run for any iteration, and at any sample rate. It is useful to run these models at a sample rate of 20 to 25% for calibration in order to reduce runtimes without compromising the statistical reliability of the sample.

Prior to executing the CT-RAMP model, the user must call these DOS commands from the main computer. To do this, do the following:

1) If running the model in distributed mode:
    a. Ensure the JPPF service settings and Run Component files are configured for distributed mode, as described in the Setup and Configuration section
    b. Remote desktop to the machine with the JPPF worker node
    c. Double click runNode1.cmd in ctramp/config folder to start the remote JPPF process

2) Open a DOS command window and navigate to C:\serpm7\Cube

3) Run the following commands to copy the model inputs and outputs to the generic ABMTEMP folder, create the MAZ to MAZ/TAP accessibility files, run CT-RAMP, create demand matrices for Cube, and copy results back to the scenario folder:

a. rmdir {OUTDIR}\..\ABMTEMP /S /Q
b. rmdir {DATADIR}\..\ABMTEMP /S /Q
c. xcopy {OUTDIR}* {OUTDIR}\..\ABMTEMP /E /I /Y
d. xcopy {DATADIR}* {DATADIR}\..\ABMTEMP /E /I /Y

e. {CUBE}\..\ctramp\runCreateAccessFiles.cmd

f. taskkill /im "java.exe" /F
g. mkdir logFiles
h. start {CUBE}\..\ctramp\runHhMgr.cmd {JDK64} {HOST_IP}
i. start {CUBE}\..\ctramp\runMtxMgr.cmd {HOST_IP} {JDK64}
j. rem start {CUBE}\..\ctramp\config\runDriver.cmd #un-rem for distributed mode
k. xcopy {CUBE}\..\ctramp\serpm_abm.properties serpm_abm.properties /E /Y
l. {CUBE}\..\ctramp\runSERPMAbm.cmd {SAMPLERATE} {VSAMPLERATE} {JDK64}
m. {CUBE}\..\ctramp\runCreateDemandMatrices.cmd {SAMPLERATE} {VSAMPLERATE} {JDK64}
n. taskkill /im "java.exe" /F

o. xcopy {OUTDIR}\..\ABMTEMP* {OUTDIR} /E /I /Y
p. xcopy {DATADIR}\..\ABMTEMP* {DATADIR} /E /I /Y
q. rmdir {OUTDIR}\..\ABMTEMP /S /Q
r. rmdir {DATADIR}\..\ABMTEMP /S /Q

Where {X} are Cube application manager tokens such as:
{ OUTDIR} = C:\SERPM7\Output\Out-2010R
{ DATADIR }= C:\SERPM7\Input\In-2010R
{CUBE} = C:\serpm7\Cube
{SAMPLERATE}=0.25
{JDK64} = "/Program Files/Java/jdk1.7.0_21"
{VSAMPLERATE} = 0.25

The commands above start by copying the scenario inputs and outputs to the ABMTEMP generic folder. Next it creates the CT-RAMP accessibility files. The Household data manager and Matrix data manager processes are then started in separate DOS windows. The JPPF driver is started for managing the distribution of tasks across the cluster. CT-RAMP is then run with the user supplied

sample rate and the visitor model is run with the user supplied visitor model sample rate. After running CT-RAMP, the Cube demand matrices are created based on the microsimulated trip list output files. Finally, the outputs from CT-RAMP are copied back to the scenario folder.

Output Files

The table below contains brief descriptions of the output files produced by the SERPM 7 travel model. See the Output Files Data Dictionary section for more details on variables in each output file.

Table 41: Output Files

File Name	Description	File Type
MPPTR00A.LIN	ASCII line file in PT format	Text
EETAB-PRD.{ALT}{Year}.MAT	EE trips for each of the five time periods	Cube matrix
TAZ3.dbf	Dynamic area type	DBF
Air_truck_panda.dbf	Airport and Truck P's and A's	DBF
EI_Auto_Panda.dbf	EI productions and attractions	DBF
TAP_LINKS.DBF	TAP to transit Node Connectors	DBF
TAP_Nodes.DBF	TAP Node Data	DBF
TURNS_{Year}.PEN	Cube format turn penalties	Text
tap_net.net	Highway network with TAP connectors added	Net
nonABM_<<time period>>_OD.mat	Airport, truck, EI, trips for <<time period>>	Cube matrix
DAILY_LOADED.net	Loaded highway network containing daily loads, and loads by time period.	Net
<<time period>>_HLOAD.NET	Loaded highway network by <<time period>> These networks contain loads by highway mode and path: 1) General Purpose lanes: DA, SR2, SR3+ 2) HOV (excludes toll and HOT): SR2 and SR3+ 3) Pay (Turnpike or HOT): DA, SR2, SR3+ 4) Trucks and external (EE+EI/IE)	Net
<<time period>>_<<ACCESSM>>_<<pathname>>_TASN.NET	Loaded transit network files for premium paths, not useful for display	Net
<<time period>>_Local_<<pathname>>_TASN.NET	Loaded transit network files for local only paths	Net
TLINK_<<time period>>_<<ACCESSM>>_<<pathname>>.DBF	Loaded transit network links for premium paths	DBF
TLINK_<<time period>>_Local.DBF	Loaded transit network link for local only paths	DBF

File Name	Description	File Type
FIRST_LAST_<<time period>>_<<ACCESSM>>_<<pathname>>.DBF	Stop to stop loads for premium paths	DBF
FIRST_LAST_<<time period>>_Local.DBF	Stop to stop loads for local only paths	DBF
<<time period>>_<<ACCESSM>>_<<pathname>>_TASN.PRN	Transit assignment report for premium paths This is a PT text file containing several reports	Text
<<time period>>_Local_TASN.PRN	Transit assignment report for local only paths This is a PT text file containing several reports	Text
TSKIM<<time period>>_Walk_Walk_WithLOC_<<alt>><<year>>.mat	TAP to TAP transit skims for <<time period>> <<alt>> and <<year>> for walk premium	Cube matrix
TSKIM<<time period>>_Walk_Walk_LocOnly_<<alt>><<year>>.mat	TAP to TAP transit skims for <<time period>> <<alt>> and <<year>> for walk local only	Cube matrix
<<time period>>HSKIMS_<<alt>><<year>>.mat	TAZ to TAZ highway skims for <<time period>> <<alt>> and <<year>>	Cube matrix
Accessibilities.csv	Accessibilities output file	CSV
aoResults.csv	Auto-ownership output file	CSV
ShadowPricingOutput_work_<<shadow iteration>>.csv	Shadow pricing output for work location by iteration	CSV
ShadowPricingOutput_school<<shadow iteration>>.csv	Shadow pricing output for school location by iteration	CSV
wsLocresults_<<iteration>>.csv	Work and school location output file for each iteration	CSV
Householdata_<<iteration>>.csv	Household output file for each iteration	CSV
Persondata_<<iteration>>.csv	Person output file for each iteration	CSV
Indivtourdata_<<iteration>>.csv	Individual tour output file for each iteration	CSV
Jointtourdata_<<iteration>>.csv	Joint tour output file for each iteration	CSV
Indivtripdata_<<iteration>>.csv	Individual trip output file for each iteration	CSV
Jointripdata_<<iteration>>.csv	Joint trip output file for each iteration	CSV

File Name	Description	File Type
TAZ_Demand_<<time period>>.mat	Output CT-RAMP trip tables by highway mode and time period	Cube matrix
TAP_Demand_<<time period>>.mat	Output CT-RAMP trip tables by transit mode and time period	Cube matrix
Up.net	Current highway network, updated with congested times and V/C. For the first feedback iteration, the update data come from seed values from an earlier model run. For subsequent iterations, MSA values from the preceding iteration is used.	Net
visitorTours.csv	Visitor tours	CSV
visitorTrips.csv	Visitor trips	CSV

Output Files Data Dictionary

Table 42: ASCII Transit Line File (standard PT file)

Table Name	Description
LINE	Record header
NAME	Short name for the line
LONGNAME	Long name for the line
HEADWAY[x]	1=AM headway, 2=OP headway, 3=PM headway
MODE	Transit Mode 111 Tri-Rail 121 Metrorail 131 Regional LRT 151 Inter-County Express Bus 181 Exclusive ROW Cir-Reg 191 Trolleys/Shuttles-Reg 192 Trolleys/Shuttles-Reg 231 LRT- MDT 241 Buses (local/express) - MDT 242 Buses (local/express) - MDT 243 BRT - MDT 251 Buses (local/express) - MDT 252 Buses (local/express) - MDT 253 Inter-County Express Bus

	281 Exclusive ROW Cir-MDT 291 Trolleys/Shuttles-MDT 292 Buses (local/express) - MDT 293 Buses (local/express) - MDT 331 LRT - BCT 341 BCT Rapid Bus BRT 342 BCT Fixed-guideway BRT 351 Buses (local/express) - BCT 352 Buses (local/express) - BCT 381 Exclusive ROW Cir-BCT 393 Trolleys/Shuttles-BCT 391 Buses (local/express) - BCT 392 Buses (local/express) - BCT 431 LRT - Palm Tran 441 BRT - Palm Tran 451 Buses (local/express) - Ptran 481 Exclusive ROW Cir-PTran 491 Trolleys/Shuttles-Ptran 492 Buses (local/express) - Ptran
ONEWAY	0=2-way, 1=1-way
OPERATOR	11 Tri-Rail 12 Tri-Rail Shuttles 13 Exclusive ROW Cir-Reg 21 MDT 22 MDT Express 23 MDT Shuttle 24 MDT 25 MDT 26 Metromover 27 Local 28 MDT BRT/LRT 31 BCT 32 BCT Breeze 33 BCT Express 34 BCT Rapid Bus 35 Palm Tran Exclusive ROW Cir 42 Palm Tran Exp 43 Palm Tran LRT/BRT
USERA4	Dwell time category 15 Tri-Rail 14 Metrorail

	14 Regional LRT 16 I-95 Inter-County Express 12 Exclusive ROW Cir-Reg 11 Tri-Rail Shuttles 33 Tri-Rail Shuttles 14 MDT LRT 27 MDT Busway Flyers 27 MDT MAX/KAT/Busway Local 28 MDT BRT/LRT 24 MDT Express 26 MDT I-95 Exp 16 I-95 Inter-County Express 22 Metromover 21 MDT Trolleys/Shuttles 23 MDT Local 25 MDT Shuttle 37 BCT LRT 36 BCT Rapid Bus 37 BCT BRT 34 BCT Breeze 35 BCT Express 32 BCT Exclusive ROW Cir 31 BCT Trolleys/Shuttles 27 BCT Local 33 BCT Local 44 Palm Tran LRT 44 Palm Tran BRT 44 Palm Tran Exp 42 Palm Tran Exclusive ROW Cir 41 Palm Tran Trolleys/Shuttles 43 Palm Tran Local
N	Node numbers
DWELL_C	Dwell Time in minutes

Table 43: EE Trip Table by time period - EETAB-PRD.{ALT}{Year}.MAT

Table Name	Description
EE_EA	Early vehicles
EE_AM	AM vehicles
EE_MD	Midday vehicles

EE_PM	PM vehicles
EE_EV	Evening vehicles

Table 44: Dynamic Area Type - zdata3.dbf

Table Name	Description
TAZ	TAZ number
SAT1	Dynamic area type 1. CBD 2. Fringe 3. OBD 4. Generally residential 5. Rural

Table 45: Airport and Truck P's and A's (vehicles) - Air_truck_panda.dbf

Table Name	Description
TAZ	Zone number
P_Airport	Daily airport productions
P_4tire	Daily four tire truck productions
P_SU	Daily single unit truck productions
P_COMP	Daily combination truck productions
A_Airport	Daily airport attractions
A_4tire	Daily four tire truck attractions
A_SU	Daily single unit truck attractions
A_COMP	Daily combination truck attractions

Table 46: External-internal P's and A's - EI_Auto_Panda.dbf (vehicles)

Table Name	Description
TAZ	Zone number
EI_P	External-internal P's
EI_A	External-internal A's

Table 47: TAP to Transit Node Connectors - TAP_LINKS.DBF

Table Name	Description
A	TAP or transit node

B	TAP or transit node
TAPMODE	Mode 1 – local 2 – express bus 3 – BRT 4 – Commuter rail 5 – People mover 6 – Heavy rail 7 – LRT
TIME_	time (always 0.01 minutes)
DISTANCE	distance (always 0.01 miles)
FTC2	2-digit facility type (always 51)

Table 48: TAP Node Data - TAP_Node.dbf

Table Name	Description
N	Sequential number
X	State plane x
Y	State plane y
TAPMODE	Same as TAP links
STATIONDESC	Station description
PKSPACES	Number of parking spaces
PKCOSTAM	Daily 9 hour parking cost
PKCOSTMD	Midday 3 hour parking cost
TTIMEPNR	Walk time from PNR lot (minutes)
TTIMEKNR	Walk time from KNR spaces (minutes)
STOPNODE	Transit stop node

Table 49: Turn penalties - TURNS_{Year}.PEN

Field	Description
1	From node
2	Through node
3	To node
4	Penalty set (1)
5	Penalty (minutes), -1=prohibitor

Table 50: Non-ABM Trip Tables - nonABM_<<time period>>_OD.mat

Table	Description
1	Airport vehicle trips
2	EI vehicle trips
3	4-tire truck trips
4	SU Truck trips
5	Combination truck trips

Table 51: Daily loaded highway network - DAILY_LOADED.net (added fields)

Field Name	Description
AM_CONGTIME	AM period congested time
AM_VCLOSE	AM period vol/LOS E Capacity
OF_CONGTIME	Off-peak period congested time
OF_VCLOSE	Off-peak period vol/LOS E Capacity
PM_CONGTIME	PM period congested time
PM_VCLOSE	PM period vol/LOS E Capacity
Day_V_1	Daily directional vehicles
EA_V_1	Early period directional vehicles
AM_V_1	AM period directional vehicles
MD_V_1	Midday period directional vehicles
PM_V_1	PM period directional vehicles
EV_V_1	Evening period directional vehicles

Table 52: Loaded highway network - <<time period>>_HLOAD}.NET (added fields)

Field Number	Description
1	General purpose drive-alone
2	Toll drive alone
3	General purpose SR2
4	HOV SR2
5	Toll SR2
6	General purpose SR3+

7	HOV SR3+
8	Toll SR3+
9	All trucks
10	External (EI/IE/EE)

Table 53: Loaded transit links (premium), TLINK_<<time period>>_<<ACCESSM>>_<<path name>>.DBF (primary fields)

Field Name	Description
A	A-node
B	B-node
MODE	Transit mode (see line records)
OPERATOR	Transit operator (see line records)
NAME	Route name
LONGNAME	Long route name
VOLUME	Link transit volume

Table 54: Loaded transit links (local only), TLINK_<<time period>>_Local.DBF, (primary fields)

Field Name	Description
A	A-node
B	B-node
MODE	Transit mode (see line records)
OPERATOR	Transit operator (see line records)
NAME	Route name
LONGNAME	Long route name
VOLUME	Link transit volume

Table 55: Stop to stop loads (premium paths), FIRST_LAST_<<time period>>_<<ACCESSM>>_<<pathname>>.DBF

Field Name	Description
I	From zone
J	To zone
FROMNODE	First transit node
TONODE	Last transit node

MODE	(0)
ACCUM	(1)
VOL	Transit volume

Table 56: Stop to stop loads (local-only paths), FIRST_LAST_<<time period>>_Local.DBF

Field Name	Description
I	From zone
J	To zone
FROMNODE	First transit node
TONODE	Last transit node
MODE	(0)
ACCUM	(1)
VOL	Transit volume

Table 57: Alternative highway net with congested times and speeds - up.net

Field Name	Description
a	Anode
b	Bnode
am_congtime	AM period congested time
am_congspd	AM period congested speed
am_vclose	AM period volume/(LOSE capacity)
pm_congtime	PM period congested time
pm_congspd	PM period congested speed
pm_vclose	PM period volume/(LOSE capacity)
of_congtime	Off-peak period congested time
of_congspd	Off-peak period congested speed
of_vclose	Off-peak period volume/(LOSE capacity)

Table 58: TAP to TAP Local Only Transit Skim Matrices

Table Name	Description
<<time period>>_XFRES	Transfers for <<time period>>
<<time period>>_TOTTIME	Total time for <<time period>>

<<time period>>_FARE	Fare for <<time period>>
<<time period>>_IWAIT	Initial wait for <<time period>>
<<time period>>_XWAIT	Transfer wait time for <<time period>>
<<time period>>_XFERTIME	Transfer time for <<time period>>

Table 59: TAP to TAP Premium Transit Skim Matrices

Table Name	Description
<<time period>>_XFRES	Transfers for <<time period>>
<<time period>>_IVT_CR	In-vehicle time commuter rail for <<time period>>
<<time period>>_IVT_LR	In-vehicle time light rail for <<time period>>
<<time period>>_IVT_BRT	In-vehicle time BRT for <<time period>>
<<time period>>_IVT_EXP	In-vehicle time express bus for <<time period>>
<<time period>>_IVT_LB	In-vehicle time local bus for <<time period>>
<<time period>>_MAINMODE	Main mode for <<time period>>
<<time period>>_FARE	Fare for <<time period>>
<<time period>>_IWAIT	Initial wait for <<time period>>
<<time period>>_XWAIT	Transfer wait time for <<time period>>
<<time period>>_XFERTIME	Transfer time for <<time period>>

Table 60: TAZ Highway Skim Matrices

Table Name	Description
<<time period>>_GP_TIME	General purpose lane travel time for <<time period>>
<<time period>>_GP_FFTIME	General purpose lane free flow time for <<time period>>
<<time period>>_GP_DIST	Drive alone toll distance for <<time period>>
<<time period>>_DAT_TIME	Drive alone toll travel time for <<time period>>
<<time period>>_DAT_FFTIME	Drive alone toll free flow time for <<time period>>
<<time period>>_DAT_DIST	Drive alone toll distance for <<time period>>
<<time period>>_DAT_TOLLCOST	Drive alone toll free toll for <<time period>>
<<time period>>_DAT_TOLLDIST	Drive alone toll toll distance for <<time period>>
<<time period>>_S2NH_TIME	Shared ride 2+ person no toll travel time for <<time period>>
<<time period>>_S2NH_FFTIME	Shared ride 2+ person no toll free flow time for <<time period>>

<<time period>>_S2NH_DIST	Shared ride 2+ person no toll distance for <<time period>>
<<time period>>_S2NH_HOVDIST	Shared ride 2+ person no toll HOV distance for <<time period>>
<<time period>>_S2TH_TIME	Shared ride 2+ person toll travel time for <<time period>>
<<time period>>_S2TH_FFTIME	Shared ride 2+ person toll free flow time for <<time period>>
<<time period>>_S2TH_DIST	Shared ride 2+ person toll distance for <<time period>>
<<time period>>_S2TH_TOLLDIST	Shared ride 2+ person toll toll distance for <<time period>>
<<time period>>_S2TH_TOLLCOST	Shared ride 2+ person toll toll cost for <<time period>>
<<time period>>_S2TH_HOVDIST	Shared ride 2+ person toll HOV distance for <<time period>>

Table 61: Accessibilities – accessibilities.csv

No.	Description	Model utilization	Attraction size variable	Travel cost	Dispersion coefficient
1	Access to non-mandatory attractions by SOV in off-peak	Car ownership	Total weighted employment for all purposes	Generalized SOV time including tolls	-0.05
2	Access to non-mandatory attractions by transit in off peak	Car ownership	Total weighted employment for all purposes	Generalized best path walk-to-transit time including fares	-0.05
3	Access to non-mandatory attractions by walk	Car ownership	Total weighted employment for all purposes	SOV off-peak distance (set to 999 if >3)	-1.00
4-6	Access to non-mandatory attractions by all modes except HOV	CDAP	Total weighted employment for all purposes	Off-peak mode choice logsums (SOV skims for ipersons) segmented by 3 car-availability groups	+1.00
7-9	Access to non-mandatory attractions by all modes except SOV	CDAP	Total weighted employment for all purposes	Off-peak mode choice logsums (HOV skims for interaction) segmented by 3 car-availability groups	+1.00
10-12	Access to shopping attractions by all modes except SOV	Joint tour frequency	Weighted employment for shopping	Off-peak mode choice logsum (HOV skims) segmented by 3 HH adult car-availability groups	+1.00
13-15	Access to maintenance attractions by all modes except SOV	Joint tour frequency	Weighted employment for maintenance	Off-peak mode choice logsum (HOV skims) segmented by 3 adult car-availability groups	+1.00

No.	Description	Model utilization	Attraction size variable	Travel cost	Dispersion coefficient
16-18	Access to eating-out attractions by all modes except SOV	Joint tour frequency	Weighted employment for eating out	Off-peak mode choice logsum (HOV skims) segmented by 3 adult HH car-availability groups	+1.00
19-21	Access to visiting attractions by all modes except SOV	Joint tour frequency	Total households	Off-peak mode choice logsum (HOV skims) segmented by 3 adult car-availability groups	+1.00
22-24	Access to discretionary attractions by all modes except SOV	Joint tour frequency	Weighted employment for discretionary	Off-peak mode choice logsum (HOV skims) segmented by 3 adult car-availability groups	+1.00
25-27	Access to escorting attractions by all modes except SOV	Allocated tour frequency	Total households	AM mode choice logsum (HOV skims) segmented by 3 adult car-availability groups	+1.00
28-30	Access to shopping attractions by all modes except HOV	Allocated tour frequency	Weighted employment for shopping	Off-peak mode choice logsum (SOV skims) segmented by 3 adult car-availability groups	+1.00
31-33	Access to maintenance attractions by all modes except HOV	Allocated tour frequency	Weighted employment for maintenance	Off-peak mode choice logsum (SOV skims) segmented by 3 adult car-availability groups	+1.00
34-36	Access to eating-out attractions by all modes except HOV	Individual tour frequency	Weighted employment for eating out	Off-peak mode choice logsum (SOV skims) segmented by 3 car-availability groups	+1.00
36-39	Access to visiting attractions by all modes except HOV	Individual tour frequency	Total households	Off-peak mode choice logsum (SOV skims) segmented by 3 car-availability groups	+1.00
40-41	Access to discretionary attractions by all modes except HOV	Individual tour frequency	Weighted employment for discretionary	Off-peak mode choice logsum (SOV skims) segmented by 3 car-availability groups	+1.00

No.	Description	Model utilization	Attraction size variable	Travel cost	Dispersion coefficient
43-44	Access to at-work attractions by all modes except HOV	Individual sub-tour frequency	Weighted employment for at work	Off-peak mode choice logsum (SOV skims) segmented by adult 2 car-availability groups (0 cars and cars equal or graeter than workers)	+1.00
45	Access to all attractions by all modes of transport in the peak	Work location, CDAP	Total weighted employment for all purposes	Peak mode choice logsums??	??
46	Access to at-work attractions by walk	Individual sub-tour frequency	Weighted employment for at work	SOV off-peak distance (set to 999 if >3)	?
47	Access to all households by all modes of transport in the peak?	?	Total weighted households for all purposes	Generalized best path walk-to-transit time including fares	??

Table 62: Auto Ownership Output –aoresults.csv

Column Name	Description
HHID	Unique Household ID
AO	Auto ownership category 0 = 0 autos 1 = 1 auto 2 = 2 autos 3 = 3 autos 4 = 4+ autos

Table 63: Work and School Location Output – wslocresults_<<iteration>>.csv

Column Name	Description
HHID	Unique Household ID
homeMGRA	Home MAZ zone
income	Household income
perid	Unique person ID
personNum	Person number in household
personType	Person type: 1 = Full-time worker

	2 = Part-time worker 3 = University student 4 = Non-working adult 5 = Non-working senior 6 = Driving age student 7 = Pre-driving student 8 = Pre-school
personAge	Age of person
employmentCategory	Employment category: 1 = Working Full Time worker 2 = Working Part Time workers, university student workers, and driving age student workers 3 = Non-working university students, non-workers, and driving age student worokers 4 = Non-working pre-driving and preschool students
studentCategory	Student category: 1 = Preschool and K-12 2 = University/College 3 = Workers/Non-workers/Preschool
workSegment	Work segment = work district: -1 through 5, and 99999 = non-workers
schoolSegment	School Segment = school district: -1 through 56, 88888 = non-school students
workLocation	Work location MAZ
workLocationDistance	Distance from home to work location
workLocationLogsum	Work location logsum
schoolLocation	School Location MAZ
schoolLocationDistance	Distance from home to school location
schoolLocationLogsum	School location logsum

Table 64: Household Data Output – householddata_<<iteration>>.csv

Column Name	Description
hhid	Unique Household ID
home_mgra	Home MAZ zone
income	Household income
autos	Household Auto ownership: 0 = 0 autos 1 = 1 auto

	2 = 2 autos 3 = 3 autos 4 = 4+ auto
transponder	Boolean for owning a transponder: 0=No 1=Yes
cdap_pattern	Coordinated daily activity pattern. See model specification document for the different patterns
jtf_choice	Joint Tour frequency choice (150 choices with 5 purposes on first tour, 5 purposes on 2nd tour (with possible no 2nd tours for some people), 3 party types (adult only, children only, adult/children combined) on first tour, and 3 party types on second tour

Table 65: Person Data Output – personData_<<iteration>>.csv

Column Name	Description
hhid	Unique Household ID
perid	Unique Person ID
person_num	Person number in household
age	Age of person
gender	Gender of person: f = female, m = male
type	Person type: Child too young for school Full-time worker Non-worker Part-time worker Retired Student of driving age Student of non-driving age University student
value_of_time	Value of time
activity_pattern	Activity pattern: H = Home M = Mandatory N = Non-mandatory
imf_choice	Individual mandatory tour frequency choice: 0 = No mandatory tours 1 = 1 work tour 2 = 2+ work tours 3 = 1 school tour

	4 = 2+ school tours 5 = 1 work, 1 school tour
inmf_choice	Individual non-mandatory tour frequency choice: 0 = NO non-mandatory tours 1 through 197 = number of non-mandatory tours by different non-mandatory tour frequency combinations: see individualnonmandatorytourfrequencyalternatives.csv in the UEC folder
fp_choice	Free Parking choice: -1 = no parking 1 = Free parking 2 = Paid parking 3 = Reimbursed parking
reimb_pct	Percentage of parking reimbursed

Table 66: Individual Tour Data Output – indivtourdata_<<iteration>>.csv

Column Name	Description
hhid	Unique Household ID
perid	Unique Person ID
person_num	Person number in household
person_type	Person type: 1 = full time worker 2 = part time worker 3 = university student 4 = non-working adult 5 = retiree 6 = driving age school children 7 = pre-driving age school children 8 = pre-school children
tour_id	Tour ID for household and person
tour_category	Tour category: AT_WORK INDIVIDUAL_NON_MANDATORY INDIVIDUAL_MANDATORY
tour_purpose	Tour purpose: Discretionary Eating Out Escort Maintenance School Shop

Column Name	Description
	University Visiting Work Work-Based
orig_mgra	Original MAZ
dest_mgra	Destination MAZ
start_period	Start Period: 1=Before 5:00AM, 2=5:00AM-5:30AM, 3 through 39 is every half hour time slots 40 = After 12:00AM
end_period	End Period: 1=Before 5:00AM, 2=5:00AM-5:30AM, 3 through 39 is every half hour time slots 40 = After 12:00AM
tour_mode	Tour Mode: 1 = Drive Alone Free 2 = Drive Alone Pay 3 = Shared Ride 2 General Purpose 4 = Shared Ride 2 HOV 5 = Shared Ride 2 Pay 6 = Shared Ride 3 General Purpose 7 = Shared Ride 3 HOV 8 = Shared Ride 3 Pay 9 = Walk 10 = Bike 11 = Walk to Local 12 = Walk to Express 13 = Walk to BRT 14 = Walk to Light Rail 15 = Walk to Commuter Rail 16 = Park Ride Local 17 = Park Ride Express 18 = Park Ride BRT 19 = Park Ride Light Rail 20 = Park Ride Commuter Rail 21 = Kiss Ride Local 22 = Kiss Ride Express 23 = Kiss Ride BRT 24 = Kiss Ride Light Rail 25 = Kiss Ride Commuter Rail

Column Name	Description
	26 = School Bus
tour_distance	Tour distance
atwork_freq	At-work sub tour frequency: 0 = Not on a work tour 1 = On a work tour with no sub tours 2 = On a work tour with one eat out sub-tour 3 = On a work tour with one business sub tour 4 = On a work tour with one other sub tour 5 = On a work tour with two business sub tours 6 = On a work tour with two other sub tours 7 = On a work tour with a combination of business/eat-out/other sub tours
num_ob_stops	Number of outbound stops (0, 1, 2, 3+)
num_ib_stops	Number of inbound stops (0, 1, 2, 3+)
util_1 through util_26	Utility for each tour mode (see tour mode for codes)

Table 67: Joint Tour Data Output – jointTourData_<<iteration>>.csv

Column Name	Description
hhid	Unique Household ID
tour_id	Tour ID for household and person
tour_category	Tour category: AT_WORK INDIVIDUAL_NON_MANDATORY INDIVIDUAL_MANDATORY
tour_purpose	Tour purpose: Discretionary Eating Out Escort Maintenance School Shop University Visiting Work Work-Based
tour_composition	Tour composition: 1 = Adults only 2 = Children only 3 = Adults and children
tour_participants	Listing of person numbers participating on the joint tour

Column Name	Description
orig_mgra	Original MAZ
dest_mgra	Destination MAZ
start_period	Start Period: 1=Before 5:00AM, 2=5:00AM-5:30AM, 3 through 39 is every half hour time slots 40 = After 12:00AM
end_period	End Period: 1=Before 5:00AM, 2=5:00AM-5:30AM, 3 through 39 is every half hour time slots 40 = After 12:00AM
tour_mode	Tour Mode: 1 = Drive Alone Free 2 = Drive Alone Pay 3 = Shared Ride 2 General Purpose 4 = Shared Ride 2 HOV 5 = Shared Ride 2 Pay 6 = Shared Ride 3 General Purpose 7 = Shared Ride 3 HOV 8 = Shared Ride 3 Pay 9 = Walk 10 = Bike 11 = Walk to Local 12 = Walk to Express 13 = Walk to BRT 14 = Walk to Light Rail 15 = Walk to Commuter Rail 16 = Park Ride Local 17 = Park Ride Express 18 = Park Ride BRT 19 = Park Ride Light Rail 20 = Park Ride Commuter Rail 21 = Kiss Ride Local 22 = Kiss Ride Express 23 = Kiss Ride BRT 24 = Kiss Ride Light Rail 25 = Kiss Ride Commuter Rail 26 = School Bus
tour_distance	Tour distance
num_ob_stops	Number of outbound stops (0, 1, 2, 3+)

Column Name	Description
num_ib_stops	Number of inbound stops (0, 1, 2, 3+)
util_1 through util_26	Utility for each tour mode (see tour mode for codes)
prob_1 through prob_26	Probability for each tour mode (see tour mode for codes)

Table 68: Individual Trip Data Output – indivTripData_<<iteration>>.csv

Column Name	Description
hhid	Unique Household ID
Person_id	Unique Person ID
person_num	Person number in household
tour_id	Tour ID for household and person
stop_id	Stop ID on tour
inbound	Boolean for whether stop is inbound: 0 = No 1 = Yes
tour_purpose	Tour purpose: Discretionary Eating Out Escort Maintenance School Shop University Visiting Work Work-Based
orig_purpose	Origin Trip Purpose: Discretionary Eating Out Escort Home Maintenance School Shop University Visiting Work Work-Based Work related
dest_purpose	Destination Trip Purpose:

Column Name	Description
	Discretionary Eating Out Escort Home Maintenance School Shop University Visiting Work Work-Based Work related
orig_mgra	Original MAZ
dest_mgra	Destination MAZ
parking_mgra	Parking MAZ if applicable
stop_period	Stop Period: 1=Before 5:00AM, 2=5:00AM-5:30AM, 3 through 39 is every half hour time slots 40 = After 12:00AM
trip_mode	Tour Mode: 1 = Drive Alone Free 2 = Drive Alone Pay 3 = Shared Ride 2 General Purpose 4 = Shared Ride 2 HOV 5 = Shared Ride 2 Pay 6 = Shared Ride 3 General Purpose 7 = Shared Ride 3 HOV 8 = Shared Ride 3 Pay 9 = Walk 10 = Bike 11 = Walk to Local 12 = Walk to Express 13 = Walk to BRT 14 = Walk to Light Rail 15 = Walk to Commuter Rail 16 = Park Ride Local 17 = Park Ride Express 18 = Park Ride BRT 19 = Park Ride Light Rail 20 = Park Ride Commuter Rail 21 = Kiss Ride Local

Column Name	Description
	22 = Kiss Ride Express 23 = Kiss Ride BRT 24 = Kiss Ride Light Rail 25 = Kiss Ride Commuter Rail 26 = School Bus
trip_board_tap	Trip boarding Transit Access Point
trip_alight_tap	Trip alighting Transit Access Point
tour_mode	Tour Mode: 1 = Drive Alone Free 2 = Drive Alone Pay 3 = Shared Ride 2 General Purpose 4 = Shared Ride 2 HOV 5 = Shared Ride 2 Pay 6 = Shared Ride 3 General Purpose 7 = Shared Ride 3 HOV 8 = Shared Ride 3 Pay 9 = Walk 10 = Bike 11 = Walk to Local 12 = Walk to Express 13 = Walk to BRT 14 = Walk to Light Rail 15 = Walk to Commuter Rail 16 = Park Ride Local 17 = Park Ride Express 18 = Park Ride BRT 19 = Park Ride Light Rail 20 = Park Ride Commuter Rail 21 = Kiss Ride Local 22 = Kiss Ride Express 23 = Kiss Ride BRT 24 = Kiss Ride Light Rail 25 = Kiss Ride Commuter Rail 26 = School Bus

Table 69: Joint Trip Data Output – jointTripData_<<iteration>>.csv

Column Name	Description
hhid	Unique Household ID
tour_id	Tour ID for household and person
stop_id	Stop ID on tour

Column Name	Description
inbound	Boolean for whether stop is inbound: 0 = No 1 = Yes
tour_purpose	Tour purpose: Discretionary Eating Out Escort Maintenance School Shop University Visiting Work Work-Based
orig_purpose	Origin Trip Purpose: Discretionary Eating Out Escort Home Maintenance School Shop University Visiting Work Work-Based Work related
dest_purpose	Destination Trip Purpose: Discretionary Eating Out Escort Home Maintenance School Shop University Visiting Work Work-Based Work related
orig_mgra	Original MAZ
dest_mgra	Destination MAZ

Column Name	Description
parking_mgra	Parking MAZ if applicable
stop_period	Stop Period: 1=Before 5:00AM, 2=5:00AM-5:30AM, 3 through 39 is every half hour time slots 40 = After 12:00AM
trip_mode	Tour Mode: 1 = Drive Alone Free 2 = Drive Alone Pay 3 = Shared Ride 2 General Purpose 4 = Shared Ride 2 HOV 5 = Shared Ride 2 Pay 6 = Shared Ride 3 General Purpose 7 = Shared Ride 3 HOV 8 = Shared Ride 3 Pay 9 = Walk 10 = Bike 11 = Walk to Local 12 = Walk to Express 13 = Walk to BRT 14 = Walk to Light Rail 15 = Walk to Commuter Rail 16 = Park Ride Local 17 = Park Ride Express 18 = Park Ride BRT 19 = Park Ride Light Rail 20 = Park Ride Commuter Rail 21 = Kiss Ride Local 22 = Kiss Ride Express 23 = Kiss Ride BRT 24 = Kiss Ride Light Rail 25 = Kiss Ride Commuter Rail 26 = School Bus
num_participants	Number of participants on joint trip
trip_board_tap	Trip boarding Transit Access Point
trip_alight_tap	Trip alighting Transit Access Point
tour_mode	Tour Mode: 1 = Drive Alone Free 2 = Drive Alone Pay 3 = Shared Ride 2 General Purpose 4 = Shared Ride 2 HOV

Column Name	Description
	5 = Shared Ride 2 Pay 6 = Shared Ride 3 General Purpose 7 = Shared Ride 3 HOV 8 = Shared Ride 3 Pay 9 = Walk 10 = Bike 11 = Walk to Local 12 = Walk to Express 13 = Walk to BRT 14 = Walk to Light Rail 15 = Walk to Commuter Rail 16 = Park Ride Local 17 = Park Ride Express 18 = Park Ride BRT 19 = Park Ride Light Rail 20 = Park Ride Commuter Rail 21 = Kiss Ride Local 22 = Kiss Ride Express 23 = Kiss Ride BRT 24 = Kiss Ride Light Rail 25 = Kiss Ride Commuter Rail 26 = School Bus

Table 70: Visitor Tour Data – visitorTours.csv

Column Name	Description
id	Unique ID of Visitor tour
segment	Visitor Segment: 0 = Business 1 = Personal
purpose	Visitor Tour Purpose: 0 = Work 1 = Recreation 2 = Dining
autoavailable	Boolean for Auto available: 0 = No 1 = Yes
partysize	Party size (1 through 10+)
income	Visitor income: 0 = < $30K 1 = $30K-$60K

Column Name	Description
	2 = $60K-100K 3 = $100K-$150K 4 = $150K+
departtime	Depart time: 1=Before 5:00AM, 2=5:00AM-5:30AM, 3 through 39 is every half hour time slots 40 = After 12:00AM
arrivetime	Arrive time: 1=Before 5:00AM, 2=5:00AM-5:30AM, 3 through 39 is every half hour time slots 40 = After 12:00AM
originmgra	Origin MGRA of tour
destinationmgra	Destination MGRA of tour
tourmode	Tour Mode: 1 = Drive Alone Free 2 = Drive Alone Pay 3 = Shared Ride 2 General Purpose 4 = Shared Ride 2 HOV 5 = Shared Ride 2 Pay 6 = Shared Ride 3 General Purpose 7 = Shared Ride 3 HOV 8 = Shared Ride 3 Pay 9 = Walk 10 = Bike 11 = Walk to Local 12 = Walk to Express 13 = Walk to BRT 14 = Walk to Light Rail 15 = Walk to Commuter Rail 16 = Park Ride Local 17 = Park Ride Express 18 = Park Ride BRT 19 = Park Ride Light Rail 20 = Park Ride Commuter Rail 21 = Kiss Ride Local 22 = Kiss Ride Express 23 = Kiss Ride BRT 24 = Kiss Ride Light Rail 25 = Kiss Ride Commuter Rail

Column Name	Description
	26 = School Bus 27 = Taxi
outboundstops	Number of outbound stops
inboundstops	Number of inbound stops

Table 71: Visitor Trip Data – visitorTrips.csv

Column Name	Description
tourid	Unique ID of Visitor tour
tripid	Unique ID of Visitor trip
originpurp	Origin Trip Purpose: -1 = Unknown 0 = Work 1 = Recreation 2 = Dining
destpurp	Destination Trip Purpose: -1 = Unknown 0 = Work 1 = Recreation 2 = Dining
originmgra	Origin MGRA of trip
Destinationmgra	Destination MGRA of trip
origintaz	Origin TAZ of trip
destinationtaz	Destination TAZ of trip
inbound	Boolean for whether trip is inbound: False True
originIsTourDestination	Boolean for whether trip origin is tour destination: False True
destinationIsTourDestination	Boolean for whether trip destination is tour destination: False True
period	Period: 1=Before 5:00AM, 2=5:00AM-5:30AM, 3 through 39 is every half hour time slots

Column Name	Description
	40 = After 12:00AM
tripMode	Trip Mode: 1 = Drive Alone Free 2 = Drive Alone Pay 3 = Shared Ride 2 General Purpose 4 = Shared Ride 2 HOV 5 = Shared Ride 2 Pay 6 = Shared Ride 3 General Purpose 7 = Shared Ride 3 HOV 8 = Shared Ride 3 Pay 9 = Walk 10 = Bike 11 = Walk to Local 12 = Walk to Express 13 = Walk to BRT 14 = Walk to Light Rail 15 = Walk to Commuter Rail 16 = Park Ride Local 17 = Park Ride Express 18 = Park Ride BRT 19 = Park Ride Light Rail 20 = Park Ride Commuter Rail 21 = Kiss Ride Local 22 = Kiss Ride Express 23 = Kiss Ride BRT 24 = Kiss Ride Light Rail 25 = Kiss Ride Commuter Rail 26 = School Bus 27 = Taxi
boardingTap	Boarding Transit Access Point
alightingTap	Alighting Transit Access Point

Table 72: TAZ Demand (Resident + Visitor) Matrices – TAZ_Demand_<<time period>>.mat

Table Name	Description
SOV_GP_<<time period>>	Drive Alone Free for <<time period>>
SOV_PAY_<<time period>>	Drive Alone Toll for <<time period>>
SR2_GP_<<time period>>	Shared Ride 2 Free for <<time period>>
SR2_HOV_<<time period>>	Shared Ride 2 HOV for <<time period>>
SR2_PAY_<<time period>>	Shared Ride 2 Toll for <<time period>>

SR3_GP_<<time period>>	Shared Ride 3 Free for <<time period>>
SR3_HOV_<<time period>>	Shared Ride 3 HOV for <<time period>>
SR3_PAY_<<time period>>	Shared Ride 3 Toll for <<time period>>
BIKE_<<time period>>	Bike for <<time period>>
WALK_<<time period>>	Walk for <<time period>>
SCHLBUS_<<time period>>	School bus for <<time period>>

Table 73: TAP Demand (Resident + Visitor) Matrices – TAP_Demand_<<time period>>.mat

Table Name	Description
WLK_LOC_<<time period>>	Walk to Local for <<time period>>
WLK_EXP_<<time period>>	Walk to Express for <<time period>>
WLK_BRT_<<time period>>	Walk to BRT for <<time period>>
WLK_LRT_<<time period>>	Walk to Light Rail for <<time period>>
WLK_CMR_<<time period>>	Walk to Commuter Rail for <<time period>>
PNR_LOC_<<time period>>	Park Ride to Local for <<time period>>
PNR _EXP_<<time period>>	Park Ride to Express for <<time period>>
PNR _BRT_<<time period>>	Park Ride to BRT for <<time period>>
PNR _LRT_<<time period>>	Park Ride to Light Rail for <<time period>>
PNR _CMR_<<time period>>	Park Ride to Commuter Rail for <<time period>>
KNR_LOC_<<time period>>	Kiss Ride to Local for <<time period>>
KNR _EXP_<<time period>>	Kiss Ride to Express for <<time period>>
KNR _BRT_<<time period>>	Kiss Ride to BRT for <<time period>>
KNR _LRT_<<time period>>	Kiss Ride to Light Rail for <<time period>>
KNR _CMR_<<time period>>	Kiss Ride to Commuter Rail for <<time period>>

Running the Population Synthesizer

The hhFile.csv and personFile.csv input files used by the model are created by the population synthesizer. This section of the document describes how to create new synthetic population files based on revised land use data. See the SERPM7_POPSYN_MEMO.docx for more information on how to setup and configure the population synthesizer, including the required SQL Server database.

To run the population synthesizer, run the **runAll_ popsyn.bat** batch file in ctramp\popSyn. The following variables need to be set in the batch file before running the script:

1) SERVERNAME = server location, i.e.: "localhost\SQLEXPRESS"

2) DATABASENAME = database name on server, i.e.: “popSyn”
3) ARCPATH = location of the folder containing the runAll_ popsyn.bat file, i.e.: “C:\popsyn”
4) CSVPATH_CONTROLS = location of the controls.csv file, i.e.: “C:\popsyn\inputs\controls.csv”
5) CSVPATH_MAZ = location of the maz.csv file, i.e.: “C:\popsyn\inputs\maz_data.csv”
6) CSVPATH_TAZ = location of the taz.csv file, i.e.: “C:\popsyn\inputs\taz_data.csv”
7) USER= if SQL authentication is used, set the username to the SQL account that has bulk load and bulk insert permissions to the data base, i.e. “popsyn”
8) PASSWORD= the password of the SQL user, i.e. “p0ps1n”

The batch file executes the following tasks:

1) Loads the following input tables into the database via the **1_importCSV.sql** script:
    a. controls.csv – Control category values; this file does not change
    b. maz_data.csv – MAZ land use data
    c. taz_data.csv – TAZ land use data
2) Creates the targets table for use in PopSyn II based on the input MAZ and TAZ data via the **2_createTargets.sql** script.
3) Runs the Java PopSyn II programs via the command line.
4) Creates the output hhFile.csv and personFile.csv files for input into CT-RAMP via the **3_createSynHHPerFiles.sql** script. These files can then be copied into the inputs folder.

Integration with Summit for User Benefits Analysis

With an activity-based model, user benefits analysis is done at the tour level as opposed to the trip level, since mode shifts at the tour level are of primary significance. In addition, Summit evaluates the changes in mode utility for each individual tour as opposed for aggregate OD pairs.

To run the model with Summit, do the following:

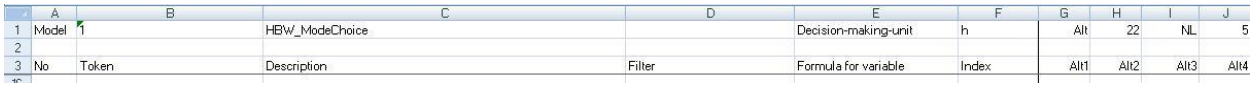
1) Base Scenario
    1. Run the base alternative and output the tour output file (indivTourData.csv and jointTourData.csv). The tour output files include utilities and probabilities for all modes, not just the selected mode.
    2. Run the SandagSummitFile program included with CT-RAMP to read the tour output file and create the base scenario Summit input file.
2) Build Scenario
    1. Leave the HH manager and matrix manager open to keep all the existing choices in memory.
    2. Replace the base transit skims with transit skims calculated from the build network.
    3. Run only tour mode choice and output the tour output file with utilities and probabilities.
    4. Run the SandagSummitFile program included with CT-RAMP to read the tour output file and create the build scenario Summit input file.
3) Run Summit and specify the base and build Summit input files.

Utility Expression Calculator Files

The ABM model is largely a sequence of discrete choice models, which use the Utility Expression Calculator (UEC) Java package to both locate input variables and specify utility equations that describe each discrete choice. The input variables and specifications are defined and stored in a Microsoft Excel workbook. The use of Excel improves the flexibility and transparency of the model system since utility coefficients, model structures, etc, can be edited via Excel rather than a text file, XML file, or source code.

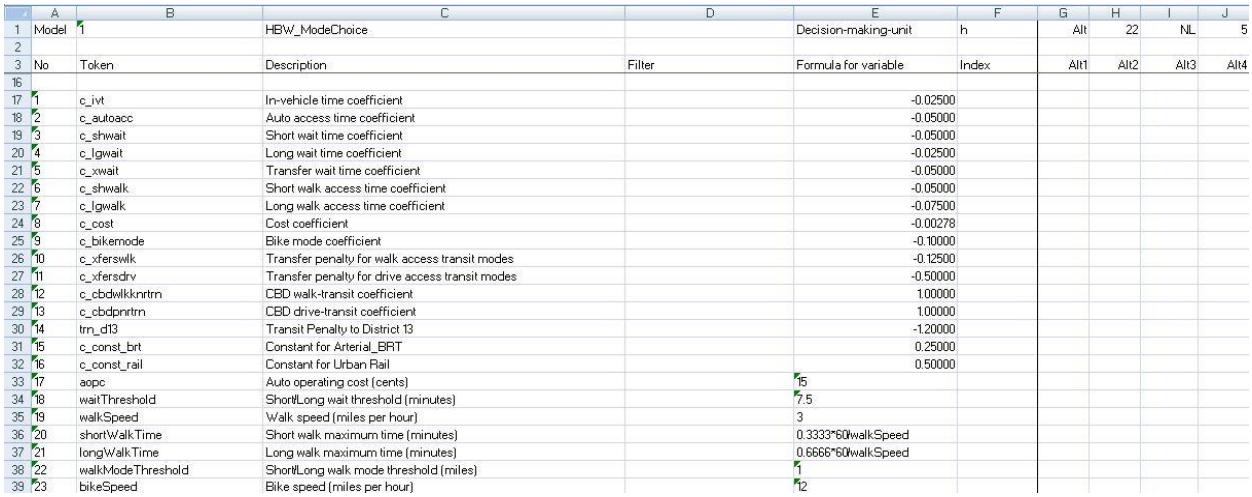
Each UEC consists of at least two worksheets. The first worksheet is the data sheet, which defines the input files used in the utility expressions, including zonal (vector) data and level-of-service "skims" (matrix data). The second, third, fourth, etc page specifies one or more multinomial or nested logit models via a unique UEC utility sheet. The sheet consists of three sections, as follows:

1) The nesting structure of the logit model -- if omitted, a multinomial structure is assumed



	A	B	C	D	E	F	G	H	I	J
1	Model	1	HBW_ModeChoice		Decision-making-unit	h	Alt	22	NL	5
2										
3	No	Token	Description	Filter	Formula for variable	Index	Alt1	Alt2	Alt3	Alt4

2) Variable names, or tokens, are defined for use in subsequent (moving down rows) utility equations



	A	B	C	D	E	F	G	H	I	J
1	Model	1	HBW_ModeChoice		Decision-making-unit	h	Alt	22	NL	5
2										
3	No	Token	Description	Filter	Formula for variable	Index	Alt1	Alt2	Alt3	Alt4
16										
17	1	c_ivt	In-vehicle time coefficient		-0.02500					
18	2	c_autoacc	Auto access time coefficient		-0.05000					
19	3	c_shwait	Short wait time coefficient		-0.05000					
20	4	c_lgwait	Long wait time coefficient		-0.02500					
21	5	c_xwait	Transfer wait time coefficient		-0.05000					
22	6	c_shwalk	Short walk access time coefficient		-0.05000					
23	7	c_lgwalk	Long walk access time coefficient		-0.07500					
24	8	c_cost	Cost coefficient		-0.00278					
25	9	c_bikemode	Bike mode coefficient		-0.10000					
26	10	c_xferswlk	Transfer penalty for walk access transit modes		-0.12500					
27	11	c_xfersdrv	Transfer penalty for drive access transit modes		-0.50000					
28	12	c_cbdwlkknrtrn	CBD walk-transit coefficient		1.00000					
29	13	c_cbdpnrtrn	CBD drive-transit coefficient		1.00000					
30	14	trn_d13	Transit Penalty to District 13		-1.20000					
31	15	c_const_brt	Constant for Arterial_BRT		0.25000					
32	16	c_const_rail	Constant for Urban Rail		0.50000					
33	17	aopc	Auto operating cost (cents)		15					
34	18	waitThreshold	Short/Long wait threshold (minutes)		7.5					
35	19	walkSpeed	Walk speed (miles per hour)		3					
36	20	shortWalkTime	Short walk maximum time (minutes)		0.3333*60/walkSpeed					
37	21	longWalkTime	Long walk maximum time (minutes)		0.6666*60/walkSpeed					
38	22	walkModeThreshold	Short/Long walk mode threshold (miles)		1					
39	23	bikeSpeed	Bike speed (miles per hour)		12					

3) Utility terms, typically a variable and a coefficient for each of the logit model's alternatives.

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	Model	1	HBW_ModeChoice		Decision-making-unit	h	Alt	22	NL	5			
2													
3	No	Token	Description	Filter	Formula for variable	Index	Alt1	Alt2	Alt3	Alt4	Alt5	Alt6	Alt7
81	65		Drive alone is not available		if(DA_Available==0,1,0)		-999.00						
82	66		Drive Alone In-vehicle time	DA_Available	c_ivt*LOV_TIME	od	1.00						
83	67		Drive Alone Cost	DA_Available	c_cost*(LOV_DIST *aopc+dailyParkingCost/2.0)	od,z	1.00						
84	68		Auto shared ride 2 GP path is not available		if(SR2GP_Available==0,1,0)			-999.00					
85	69		Shared Ride 2 In-vehicle time	SR2GP_Available	c_ivt*LOV_TIME	od		1.00					
86	70		Shared Ride 2 Cost	SR2GP_Available	c_cost*(LOV_DIST *aopc+dailyParkingCost/2.0*0.5)	od,z		1.00					
87	71		Auto shared ride 2 HOV path is not available		if(SR2HOV_Available==0,1,0)				-999.00				
88	72		Shared Ride 2 In-vehicle time	SR2HOV_Available	c_ivt*HOV_TIME	od			1.00				
89	73		Shared Ride 2 Cost	SR2HOV_Available	c_cost*(HOV_DIST *aopc+dailyParkingCost/2.0*0.5)	od,z			1.00				
90	74		Auto shared ride 2+ GP path is not available		if(SR3GP_Available==0,1,0)					-999.00			
91	75		Shared Ride 2 In-vehicle time	SR3GP_Available	c_ivt*LOV_TIME	od				1.00			
92	76		Shared Ride 2 Cost	SR3GP_Available	c_cost*(LOV_DIST *aopc+dailyParkingCost/2.0*0.33)	od				1.00			
93	77		Auto shared ride 2+ HOV path is not available		if(SR3HOV_Available==0,1,0)						-999.00		
94	78		Shared Ride 2 In-vehicle time	SR3HOV_Available	c_ivt*HOV_TIME	od					1.00		
95	79		Shared Ride 2 Cost	SR3HOV_Available	c_cost*(HOV_DIST *aopc+dailyParkingCost/2.0*0.33)	od,z					1.00		
101	85		Walk-Local is not available		if(WLK_LOC_Available==0,1,0)							-999.00	
102	86		Walk-Local in-vehicle time	WLK_LOC_Available	c_ivt*WLK_LOC_IVT	od						1.00	
103	87		Walk-Local short first wait	WLK_LOC_Available	c_shwait*min(WLK_LOC_IWT,waitThreshold)	od						1.00	
104	88		Walk-Local long first wait	WLK_LOC_Available	c_lgwait*max(WLK_LOC_IWT-waitThreshold,0.0)	od						1.00	
105	89		Walk-Local transfer wait	WLK_LOC_Available	c_xwait*WLK_LOC_XWT	od						1.00	
106	90		Walk-Local transfer penalty	WLK_LOC_Available	c_xferswlk*WLK_LOC_XFR	od						1.00	
107	91		Walk-Local fare	WLK_LOC_Available	c_cost*locFare							1.00	
108	92		Walk-Local fare, All IVT on Streetcar, Add difference	WLK_LOC_Available*IF(WLK_LOC_	c_cost*(streetcarFare-locFare)							1.00	
109	93		Walk-Local short access walk	WLK_LOC_Available*(shortToShort+	c_shwalk*min(WLK_LOC_ACC,shortWalkTime)	od						1.00	
110	94		Walk-Local long access walk	WLK_LOC_Available*(longToShort+	c_lgwalk*max(shortWalkTime,min(WLK_LOC_ACC,lor	od						1.00	
111	95		Walk-Local short egress walk	WLK_LOC_Available*(shortToShort+	c_shwalk*min(WLK_LOC_EGR,shortWalkTime)	od						1.00	
112	96		Walk-Local long egress walk	WLK_LOC_Available*(longToLong+s	c_lgwalk*max(shortWalkTime,min(WLK_LOC_EGR,lor	od						1.00	
113	97		Walk-Local auxiliary walk	WLK_LOC_Available	c_shwalk*WLK_LOC_AUX	od						1.00	
114	98		Walk-Local CBD	WLK_LOC_Available	c_cbdwlknrtrn*cbd	z						1.00	
115	99		Walk-Local-Dist13	WLK_LOC_Available	trn_d13*iif(PRODUCTION_DISTRICT==13,1,0)	od						1.00	

When the UEC is solved, it returns an array of doubles dimensioned to the number of alternatives specified in the utility sheet. The array contains the sum-product of each of the formulas and coefficients for each alternative, which is the utility for each alternative. This array can then be used with a logit model object to compute alternative probabilities and then simulate choices.

Model Logging/Trace Results

CT-RAMP writes a series of log files to the logFiles folder during a model run. These log files are extremely useful for understanding the model as well as for debugging a model run. The main log file is the event.log file and it is the starting point for reviewing the logs. In addition to event.log, the event_hh.log logs the household data manager work, event_mtx.log logs the matrix data manager work, and event-node<X>.log logs the work on each remote node. There are also model component specific log files for each node such as event-node1-ao.log for the auto ownership model being run on remote machine node 1. Finally, the cmd run log files are placed in the /cube folder as "abm.log" and "visitors.log" for the Resident and Visitor models, respectively.

In addition to writing log files, CT-RAMP can trace model calculations for a user specified household. To trace results for household 2949465 for example, set Debug.Trace.HouseholdIdList = 2949465 in the serpm_abm.properties file. This tells CT-RAMP to write out all calculations for every person in household 2949465, including the results of the UEC calculations for each model. Figure 8 below contains a sample of the household trace results. The first screenshot shows a sample of the trace results for the household and person number 1 in the household. As is shown below, key attributes of the household and person are traced such as the household income, size, number of workers, home TAZ, person age, person employment category (worker, retired, etc). As is shown in the second screenshot, which is the auto ownership model trace for household 2949465, the value of the coefficient for each alternative times the value of each expression is logged. The 111 expressions and five alternatives (0, 1, 2, 3, 4+ autos) traced correspond exactly with the expressions and alternatives in the auto ownership UEC file, which makes tracing and debugging easier.

Figure 8: Household Trace Results

```
23-Jun-2011 15:41:44:964, INFO, HHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHH
23-Jun-2011 15:41:44:965, INFO, Pre Work/School Location Choice HHId=2949465 Object
23-Jun-2011 15:41:44:966, INFO, HHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHH
23-Jun-2011 15:41:44:966, INFO,      hhId:                                                      2949465
23-Jun-2011 15:41:44:967, INFO,      debugChoiceModels:                                            True
23-Jun-2011 15:41:44:968, INFO,      hhIncomeInDollars:                                           89000
23-Jun-2011 15:41:44:969, INFO,      hhSize:                                                          1
23-Jun-2011 15:41:44:970, INFO,      hhType:                                                          6
23-Jun-2011 15:41:44:970, INFO,      hhWorkers:                                                       1
23-Jun-2011 15:41:44:971, INFO,      homeTaz:                                                         4
23-Jun-2011 15:41:44:972, INFO,      homeWalkSubzone:                                                 1
23-Jun-2011 15:41:44:972, INFO,      aoModelAutos:                                                    0
23-Jun-2011 15:41:44:973, INFO,      freeParkingAvailable:                                            0
23-Jun-2011 15:41:44:974, INFO,      cdapModelPattern:                                             null
23-Jun-2011 15:41:44:974, INFO,      imtfModelPattern:                                                0
23-Jun-2011 15:41:44:975, INFO,      jtfModelPattern:                                              null
23-Jun-2011 15:41:44:976, INFO,      randomCount:                                                     2
23-Jun-2011 15:41:44:990, INFO, Joint Tours[]:
23-Jun-2011 15:41:44:990, INFO, HHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHH
23-Jun-2011 15:41:44:990, INFO,
23-Jun-2011 15:41:44:991, INFO,
23-Jun-2011 15:41:44:993, INFO, PPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPP
23-Jun-2011 15:41:44:994, INFO, HH=2949465, PersonNum=1, PersonType=Full-time worker
23-Jun-2011 15:41:44:994, INFO, PPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPP
23-Jun-2011 15:41:44:995, INFO,      persNum:                                                                                                         1
23-Jun-2011 15:41:44:995, INFO,      persId:                                                                                                    8887251
23-Jun-2011 15:41:44:996, INFO,      persAge:                                                                                                        58
23-Jun-2011 15:41:44:996, INFO,      persGender:                                                                                                      2
23-Jun-2011 15:41:44:997, INFO,      persEmploymentCategory:                                                                                          1
23-Jun-2011 15:41:44:998, INFO,      persStudentCategory:                                                                                             3
23-Jun-2011 15:41:44:998, INFO,      personType:                                                                                                      1
23-Jun-2011 15:41:44:999, INFO,      workLoc:                                                                                                         0
23-Jun-2011 15:41:44:999, INFO,      schoolLoc:                                                                                                       0
23-Jun-2011 15:41:45:008, INFO, PPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPP
```

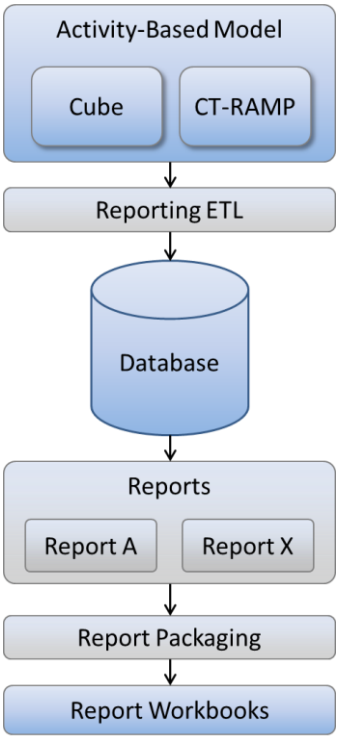
```

23-Jun-2011 15:42:20:831, INFO, *******************************************************************************************
23-Jun-2011 15:42:20:831, INFO, Utility Expressions for Household Auto Ownership Choice, HH_2949465
23-Jun-2011 15:42:20:832, INFO, *******************************************************************************************
23-Jun-2011 15:42:20:832, INFO, For each model expression, 'coeff * expressionValue' pairs are listed for each available alternative.  At the end, total utility is listed.
23-Jun-2011 15:42:20:833, INFO, The last line shows total utility for each available alternative.
23-Jun-2011 15:42:20:833, INFO, Exp                               1                              2                              3                              4                              5
23-Jun-2011 15:42:20:834, INFO, ---------------------------------------------------------------------------------------------------------------------------------------------------------------
23-Jun-2011 15:42:20:835, INFO, 1            0.00000 *   1.29179e+01        0.00000 *   1.29179e+01        0.00000 *   1.29179e+01        0.00000 *   1.29179e+01        0.00000 *   1.29179e+01
23-Jun-2011 15:42:20:835, INFO, 2            0.00000 *   1.19723e+01        0.00000 *   1.19723e+01        0.00000 *   1.19723e+01        0.00000 *   1.19723e+01        0.00000 *   1.19723e+01
23-Jun-2011 15:42:20:836, INFO, 3            0.00000 *   0.00000e+00        0.00000 *   0.00000e+00        0.00000 *   0.00000e+00        0.00000 *   0.00000e+00        0.00000 *   0.00000e+00
23-Jun-2011 15:42:20:837, INFO, 4            0.00000 *   1.00000e+00        0.00000 *   1.00000e+00        0.00000 *   1.00000e+00        0.00000 *   1.00000e+00        0.00000 *   1.00000e+00
23-Jun-2011 15:42:20:838, INFO, 5            0.00000 *   1.00000e+00        0.00000 *   1.00000e+00        0.00000 *   1.00000e+00        0.00000 *   1.00000e+00        0.00000 *   1.00000e+00
23-Jun-2011 15:42:20:838, INFO, 6            0.00000 *   1.00000e+00        0.00000 *   1.00000e+00        0.00000 *   1.00000e+00        0.00000 *   1.00000e+00        0.00000 *   1.00000e+00
23-Jun-2011 15:42:20:839, INFO, 7            0.00000 *   0.00000e+00        0.00000 *   0.00000e+00        0.00000 *   0.00000e+00        0.00000 *   0.00000e+00        0.00000 *   0.00000e+00
23-Jun-2011 15:42:20:840, INFO, 8            0.00000 *   0.00000e+00        0.00000 *   0.00000e+00        0.00000 *   0.00000e+00        0.00000 *   0.00000e+00        0.00000 *   0.00000e+00
23-Jun-2011 15:42:20:840, INFO, 9            0.00000 *   0.00000e+00        0.00000 *   0.00000e+00        0.00000 *   0.00000e+00        0.00000 *   0.00000e+00        0.00000 *   0.00000e+00
.....
23-Jun-2011 15:42:20:858, INFO, 33          -3.93320 *   0.00000e+00       -0.24080 *   0.00000e+00       -0.10640 *   0.00000e+00       -0.10640 *   0.00000e+00        0.00000 *   0.00000e+00
23-Jun-2011 15:42:20:859, INFO, 34           1.98370 *   0.00000e+00        0.44120 *   0.00000e+00        0.00000 *   0.00000e+00       -0.06940 *   0.00000e+00       -0.12800 *   0.00000e+00
23-Jun-2011 15:42:20:860, INFO, 35           0.62250 *   0.00000e+00        0.22150 *   0.00000e+00        0.00000 *   0.00000e+00       -0.06940 *   0.00000e+00       -0.12800 *   0.00000e+00
23-Jun-2011 15:42:20:861, INFO, 36          -1.23000 *   0.00000e+00       -0.09910 *   0.00000e+00        0.00000 *   0.00000e+00        0.10180 *   0.00000e+00        0.13420 *   0.00000e+00
23-Jun-2011 15:42:20:861, INFO, 37          -1.57870 *   0.00000e+00       -0.36710 *   0.00000e+00        0.00000 *   0.00000e+00        0.17920 *   0.00000e+00        0.30100 *   0.00000e+00
23-Jun-2011 15:42:20:862, INFO, 38          -0.18640 *   0.00000e+00        0.00000 *   0.00000e+00        0.00000 *   0.00000e+00        0.00000 *   0.00000e+00        0.00000 *   0.00000e+00
23-Jun-2011 15:42:20:863, INFO, 39           0.00000 *   0.00000e+00        0.00000 *   0.00000e+00        0.00000 *   0.00000e+00        0.00000 *   0.00000e+00        0.00000 *   0.00000e+00
.....
23-Jun-2011 15:42:20:911, INFO, 105          0.00000 *   0.00000e+00        0.00000 *   0.00000e+00        0.00000 *   0.00000e+00        0.00000 *   0.00000e+00        0.01850 *   0.00000e+00
23-Jun-2011 15:42:20:912, INFO, 106          0.19100 *   1.00000e+00        0.28120 *   1.00000e+00       -0.24170 *   1.00000e+00       -1.88700 *   1.00000e+00       -1.68620 *   1.00000e+00
23-Jun-2011 15:42:20:912, INFO, 107          0.12600 *   0.00000e+00        0.20710 *   0.00000e+00        0.13200 *   0.00000e+00       -0.65320 *   0.00000e+00       -1.29510 *   0.00000e+00
23-Jun-2011 15:42:20:913, INFO, 108         -0.41000 *   0.00000e+00       -0.00300 *   0.00000e+00        0.34100 *   0.00000e+00       -0.24100 *   0.00000e+00       -0.77100 *   0.00000e+00
23-Jun-2011 15:42:20:913, INFO, 109         -0.46900 *   0.00000e+00       -0.13400 *   0.00000e+00        0.36800 *   0.00000e+00       -0.21400 *   0.00000e+00       -0.55900 *   0.00000e+00
23-Jun-2011 15:42:20:914, INFO, 110          0.34300 *   0.00000e+00       -0.18800 *   0.00000e+00        0.31600 *   0.00000e+00       -0.02400 *   0.00000e+00       -0.70000 *   0.00000e+00
23-Jun-2011 15:42:20:915, INFO, 111          0.11100 *   0.00000e+00       -0.19700 *   0.00000e+00        0.38200 *   0.00000e+00       -0.07600 *   0.00000e+00       -0.93100 *   0.00000e+00
23-Jun-2011 15:42:20:916, INFO, ---------------------------------------------------------------------------------------------------------------------------------------------------------------
23-Jun-2011 15:42:20:917, INFO, Alt Utility                -4.56485e-01                   2.81200e-01                  -1.16847e+00                  -3.38567e+00                  -3.56937e+00
```

CHAPTER 3 SERPM ABM Reporting Database

Overview

The purpose of this memo is to describe the activity-based modeling reporting system that has been developed for SERPM. The system has four primary components:

1) Reporting ETL (extract, transform, and load) program to load all Cube and CT-RAMP Java model outputs into the database. This has been written in Cube scripts and Java.
2) Database to store all data used in reporting, including all networks, select network skims, household, person, tour, and trip data, zonal data, etc. This was implemented with SQL Server.
3) Report scripts to generate reports. These will be written in T-SQL. Reporting scripts can be revised and additional reports added as additional scripts. Additional scripts should be listed in the **serpm_reports.properties** file.
4) Report packaging program to generate Excel workbooks with one report per worksheet. This has been written in Java.



Each of the four system components is described below. A set of initial reports is described in the Reports section as well.

Reporting ETL Program

The ETL program extracts, transforms, and loads the model data into the database. This involves creating additional CT-RAMP output fields and tables for reporting as well as actually loading the model outputs into the database. A key additional output are the **TRIP** and **TOUR** tables that contain all micro-simulated trip and tour data (individual and joint - exploded by individual) and travel impedance attributes such as travel time, distance, and cost, all joined to the personal attributes of the individual making the trip. These tables make queries easier and faster since they usually contain most of the required information.

Database

The goal of the database is to store all relevant model inputs and outputs so the user does not need to interact with Cube or CT-RAMP to generate model reports. The user may interact with Cube and CT-RAMP directly if desired, but it is not required. This setup allows a consistent reporting framework to be used for analysis across model runs. As such, the database must store all relevant model inputs and outputs for each model run.

Each model run alternative will be a separate scenario in the database, and each scenario will be stored as a separate schema (or set of tables for a given scenario) in the database. The name of the schema will be used as a prefix for all the tables that belong to that scenario. If the user selects a name that does not begin with a letter character, an additional "S_" prefix will be added to the schema. For example, the full name of the TRIP table for a model run called "2010R" (year=2010, alternative=R) would be S_2010R.TRIP. If the user named the run "R2010", the TRIP table would be

R2010.TRIP. This is done to ensure the schema names are compatible with MS SQL conventions. Regardless of the schema name, it would be stored in the ABM database. The scenario name can contain the study name as well as the alternative name. For example, the 2030STUDYALTA scenario would contain all tables for the 2030 study, alternative A model run (the schema name would be "S_2030STUDYALTA"). When querying a table, the user must specify the schema (or scenario) name as well, unless the default schema is set for the user.

The following is the database schema description. Fields and tables shown in **bold** denote additional outputs created at reports stage.

Table 74 – SERPM Reporting Database Tables

Table	Table Name	Description	Object	Primary Key(s)
Inputs				
Synthetic Households	SYNHH	Input synthesized households from the PopSyn. This input file is copied to the database to ensure the actual file used to run the model is stored.	Household	HHID
Synthetic Persons	SYNPERSON	Input synthesized persons from the PopSyn. This input file is copied to the database to ensure the actual file used to run the model is stored.	Person	PERID
TAZ Data	TAZ	Input TAZ data such as district code, parking cost, EJ identifier, etc	TAZ	TAZ
MGRA Data	MGRA	Input MGRA data such as employment, etc	MGRA	MGRA
TAP Data	TAP	Input TAP data	TAP	TAP
MGRA to TAP Data	MGRATOTAP	MGRA to TAP distances, etc. This file is read by Java and formatted into a CSV for loading into the database.	MGRA-TAP	MGRA, TAP
MGRA to Stop Data	MGRATOSTOP	MGRA to all transit stops distances, etc. This file is read by Java and formatted into a CSV for loading	MGRA-STOP	MGRA, STOP

Table	Table Name	Description	Object	Primary Key(s)
		into the database.		
MGRA to MGRA Data	MGRATOMGRA	MGRA to MGRA distances, etc. This file is read by Java and formatted into a CSV for loading into the database.	MGRA-MGRA	ORIG_MGRA, DEST_MGRA
TAZ to TAP Data	TAZTOTAP	TAZ to TAP distances, etc. This file is read by Java and formatted into a CSV for loading into the database.	TAZ-TAP	TAZ,TAP
Microsimulated Outputs				
Accessibilities	ACCESSIBILITIES	Model results for accessibilities	MGRA	MGRA
Household Data	HHDATA	Model results for household level choice models	Household	HH_ID
Person Data	PERSONDATA	Model results for person level choice models	Person	PERSON_ID
Work and School Location	WSLOCATION	Model results for the usual work and school location choice models	Person	PERSONID
Individual Tours	INDIVTOUR	Modeled individual tours	Tour	HHID, PERID, TOUR_CATEGORY, TOUR_ID, TOUR_PURPOSE
Joint Tours	JOINTTOUR	Modeled joint tours	Tour	HHID, TOUR_CATEGORY, TOUR_ID, TOUR_PURPOSE
Individual Trips	INDIVTRIP	Modeled individual trips	Trip	HHID, PERID, TOUR_CATEGORY, TOUR_ID, TOUR_PURPOSE, INBOUND, STOP_ID
Joint Trips	JOINTTRIP	Modeled joint trips	Trip	HHID, TOUR_CATEGORY, TOUR_ID, TOUR_PURPOSE, INBOUND, STOP_ID
CBD Vehicle Trips	CBDVEHICLES	Number of vehicle	MGRA	MGRA

Table	Table Name	Description	Object	Primary Key(s)
		trips to CBD by time period		
Reporting Tables				
All microsimulated trips	TRIP	All microsimulated trips (individual, joint – exploded to individuals participating in joint trip) and trip impedance attributes.	Trip	RECID
District/County Definitions	DISTRICTDEFINITIONS	Mapping of TAZs to counties and districts. Defined in the District.Definitions variable within the serpm_abm.properties file	TAZ	TAZ2010
PECAS Occupations	PECASCODES	Mapping of PECAS codes to occupations. Defined in the PopulationSynthesizer.OccupCodes variable within the serpm_abm.properties file	Code	OCCCEN1

Table 75 – SERPM Reporting Database Fields

Table	Field	Type	Description
SYNHH	HHID	Int	ABM HH ID. See households.csv input file.
	HOUSEHOLD_SERIAL_NO	Int	PUMS HH ID
	TAZ	Int	Home TAZ
	MGRA	Int	Home MGRA
	HINCCAT1	Int	HH Income category
	HINC	Int	Household income
	HWORKERS	Int	Number of workers
	VEH	Int	Number of vehicles (from PUMS)
	PERSONS	Int	Number of persons
	HHT	Int	Household type
	BLDGSZ	Int	Building size

Table	Field	Type	Description
	UNITTYPE	Int	Unit type
SYNPERSON	HHID	Int	ABM HH ID. See persons.csv input file.
	PERID	Int	ABM person ID
	HOUSEHOLD_SERIAL_NO	Int	PUMS HH ID
	PNUM	Int	Person number in HH
	AGE	Int	Age
	SEX	Int	Sex
	MILITARY	Int	Military Service code
	PEMPLOY	Int	PopSyn employment type
	PSTUDENT	Int	PopSyn student type
	PTYPE	Int	PopSyn person type
	EDUC	Int	Educational attainment
	GRADE	Int	Grade level attending
	OCCCEN1	Int	Occupation (Census) for 1% file
	INDCEN	Int	Industry (Census)
	WKW	Int	Weeks worked
	WKHP	Int	Hours per week
TAZ	TAZ	Int	TAZ
	TERM	Int	Terminal time (minutes) from zone.term file
MGRA	MGRA	Int	MAZ
	TAZ	int	TAZ
	HH	Real	total number of households
	POP	Real	total population
	EMP_SELF	Real	Self-employed
	EMP_AG	Real	Agriculture
	EMP_CONST_NON_BLDG_PROD	Real	Construction Non-Building production (including mining)
	EMP_CONST_NON_BLDG_OFFICE	Real	Construction Non-Building office support (including mining)

Table	Field	Type	Description
	EMP_UTILITIES_PROD	Real	Utilities production
	EMP_UTILITIES_OFFICE	Real	Utilities office support
	EMP_CONST_BLDG_PROD	Real	Construction of Buildings production
	EMP_CONST_BLDG_OFFICE	Real	Construction of Buildings office support
	EMP_MFG_PROD	Real	Manufacturing production
	EMP_MFG_OFFICE	Real	Manufacturing office support
	EMP_WHSLE_WHS	Real	Wholesale and Warehousing
	EMP_TRANS	Real	Transportation Activity
	EMP_RETAIL	Real	Retail Activity
	EMP_PROF_BUS_SVCS	Real	Professional and Business Services
	EMP_PROF_BUS_SVCS_BLDG_MAINT	Real	Professional and Business Services (Building Maintenance)
	EMP_PVT_ED_K12	Real	Private Education K-12
	EMP_PVT_ED_POST_K12_OTH	Real	Private Education Post-Secondary (Post K-12) and Other
	EMP_HEALTH	Real	Health Services
	EMP_PERSONAL_SVCS_OFFICE	Real	Personal Services Office Based
	EMP_AMUSEMENT	Real	Amusement Services
	EMP_HOTEL	Real	Hotels and Motels
	EMP_RESTAURANT_BAR	Real	Restaurants and Bars
	EMP_PERSONAL_SVCS_RETAIL	Real	Personal Services Retail Based
	EMP_RELIGIOUS	Real	Religious Activity
	EMP_PVT_HH	Real	Private Households
	EMP_STATE_LOCAL_GOV_ENT	Real	State and Local Government Enterprises Activity
	EMP_SCRAP_OTHER	Real	Scrap other
	EMP_FED_NON_MIL	Real	Federal Non-Military Activity
	EMP_FED_MIL	Real	Federal Military Activity
	EMP_STATE_LOCAL_GOV_BLUE	Real	State and Local Government Non-Education Activity production
	EMP_STATE_LOCAL_GOV_WHITE	Real	State and Local Government Non-Education Activity office support
	EMP_PUBLIC_ED	Real	Public Education K-12 and other
	EMP_OWN_OCC_DWELL_MGMT	Real	Owner-Occupied Dwellings Management and Maintenance

Table	Field	Type	Description
			Activity
	EMP_FED_GOV_ACCTS	Real	Federal Government Accounts
	EMP_ST_LCL_GOV_ACCTS	Real	State and Local Government Accounts
	EMP_CAP_ACCTS	Real	Capital Accounts
	EMP_TOTAL	Real	Total employment
	ENROLLGRADEKTO8	Real	Grade School K-8 enrollment
	ENROLLGRADE9TO12	Real	Grade School 9-12 enrollment
	COLLEGEENROLL	Real	Major College enrollment
	OTHERCOLLEGEENROLL	Real	Other College enrollment
	ADULTSCHENRL	Real	Adult School enrollment
	ECH_DIST	Real	Elementary school district
	HCH_DIST	Real	High school district
	PARKAREA	Real	1: Trips with destinations in this MAZ may choose to park in a different MAZ, parking charges apply (downtown) 2: Trips with destinations in parkarea 1 may choose to park in this MAZ, parking charges might apply (quarter mile buffer around downtown) 3: Only trips with destinations in this MAZ may park here, parking charges apply (outside downtown paid parking, only show cost no capacity issue) 4: Only trips with destinations in this MAZ may park here, parking charges do not apply (outside downtown, free parking)
	HSTALLSOTH	Real	Number of stalls allowing hourly parking for trips with destinations in other MAZs
	HSTALLSSAM	Real	Number of stalls allowing hourly parking for trips with destinations in the same MAZ
	HPARKCOST	Real	Average cost of parking for one hour in hourly stalls in this MAZ, dollars
	NUMFREEHRS	Real	Number of hours of free parking allowed before parking charges begin in hourly stalls

Table	Field	Type	Description
	DSTALLSOTH	Real	Stalls allowing daily parking for trips with destinations in other MAZs
	DSTALLSSAM	Real	Stalls allowing daily parking for trips with destinations in the same MAZ
	DPARKCOST	Real	Average cost of parking for one day in daily stalls, dollars
	MSTALLSOTH	Real	Stalls allowing monthly parking for trips with destinations in other MAZs
	MSTALLSSAM	Real	Stalls allowing monthly parking for trips with destinations in the same MAZ
	MPARKCOST	Real	Average cost of parking for one day in monthly stalls, amortized over 22 workdays, dollars
	TOTINT	Real	Total intersections (calculated by maz_densities.s script)
	DUDEN	Real	Dwelling unit density (calculated by maz_densities.s script)
	EMPDEN	Real	Employment density (calculated by maz_densities.s script)
	POPDEN	Real	Population density (calculated by maz_densities.s script)
	RETEMPDEN	Real	Retail employment density (calculated by maz_densities.s script)
	TOTINTBIN	Real	Total intersection bin (calculated by maz_densities.s script)
	EMPDENBIN	Real	Employment density bin (calculated by maz_densities.s script)
	DUDENBIN	Real	Dwelling unit density bin (calculated by maz_densities.s script)
	POINT_X	Real	Centroid X coordinate
	POINT_Y	Real	Centroid Y coordinate
	ACRES	Real	MAZ acres
TAP	TAP	Int	TAP
	LOTID	Int	Lot ID from tap.pytype input file
	PTYPE	Int	Parking type from tap.pytype input file
	TAZ	Int	Nearest TAZ from tap.pytype input

Table	Field	Type	Description
			file
	CAPACITY	Int	Lot capacity from tap.pytype input file
	DISTANCE	Int	Distance lot to TAP from input tap.pytype file
MGRATOTAP	TAP	Int	
	DISTANCE	Int	Distance (feet) from input wkacc file.
	MGRA	Int	
MGRATOMGRA	ORIG_MGRA	Int	Origin MGRA
	DEST_MGRA	Int	Destination MGRA
	DISTANCE	Int	Distance (feet) from input mgrataz.walk file.
TAZTOTAP	TAP	Int	TAP
	DISTANCE	Int	Distance (feet) from input access.prp file.
	TIME	Int	Time (minutes * 100) from input access.prp file.
	TAZ	Int	TAZ
ACCESSIBILITIES	COLUMN_1	Real	Accessibility measure 1. See alts tab in Accessibilities.xls UEC file.
	…		Accessibility measure 2-46.
	COLUMN_47	Real	Accessibility measure 47.
	MGRA	Int	MGRA
HHDATA	HOME_MGRA	Int	Home MGRA
	INCOME	Int	Income
	AUTOS	Int	Estimated autos
	TRANSPONDER	Int	Has transponder

Table	Field	Type	Description
	CDAP_PATTERN	Varchar	CDAP model result (CDAP result by person and number of joint tours)
	JTF_CHOICE	Int	Joint tour frequency model choice
	HH_ID	Int	HH ID. See personData.csv.
PERSONDATA	PERSON_ID	Int	Person ID
	AGE	Int	Age
	PERSON_NUM	Int	Number of person in household
	GENDER	Varchar	Gender
	TYPE	Varchar	Person type
	VALUE_OF_TIME	Real	Value-of-time
	ACTIVITY_PATTERN	Varchar	CDAP model result
	IMF_CHOICE	Int	Individual mandatory tour frequency choice model result
	INMF_CHOICE	Int	Individual non-mandatory tour frequency choice model result
	FP_CHOICE	Int	Free parking choice model result
	REIMB_PCT	Real	Parking reimbursement percent
	HHID	Int	HH ID. See wsLocResults.csv.
WSLOCATION	HOMEMGRA	Int	Home MGRA
	INCOME	Int	Income
	PERSONID	Int	Person ID
	PERSONNUM	Int	Person number in HH
	PERSONTYPE	Int	Person type
	PERSONAGE	Int	Age
	EMPLOYMENTCATEGORY	Int	Employment category (PopSyn PEMPLOY)
	STUDENTCATEGORY	Int	Student category (PopSyn PSTUDENT)
	WORKSEGMENT	Int	Work location choice segment. See size terms – work tab in Accessibilities.xls.
	SCHOOLSEGMENT	Int	School location choice segment. See the size terms – school tab in

Table	Field	Type	Description
			Accessibilities.xls and the K-12 school districts for the home MGRA in the MGRA table.
	WORKLOCATION	Int	Work MGRA
	WORKLOCATIONDISTANCE	Real	Distance from home
	WORKLOCATIONLOGSUM	Real	Logsum from home
	SCHOOLLOCATION	Int	School MGRA
	SCHOOLLOCATIONDISTANCE	Real	Distance from home
	SCHOOLLOCATIONLOGSUM	Real	Logsum from home
	HH_ID	Int	HH ID. See indivTourData.csv.
INDIVTOUR	PERSON_ID	Int	Person ID
	PERSON_NUM	Int	Person number in HH
	PERSON_TYPE	Int	Person type
	TOUR_ID	Int	Tour ID
	TOUR_CATEGORY	Varchar	Tour category
	TOUR_PURPOSE	Varchar	Tour purpose. See Definition table.
	ORIG_MGRA	Int	Origin MGRA
	DEST_MGRA	Int	Destination MGRA
	START_PERIOD	Int	Tour start time period. See Definition table.
	END_PERIOD	Int	Tour end time period. See Definition table.
	TOUR_MODE	Int	Tour mode choice result. See TourModeChoice.xls. See Definition table.
	TOUR_DISTANCE	Real	Tour distance
	ATWORK_FREQ	Int	At-work tour frequency
	NUM_OB_STOPS	Int	Number of outboard stops on tour
	NUM_IB_STOPS	Int	Number of inbound stops on tour
	UTIL_1	Real	Utility tour primary destination choice sample 1
	…	Real	Utility tour primary destination choice sample 2 -25
	UTIL_26	Real	Utility tour primary destination choice sample 26

Table	Field	Type	Description
	PROB_1	Real	Probability tour primary destination choice sample 1
	…	Real	Probability tour primary destination choice sample 2 -25
	PROB_26	Real	Probability tour primary destination choice sample 26
	HH_ID	Int	HH ID. See jointTourData.csv.
JOINTTOUR	TOUR_ID	Int	Tour ID
	TOUR_CATEGORY	Varchar	Tour category
	TOUR_PURPOSE	Varchar	Tour purpose. See Definition table.
	TOUR_COMPOSITION	Int	Number of participants
	TOUR_PARTICIPANTS	Varchar	HH Person numbers
	ORIG_MGRA	Int	Origin MGRA
	DEST_MGRA	Int	Destination MGRA
	START_PERIOD	Int	Tour start time period. See Definition table.
	END_PERIOD	Int	Tour end time period. See Definition table.
	TOUR_MODE	Int	Tour mode choice result. See TourModeChoice.xls. See Definition table.
	TOUR_DISTANCE	Real	Tour distance
	NUM_OB_STOPS	Int	Number of outboard stops on tour
	NUM_IB_STOPS	Int	Number of inbound stops on tour
	UTIL_1	Real	Utility tour primary destination choice sample 1
	…	Real	Utility tour primary destination choice sample 2 -25
	UTIL_26	Real	Utility tour primary destination choice sample 26
	PROB_1	Real	Probability tour primary destination choice sample 1
	…	Real	Probability tour primary destination choice sample 2 -25
	PROB_26	Real	Probability tour primary destination choice sample 26

Table	Field	Type	Description
	HH_ID	Int	HH ID. See indivTripData.csv.
INDIVTRIP	PERSON_ID	Int	Person ID
	PERSON_NUM	Int	Person number in HH
	TOUR_ID	Int	Tour ID
	STOP_ID	Int	Stop ID
	INBOUND	Int	Is inbound stop
	TOUR_PURPOSE	Varchar	Tour purpose. See Definition table.
	ORIG_PURPOSE	Varchar	Trip origin purpose. See Definition table.
	DEST_PURPOSE	Varchar	Trip destination purpose. See Definition table.
	ORIG_MGRA	Int	Origin MGRA
	DEST_MGRA	Int	Destination MGRA
	PARKING_MGRA	Int	Parking MGRA if applicable
	STOP_PERIOD	Int	Trip arrival time. See Definition table.
	TRIP_MODE	Int	Trip mode choice result. See TripModeChoice.xls. . See Definition table.
	TRIP_BOARD_TAP	Int	Trip boarding TAP if applicable
	TRIP_ALIGHT_TAP	Int	Trip alighting TAP if applicable
	TOUR_MODE	Int	Tour mode. See Definition table.
	TRIP_TIME	**Real**	**Trip travel time**
	TRIP_DISTANCE	**Real**	**Trip distance**
	TRIP_COST	**Real**	**Trip cost**
	TRIP_PURPOSE_NAME	**Varchar**	**Trip purpose name**
	TRIP_MODE_NAME	**Varchar**	**Trip mode name**
	RECID	**Int**	**Record ID calculated on insert**
	TRIB_BOARD_TAZ	**Int**	**Trip boarding TAZ if applicable**
	TRIP_ALIGHT_TAZ	**Int**	**Trip alighting TAZ if applicable**
	ORIG_TAZ	**Int**	**Origin TAZ**
	DEST_TAZ	**Int**	**Destination TAZ**
	HH_ID	Int	HH ID. See jointTripData.csv.

Table	Field	Type	Description
JOINTTRIP	TOUR_ID	Int	Tour ID
	STOP_ID	Int	Stop ID
	INBOUND	Int	Is inbound stop
	TOUR_PURPOSE	Varchar	Tour purpose. See Definition table.
	ORIG_PURPOSE	Varchar	Trip origin purpose. See Definition table.
	DEST_PURPOSE	Varchar	Trip destination purpose. See Definition table.
	ORIG_MGRA	Int	Origin MGRA
	DEST_MGRA	Int	Destination MGRA
	PARKING_MGRA	Int	Parking MGRA if applicable
	STOP_PERIOD	Int	Trip arrival time. See Definition table.
	TRIP_MODE	Int	Trip mode choice result. See TripModeChoice.xls. . See Definition table.
	NUM_PARTICIPANTS	Int	Number of participants
	TRIP_BOARD_TAP	Int	Trip boarding TAP if applicable
	TRIP_ALIGHT_TAP	Int	Trip alighting TAP if applicable
	TOUR_MODE	Int	Tour mode. See Definition table.
	TRIP_TIME	**Real**	**Trip travel time**
	TRIP_DISTANCE	**Real**	**Trip distance**
	TRIP_COST	**Real**	**Trip cost**
	TRIP_PURPOSE_NAME	**Varchar**	**Trip purpose name**
	TRIP_MODE_NAME	**Varchar**	**Trip mode name**
	RECID	**Int**	**Record ID calculated on insert**
	TRIB_BOARD_TAZ	**Int**	**Trip boarding TAZ if applicable**
	TRIP_ALIGHT_TAZ	**Int**	**Trip alighting TAZ if applicable**
	ORIG_TAZ	**Int**	**Origin TAZ**
	DEST_TAZ	**Int**	**Destination TAZ**
	HH_ID	Int	HH ID. See jointTripData.csv.
TRIP	**HH_ID**	**Int**	**HH ID. See jointTripData.csv.**
	TOUR_ID	**Int**	**Tour ID**

Table	Field	Type	Description
	STOP_ID	Int	Stop ID
	INBOUND	Int	Is inbound stop
	TOUR_PURPOSE	Varchar	Tour purpose. See Definition table.
	ORIG_PURPOSE	Varchar	Trip origin purpose. See Definition table.
	DEST_PURPOSE	Varchar	Trip destination purpose. See Definition table.
	ORIG_MGRA	Int	Origin MGRA
	DEST_MGRA	Int	Destination MGRA
	PARKING_MGRA	Int	Parking MGRA if applicable
	STOP_PERIOD	Int	Trip arrival time. See Definition table.
	TRIP_MODE	Int	Trip mode choice result. See TripModeChoice.xls. . See Definition table.
	NUM_PARTICIPANTS	Int	Number of participants
	TRIP_BOARD_TAP	Int	Trip boarding TAP if applicable
	TRIP_ALIGHT_TAP	Int	Trip alighting TAP if applicable
	TOUR_MODE	Int	Tour mode. See Definition table.
	TRIP_TIME	Real	Trip travel time
	TRIP_DISTANCE	Real	Trip distance
	TRIP_COST	Real	Trip cost
	TRIP_PURPOSE_NAME	Varchar	Trip purpose name
	TRIP_MODE_NAME	Varchar	Trip mode name
	RECID	Int	Record ID calculated on insert
	TRIP_BOARD_TAZ	Int	Trip boarding TAZ if applicable
	TRIP_ALIGHT_TAZ	Int	Trip alighting TAZ if applicable
	ORIG_TAZ	Int	Origin TAZ
	DEST_TAZ	Int	Destination TAZ
	PERID	Int	ABM person ID
	HOUSEHOLD_SERIAL_NO	Int	PUMS HH ID
	PNUM	Int	Person number in HH
	AGE	Int	Age
	SEX	Int	Sex

Table	Field	Type	Description
	MILITARY	Int	Military Service code
	PEMPLOY	Int	PopSyn employment type
	PSTUDENT	Int	PopSyn student type
	PTYPE	Int	PopSyn person type
	EDUC	Int	Educational attainment
	GRADE	Int	Grade level attending
	OCCCEN1	Int	Occupation (Census) for 1% file
	INDCEN	Int	Industry (Census)
	WKW	Int	Weeks worked
	WKHP	Int	Hours per week
	OCC_CODE	Int	Pecas Occupation Code
TOUR	HH_ID	Int	HH ID. See jointTripData.csv.
	TOUR_ID	Int	Tour ID
	TOUR_PURPOSE	Varchar	Tour purpose. See Definition table.
	ORIG_MGRA	Int	Origin MGRA
	DEST_MGRA	Int	Destination MGRA
	START_PERIOD	Int	Tour start time period. See Definition table.
	END_PERIOD	Int	Tour end time period. See Definition table.
	NUM_OB_STOPS	Int	Number of outboard stops on tour
	NUM_IB_STOPS	Int	Number of inbound stops on tour
	TOUR_MODE	Int	Tour mode. See Definition table.
	TOUR_TIME	Real	Tour travel time summed up from composing trips
	TOUR_DISTANCE	Real	Tour travel distance summed up from composing trips
	TOUR_TYPE	Varchar	Trip cost
	RECID	Int	Record ID calculated on insert
	ORIG_TAZ	Int	Origin TAZ
	DEST_TAZ	Int	Destination TAZ
	PERID	Int	ABM person ID

Table	Field	Type	Description
	HOUSEHOLD_SERIAL_NO	Int	PUMS HH ID
	PNUM	Int	Person number in HH
	AGE	Int	Age
	SEX	Int	Sex
	MILITARY	Int	Military Service code
	PEMPLOY	Int	PopSyn employment type
	PSTUDENT	Int	PopSyn student type
	PTYPE	Int	PopSyn person type
	EDUC	Int	Educational attainment
	GRADE	Int	Grade level attending
	OCCCEN1	Int	Occupation (Census) for 1% file
	INDCEN	Int	Industry (Census)
	WKW	Int	Weeks worked
	WKHP	Int	Hours per week
	NUM_PARTICIPANTS	Int	Number of participants
	TOUR_PARTICIPANTS	Varchar	List of PNUM values of tour participants

For custom analysis, the user may directly connect to the database through either a database management tool such as SQL Server Management Studio (Express) or through a data analysis package such as Excel, R, Stata, etc. If connecting through a data analysis package, the database is usually accessed by either setting up an ODBC Data Source (under Administrative tools under Control Panel) or on-the-fly via a connection string. Once connected to the database, the user can execute SQL statements against the database to perform custom analysis.

Reports

A set of sample reports is included. The scripts can be run on the command line using the tsql command line tool, in SQL Server's database management tool SQL Server Management Studio (to do so, one must first replace the schema tokens with the actual database schema name), or by the report packager, as part of the batch file call described below. These reports is not meant to be exhaustive. Rather, since these reports cover much of the reporting requirements, they are a good set of reports to start with. The reports, listed by worksheet name (and corresponding SQL script), are:

1) CBD Trips By Occupation (cbdtrips.csv)

Outputs a count of all of the trips to the CBD area, as defined in the CBDVEHICLES table, grouped by occupation (White Collar, Services, Health, Retail and Food, Blue Collar, Military, Other). Takes the TRIP and CBDVEHICLES tables as inputs.

2) Time Spent Traveling By Type (time_spent_traveling_type.sql)

Outputs the average amount of time spent traveling grouped by income level (LO,MED-LO,MED,MED-HI,HI, Other (negative)) and occupation (White Collar, Services, Health, Retail and Food, Blue Collar, Military, Other). Combines joint and individual trips – in joint trips each participating individual's time is added to the total. Takes the TRIP and HHDATA tables as inputs.

3) Primary Tour Distance (primary_tour_distance.sql)

Outputs the average distances of primary tours (starting from home mgra) for all combinations of person types (Full Time Worker, Part Time Worker, University Student, NonWorking Adult, Retiree, Driving Age School Children, Pre-Driving Age School Children, Pre-School Children) and tour purposes (Mandatory, Maintenance, Discretionary). Takes the TOUR and HHDATA tables as inputs.

4) Primary Tour Time (primary_tour_time.sql)

Outputs the average times of primary tours (starting from home mgra) for all combinations of person types (Full Time Worker, Part Time Worker, University Student, NonWorking Adult, Retiree, Driving Age School Children, Pre-Driving Age School Children, Pre-School Children) and tour purposes (Mandatory, Maintenance, Discretionary). Takes the TOUR and HHDATA tables as inputs.

5) Individual Tours By Type (indiv_tours_type.sql)

Outputs the total number of individual tours taken (and the number of people on those tours) by specific person types (Full Time Worker, Part Time Worker, University Student, NonWorking Adult, Retiree, Driving Age School Children, Pre-Driving Age School Children, Pre-School Children) in a given county (Palm Beach, Broward and Miami-Dade). Takes the TOUR, SYNHH and DISTRICTDEFINITIONS tables as inputs.

6) Joint Tours By Type (joint_tours_type.sql)

Outputs the total number of joint tours taken (and the number of people on those tours) by specific person types (Full Time Worker, Part Time Worker, University Student, NonWorking Adult, Retiree, Driving Age School Children, Pre-Driving Age School Children, Pre-School Children) in a given county (Palm Beach, Broward and Miami-Dade). Takes the TOUR, SYNHH and DISTRICTDEFINITIONS tables as inputs.

7) Persons By County (person_county.sql)

Outputs the total number of persons residing in each county (Palm Beach, Broward and Miami-Dade) by type (Full Time Worker, Part Time Worker, University Student, NonWorking Adult, Retiree, Driving Age School Children, Pre-Driving Age School Children, Pre-School Children). Takes the SYNPERSON, SYNHH and DISTRICTDEFINITIONS tables as inputs.

8) Total Tours By Mode Purpose and Person Type (total_tours_type.sql)

Outputs the total number of tours occurring by each mode, purpose (Mandatory, Maintenance, Discretionary) and person type (Full Time Worker, Part Time Worker, University Student,

NonWorking Adult, Retiree, Driving Age School Children, Pre-Driving Age School Children, Pre-School Children) by county (Palm Beach, Broward and Miami-Dade). Takes the TOUR and DISTRICTDEFINITIONS tables as inputs.

9) Total Trips By Mode Purpose and Person Type (total_trips_type.sql)

Outputs the total number of trips occurring by each mode, purpose (Mandatory, Maintenance, Discretionary) and person type (Full Time Worker, Part Time Worker, University Student, NonWorking Adult, Retiree, Driving Age School Children, Pre-Driving Age School Children, Pre-School Children) by county (Palm Beach, Broward and Miami-Dade). Takes the TRIP and DISTRICTDEFINITIONS tables as inputs.

10) Tours By Origin District and Purpose (tours_district.sql)

Outputs number of persons traveling between each district pairing grouped by tour purposes (Mandatory, Maintenance, Discretionary). Takes the TOUR and DISTRICTDEFINITIONS tables as inputs.

11) Cleanup (cleanup.sql)

This script deletes any output tables that were created as a result of running the above scripts – this is run before the reports scripts, deleting any of the old output tables. If additional reports scripts are added, they should also be appended to this file.

Report Packaging

The report packaging Java program will query the report tables in the database and write each report table to a separate tab in an Excel workbook. The report packager can be run on the command line and will take as input a Java properties file that specifies the following:

1) Directory where SQL scripts are contained
2) SQL report scripts to run
3) For each SQL report script:
    a. Database table name
    b. Worksheet (Excel tab) name
4) Name of the output workbook
5) Server, database and instance name (instance must be declared separately)
6) Username and password on database

Running the Reporting System

At the end of a model run, the reporting ETL component can be run to build the model run database for later use. In addition, the reporting scripts and report packaging can also be run at the end of the model run to generate the reports. Users can also just build the database at the end of the model run, and then run reports later as needed.

Running the reporting system requires a complete run of the SERPM model to be available in the project directory. SERPM model input and output files are used to create the reports.

Setting up

The **serpm_reports.properties** file must be configured prior to running the reporting system. This file, in addition to the batch file described below contains all of the necessary setting to run the reporting system. The properties file contains the following variables which must be set:

1) **sql.scripts.directory**, the directory where the desired SQL reports are stored. For example: C:/serpm7/reports_sql
2) **sql.scripts.schema.key** , the token that will be replaced as the schema name defined in database.schema (below). For example: @@SCHEMA@@
3) **sql.scripts**, the names of the report scripts to be run. The listed files must not be separated by anything other than a comma (no spaces or linebreaks). For example: cbdtrips.sql,cbdjointtrips.sql…
4) **worksheet.tables**, the tables in the database that will be created as part of the scripts above to be transferred as worksheets. The listed names must not be separated by anything other than a comma (no spaces or linebreaks). For example: tempcbdtrips,tempjointcbdtrips…
5) **worksheet.names**, the names of the worksheets within the excel file that represent the reports that were ran. The listed names must not be separated by anything other than a comma (no spaces or linebreaks). For example: Individual CBD Trips By Occupation,Joint CBD Trips By Occupation…
6) **workbook.name**, the name and path of the excel file to house all of the worksheets described above. For example: C:/serpm7/test_outputs/SERPM_REPORT.xls
7) **database.schema**, the name of the database schema for which reports will be created. Must match the schema name given in the **runReports.bat** file described below. Example: 2010R
8) **database.ipaddress**, the IP of the server on which the database resides. For example: DOTSD4HQSQL
9) **database.instance**, the instance of the server on which the database resides. For example: DOTSD4PLEMO (must be passed separately, i.e. DOTSD4HQSQL/ DOTSD4PLEMO will not work)
10) **database.port**, the port on which the database is accepting interactions. For example: 1433
11) **database.name**, the name of the database to be created (if it does not exist) or appended. For example: SERPMABM
12) **database.username**, the username for the database. For example: serpm. (Note: the user must have create, insert and modify rights on the database)
13) **database.password**, the password for the user listed above.

Executing the batch file

The **runSERPMReports.bat** batch file contains all of the necessary call for a complete run of the reporting system. The procedure can be divided into three steps: 1) setting up environment variables, 2) exporting data 3) building the database and running reports.

1. Setting up environment variables

The batch file sets the following variables, which may be changed by the user to reflect the run environment and desired options:

1) **MATRIX_SERVER**, The IP of the matrix server that will read the skims. For example: 156.75.49.36
2) **JDK_PATH**, The path to the Java JDK on the machine running the reports batch file. For example, "C:\Program Files\Java\jdk1.7.0_21" (with quotes).
3) **FEEDBACK_ITERATION**, The feedback iteration number desired for the report. For example: 1
4) **SCHEMA**, The desired name of the schema to be created in the database. For example: 2010R (This name works, but the system will create schema "S_2010R" to comply with SQL Server naming conventions).
5) **PROJECT_DIRECTORY**, the SERPM 7 project directory containing the model run. For example: C:\serpm7\ctramp. The reports module reads the location of model input and output files from the **serpm_abm.properties** file, which references the ABMTEMP temporary scenario.
6) **OUTPUT_DIRECTORY**, the directory in which all generated CSV and SQL files will be placed. For example: \\dotsd4hqmodel2.d4b.dot.state.fl.us\serpm7\Output\OUT-%SCHEMA%\Reports. It is required to use a fully qualified machine name and shared folder in order for this location to be accessible to the SQL Server database.
7) **[DELETESCRIPTS]**, A flag for specifying if individual table-building SQL scripts should be deleted. If TRUE, only the final "sql_table_import.sql" script is retained. All CSV files are still created. For example: TRUE. This flag is optional - if it is not passed, all SQL scripts will be kept by default.
8) **[BUILDFILE]**, A path to the sql_table_import.sql file needed to build the database. If the database is already built, and only a report run is required, this field may be set to and empty string (""), or not passed on to the **runSERPMReportBuilder.bat** file.

2. Exporting data

After the variables are defined, the batch file proceeds to kill any active java processes (to ensure the environment variables are reset) and starts the Matrix Manger using the **runMtxMgr.bat** script. This script requires the **MATRIX_SERVER** and **JDK_PATH** variables to be passed with the call, in that order.

Once the Matrix Manager is running, the **runSERPMDataExport.bat** batch file is called to generate the CSV and SQL files for the reporting database. The batch file takes up to 5 parameters – **PROJECT_DIRECTORY**, **OUTPUT_DIRECTORY**, **FEEDBACK_ITERATION**, **SCHEMA** and **[DELETE_SCRIPTS]** (optional). The script continues to call DataExporter.java to create the necessary files and translate model outputs and inputs into CSV files for future ingestion into the database.

3. Building the database and running reports

Once the **runSERPMDataExport.bat** finishes, the **runSERPMReports.bat** file runs the **runSERPMReportBuilder.bat** batch file. This file takes up to two parameters – **PROJECT_DIRECTORY** and **[BUILDFILE]** (optional). The second parameter points to the sql_table_import.sql file. This is the file to build the database and must retain this name – however, the parameter may be skipped if the database is already built and only running the reports is of interest.

Creating Additional Reports

The reports included are meant to be an example set, and can be readily extended. There are three basic steps that are needed to create additional reports:

1) Create a *.sql script (using the proper **sql.scripts.schema.key** token, such as @@SCHEMA@@ when addressing database tables, see example reports) with the desired report query being placed into a table on the database (keeping in mind Excel's rows/columns limitations). NULL values are not allowed by the current setup – thus no report tables are allowed to contain NULL values. (This will result in a Null Pointer exception in Java upon attempting to insert the report data into the worksheet).

2) Once the *.sql script is created, the table containing the query must be listed in **serpm_reports.properties** file, along with the *.sql filename that is used to create it and a corresponding worksheet name in which to write it.

3) Add a cleanup step for the output table generated by the new report. For example:

```
IF OBJECT_ID('@@SCHEMA@@.newreporttable') IS NOT NULL
      DROP TABLE @@SCHEMA@@. newreporttable';
```

1) Finally, **runSERPMReports.bat** can be run (with **BUILDFILE="",** if a new report is the only change made, to save time and not re-build the database). The new *.sql report will be automatically added as a worksheet to the *.xls output file specified in the **serpm_reports.properties** file.

To facilitate the creation of additional scripts, the **TRIP** and **TOUR** summary tables can be used for most queries, as they contain the individual and joint trip and tour information joined on person attributes, thus exposing the primary outputs of the model runs.

Note: when building new temporary tables that are meant to replace old ones, but have a different number/set of columns; it may be necessary to flush the older tables from the database to prevent a column mismatch error.

CHAPTER 4 Survey and Activity Model Visualization System

4.1 Overview

This document describes the SERPM activity-based model (ABM) model visualization system, called SERPMVIZ. SERPMVIZ consists of two primary components: a database and a query/visualization dashboard. The database stores the key outputs for a model or survey results and the dashboard is used to generate tables, reports, charts, maps, animations, etc. of these results. The dashboard allows for querying and visualizing each model run or survey dataset independently. SERPMVIZ includes a collection of default queries and visuals that provide examples of many of the types of questions that will be asked of the model and/or survey dataset.

The goal of the system is to be the primary starting point for most model analysis type questions. It is not designed to answer every type of question since the nature of some questions is difficult to answer in a flexible easy-to-use system. However, often difficult questions can be answered through a series of queries that are then export from the system and pieced together externally to provide an answer.

The visualization system, as shown in Figure 9, consists of two core components:

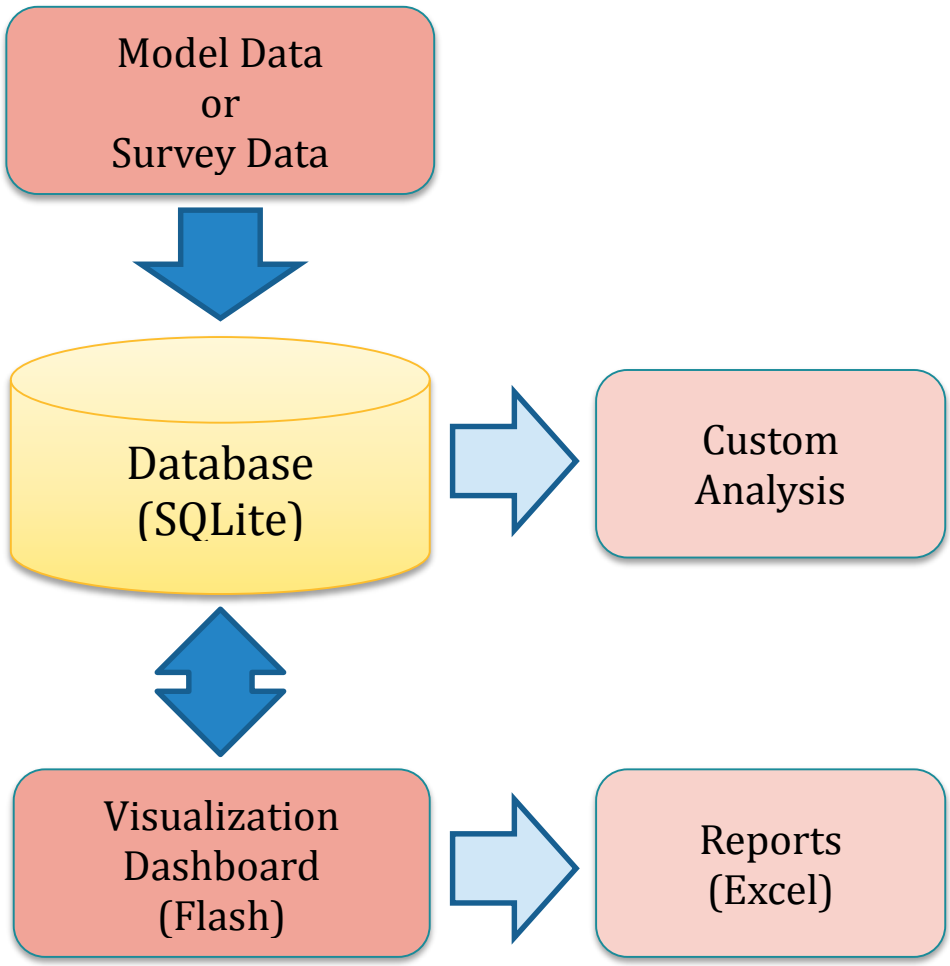
1) Database (implemented in SQLite)
2) Visualization dashboard (implemented in Adobe Flash)

After loading the results into the database, the user of the system can explore results in two ways. The most common and suggested approach is via the interactive model dashboard. From here, the user can create queries, quickly view and interactively analyze results, and generates reports. The reports, which are either tables or maps, can be saved to common formats such as CSV, Excel, PNG, and GIF. The second, and more advanced approach, is to connect to the database directly and perform custom analysis. Directly connecting to the database can be done with the "Get External Data" functions in Excel or Access, via ODBC or similar database connection technology in a data analysis package such as Stata or R, or with a database management tool such as Spatialite.

The components of the visualization system are described below. The description of the components starts with SERPMVIZ – the visualization dashboard, since it is the starting point for most analysis. This document then describes the database and data access layer back-end components, including what data is output by a model run and how to load that data into the system.

Finally, for an introduction to the SQL database query language please refer to the SERPMVIZ training material. The SERPMVIZ training material is also a good supplement to this document since the training material contains additional step by step examples.

Figure 9 - Model Visualization System

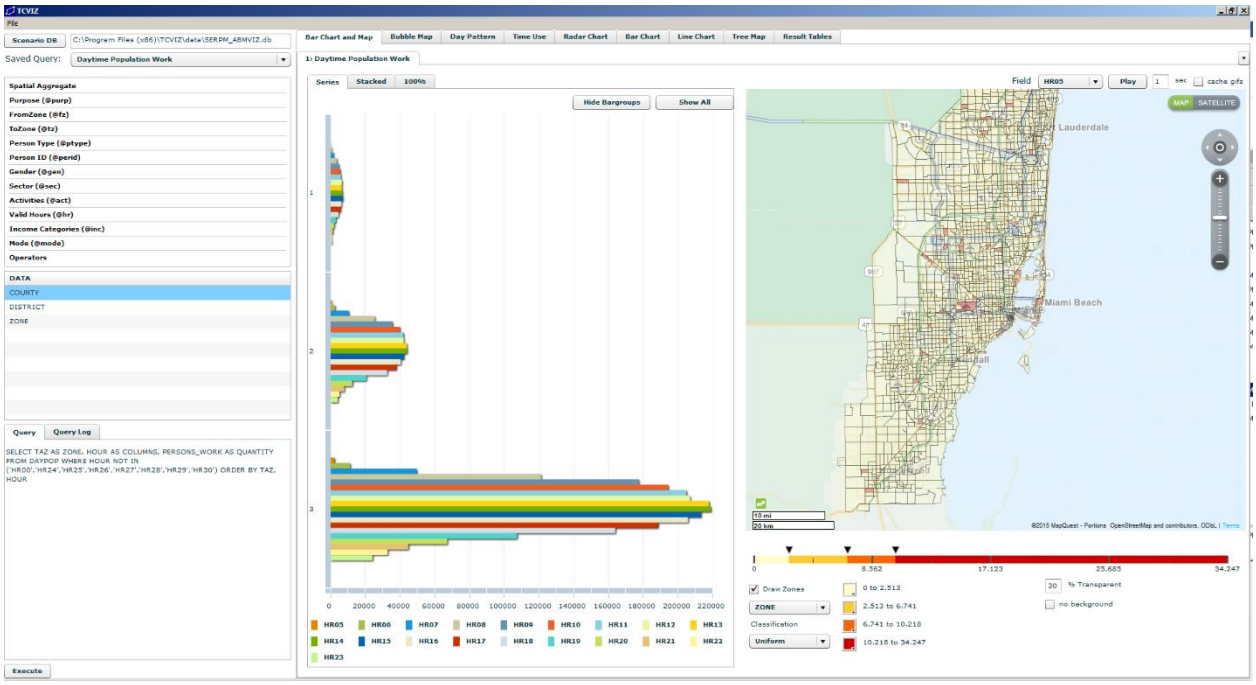


4.2 Visualization Dashboard

The visualization dashboard, called SERPMVIZ and shown in Figure 10, is a model visualization and reporting dashboard. It is designed to be the starting point for querying and visualizing model results. A series of default queries and visuals are included in the installer to get the user started. The primary workflow for SERPMVIZ is to select a model database, run a default (or custom) query, visualize the results, and save the resulting table and visual for later use. Especially useful in SERPMVIZ is the interactive nature of the application, which makes exploratory data analysis for debugging model results, understanding model relationships, and creating data summaries / maps / visuals / animations fairly quick and easy.

SERPMVIZ is developed in Adobe Flash and is deployed as a Windows desktop application that is run by Adobe AIR (Adobe Integrated Runtime). Adobe AIR is basically Flash on the desktop, with the additional ability to read and write files, which is not allowed by the browser-based Flash player due to Internet security concerns. Adobe Flash technology was selected for the dashboard since it is a popular and extensive framework for interactive visualization. The remainder of this section of the document describes the visualization dashboard.

Figure 10 - Model Visualization Dashboard



4.3 Startup

When SERPMVIZ starts, it first reads the Settings.xml file (see below) in order to get the location of the database server and the data access layer. The next step is to read the DefaultQueries.xml file (see below) in order to provide the user with a collection of default queries and visuals to start from. The third step is to run a series of queries to populate the user interface. These queries include: getting all the table names, getting all the possible aggregators and filters (for the Query Builder), getting the zone table, getting the links table, and a few other small pieces of information. After receiving all the query results from the database, the user interface controls are populated with data and the dashboard is ready for use.

The Settings.xml file in the installation folder contains the following settings:

Table 76 - Startup Settings

TABLE	DESCRIPTION	EXAMPLE
<mapParams>	MapQuest map parameters	
mapCenterLat	initial map center latitude	27.4702
mapCenterLng	initial map center longitude	-80.3939
mapLatOffset	lattitude offset (for matching data+map)	0.0000
mapLngOffset	longitude offset (for matching data+map)	0.0000
mapZoom	initial map zoom level	10
<aggOps>	Aggregation operators	
operators	list of operators available	SUM
<query_builder>	Parameters for Query Builder interface	
name	Name of filter	Purpose (@purp)
token	Filter token	@purp
sql	SQL query defining filter content	SELECT DISTINCT LABEL as LABELS, GROUPLABEL, VALUE as DATA FROM definitions WHERE field = 'TPUR1' ORDER BY CAST(DATA AS NUMERIC)
<timeuse_viz>	Parameters for timeuse plots	
colors	List of colors for purposes	#8DD3C7
purposes	List of purposes to use	HOME
<activity_pattern_viz>	Parameters for daily activity patterns	
defaultActivityPatterns	Default pattern "skeletons"	OS%W%H
mandatoryPurposeCodes	List of purposes that should be shown in red	21

The DefaultQueries.xml file in the installation folder contains a series of default queries that can be run in SERPMVIZ. This file can be edited in order to add/remove default queries. Edits must be done carefully to ensure the XML is well formed (i.e. there are no errors in the formatting, opening and closing tags, etc). The default queries XML file is a hierarchy of simple table and query objects. At the top of the hierarchy is a <tables> object that contains multiple <table> objects. Each <table> object contains a <name> and multiple <query> objects. The <name> attribute of the table determines which visual the SQL query will appear under (mapping shown in Table 78). Each <query> object contains a <name> and a <sql> statement. An example of the DefaultQueries.xml file is below.

```
<?xml version="1.0"?>
<tables>
      <table>
        <name> barMapTabs </name>
            <query>
                  <name>Household TAZs for a Trip Purpose</name>
                  <sql>
SELECT
HHTAZ as ZONE,
COUNT(*) as QUANTITY
FROM linkedtrips
WHERE TPUR1 IN (@purp)
GROUP BY ZONE
                  </sql>
            </query>
      </table>
</tables>
```

This query relies on the "@purp" token passed in by the selection of the appropriate purpose in the Query Builder (see Figure 14), which in turn was filled in using the example query shown in Table 76.

NOTE: When editing the DefaultQueries.xml file, make sure to replace any XML special characters with XML entities. The XML special characters are in Table 77.

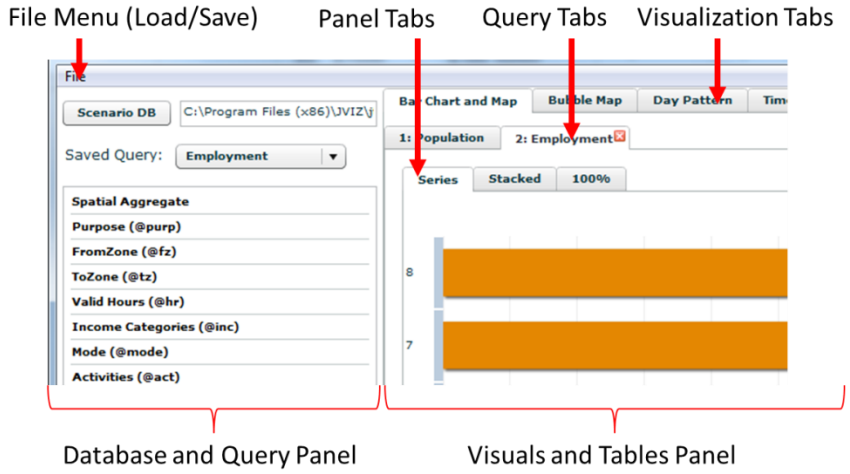
Table 77- XML Special Characters

XML Special Characters	
Character	Replacement
<	<
>	>
&	&
'	'
"	"

4.4 User Interface

The SERPMVIZ user interface is organized into two halves – a Database and Query Panel on the left, and a Visuals and Tables Panel on the right. The Visuals and Tables panel is composed of ten tabbed displays (Visualization Tabs), each of which can produce Query Tabs for every query that is run (Query Tabs can be closed by pressing the red "x" in the top right corner). Furthermore, additional Panel Tabs are present in some visuals. These are shown in Figure 11. The File menu is located in the upper left corner and can be used to save and load queries, tables and snapshots.

Figure 11 – User Interface Layout



The Database and Query Panel is used for selecting a scenario and building and executing queries, while the Visuals and Tables Panel are for data visualization and analysis. Each data Visuals and Tables Panel tab has a table signature – the required query table fields (columns) in order to populate the visual. Table signatures are described in more detailed later in this document. The list of tabs is provided in Table 78. Optional columns are shown in [] brackets. These may be used to add additional dimensions to the visuals.

Table 78 – User Interface Tabs

Tab	Description	Table Signature Fields	XML Table <name>
Bar Chart and Map	Interactive bar chart and map visual	ZONE, QUANTITY	barMapTabs
Bubble Map	Interactive map visual	ZONE, QUANTITY, [QUANTITY2],[QUANTITY3]	bubbleMapTabs
Time Use	Interactive time use visual	HR, ORIG_PURPOSE, QUANTITY	timeusePlots
Tour Tracing	Interactive tour tracing visual	home_taz, orig_taz, dest_taz, depart_hour, trip_mode_name, dest_purpose	tourTracePanel
Tree Map	Interactive tree map visual	MAINGROUP, SUBGROUP, QUANTITY	treeMapTabs
Radar Charts	Interactive radar charts visual	AXIS, CHART, QUANTITY	radarChartTabs
Bar Chart	Interactive bar chart visual	BARGROUP, COLUMNS, QUANTITY	barChartTabs
Line Chart	Interactive bar chart visual	XVAL, QUANTITY, [PIEGROUP],[QUANTITY2]	lineChartTabs

Tables	View query results in table form	NA	tableTabs

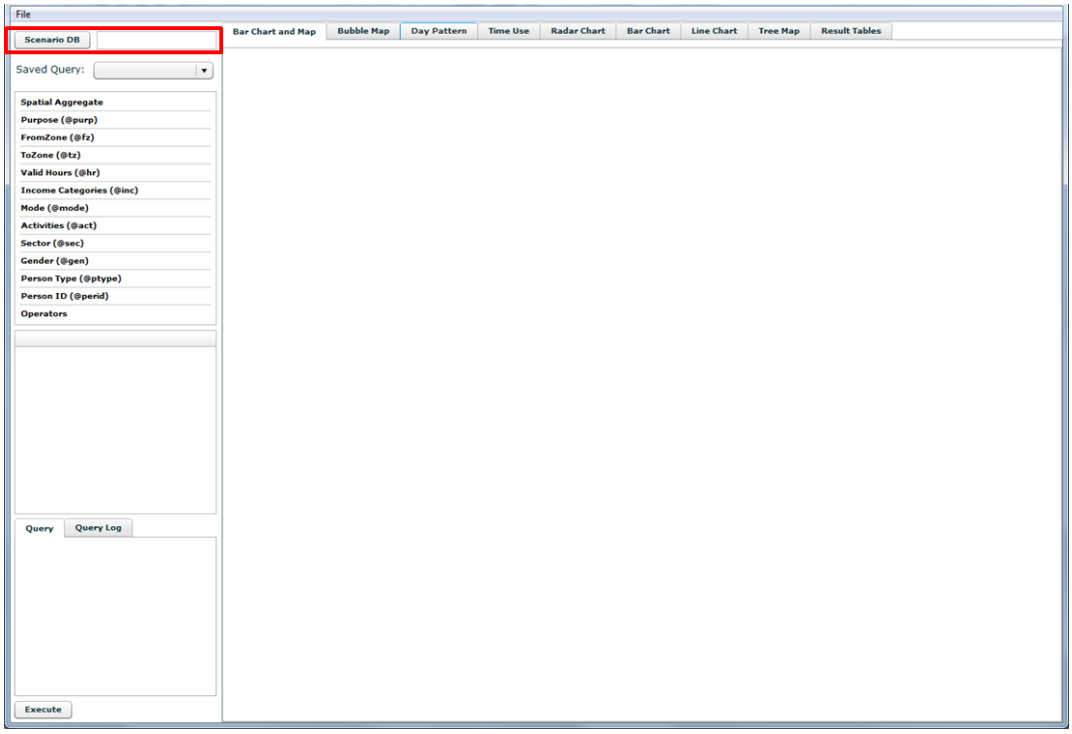
Database and Query

The Database and Query panel allows the user to select a database and input a SQL query to be visualized. The panel is dynamic and has an area which describes the query to be performed. A query is the SQL code that will create a table from the data stored in the database. A number of default queries are listed in the pull down menu, or the user can create their own query, which is subject to certain constraints (described below). The Database and Query panel will update the availiable Saved Queries and the last selected query depending on the tab selected.

The Database and Query tab has the following components:

1) Database Selection (see Figure 12). The database selection controls allows the user to specify what databases to query. The "Get DB" button allows the user to select the appropriate SQLite DB file. The filename is displayed in the "Scenario Database" textbox to the right.

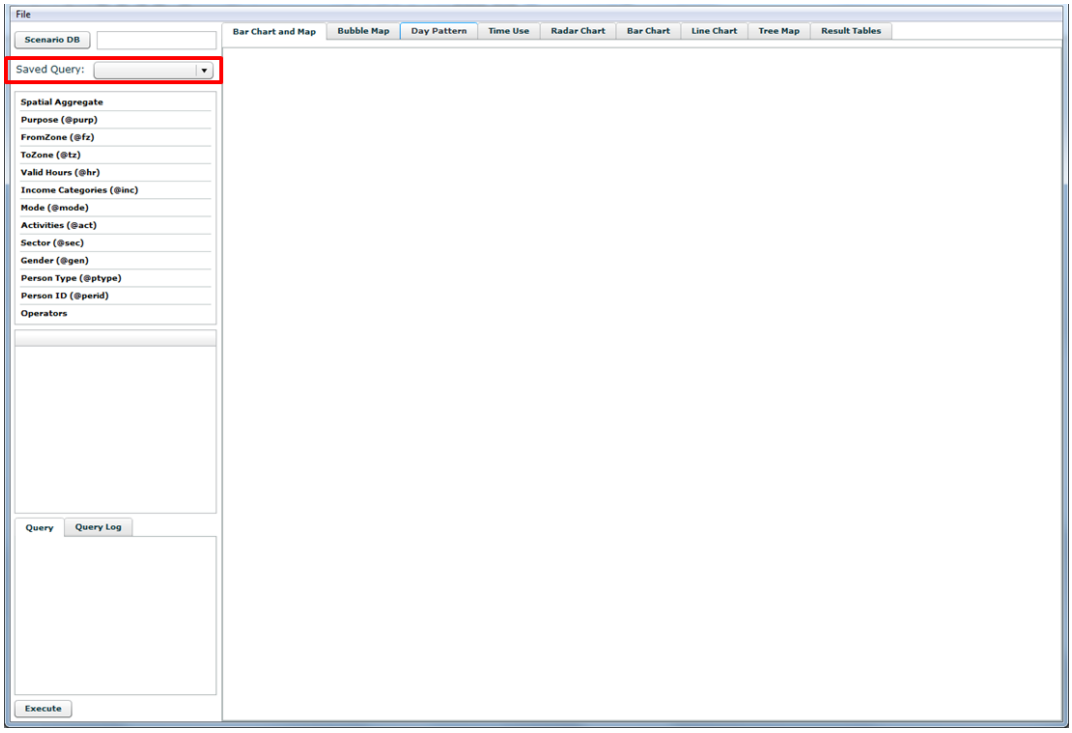
Figure 12 – Database Selection



2) Select from the default queries that are available for each tab visual (as denoted in Table 78) (see

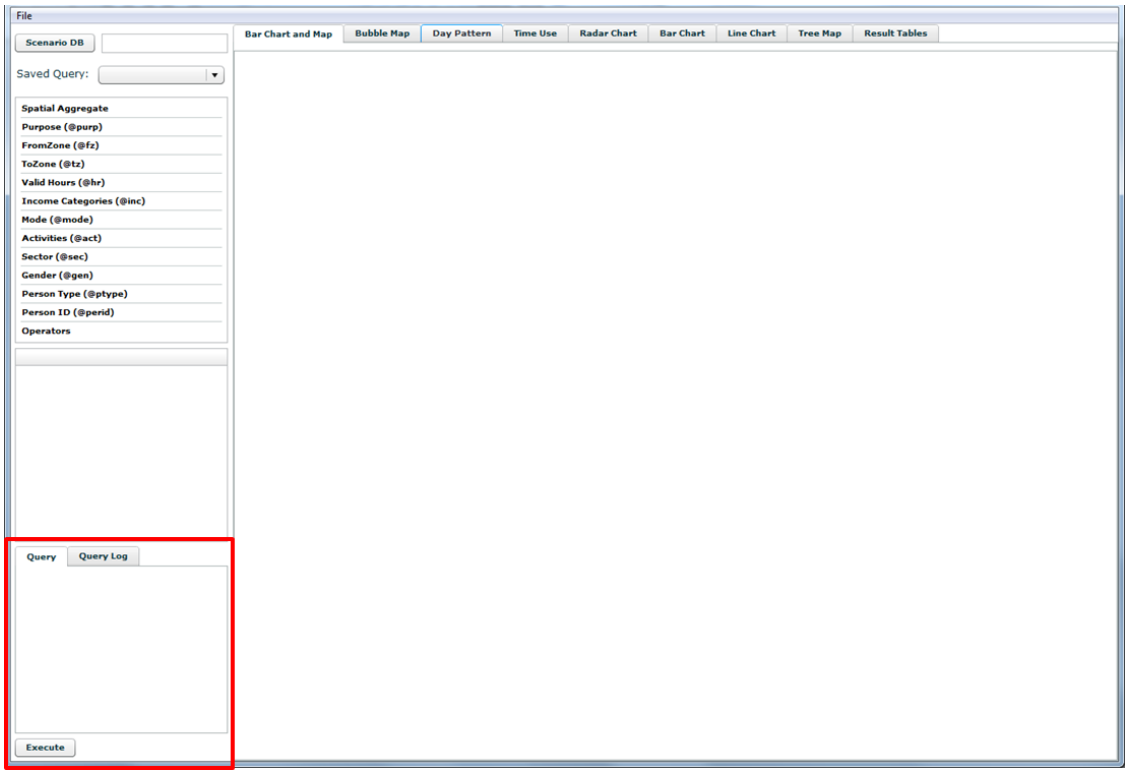
3) Figure 13). These drop-downs allow the users to select from the default queries prepared in advance and stored in the "DefaultQueries.xml" file. These queries can be modified to produce the desired visual or replaced with a completely new query.

Figure 13 – Saved Query Selection



The Query Builder interface (see Figure 14). This interface allows the user to parameterize the constructed query using the results of the queries specified in the Settings.xml file. For example, if the user wishes to aggregate or filter the query based on a specific travel purpose, they may select the "Purpose @purp" link from the left and the available purposes (obtained via the appropriate query in the Settings.xml file) will appear. The user can then select one or multiple purposes that will be used in lieu of the token in the main query when it is executed. The "Spatial Aggregate" and "Operators" lists are special – they may be used without corresponding tokens – these will work for any query that has a ZONES column in the resulting table. If no specific value is selected, the query will run with the first item in the list as the default. In addition to these two, there are ten others currently included in the "Settings.xml" file, but others may be added by editing this file by adding a query name (with corresponding token to remind users), the token and the SQL query necessary to retrieve the list. The tokens will only be substituted for the "DATA" column, thus the SQL query must include this column (see Table 76 for query example).

Figure 14 – Query Builder



4) The execution/input pane, shown in Figure 15 shows the actual query (with tokens) to be executed. It may be edited within the "Query" tab. The "Query Log" tab shows all the queries that have been executed this session. The "Execute" button executes the query and displays the corresponding visual upon the query's completion.

In summary, to specify and execute a saved query, do the following:

a. Select the visual tab to query from the top and select the saved query from the Saved Query drop down selector(if running or editing a stored query).
b. Select the appropriate Query Builder parameters (Spatial Aggregate, Operator and "@xxx" tokens) to fill in the query (if necessary).
c. Edit query further if necessary using the "Query" tab.
d. Execute the query.

The mouse cursor is changed to a busy cursor while the query is executing. When the database completes, SERPMVIZ will return the results to the Tables tab (see the next section).

To specify and execute a saved query to populate a visual, do the following:

a. Repeat the steps above for executing a saved query but ensure the query result contains the appropriate table signature. The required fields for the visual table signatures are listed in Table 79 below.

Figure 15 – Query Pane

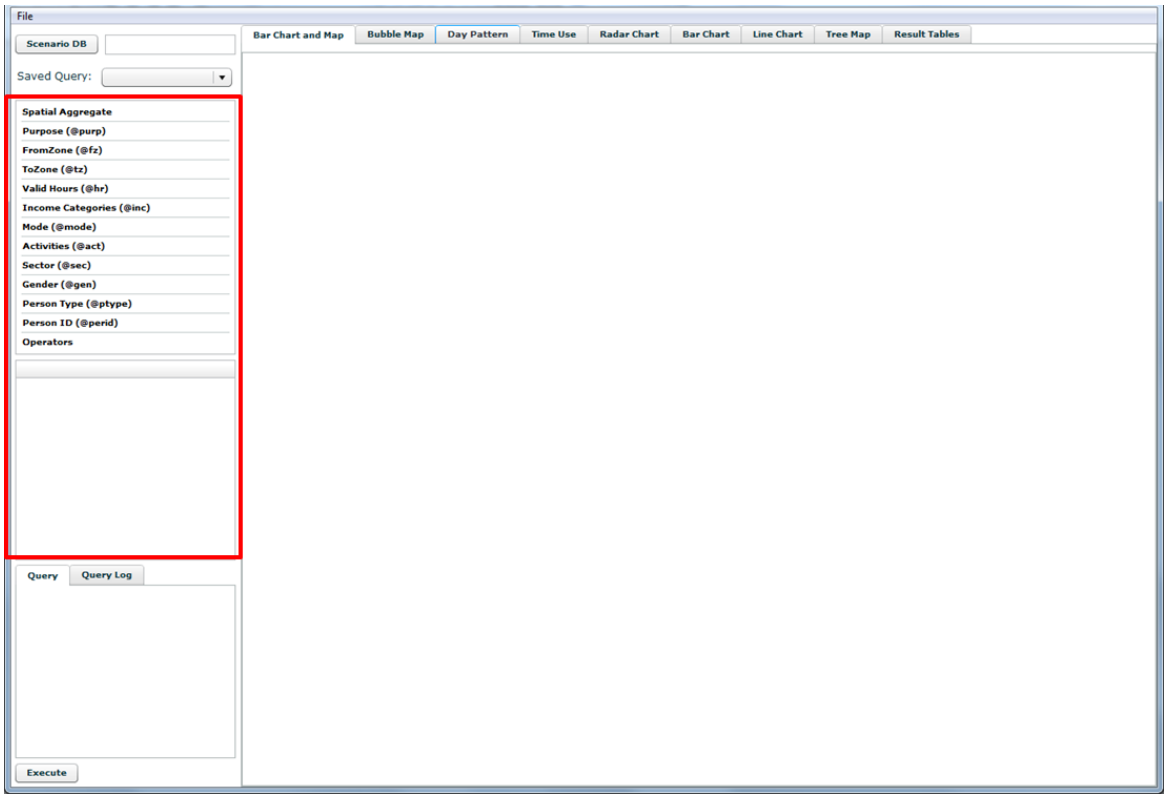


Table 79 - Table Signature Fields

Visual	Description	Table Signature Fields
Bar Chart and Map	Interactive bar chart and map visual	ZONE, QUANTITY
Bubble Map	Interactive map visual	ZONE, QUANTITY, [QUANTITY2],[QUANTITY3]
Time Use	Interactive time use visual	HR, ORIG_PURPOSE, QUANTITY
Day Pattern	Interactive tour tracing visual	home_taz, orig_taz, dest_taz, depart_hour, trip_mode_name, dest_purpose
Tree Map	Interactive tree map visual	MAINGROUP, SUBGROUP, QUANTITY
Radar Charts	Interactive radar charts visual	AXIS, CHART, QUANTITY
Bar Chart	Interactive bar chart visual	BARGROUP, COLUMNS, QUANTITY
Line Chart	Interactive bar chart visual	XVAL, QUANTITY,

		[PIEGROUP],[QUANTITY2]

SERPMVIZ can pivot query results as well. Pivoting a table means the unique values in one column are pivoted to become columns themselves. For example, if a query returns the number of trips by ZONE and MODE, then the query results can be pivoted so the zones remain as the rows and the unique modes become columns. Results of a query will be pivoted if one of the field names is labeled COLUMNS, as shown in the example below:

SELECT ZONE, MODE, QUANTITY FROM TRIPS

can be rewritten as:

SELECT ZONE, MODE AS COLUMNS, QUANTITY FROM TRIPS

to get a column for each unique MODE. This is used quite often in the default queries since many of the visuals draw a bar for each column.

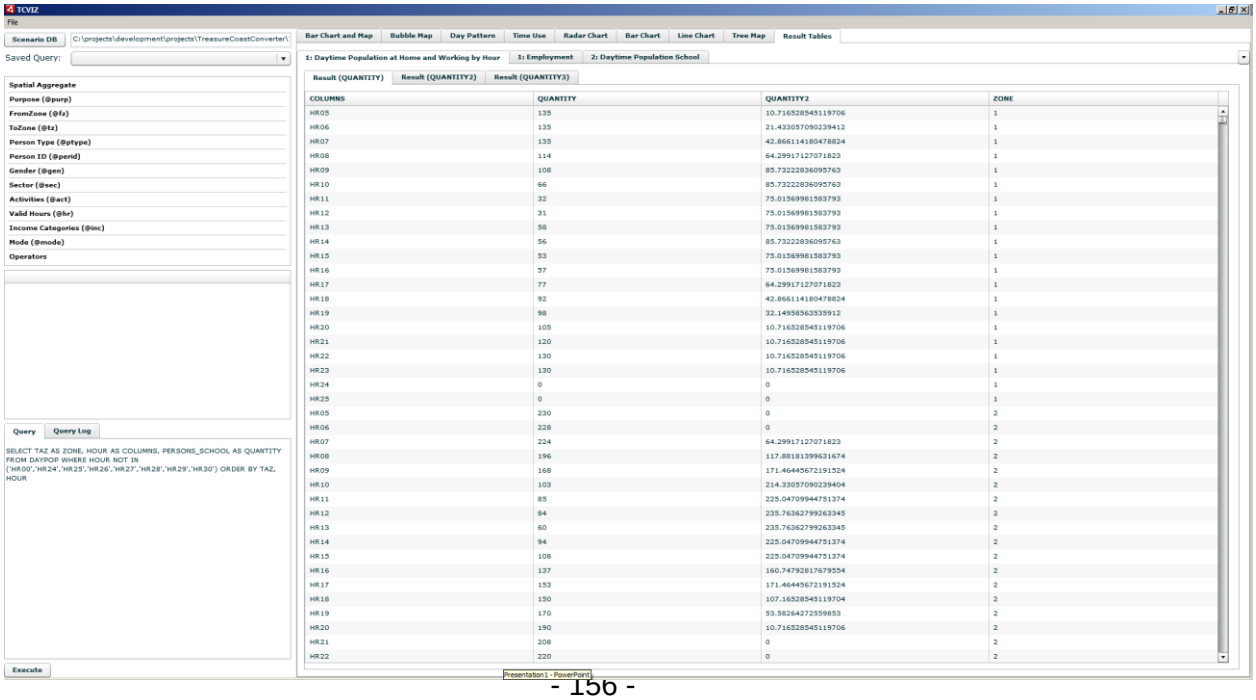
To specify and execute a pivot query, do the following:

a. Repeat the steps above for executing a saved query but ensure the query result contains a field labeled as COLUMNS.

Tables

As shown in Figure 16, the Tables tab allows the user to review and export the query results in table form. The results of each query will be shown, with a corresponding title (based on the query name in the Saved Query dropdown). If the query contained additional "QUANTITY" terms, i.e. "QUANTITY2" and "QUANTITY3", and these were not aggregated into one table, their results will be shown in corresponding tabs.

Figure 16 – Tables Tab



To copy the current table to the Windows clipboard, do the following:

1) Go to File menu + Copy Table menu item

To save all tables to an Excel XML spreadsheet file, do the following:

2) Go to File menu + Save Tables menu item

Bar Chart and Map

The Bar Chart and Map tab contains a series of interactive visuals related to spatial information. As shown in Figure 17, the bar chart displays grouped bars, with each group corresponding to a spatial aggregator, such as county, and each bar corresponding to the QUANTITIY field or a COLUMNS pivoted field, such as trip mode. A query will populate the Bar Chart and Map tab if it has the following table signature:

ZONE, QUANTITY or

ZONE, COLUMNS, QUANTITY

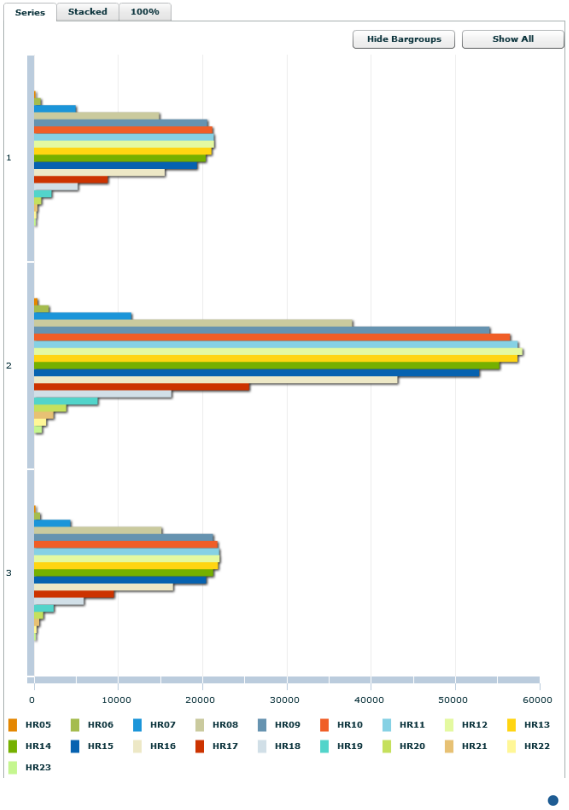
When the query is returned from the database, the user selected spatial aggregator field is joined based on the ZONE field and both the Query Result and Spatial Aggregation tabs are populated on the Tables tab. The bar chart is drawn based on the information in the Spatial Aggregation tab. The bar chart controls are:

1) Series/Stacked – changes the chart to show each of the "COLUMNS" as a separate bar (series) or as a proportion of the entire group (@spag).

2) Bars - mousing over a bar highlights the spatial aggregation area on the map and shows the value of the selected bar.

3) File menu + Save Visuals - exports the plots to PNG format.

Figure 17 – Bar Chart and Map – Bar Chart



In addition to the bar chart, the Bar Chart and Map tab contains a zone map for the display of spatial information. The map is based on the MapQuest Open Street Map tiles specific to the area and can display a chloropleth based on the zone shape data provided (see Figure 18). The map is visualized based on the same data as the bar chart. Each of the map visualization types are explained below. Common controls to all the map visuals are:

a. Field drop down selector – specifies which field to draw zonal-based results

for.

b. Play button – steps through the fields in the fields drop down selector, redrawing the map each time.

c. Play interval – specifies the interval in seconds to step through the fields.

d. Cache gifs checkbox – If selected, the map image is cached every time the map changes. The image cache is reset when the checkbox is unchecked.

e. File menu + Save Animated Gif menu item – Saves all the images in the cache to an animated gif.

f. Mouse wheel or the + and – buttons – zoom in and out of the map. Click and drag or arrow buttons pan the map. The map can also be switched to satellite imagery.

g. Double-clicking a zone on the map – brings up the "Use zone as" dialog to substitute the clicked zone number or all the zones in the zone's spatial aggregation into the From Zone and/or To Zone selectors on the Scenario and Query tab.

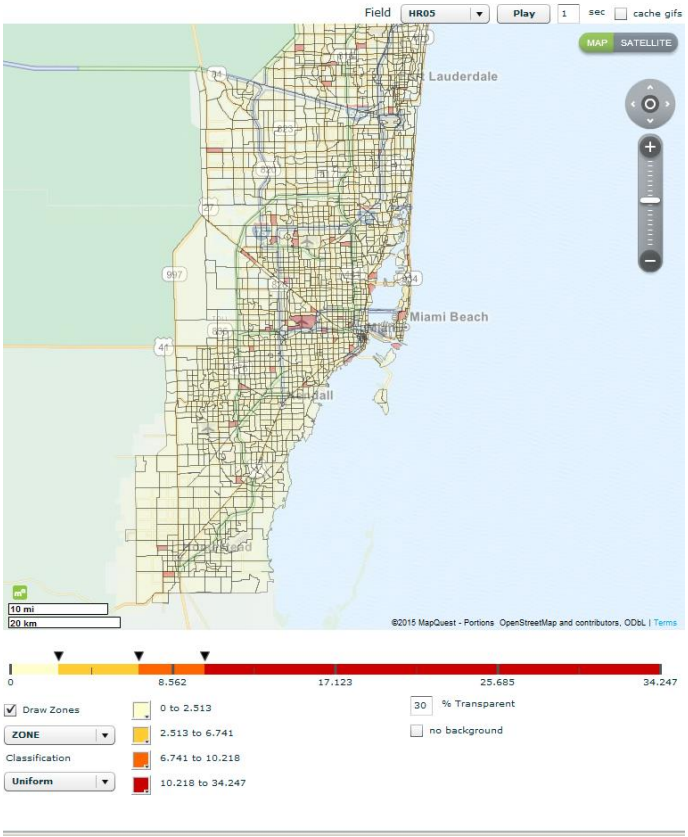
The Map (Zones) - Choropleth map visual (Figure 18) has the following controls:

a. Draw Zones as drop down selector – draw zones according to zone or spatial aggregation value.

b. Classification drop down selector – change the classification scheme to either Percentile (quantiles) or Uniform (equal interval).

c. Classification Slider – define custom binds by moving the sliders. The classification scheme

Figure 18 – Bar Chart and Map – Map (Zones)



is not recalculated
after every query.

d. Color Selectors – select the color for the classes and zone border

e. % Transparent text input – determines how transparent the overlaying cloropleth zones appear on the map layer

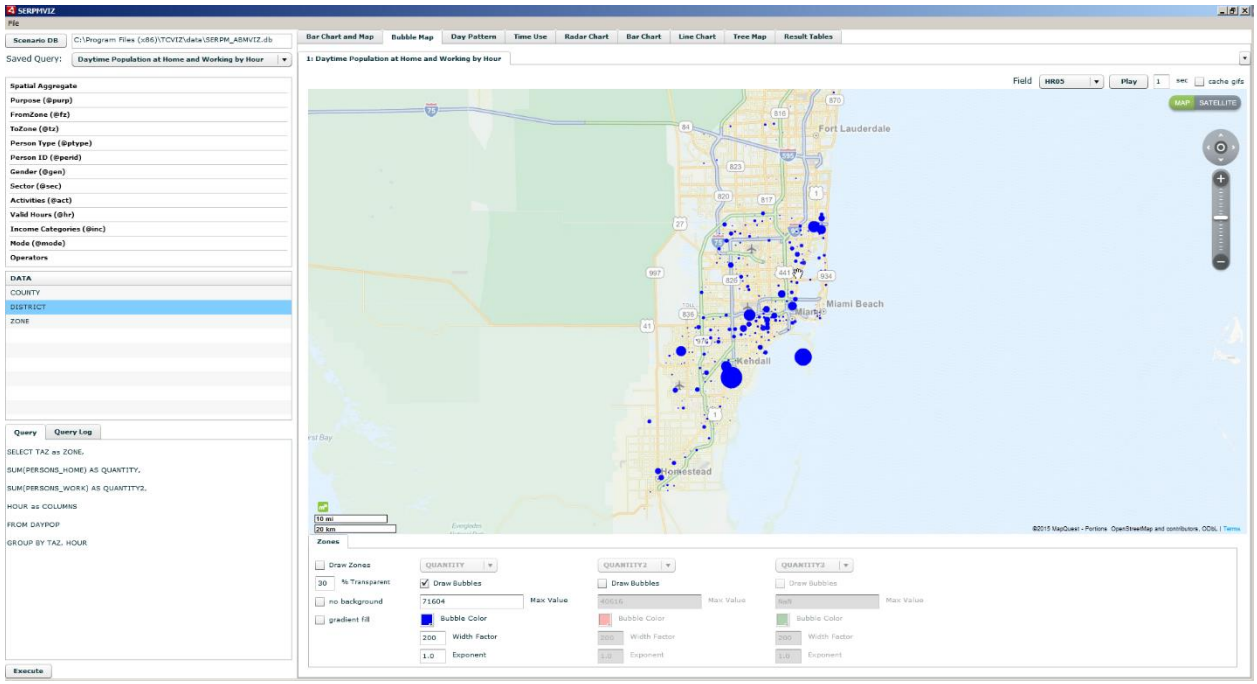
f. No background check box – if selected, the MapQuest tiles will not appear, leaving a plain white background.

Bubble Map

Figure 19 displays the Bubble Map tab. This tab is capable of displaying zone-based data as filled circles (bubbles) of various shape and color. For example, Figure 19 below shows the quantity of trips being made by three population sectors at a given hour of the day. Larger bubbles imply greater quantities of trips, while the color coding shows that there is some significant clustering of population types occurring. Up to three different groups can be plotted, designated as QUANTITY, QUANTITY2 and QUANTITY3. This visualization will also be populated with a Bar Chart and Map query, as the signatures are compatible. A query will populate the Bubble Map tab if it has the following table signature:

ZONE, QUANTITY (same as Bar Chart and Map)
or
ZONE, QUANTITY, COLUMNS (same as Bar Chart and Map)
or
ZONE, QUANTITY, QUANTITY2, COLUMNS
or
ZONE, QUANTITY, QUANTITY2, QUANTITY3, COLUMNS

Figure 19 - Bubble Map



Common controls to all the map visuals are:

a. Field drop down selector – specifies which field to draw zonal-based results for.

b. Play button – steps through the fields in the fields drop down selector, redrawing the map each time.

c. Play interval – specifies the interval in seconds to step through the fields.

d. Cache gifs checkbox – If selected, the map image is cached every time the map changes. The image cache is reset when the checkbox is unchecked.

e. File menu + Save Animated Gif menu item – Saves all the images in the cache to an animated gif.

f. Mouse wheel or the + and – buttons – zoom in and out of the map. Click and drag or arrow buttons pan the map. The map can also be switched to satellite imagery.

g. Draw zones checkbox – If checked, the zones are drawn

h. no background checkbox – if checked, the background MapQuest map is not displayed.

i. gradient fill textbox – selecting this will fill the bubbles with a radial gradient.

j. % Transparent text input – determines how transparent the overlaying zones appear on the map layer.

k. Draw bubbles checkbox – will display the bubbles for a given group, provided the QUANTITY field of that group is filled in in the results table.

l. QUANTITY dropdown – this can be used to assign different QUANTITY or pivot (COLUMNS) to a particular group (1,2 or 3).

m. Max Value text box – sets the normalizing value for the group. If all groups need to be scaled to one value, this should be set to the same number, otherwise it is initially populated with the maximum value of that group.

n. Bubble Color selector – chooses a color for the bubble group.

o. Width Factor text box – sets the multiplier value for the bubble scale for the group.

p. Exponent text box – sets the exponent of the scale multiplier for the group.

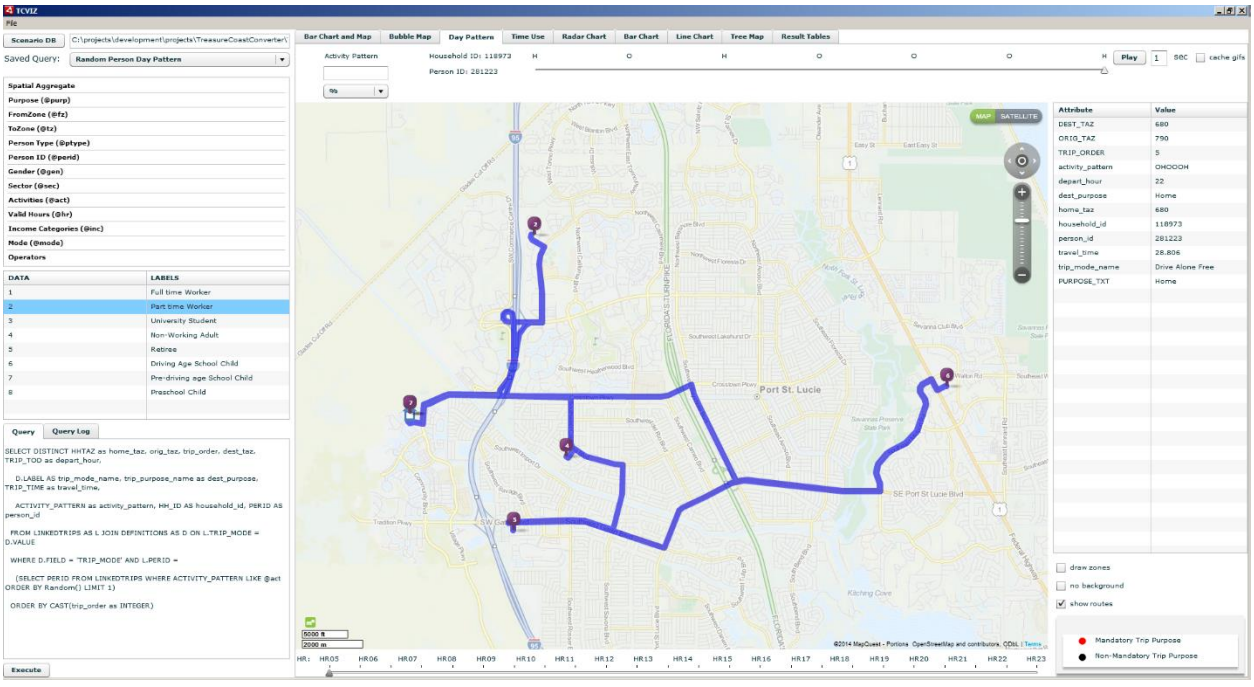
q. Editable title (on tab)

Day Pattern

As shown in Figure 20, the Day Pattern tab contains an interactive visual that traces a person's activity pattern and tour in the queried scenario. It draws on the map the location of destinations, as well as a straight line between origins and destinations. A query will populate the Day Pattern tab if it has the following table signature:

home_taz, orig_taz, dest_taz, depart_hour, trip_mode_name, dest_purpose

Figure 20 – Day Pattern



The tour tracing visual has the following controls:

a. Get Random Person button – click to get the activity pattern information for a random person in the database. The results of the query are returned to the Tables tab and the tour trace visual is setup.

b. Random person pattern dropdown and text input box – the dropdown is populated by the patterns listed in the Settings.xml file "defaultActivityPatterns" list. These ensure that the chosen random person has a particular activity pattern type. The text input box may be used to specify any pattern, valid activity types are also specified in the Settings.xml file, via the "Activities (@act)" table. In addition to the specified codes, "%" can be used as a wildcard character. The pattern matching rules are those used by SQLite.

c. Specific person dropdown Saved Query – can be filled in by replacing the "@perid" token manually, or selecting a Person ID from the query builder list, as shown in Figure 20.

d. Trip ID slider – the slider at the top moves back and forth through the person's simulated trips. When the slider changes, so does the spatial trace on the map.

e. Hour slider – the slider at the bottom moves back and forth through the hours of the day.

f. Attribute data grid – displays the person, household, tour, and trip attributes.

g. Draw zones checkbox – If checked, the zones are drawn

h. "no background" checkbox – if checked, the background MapQuest map is not displayed.

i. Legend – each destination is colored red if it is a mandatory purpose destination (such as work) or blue if it is not. Each trip is colored green if the trip mode is auto or yellow if it is not auto (such as walk or transit). The definitions for these are listed in listed in the Settings.xml file "mandatoryPurposeCodes" list.

r. Play button – steps through the trips on the day, redrawing the map each time.

s. Play interval – specifies the interval in seconds to step through the trips.

t. Cache gifs checkbox – If selected, the map image is cached every time the map changes. The image cache is reset when the checkbox is unchecked.

u. File menu + Save Animated Gif menu item – Saves all the images in the cache to an animated gif.

v. Mouse wheel or the + and – buttons – zoom in and out of the map. Click and drag or arrow buttons pan the map. The map can also be switched to satellite imagery.

w. "show routes" checkbox – using this option will display the most likely route taken by

the random individual on the MapQuest street network.

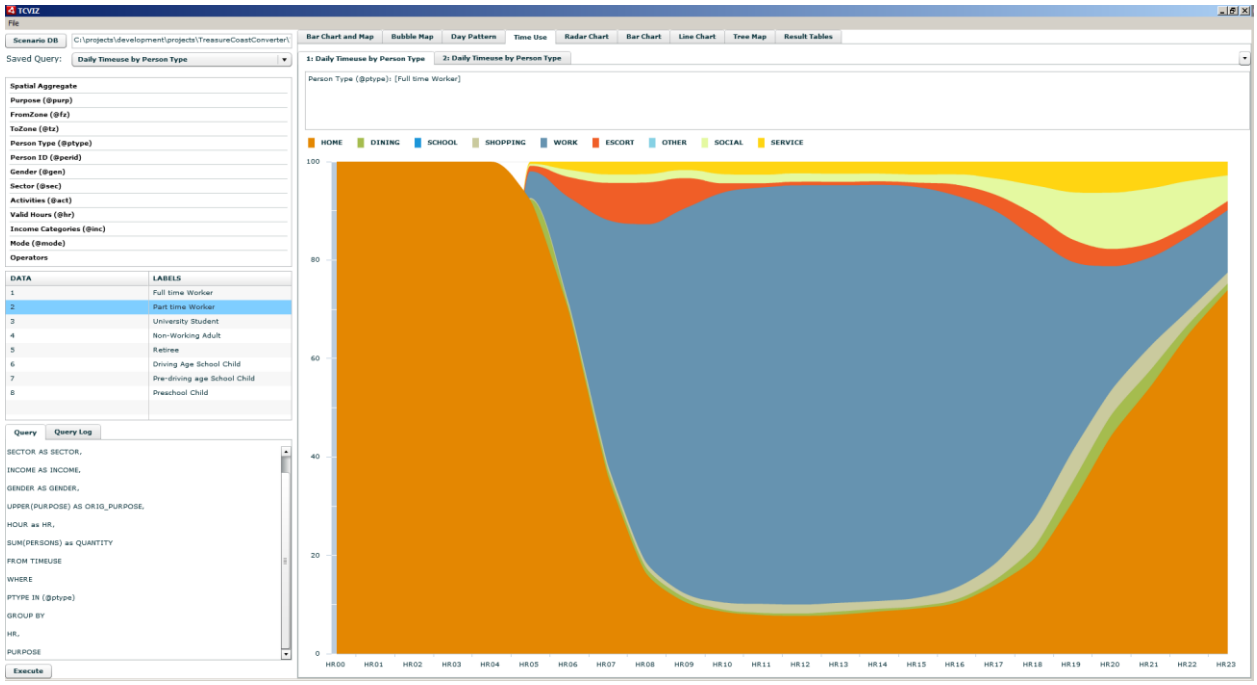
x. Mode Icons – clicking on the mode icons will reveal detail regarding the trip, such as detailed mode name, origin and destination TAZs, purpose, departure time and travel time.

Time Use

As shown in Figure 21, the Time Use tab contains an interactive visual that displays person time use by activity purpose and hour of the day. The vertical axis displays the percent of activity and the horizontal axis displays the hour of the day. The vertical axis sums to 100%. A query will populate the Time Use tab if it has the following table signature:

HR, ORIG_PURPOSE, QUANTITY

Figure 21 – Time Use



The time use visual has the following controls:

a. Editable title (on tab)

b. Mouse over the boundary of one of the area regions – displays a tooltip with the QUANTITY and percent of activity purpose for the hour of the day.

c. File menu + Save Visuals menu item – exports the visuals in PNG format.

Radar Charts

As shown in Figure 22, the Radar Charts tab visualizes a series of radar (or triangle or quadrilateral in the case of three or four axes, respectively) charts. Radar charts display multiple data series on different axes in order to compare different entities across multiple measures. For example, a zone's attractiveness could be compared by accessibility to jobs, jobs/housing balance and availability of transit. The radar charts visual in SERPMVIZ draws a series of radar charts in order to compare entities according to the user specified axes. The value (or quantity) or each axis is scaled so the max value across the charts is 1 (or 100%). Each axis is independent of one another. A query will populate the Radar Chart tab if it has the following table signature:

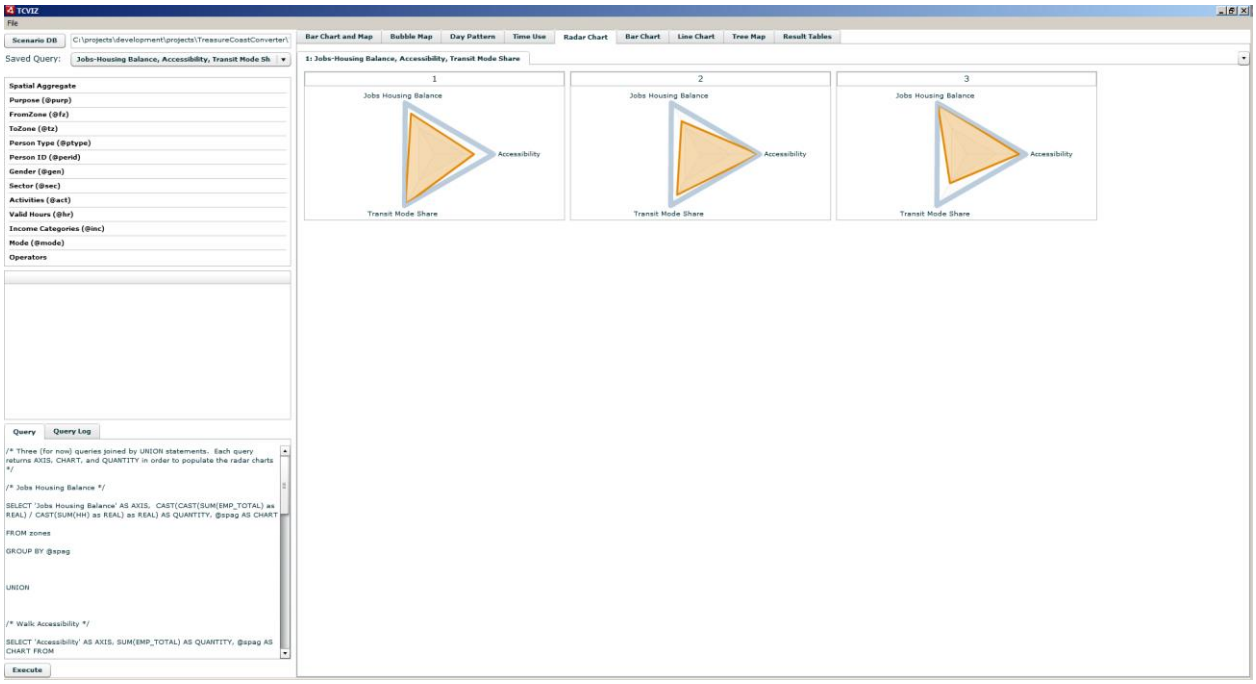
AXIS, CHART, QUANTITY

a. The AXIS field is a string and it specifies the axis label on each chart.

b. The CHART field is a string and it specifies the chart title on each chart.

c. The QUANTITIY field specifies the quantity for the record's AXIS, CHART combination.

Figure 22 – Radar Charts



The radar charts visual has the following controls:

a. Title – editable title (on tab)

b. Mouse over a chart item – displays a tooltip of the absolute and scaled quantity

c. File menu + Save Visuals menu item – exports the visuals to PNG format.

Bar Chart

As shown in Figure 23, the Bar Chart tab displays grouped bars, with each group corresponding to a BARGROUP, such as auto household auto sufficiency group, and each bar corresponding to a COLUMNS pivoted field, such as trip mode. A query will populate the Bar Chart tab if it has the following table signature:

BARGROUP, COLUMNS, QUANTITY

Figure 23 – Bar Chart (Series, Stacked, 100%)

The bar chart visual has the following controls:

a. Plot title – editable plot title (on tab)

b. Series/Stacked/100% tabs – changes the chart to show each of the "COLUMNS" as a

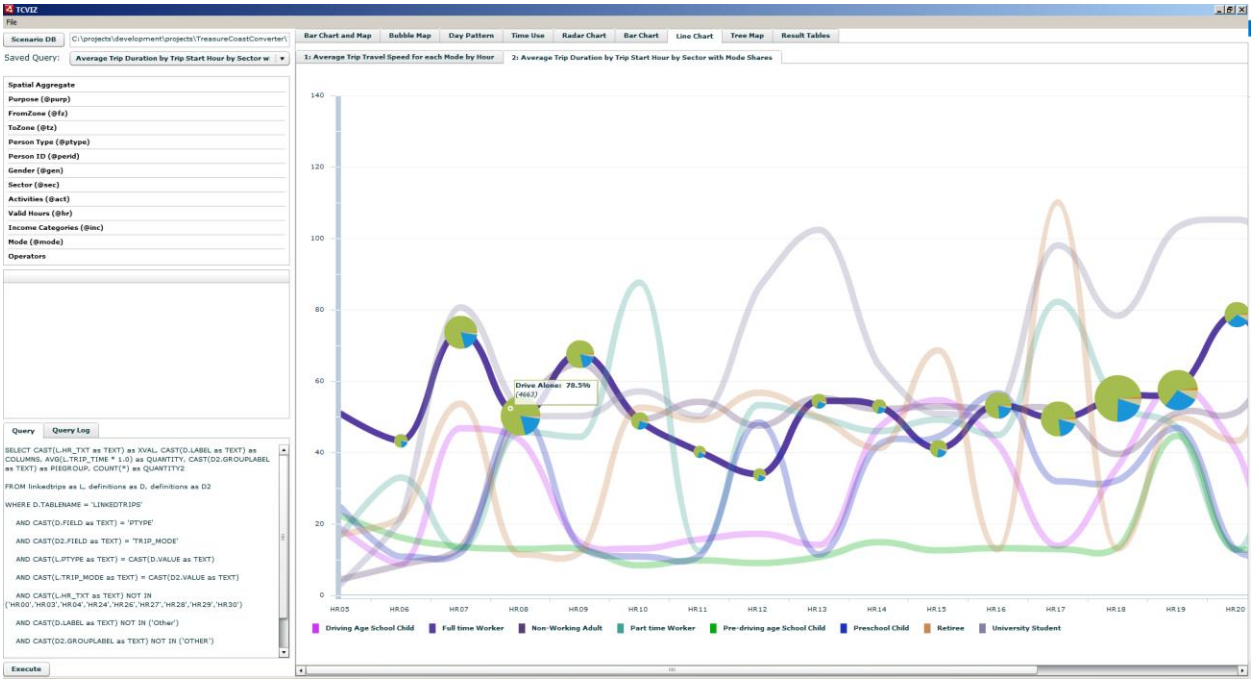
separate bar (Series) or as a total of the entire group (Stacked) or as a percentage share (100%). Figure 23 shows the same data in the three different views.

c. File menu + Save Visuals - exports the plots to PNG format.

Line ChartFigure 24 shows a result of a Line Chart query. In this query each plotted line corresponds to the COLUMNS pivot, with its QUANTITY value plotted along the XVAL variable. An additional dimension can be added by including the QUANTITY2 and PIEGROUP values to the table signatures. This will add pie charts, where the QUANTITY2 value of the particular PIGROUP is displayed at each XVAL. In the case of Figure 24, the pie charts represent the mode shares at each hour of the day. The size of the pie is proportional to the total sum of the values in QUANTITY2 at that XVAL. A query will populate the Line Chart tab if it has the following table signature:

XVAL, COLUMNS, QUANTITY
or
XVAL, COLUMNS, QUANTITY, PIEGROUP, QUANTITY2

Figure 24 – Line Chart



The line chart visual has the following controls:

a. Plot title – editable plot title (on tab)

b. Clicking on a line will highlight it and, if PIEGROUP and QUANTITY2 exist, display the appropriate pie charts along the line. Double clicking outside a line will clear the selection.

c. File menu + Save Visuals - exports the plots to PNG format

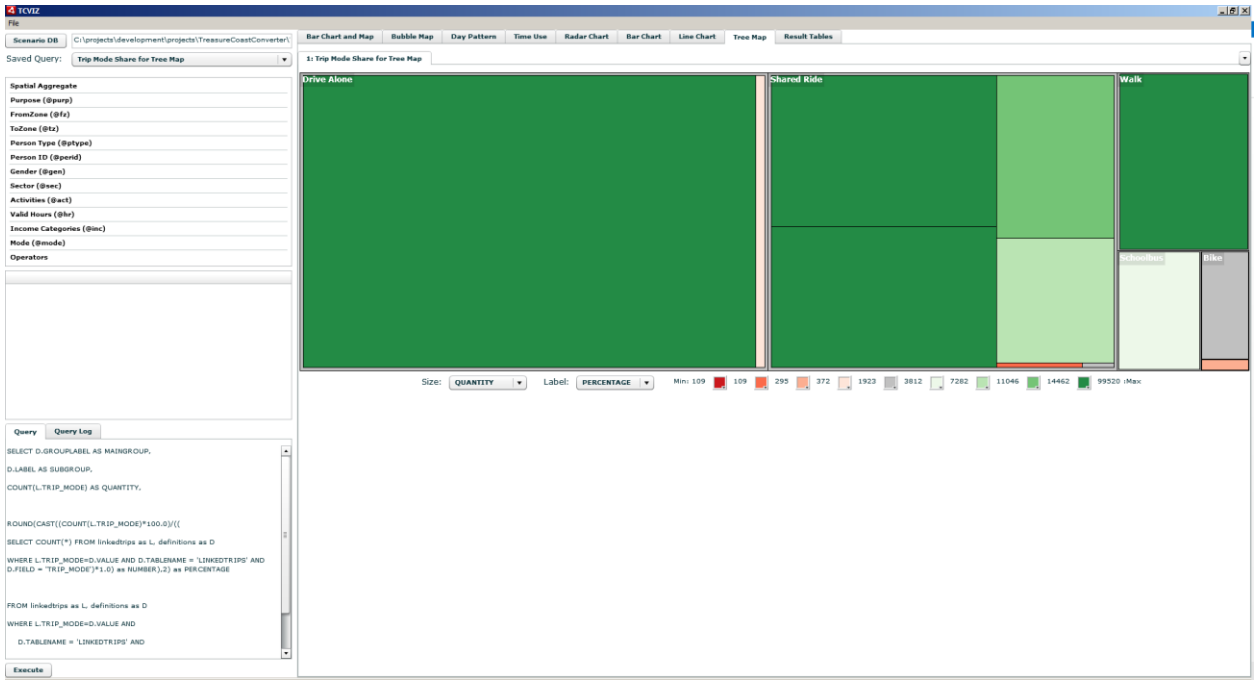
Tree Map

As shown in

Figure 25, the Tree Map tab displays a tree map, which is similar to a pie chart. A tree map shows quantities as rectangles, where each rectangle is sized according to a value. In addition, each rectangle can be colored according to a different value. A tree map is good for displaying hierarchical (nested) data. For example, a tree map can be configured to show two rectangles – one for auto and one for transit. Inside the auto rectangle is three more rectangles – one for drive alone, one for two person shared ride, and one for three plus persons shared ride. The SERPMVIZ tree map allows the user to interactively zoom in and out of the hierarchical data. A query will populate the Tree Map tab if it has the following table signature:

MAINGROUP, SUBGROUP, QUANTITY

Figure 25 – Tree Map



The tree map visual has the following controls:

a. Title – editable title (on tab)

b. Size drop down field selector – select the query field to use for the rectangle size

c. Label drop down field selector – select the query field to use for the rectangle labels.

d. Color selectors – select the colors for the color field ranges. The min, max, and range of color break values are automatically set when a color field is specified.

e. Click a rectangle – zooms to other level of the hierarchy (i.e. SUBGROUP if MAINGROUP

is current or MAINGROUP if SUBGROUP is current).

Mouse over a rectangle – displays the SUBGROUP, MAINGROUP, SIZE and LABEL values. (In

f. Figure 25 the label is the percentage share).

g. File menu + Save Visuals menu item – exports the visuals to PNG format.

File Menu

The SERPMVIZ file menu contains a number of useful functions for exporting data and loading/saving SERPMVIZ settings. The file menu items are:

1) Load Settings – loads from a *.set file a complete SERPMVIZ configuration, including query text, parameter settings, map classes, map colors, etc and execute the stored query against the current scenario and reference databases.

2) Save Settings – saves the current configuration including the query text, parameter settings, map classes, map colors, etc. to a binary *.set file. The scenario database name and reference database name are not saved to the file in order to allow the user to reuse settings across scenarios.

3) Copy Table – copies the current data table on the Tables tab to the Windows clipboard.

4) Save Tables – saves the data tables on the Tables tab to an Excel XML spreadsheet file.

5) Save Visuals – saves all the visuals to a series of PNG files.

6) Save Animated Gif – saves the cached gif images (see the Bar Chart and Maps section above) to an animated gif file using the freely-distributable ImageMagick convert program. The convert program is a command line utility that converts a series of images to an animated gif. An example command line call is:

   convert -delay 200 filename-.png out.gif*

   where:
   200 = 2 second delay between images
   filename-*.png = the file name pattern to match for input images. The files need to be
   named with a number to order the images (i.e. filename-01.png and filename-02.png.
   out.gif = the output animated gif file

7) Exit – exit the application.

4.5 Installation/Setup

The distributed SERPMVIZ package consists of a single installer executable and a supporting SQLite database. The executable installs into the **C:\Program Files (x86)\SERPMVIZ** folder by default. The user documentation can be found in this folder.

The database supplied should be unzipped and can be placed anywhere – the user will have to open it via an Open File dialog initiated by the "GetDB" button. However, the supplied settings files (which are just saved queries and SERPMVIZ settings), located in the **SERPMVIZ\queries** folder, rely on the database being here **C:\Program Files (x86)\SERPMVIZ\SERPM_ABMVIZ.db**. These are not required to run SERPMVIZ and are only meant to simplify workflow, reducing the need to re-set all the visualization parameters for each query. The included set is not meant to be exhaustive and is included primarily for example purposes.

The SERPMVIZ folder also contains the **SERPMVIZ\db_generator** folder, which contains two python scripts (buildSERPMDB.py and shapefile.py) that can be used to rebuild the SQLite database, as detailed in the Database Back-End Section. These are not required when using the supplied SQLite database and are used only to rebuild the database from supplied data tables.

In summary, the **SERPMVIZ** folder will contain:

1) This documentation file
2) **SERPMVIZ\queries** folder for saved settings files
3) **SERPMVIZ\db_generator** for DB building scripts
4) (optional) SQLite database file (SERPM_ABMVIZ.db)
5) Settings.xml file (described in Startup section)
6) DefaultQueries.xml file (described in Startup section)
7) SERPMVIZ executable
8) Other necessary files

To install and run SERPMVIZ if it has not been installed before:

1) Run the SERPMVIZ.EXE installer application
2) Choose install directory (**C:\Program Files (x86)** by default)
3) Start SERPMVIZ via Start + Programs + SERPMVIZ

To install SERPMVIZ on a machine with a prior version of SERPMVIZ:

1) Attempt to install as above. Adobe AIR should pop up an "upgrade/replace" dialog.
2) In case of a popup stating "Sorry, an error has occurred", uninstall SERPMVIZ from the Windows control panel. Delete the SERPMVIZ folder from **C:\Program Files (x86)** and re-run the executable.
3) If the above do not work, the user can attempt replace the SERPMVIZ.swf file in **C:\Program Files (x86)\SERPMVIZ**

Administrative rights and an internet connection are required to install and run SERPMVIZ.

4.6 Database Back-End

The model run analysis that is done in the visualization dashboard is made possible by the database back-end. The database back-end consists of a single SQLite database file that contains the necessary tables for the visualization component.

Build Script for Database

The SQLite database file is created via the "buildSERPMDB.py" python script. This script combines the necessary survey inputs into a single SQLite DB file. The necessary inputs are listed in Table 80.

Table 80 – Database Input Files

Filename	Description
synhh.csv	Household data file
synperson.csv	Person data file
temptrip.csv	Linked Trips data file
maz_data.csv	MAZ data
walkLOS.csv	Walk times (for accessibility calculations)
V7Zone_District_County.csv	County/District TAZ mappings
definitions.csv	Definitions table converting codes to descriptions
shapefile.py	Python file to read shapefile
SERPM_SIMPLIFIED.shp	Shapefile to provide zone geometries

The compiled database will contain the tables shown in Table 81, which can then be queried via the SERPMVIZ interface.

Table 81 - Database Tables

Table Name	Description
DAYPOP	An hourly summary of the quantity of different population sectors that are participating in an activity during the given hour
DEFINITIONS	Definitions table that defines the codings used in all other tables
HOUSEHOLDS	Contains the survey household records
LINKEDTRIPS	Contains the trips contained in the model run and linked by household and person identifiers
WALKLOS	Travel time skims for auto and bus travel times between zones
PERSONS	Contains the survey person records
SYNPERSONS	Contains the model person records
TIMEUSE	An hourly summary of the activities different population sectors are participating during the given hour
ZONES	Contains zone data for the region

4.6 References

1) Adobe Flex, http://www.adobe.com/products/flex
2) Adobe Data Visualization Library, http://livedocs.adobe.com/flex/3/datavis_flex3.pdf
3) Adobe Flash Player, http://get.adobe.com/flashplayer
4) Adobe AIR Runtime, http://get.adobe.com/air
5) IBM ILOG Elixir, http://www-03.ibm.com/software/products/en/elixir-enterprise/
6) ImageMagick, http://www.imagemagick.org
7) MapQuest.com, http://www.mapquest.com
8) SQLite, http://www.sqlite.org/