APPENDIX B

FULL MODEL SCRIPTS

- **External-External**
- **Trip Generation**
- **Highway Network**
- **Highway Paths**
- **Trip Distribution**
- **Transit Network**
- **Pre Mode**
- **Mode**
- **Assignment**
- **Evaluation**

EXTERNAL- EXTERNAL

1. EXMAT00A.S

```
; Do not change filenames or add or remove FILEI/FILEO statements using
an editor. Use Cube/Application Manager.
RUN PGM=MATRIX PRNFILE="D:\FSUTMS\D4\TCRPM\CUBE\EXT.PRN" MSG='Creates
EETRIP Table from Text File EETRIPS.{YEAR}{ALT} and Generates
FSUTMS.CTL for User-written Programs'
FILEI MATI[1] = "{DATADIR}\EETRIPS.DBF",
 PATTERN=IJ:V, FIELDS=ORZ,DSZ,TRIPS
FILEO PRINTO[1] = "D:\FSUTMS\D4\TCRPM\CUBE\FSUTMS.CTL"
FILEO MATO[1] = "{SCENARIO_DIR}\EETAB.{ALT}{YEAR}.MAT",
  MO=1 NAME=EETRIPS

par zones={zonesa}  MAXCOMPERRS=999999  zonemsg=100

IF (I=1)
PRINT
LIST="~SELECT","\n3","\n~FILEIN","\n{YEAR}{Alt}","\n~FILEOUT","\n{Alt}{
YEAR}","\n~{descr}",
      "\ntest","\n~MODE","\nHIGHWAY","\n~MODULE","\nEXT       1
RANGE","\nHEVAL     8 RANGE",
      PRINTO=1
ENDIF

MW[1]=MI.1.1
ENDRUN
```

TRIP GENERATION

1. GNPIL00A.S

```
; Do not change filenames or add or remove FILEI/FILEO statements
using an editor. Use Cube/Application Manager.
*if exist PBSCH.{ALT}{YEAR} DEL PBSCH.{ALT}{YEAR}
*if exist PBSC*.TEM DEL PBSC*.TEM
*if exist tripgen.out DEL tripgen.out
*copy {datadir}\*.{year}{alt}
*copy {datadir}\*.MAS
*copy {datadir}\*.SYN
*copy {datadir}\*.CTL
*copy {datadir}\*.ASC
*{path1}\TG1.EXE
*if exist STP60.TEM del STP60.TEM
*copy /a PBSCH.TEM PBSCH.{ALT}{YEAR}
*copy /a A1DECK.TEM A1DECK.{ALT}{YEAR}
*if exist PBSC*.TEM DEL PBSC*.TEM
*if exist HH.TEM del HH.TEM
*if exist PRODS.TEM del PRODS.TEM
*if exist ATTRS.TEM del ATTRS.TEM
*if exist AUTO.TEM del AUTO.TEM
*if exist LSTGEN.ERR copy /a GEN.OUT+LSTGEN.ERR
*if exist LSTGEN.ERR DEL LSTGEN.ERR
```

2. GNMAT00A.S

```
; Do not change filenames or add or remove FILEI/FILEO statements
using an editor. Use Cube/Application Manager.
RUN PGM=MATRIX PRNFILE="D:\FSUTMS\D4\TCRPM\CUBE\GNMAT00A.PRN"
MSG='Convert LSTGEN PRODS & ATTRS to DBF Format'
FILEI ZDATI[2] = "D:\FSUTMS\D4\TCRPM\CUBE\ATTRS.ASC",
 Z=3-7, A1=11-17, A2=18-24, A3=25-31, A4=32-38, A5=39-45, A6=46-
52, A7=53-59, A8=60-66, A9=67-73, A10=74-80, A11=91-97
FILEO RECO[1] = "{SCENARIO_DIR}\PANDA.DBF",

FIELDS=Z,P1,P2,P3,P4,P5,P6,P7,P8,P9,P10,P11,A1,A2,A3,A4,A5,A6,A7,
A8,A9,A10,A11
FILEI ZDATI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\PRODS.ASC",
 Z=3-7, P1=11-17, P2=18-24, P3=25-31, P4=32-38, P5=39-45, P6=46-
52, P7=53-59, P8=60-66, P9=67-73, P10=74-80, P11=91-97
PAR ZONES={ZONESA} ZONEMSG=100

P1=ZI.1.P1,P2=ZI.1.P2,P3=ZI.1.P3,P4=ZI.1.P4,P5=ZI.1.P5,P6=ZI.1.P6
,P7=ZI.1.P7,P8=ZI.1.P8,P9=ZI.1.P9,P10=ZI.1.P10,P11=ZI.1.P11

A1=ZI.2.A1,A2=ZI.2.A2,A3=ZI.2.A3,A4=ZI.2.A4,A5=ZI.2.A5,A6=ZI.2.A6
,A7=ZI.2.A7,A8=ZI.2.A8,A9=ZI.2.A9,A10=ZI.2.A10,A11=ZI.2.A11

; PRP1=HBW, PRP2=HBSHP, PRP3=HBSR, PRP4=HBSCH-PR, PRP5=HBO,
PRP6=NHBW,
; PRP7=NHBO, PRP8=ARPRT, PRP9=TRK-4TIRE,PRP10=TRK-SU, PRP11=TRK-
COMB
  hbw=hbw+p1
```

```
  hbsh=hbsh+p2
  hbsr=hbsr+p3
  hbscprv=hbscprv+p4
  hbo=hbo+p5
  nhbw=nhbw+p6
  nhbo=nhbo+p7
  airport=airport+p8
  trk4tire=trk4tire+p9
  trksingle=trksingle+p10
  trkcombo=trkcombo+p11
  PSUM=P1+P2+P3+P4+P5+P6+P7+P8+P9+P10+P11
  ASUM=A1+A2+A3+A4+A5+A6+A7+A8+A9+A10+A11
IF (PSUM>0|ASUM>0) WRITE RECO=1
ENDRUN
```

3. GNNET00A.S

```
; Do not change filenames or add or remove FILEI/FILEO statements
using an editor. Use Cube/Application Manager.
RUN PGM=NETWORK
FILEO PRINTO[1] = "D:\FSUTMS\D4\TCRPM\CUBE\FTAZAT.ASC"
FILEI LINKI[1] = "{hnet}"
ARRAY ATYPE={ZONESI}


PHASE=LINKMERGE
  IF(LI.1.A < {ZONESI})

    IF(ATYPE[LI.1.A]=0)
;    PRINT LIST= LI.1.A,LI.1.AREA_TYPE CSV=T PRINTO=1
     ATYPE[LI.1.A]=LI.1.AREA_TYPE
     PRINT LIST=LI.1.A(3.0),ATYPE[LI.1.A](2.0) CSV=T PRINTO=1

    ENDIF
;    atype=LI.1.AREA_TYPE
  ENDIF


;LOOP II=1,{ZONESI}
; PRINT LIST=II,ATYPE[II] CSV=T PRINTO=1
;ENDLOOP
ENDRUN
```

HIGHWAY NETWORK

1. NTMAT00E.S

```
; Do not change filenames or add or remove FILEI/FILEO statements
using an editor. Use Cube/Application Manager.
RUN PGM=MATRIX PRNFILE="D:\FSUTMS\D4\TCRPM\CUBE\NTMAT00B.PRN"
MSG='Create SPDCAP Lookup, Modified from TC Export'
FILEI RECI = "D:\FSUTMS\D4\TCRPM\INPUT.00T\SPDCAP.ASC"
FILEO PRINTO[2] = "D:\FSUTMS\D4\TCRPM\CUBE\SPDCAP.ERR"
FILEO PRINTO[1] = "D:\FSUTMS\D4\TCRPM\CUBE\SPDCAP.CSV"

ARRAY SPDLOOKUP=999999 CAPLOOKUP=999999
LAT=SUBSTR(RECI,1,2)
HAT=SUBSTR(RECI,1,2)
LFT=SUBSTR(RECI,3,2)
HFT=SUBSTR(RECI,3,2)
LLN=SUBSTR(RECI,5,2)
HLN=SUBSTR(RECI,5,2)
CAP=SUBSTR(RECI,7,8)
SPD=SUBSTR(RECI,15,6)
    LATVAL=VAL(LAT)
    HATVAL=VAL(HAT)
    LFTVAL=VAL(LFT)
    HFTVAL=VAL(HFT)
    LLNVAL=VAL(LLN)
    HLNVAL=VAL(HLN)
    CAPVAL=VAL(CAP)
    SPDVAL=VAL(SPD)
; PLACE INITIAL CAPACITIES & SPEEDS INTO AN ARRAY
    LOOP ATYPE=LATVAL,HATVAL
      LOOP FTYPE=LFTVAL,HFTVAL
        LOOP LANES=LLNVAL,HLNVAL
        INDEXVAL=ATYPE*10000+FTYPE*100+LANES
        CAPLOOKUP[INDEXVAL]=CAPVAL
        SPDLOOKUP[INDEXVAL]=SPDVAL
        ENDLOOP
      ENDLOOP
    ENDLOOP

IF (I=0)
  PRINT LIST='SPEED OR CAPACITY ERRORS WHERE THE SPDCAP RESULT IS
LESS THAN ZERO'
  LOOP IVAL=1,999999
     IF (CAPLOOKUP[IVAL]>0|SPDLOOKUP[IVAL]>0)   PRINT CSV=T,
LIST=IVAL(6.0),CAPLOOKUP[IVAL],SPDLOOKUP[IVAL],PRINTO=1
     IF (CAPLOOKUP[IVAL]<0)
        CAPERRCNT=CAPERRCNT+1
        PRINT CSV=T, LIST='SPDCAP ERROR FOR ATFTLN=',IVAL(6.0),'
CAPACITY=',CAPLOOKUP[IVAL](9.2),PRINTO=2
     ENDIF
     IF (SPDLOOKUP[IVAL]<0)
        SPDERRCNT=SPDERRCNT+1
        PRINT CSV=T, LIST='SPDCAP ERROR FOR ATFTLN=',IVAL(6.0),'
SPEED=',SPDLOOKUP[IVAL](9.2),PRINTO=2
     ENDIF
  ENDLOOP
```

```
  PRINT LIST='\n************Error Report Summary*************',
             '\nTOTAL LESS THAN ZERO CAPACITY
ERRORS=',CAPERRCNT(8.0C),
             '\nTOTAL LESS THAN ZERO SPEED ERRORS
=',SPDERRCNT(8.0C), printo=2
ENDIF
  if (caperrcnt>0&{spdcapabort,CheckBox,"Abort on SPDCAP
Errors",0}=1) abort
  if (spderrcnt>0&{spdcapabort}=1) abort


ENDRUN
```

2. NTMAT00C.S

```
; Do not change filenames or add or remove FILEI/FILEO statements
using an editor. Use Cube/Application Manager.
RUN PGM=MATRIX PRNFILE="D:\FSUTMS\D4\TCRPM\CUBE\NTMAT00E.PRN"
MSG='PROCESS VFACTORS into a LOOKUP'
FILEO PRINTO[1] = "D:\FSUTMS\D4\TCRPM\CUBE\VFACTORS.CSV"
FILEI RECI = "{DATADIR}\VFACTORS.{year}{alt}",_FT=6-7, _UROAD=18-
24, _CONFAC=35-41, _BPRCOEF=53-59, _BPREXP=71-77

; fld=1-77
;_FT=6-7, _UROAD=18-24, _CONFAC=35-41, _BPRCOEF=53-59,
_BPREXP=71-77

;FT=SUBSTR(RECI,6,2)
;UROAD=SUBSTR(RECI,18,7)
;CONFAC=SUBSTR(RECI,35,7)
;BPRCOEF=SUBSTR(RECI,53,7)
;BPREXP=SUBSTR(RECI,71,7)

;PRINT CSV=T,
LIST=FT(2.0),UROAD(7.4),CONFAC(7.4),BPRCOEF(7.4),BPREXP(7.4),
PRINTO=1

PRINT CSV=T
LIST=RI._FT(2.0),RI._UROAD(7.4),RI._CONFAC(7.4),RI._BPRCOEF(7.4),
RI._BPREXP(7.4), PRINTO=1


ENDRUN
```

3. NTNET00C.S

```
; Do not change filenames or add or remove FILEI/FILEO statements
using an editor. Use Cube/Application Manager.
RUN PGM=NETWORK PRNFILE="D:\FSUTMS\D4\TCRPM\CUBE\NTNET00C.PRN"
FILEI LOOKUPI[2] = "D:\FSUTMS\D4\TCRPM\CUBE\SPDCAP.CSV"
FILEI LOOKUPI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\VFACTORS.CSV"
FILEO PRINTO[2] = "D:\FSUTMS\D4\TCRPM\CUBE\NODE DATA.PRN"
FILEO PRINTO[1] = "D:\FSUTMS\D4\TCRPM\CUBE\NETERRORS.PRN"
FILEO NETO = "D:\FSUTMS\D4\TCRPM\CUBE\BASE.NET",
 exclude=uf,cf,bprlos,bprexp
```

```
FILEI LINKI[1] = "{hnet}"
PAR  ZONES={zonesa} LIST_ERRS=0 MAX_IP_ERRS=999999

PHASE=LINKMERGE
  if (ttime>0&tspeed=0) tspeed=60*tdist/ttime
  if (ttime=0&tspeed>0) ttime=60*tdist/tspeed
  if (distance=0)
     DISTANCE= SQRT((a.x-b.x)^2+(a.y-b.y)^2)/{units}
  ENDIF

/*
;  READ,
;        FILE = "C:\FSUTMS\D4\TCRPM\CUBE\SPDCAP.TXT"
;  IF (STCARD='S'&TIME>0)
;     SPEED=TIME
;     TIME=0
;  endif
;  IF (STCARD='S'&TIME2>0)
;     SPEED2=TIME2
;     TIME2=0
;  ENDIF
*/
  LOOKUP NAME=VFACTORS,
      LOOKUP[1]=1, RESULT=2, ; UROAD BY FT
      LOOKUP[2]=1, RESULT=3, ; CONFAC BY FT
      LOOKUP[3]=1, RESULT=4, ; BPR COEFFICIENT
      LOOKUP[4]=1, RESULT=5, ; BPR EXPONENT
      INTERPOLATE=N,lookupi=1
  LOOKUP, NAME=SPDCAP,
      LOOKUP[1]=1, RESULT=2,
      LOOKUP[2]=1, RESULT=3,
      INTERPOLATE=N, LOOKUPI=2
  NUM_LANES=LANES
  UROADFACTOR=VFACTORS(1,FACILITY_TYPE)
  CONFAC=VFACTORS(2,FACILITY_TYPE)
  BPRCOEFFICIENT=VFACTORS(3,FACILITY_TYPE)
  BPREXPONENT=VFACTORS(4,FACILITY_TYPE)
; PUT SPEEDS AND CAPACITIES ON NETWORK
  _INDEXVAL=10000*AREA_TYPE+100*FACILITY_TYPE+NUM_LANES
  CAP_HR=SPDCAP(1,_INDEXVAL) * NUM_LANES
  SPEED=SPDCAP(2,_INDEXVAL)
  DAILYCAP=CAP_HR*UROADFACTOR/CONFAC
  IF(speed>0)
     TIME=DISTANCE/SPEED*60
  ELSE
     SPEED=60*DISTANCE/TIME
  ENDIF
;  IF(speed2>0)
;     TIME2=DISTANCE/SPEED2*60
;  ELSE
;     SPEED2=SPEED
;     TIME2=DISTANCE/SPEED2*60
;  ENDIF

;  IF (FACILITY_TYPE=0) FACILITY_TYPE=49

   IF (DISTANCE=0) DISTANCE=TDIST
```

```
   IF (TIME=0) TIME=99
;  IF (FACILITY_TYPE=0) FACILITY_TYPE=49
;  IF (AREA_TYPE=0) AREA_TYPE=31

; THIS ADDED TO REPORT LINKS WITHOUT SPEEDS OR CAPACITIES
 ; if (speed=0|cap_hr=0)
  ;  _STOPMODEL=1
  ;  _AT=AREA_TYPE
  ;  _FT=FACILITY_TYPE
  ;
PRINT,LIST="A/B/AT/FT/LN/SPEED/CAP",A(6.0),B(6.0),AREA_TYPE(4.0),
FACILITY_TYPE(4.0),NUM_LANES(3.0),SPEED(6.2),DAILYCAP(7.0c)
PRINTO=1
  ;ENDIF
;LOG VAR=_STOPMODEL


ENDRUN
```

4. NTNET00D.S

```
; Do not change filenames or add or remove FILEI/FILEO statements
using an editor. Use Cube/Application Manager.
RUN PGM=NETWORK MSG='Flags Toll Accel/Decel Nodes'
FILEO PRINTO[1] = "D:\FSUTMS\D4\TCRPM\CUBE\TEMPNODE.DAT"
FILEI LINKI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\BASE.NET"


IF (A=1001) PRINT LIST=A,CARTOLL,CARTOLL, PRINTO=1
if (li.1.cartoll>0&LI.1.TOLLTYPE=1)
  a_accel=0
  a_decel=1
  b_accel=1
  b_decel=0
  PRINT form=0, LIST=A,' ',a_accel,' ',a_decel,printo=1
  PRINT form=0, LIST='NODE= ',A,' ACCEL_NODE= ',a_accel,'
DECEL_NODE= ',a_decel
  PRINT form=0, LIST=B,' ',b_accel,' ',b_decel,printo=1
  PRINT form=0, LIST='NODE= ',B,' ACCEL_NODE= ',b_accel,'
DECEL_NODE= ',b_decel
endif


ENDRUN
```

5. NTNET00D.S

```
; Do not change filenames or add or remove FILEI/FILEO statements
using an editor. Use Cube/Application Manager.
RUN PGM=NETWORK PRNFILE="D:\FSUTMS\D4\TCRPM\CUBE\NTNET00D.PRN"
MSG='Assign Accel/Decel Attributes to Links'
FILEO NETO = "D:\FSUTMS\D4\TCRPM\CUBE\TMPNET2.NET",
exclude=_toll_dec,_toll_acc
FILEI LINKI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\BASE.NET"
FILEI NODEI[2] = "D:\FSUTMS\D4\TCRPM\CUBE\TEMPNODE.DAT",
 VAR=N,_TOLL_ACC,_TOLL_DEC
```

```
PROCESS  PHASE=INPUT, FILEI=LI.1
  TOLL_ACC=0
  TOLL_DEC=0
ENDPROCESS

PROCESS PHASE=LINKMERGE
  IF(B._TOLL_DEC=1) TOLL_DEC=1
  IF(A._TOLL_ACC=1) TOLL_ACC=1
ENDPROCESS

ENDRUN
```

6. NTNET00F.S

```
; Do not change filenames or add or remove FILEI/FILEO statements
using an editor. Use Cube/Application Manager.
RUN PGM=NETWORK
FILEO PRINTO[1] = "D:\FSUTMS\D4\TCRPM\CUBE\SEGDATA.PRN"
FILEO NETO = "D:\FSUTMS\D4\TCRPM\CUBE\SIGNET.NET"
FILEO NODEO = "D:\FSUTMS\D4\TCRPM\CUBE\SIGS.DBF"
FILEI LINKI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\TMPNET2.NET"

ARRAY SIGS=300,segd=300,FT1=300
Phase=LinkMerge
  if(li.1.DISTANCE==0)
    Delete
  else
    ; SEGMENTS Calculated by KDK using TransCAD segment tool
    SEGID=li.1.SEGID
    _FTONE=INT(li.1.FACILITY_TYPE/10)

    if(TWOWAY==1)
      _d=DISTANCE*0.5
      _sad=0.5
    else
      _d=DISTANCE
      _sad=1.0
    endif
    segd[SEGID] = segd[SEGID]+_d
    if(b.SIGNAL<>0) SIGS[SEGID]=SIGS[SEGID]+_sad
    FT1[SEGID] = _FTONE
; Remark: Handle option of FT=1,7,9 in the program    if
(FT1[SEGID] == 9) FT1[SEGID]=3
  endif
endphase
PHASE=SUMMARY

 loop _k=1,300
      if(segd[_k]>0) print
list=_K(6.0),segd[_k](6.2),SIGS[_k](6.2),FT1[_k](4.0) PRINTO=1
 endloop
endphase
ENDRUN
```

7. NTMAT00F.S

```
; Do not change filenames or add or remove FILEI/FILEO statements
using an editor. Use Cube/Application Manager.
RUN PGM=MATRIX
FILEI LOOKUPI[1] =
"D:\FSUTMS\D4\TCRPM\INPUT.00T\SPEC_SEGDATA.CSV"
FILEO RECO[1] = "D:\FSUTMS\D4\TCRPM\CUBE\SEGDATA1.DBF",
 FIELDS=segid(8.0),length(4.2),numsig(6.2),ft1(8.0),
cycle(8.0),GC(4.2),df(4.2)
FILEI RECI = "D:\FSUTMS\D4\TCRPM\CUBE\SEGDATA.PRN"

  LOOKUP  LOOKUPI=1 NAME=SPECIAL,
          LOOKUP[1]=1, RESULT=2,  ; CYCLE
          LOOKUP[2]=1, RESULT=3,  ; GC
          LOOKUP[3]=1, RESULT=4,   ; DF
          FAIL=0,0,
          INTERPOLATE =N

   segid1=SUBSTR(RECI,3,6)
   length1=SUBSTR(RECI,7,6)
   numsigs1=SUBSTR(RECI,13,6)
   fac_type1=SUBSTR(RECI,19,4)


  segid=val(segid1)
  length=val(length1)
  numsig=val(numsigs1)
  ft1=val(fac_type1)

  CYCLE=SPECIAL(1,segid)
  GC=SPECIAL(2,segid)
  DF=SPECIAL(3,segid)

WRITE RECO=1

ENDRUN
```

8. NTNET00G.S

```
;;<<PROCESS TEMPLATE>><<NETWORK>>;;
;{note1,note,10,"Calculate and Report Node Attribute for Number
of Intersecting Links"}  <Count Intersecting Links >>>
;Input  Highway Network File 1: {neti1,filename,"Enter Input
Highway Network File
Name",x,"D:\FSUTMS\D4\TCRPM\CUBE\TMPNET2.NET","Network File
(*.net)|*.net"}
;Output Count Report:   {prto1,filename,"Enter Output Count
Report",x,"D:\FSUTMS\D4\TCRPM\INPUT.00T\NODELINKS","Report File
(*.prn)|*.prn"}
;;<<End Parameters>>;;
; Do not change filenames or add or remove FILEI/FILEO statements
using an editor. Use Cube/Application Manager.
RUN PGM=NETWORK
FILEO PRINTO[2] = "D:\FSUTMS\D4\TCRPM\CUBE\LASTLINKNODES.PRN"
FILEO NETO = "D:\FSUTMS\D4\TCRPM\CUBE\NETWITHSIG.NET"
```

```
FILEI NODEI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\SIGS.DBF"
FILEO PRINTO[1] = {PRTO1.Q}
FILEI LINKI[1] = {NETI1.Q}

array _nc=99999,_FT11=99999,_FT12=99999,_FT13=99999,_FT14=99999,
_CL=99999, _LASTLINK=99999

phase=linkmerge
  _ANODE=LI.1.A
  _BNODE=LI.1.B
  _FT1=INT(LI.1.FACILITY_TYPE/10)
  _CLEN= B.CYC_LEN
  _sig=B.SIGNAL
;  IF(_SEGID<>0)
    IF(B.SIGNAL<>0)
      _NC[_BNODE]=_NC[_BNODE]+1
      _CL[_BNODE]=_CLEN
     IF( _NC[_BNODE]=1)
       _FT11[_BNODE]= _FT1
     ELSEIF (_NC[_BNODE]=2)
       _FT12[_BNODE]= _FT1
     ELSEIF (_NC[_BNODE]=3)
       _FT13[_BNODE]= _FT1
     ELSE
       _FT14[_BNODE]= _FT1
     ENDIF

          IF(_BNODE>_MAXNODE) _MAXNODE=_BNODE
    ENDIF

   IF(LI.1.SEGID<>0) _LASTLINK[_BNODE]= _LASTLINK[_BNODE]+1

;  ENDIF

ENDPHASE

PHASE=SUMMARY
  LOOP _NCNT=1,_MAXNODE
      IF (_NC[_NCNT]>0) PRINT LIST=_NCNT,_NC[_NCNT],
_FT11[_NCNT], _FT12[_NCNT],_FT13[_NCNT],_FT14[_NCNT],_CL[_NCNT],
PRINTO=1
      IF (_LASTLINK[_NCNT]=1) PRINT
LIST=_NCNT,_LASTLINK[_NCNT],PRINTO=2
  ENDLOOP
ENDPHASE

ENDRUN
```

9. NTNET00H.S

```
; Do not change filenames or add or remove FILEI/FILEO statements
using an editor. Use Cube/Application Manager.
RUN PGM=NETWORK
FILEO NETO = "D:\FSUTMS\D4\TCRPM\INPUT.00T\NETSIG1.NET"
FILEI LOOKUPI[2] =
"D:\FSUTMS\D4\TCRPM\INPUT.00T\DEFAULT_SEGMENT.CSV"
```

```
FILEI LOOKUPI[1] = "D:\FSUTMS\D4\TCRPM\INPUT.00T\NODELINKS"
FILEI LINKI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\TMPNET2.NET"

array _nc=99999

  LOOKUP  LOOKUPI=1 LIST=Y NAME=NODEDATA,
          LOOKUP[1]=1, RESULT=2,  ; NUMBER OF LINKS CONNECTED TO
THE NODE
          LOOKUP[2]=1, RESULT=3,  ; FACILITY TYPE FOR 1ST LINK
          LOOKUP[3]=1, RESULT=4,  ; FACILITY TYPE FOR 2ND LINK
          LOOKUP[4]=1, RESULT=5,  ; FACILITY TYPE FOR 3RD LINK
          LOOKUP[5]=1, RESULT=6,  ; FACILITY TYPE FOR 4TH LINK
          LOOKUP[6]=1, RESULT=7,  ; SIGNAL CYCLE LENGTH
          FAIL=0,0,
          INTERPOLATE =N

 LOOKUP  LOOKUPI=2 LIST=Y NAME=DEFAULT,
          LOOKUP[1]=1, RESULT=2,  ; CYCLE
          LOOKUP[2]=1, RESULT=3,  ; GC
          LOOKUP[3]=1, RESULT=4,  ; DF
          LOOKUP[4]=1, RESULT=5,  ; SEG LENGTH =5.0
          LOOKUP[5]=1, RESULT=6,  ; NUMSIG =5
          INTERPOLATE =N

phase=linkmerge
  _ANODE=LI.1.A
  _BNODE=LI.1.B

 IF(B.SIGNAL<>0)

  _NUM_LINKS=NODEDATA(1,_BNODE)
  _FT11=NODEDATA(2,_BNODE)
  _FT12=NODEDATA(3,_BNODE)
  _FT13=NODEDATA(4,_BNODE)
    IF(_FT13=0) _FT13=_FT11
  _FT14=NODEDATA(5,_BNODE)
    IF(_FT14=0) _FT14=_FT11

  _FT_HIGH=MIN(_FT11,_FT12,_FT13,_FT14)
  _FT_LOW =MAX(_FT11,_FT12,_FT13,_FT14)

   _GC1=DEFAULT(2,_FT_HIGH)
   _GC2=DEFAULT(2,_FT_LOW)
  if(GC_RATIO=0)
   IF(_NUM_LINKS>2)

     _GC1_1= _GC1/(_GC1+_GC2)
     _GC2_2= 1-_GC1_1

     _FT1=INT(LI.1.FACILITY_TYPE/10)

     IF(_FT1=_FT_HIGH & _NC[_BNODE]<=2)
       _NC[_BNODE]=_NC[_BNODE]+1
       GC_RATIO=_GC1_1
     ELSE
       GC_RATIO=_GC2_2
     ENDIF
```

```
      ELSE
        GC_RATIO=_GC1
      ENDIF
     ENDIF
    ENDIF

  ENDPHASE

  ENDRUN
```

10. NTNET00J.S

```
; Do not change filenames or add or remove FILEI/FILEO statements
using an editor. Use Cube/Application Manager.
RUN PGM=NETWORK PRNFILE="D:\FSUTMS\D4\TCRPM\CUBE\CYCLEN.PRN"
FILEO NETO = "D:\FSUTMS\D4\TCRPM\CUBE\TEMPSIG.NET"
FILEO PRINTO[1] = "D:\FSUTMS\D4\TCRPM\CUBE\NODECYC.CSV"
FILEI NODEI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\SIGS1.DBF"
FILEI LINKI[1] = "D:\FSUTMS\D4\TCRPM\INPUT.00T\NETSIG1.NET"

array _NC=99999,_FT11=99999,_FT12=99999,_FT13=99999,_FT14=99999,
_CL=99999,_SGD=99999

phase=linkmerge
  _SEGID=LI.1.SEGID
  _CLEN= B.CYC_LEN
  _sig=B.SIGNAL

    IF(_SEGID<>0)
      _ALEN=_ALEN+1
      _SGD[_SEGID]=_SEGID
      IF(_CLEN > _CL[SEGID])
       _CL[_SEGID]=_CLEN
      ENDIF
      IF(_BNODE>_MAXNODE) _MAXNODE=_BNODE
    ENDIF

ENDPHASE

PHASE=SUMMARY
  LOOP _NCNT=1,_ALEN
      IF (_CL[_NCNT]>0) PRINT LIST=_SGD[_NCNT], _CL[_NCNT], CSV=T,
PRINTO=1
  ENDLOOP
ENDPHASE

ENDRUN
```

11. NTMAT00J.S

```
; Do not change filenames or add or remove FILEI/FILEO statements
using an editor. Use Cube/Application Manager.
RUN PGM=MATRIX PRNFILE="D:\FSUTMS\D4\TCRPM\CUBE\NTMAT00A.PRN"
MSG='Create SEGMENT.CSV for use in LOOKUP and apply defaults'
FILEI LOOKUPI[2] = "D:\FSUTMS\D4\TCRPM\CUBE\NODECYC.CSV"
```

```
FILEI LOOKUPI[1] = "{Default Segment Data}"
FILEO PRINTO[1] = "D:\FSUTMS\D4\TCRPM\CUBE\SEGMENT.CSV"
FILEI RECI = "D:\FSUTMS\D4\TCRPM\CUBE\SEGDATA1.DBF"


RECS=RECI.NUMRECORDS

SEGID=ri.SEGID
LENGTH=ri.LENGTH
NUMSIG=ri.NUMSIG
FT1=ri.FT1
CYCLE=ri.CYCLE
GC=ri.GC
DF=ri.DF

 LOOKUP  LOOKUPI=2 NAME=CYCLELEN,
         LOOKUP[1]=1, RESULT=2,
         FAIL=0,0,
         INTERPOLATE =N

  LOOKUP  LOOKUPI=1 LIST=Y NAME=DEFAULT,
          LOOKUP[1]=1, RESULT=2,  ; CYCLE
          LOOKUP[2]=1, RESULT=3,  ; GC
          LOOKUP[3]=1, RESULT=4,  ; DF
          LOOKUP[4]=1, RESULT=5,  ; SEG LENGTH =5.0
          LOOKUP[5]=1, RESULT=6,  ; NUMSIG =5
          INTERPOLATE =N

; flag invalid FT1
if (FT1 < 2 | FT1 > 7)
  PRINT LIST='*** FATAL Illegal FT1 of ',FT1,' on SEGID=',SEGID,'
*** MUST BE FIXED'
  ABORT
endif

; set default values where needed
if (GC=0)
  GC=DEFAULT(2,FT1)
  gck=gck+1
endif


 CYCLE1=CYCLELEN(1,SEGID)

if(CYCLE1<>0)CYCLE=CYCLE1

if (CYCLE=0)
  CYCLE=DEFAULT(1,FT1)
  cyclek=cyclek+1
endif

if (DF=0)
  DF=DEFAULT(3,FT1)
  dfk=dfk+1
endif

if (LENGTH=0)
```

```
  LENGTH=DEFAULT(4,FT1)
  slenk=slenk+1
endif

if (NUMSIG=0)
  NUMSIG=DEFAULT(5,FT1)
  nsigk=nsigk+1
endif


GCM=1-GC
DD=0.5*DF*CYCLE*GCM^2

PRINT CSV=T, PRINTO=1,
LIST=SEGID,LENGTH,NUMSIG,FT1,CYCLE,GC,DF,DD


; at end of file, print number of segments with default values
if (I=0)

PRINT LIST='segments with default GC= ',gck
PRINT LIST='segments with default CYCLE= ',cyclek
PRINT LIST='segments with default DF= ',dfk
PRINT LIST='segments with default LENGTH= ',slenk
PRINT LIST='segments with default NUMSIG= ',nsigk

endif

ENDRUN
```

12. NTNET00K.S

```
; Do not change filenames or add or remove FILEI/FILEO statements
using an editor. Use Cube/Application Manager.
RUN PGM=NETWORK
FILEO NETO = "D:\FSUTMS\D4\TCRPM\CUBE\TEMPSIG1.NET"
FILEI LOOKUPI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\SEGMENT.CSV"
FILEI LINKI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\TEMPSIG.NET"

  LOOKUP  LOOKUPI=1 LIST=Y NAME=SEGDATA,
          LOOKUP[1]=1, RESULT=2,  ; SEG LENGTH
          LOOKUP[2]=1, RESULT=3,  ; NUMSIG
          LOOKUP[3]=1, RESULT=4,  ; FT1
          LOOKUP[4]=1, RESULT=5,  ; CYCLE
          LOOKUP[5]=1, RESULT=6,  ; GC
          LOOKUP[6]=1, RESULT=7,  ; DF
          LOOKUP[7]=1, RESULT=8,  ; DD
          FAIL=0,0,
          INTERPOLATE =N

PHASE=LINKMERGE
  _SEGID=LI.1.SEGID
  IF(_SEGID<>0)
   CYCLELENGTH=SEGDATA(4,_SEGID)
  ENDIF
ENDPHASE
```

```
ENDRUN
```

13. NTMAT00B.S

```
; Do not change filenames or add or remove FILEI/FILEO statements
using an editor. Use Cube/Application Manager.
RUN PGM=MATRIX PRNFILE="D:\FSUTMS\D4\TCRPM\CUBE\NTMAT00C.PRN"
MSG='rewrite SPDADJ.DBF file into script commands for later use
via READ statement'
FILEO PRINTO[1] = "D:\FSUTMS\D4\TCRPM\CUBE\MERGE1.DAT"
FILEI RECI = "{SPDADJ Factors}"



PRINT PRINTO=1,
   LIST='IF (AREA_TYPE=',ri.AT1(2.0),'-',ri.AT2(2.0),' &
FACILITY_TYPE=',ri.FT1(2.0),'-',ri.FT2(2.0),')
_ADJF=',ri.FACTOR(6.4)


/*
AT1 AT2 FT1 FT2 FACTOR
11 14 21 26 1.1000
11 14 31 38 1.1000
11 14 41 48 0.9500
11 14 61 68 1.1000
21 21 21 26 1.0000
21 21 31 38 1.1000
21 21 41 48 0.9000
21 21 61 68 1.0000
31 35 11 17 0.9500
31 35 21 26 1.0200
31 35 31 38 1.1000
31 35 41 48 1.0500
31 35 61 68 1.0000
31 35 71 79 0.9500
31 35 97 98 0.9500
31 35 81 85 0.9500
41 45 11 17 1.0000
41 45 21 26 1.0500
41 45 31 38 1.1000
41 45 41 48 0.9500
41 45 61 68 1.0000
41 45 71 79 0.9500
41 45 97 98 0.9500
51 52 11 17 0.9500
51 52 21 26 0.9000
51 52 31 38 0.9000
51 52 41 48 1.0000
51 52 71 79 0.9500
51 52 97 98 0.9500
51 52 91 95 0.9500
*/
```

```
ENDRUN
```

14. NTMAT00D.S

```
; Do not change filenames or add or remove FILEI/FILEO statements
using an editor. Use Cube/Application Manager.
RUN PGM=MATRIX PRNFILE="D:\FSUTMS\D4\TCRPM\CUBE\NTMAT00F.PRN"
MSG='Extract Appropriate Curve Data'
FILEO PRINTO[2] = "D:\FSUTMS\D4\TCRPM\CUBE\TSPDS.CSV"
FILEO PRINTO[1] = "D:\FSUTMS\D4\TCRPM\CUBE\CURVES.TXT"
FILEI RECI = "{DATADIR}\SDLAYUPD.{YEAR}{ALT}";,
;CARD=1-1, LOWMODE=2-3, HIGHMODE=4-5,
;LOWAT=6-7, HIGHAT=8-9, LOWFT=10-11, HIGHFT=12-13, CURVE=15-16,
SELECT=2,1
;FIELDS=1-1,2-3,4-5,6-7,8-9,10-11,12-13,15-16,

CARD=VAL(SUBSTR(RECI,1,1))
LOWMODE=VAL(SUBSTR(RECI,2,2))
HIGHMODE=VAL(SUBSTR(RECI,4,2))
LOWAT=VAL(SUBSTR(RECI,6,2))
HIGHAT=VAL(SUBSTR(RECI,8,2))
LOWFT=VAL(SUBSTR(RECI,10,2))
HIGHFT=VAL(SUBSTR(RECI,12,2))
CURVE=VAL(SUBSTR(RECI,15,2))


ARRAY LOWASPD=15,LOWTSPD=15,HIASPD=15,HITSPD=15
IF (CARD=2)

  LOOP _WW=LOWMODE,HIGHMODE
      PRINT LIST="IF (AREA_TYPE=",LOWAT(2.0),"-",HIGHAT(2.0),
           "&FACILITY_TYPE=",LOWFT(2.0),"-",HIGHFT(2.0),
           ") _CURVEMODE",_WW(2.0L),"=",CURVE(2.0), printo=1
  ENDLOOP

ENDIF
IF (CARD=3)
  CURVE=VAL(SUBSTR(RECI,3,2))
  LOWASPD[CURVE]=VAL(SUBSTR(RECI,5,6))
  LOWTSPD[CURVE]=VAL(SUBSTR(RECI,11,6))
  HIASPD[CURVE]=VAL(SUBSTR(RECI,17,6))
  HITSPD[CURVE]=VAL(SUBSTR(RECI,23,6))
  IF (CURVE>MAXCURVE) MAXCURVE=CURVE
ENDIF

IF (I=0)
     PRINT CSV=T,
LIST=LOWASPD[1],LOWASPD[2],LOWASPD[3],LOWASPD[4],LOWASPD[5],

LOWASPD[6],LOWASPD[7],LOWASPD[8],LOWASPD[9],LOWASPD[10],

LOWASPD[11],LOWASPD[12],LOWASPD[13],LOWASPD[14],LOWASPD[15],

LOWTSPD[1],LOWTSPD[2],LOWTSPD[3],LOWTSPD[4],LOWTSPD[5],

LOWTSPD[6],LOWTSPD[7],LOWTSPD[8],LOWTSPD[9],LOWTSPD[10],
```

```
LOWTSPD[11],LOWTSPD[12],LOWTSPD[13],LOWTSPD[14],LOWTSPD[15],
PRINTO=2
     PRINT CSV=T,
LIST=HIASPD[1],HIASPD[2],HIASPD[3],HIASPD[4],HIASPD[5],

HIASPD[6],HIASPD[7],HIASPD[8],HIASPD[9],HIASPD[10],

HIASPD[11],HIASPD[12],HIASPD[13],HIASPD[14],HIASPD[15],

HITSPD[1],HITSPD[2],HITSPD[3],HITSPD[4],HITSPD[5],

HITSPD[6],HITSPD[7],HITSPD[8],HITSPD[9],HITSPD[10],

HITSPD[11],HITSPD[12],HITSPD[13],HITSPD[14],HITSPD[15], PRINTO=2


ENDIF
ENDRUN
```

15. NTNET00B.S

```
; Do not change filenames or add or remove FILEI/FILEO statements
using an editor. Use Cube/Application Manager.
RUN PGM=NETWORK PRNFILE="D:\FSUTMS\D4\TCRPM\CUBE\NTNET00B.PRN"
MSG='Compute FFSPD values and adjusted SPD1 Values'
FILEI LOOKUPI[3] = "D:\FSUTMS\D4\TCRPM\CUBE\LASTLINKNODES.PRN"
FILEI NODEI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\SIGS1.DBF"
FILEO PRINTO[2] = "D:\FSUTMS\D4\TCRPM\CUBE\ACTIVECENTROIDS.CSV"
FILEO PRINTO[1] = "D:\FSUTMS\D4\TCRPM\CUBE\TRNSPEEDS.PRN"
FILEI LOOKUPI[2] = "D:\FSUTMS\D4\TCRPM\CUBE\TSPDS.CSV"
FILEI LOOKUPI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\SEGMENT.CSV"
FILEI LINKI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\TEMPSIG1.NET"
FILEO NETO = "D:\FSUTMS\D4\TCRPM\CUBE\UNLOADED BASE.NET",
 EXCLUDE=FFSPD,POSTSPD,FFOLD,SPD1

ARRAY _USECNT=9999
;==================
;==================
;SEGID LENGTH NUMSIG FT1 CYCLE GC DF DD
; lookup parameters by SEGID
  LOOKUP  LOOKUPI=1 LIST=Y NAME=SEGDATA,
          LOOKUP[1]=1, RESULT=2,  ; SEG LENGTH
          LOOKUP[2]=1, RESULT=3,  ; NUMSIG
          LOOKUP[3]=1, RESULT=4,  ; FT1
          LOOKUP[4]=1, RESULT=5,  ; CYCLE
          LOOKUP[5]=1, RESULT=6,  ; GC
          LOOKUP[6]=1, RESULT=7,  ; DF
          LOOKUP[7]=1, RESULT=8,  ; DD
          INTERPOLATE =N

 LOOKUP  LOOKUPI=3 NAME=LASTNODE,
         LOOKUP[1]=1, RESULT=2,
         FAIL=0,0,
         INTERPOLATE =N
```

```
PROCESS PHASE=LINKMERGE
IF (FACILITY_TYPE!=50-59)
_slen=SEGDATA(1,SEGID) ; lookup slen from curve 1 for the current
link SEGID
_nsig=SEGDATA(2,SEGID) ; lookup nsig from curve 2 for the current
link SEGID
_DF=SEGDATA(6,SEGID)   ; lookup DD from curve 7 for the current
link SEGID
_CYCLE=LI.1.CYCLELENGTH

_LSTNOD=LASTNODE(1,LI.1.B)

_GCM=1-li.1.GC_RATIO

IF(_GCM <> 1)
 _DD=0.5*_DF*_CYCLE*_GCM^2
ELSE
 _DD=SEGDATA(7,SEGID)
ENDIF

_secphr=1.0/3600

_FT1=INT(0.1*FACILITY_TYPE) ; Compute FT1 from FT2, note FT1 is
already on this input network

IF (_FT1 <> 5)
  IF (li.1.POSTSPD > 50.0)
    FFSPD=0.88*POSTSPD + 14
  ELSEIF(_FT1=4 & AREA_TYPE<50 & SEGID=0 & POSTSPD<=35) ;
COLLECTORS (URBAN FACILITY TYPE IN 40's)
    FFSPD=0.59*POSTSPD + 13.12
  ELSE
    FFSPD=0.79*POSTSPD + 12
  ENDIF


  FFOLD=FFSPD
  IF (li.1.SEGID <> 0 & _LSTNOD<>1)
    FFSPD=_slen/( (_slen/FFSPD) + _nsig*_DD*_secphr)
    IF (FFSPD=0)
      _ecount=_ecount+1
      PRINT LIST='FFSPD computed = ',FFSPD
      PRINT LIST='FFOLD= ',FFOLD
      PRINT LIST='SEGID= ',SEGID
      PRINT LIST='slen= ',_slen
      PRINT LIST='nsig= ',_nsig
      PRINT LIST='DD= ',_DD
      PRINT LIST='secphr= ',_secphr
    ENDIF
    _icount=_icount+1
  ELSE
    _ffcount=_ffcount+1
  ENDIF
ELSE
  FFSPD=POSTSPD
ENDIF
```

```
; Apply adjustment factors and convert to Integer speeds for use
in FSUTMS

_ADJF=1.0

READ,
        FILE = "D:\FSUTMS\D4\TCRPM\CUBE\MERGE1.DAT"

IF (li.1.POSTSPD <> 0)
  SPD1=round(100*FFSPD*_ADJF)/100
ENDIF


POSTEDSPEED=POSTSPD

IF (LI.1.POSTSPD<>0)
  FREEFLOWSPEED=SPD1
ELSE
  FREEFLOWSPEED=FFSPD
ENDIF

IF (POSTSPD>0) POSTEDTIME=60*DISTANCE/POSTSPD
IF (FREEFLOWSPEED>0) FREEFLOWTIME=60*DISTANCE/FREEFLOWSPEED
IF (FREEFLOWTIME>0) TIME=FREEFLOWTIME, time2=FREEFLOWTIME
if ({rndnettime}=1) TIME=ROUND(100*TIME)/100 ,
time2=ROUND(100*TIME2)/100
IF (FACILITY_TYPE=99) TIME=0 TIME2=0
  if (time>0)
    _AUTOSPEED=60*DISTANCE/TIME
  else
    _AUTOSPEED=15
  ENDIF

ENDIF



IF (A<={ZONESA}) _USECNT[A]=1
  LOOKUP name=curve,
     lookup[1]=1, result=16,
     lookup[2]=2, result=17,
     lookup[3]=3, result=18,
     lookup[4]=4, result=19,
     lookup[5]=5, result=20,
     lookup[6]=6, result=21,
     lookup[7]=7, result=22,
     lookup[8]=8, result=23,
     lookup[9]=9, result=24,
     lookup[10]=10, result=25,
     lookup[11]=11, result=26,
     lookup[12]=12, result=27,
     lookup[13]=13, result=28,
     lookup[14]=14, result=29,
     lookup[15]=15, result=30,
     interpolate=y, list=y, lookupi=2
```

```
read,
        FILE = "D:\FSUTMS\D4\TCRPM\CUBE\CURVES.TXT"

     ; APPLY DEFAULT CURVES
     IF (_AUTOSPEED>0)
       _TSPD1=CURVE(_CURVEMODE1,_AUTOSPEED)
       _TSPD4=CURVE(_CURVEMODE4,_AUTOSPEED)
       _TSPD5=CURVE(_CURVEMODE5,_AUTOSPEED)
       _TSPD6=CURVE(_CURVEMODE6,_AUTOSPEED)
       _TSPD8=CURVE(_CURVEMODE8,_AUTOSPEED)
       _TSPD12=CURVE(_CURVEMODE12,_AUTOSPEED)

       LBTIME=60*DISTANCE/_TSPD4
       EBTIME=60*DISTANCE/_TSPD6
       RLTIME=60*DISTANCE/_TSPD8
       SHTIME=60*DISTANCE/_TSPD12

     ENDIF
     ; OVERRIDES FROM NETWORK AND YET MISSING VALUES
     IF (TTIME>0&TMODE= 4) LBTIME=TTIME
     IF (TTIME>0&TMODE= 6) EBTIME=TTIME
     IF (TTIME>0&TMODE= 8) RLTIME=TTIME
     IF (TTIME>0&TMODE=12) SHTIME=TTIME
     IF (LBTIME=0) LBTIME=99
     IF (EBTIME=0) EBTIME=99
     IF (SHTIME=0) SHTIME=99
     IF (RLTIME=0) RLTIME=99
     _WKSPEED=60*DISTANCE/LBTIME
     _LBSPEED=60*DISTANCE/LBTIME
     _EBSPEED=60*DISTANCE/EBTIME
     _RLSPEED=60*DISTANCE/RLTIME
     _SHSPEED=60*DISTANCE/SHTIME

; WALK CONTROLS
     WALKSPEED=CURVE(_CURVEMODE1,2.5)
     WALKTIME=60*DISTANCE/WALKSPEED
     IF (FACILITY_TYPE=10-19,70-99)
        WALKDISTANCE=9.99
        WALKTIME=9.99
     ENDIF

PRINT LIST="A=",A(4.0)," B=",B(4.0),
           "  SPEEDS-  Auto=",_AUTOSPEED(4.1),
           " Walk=",WALKSPEED(4.1),
           " Local Bus=",_LBSPEED(4.1),
           " Express Bus=",_EBSPEED(4.1),
           " Shuttle Bus=",_SHSPEED(4.1),
           " Rail=",_RLSPEED(4.1),
                           PRINTO=1
ENDPROCESS

PROCESS  PHASE=SUMMARY

IF (_ecount>0) PRINT LIST=_ecount,' links with segids were
assigned speed=0'

PRINT LIST=_ffcount,' Free-flow links'
```

```
PRINT LIST=_icount,' intersection controlled links'

LOOP _WW=1,{ZONESA}

PRINT CSV=T, LIST=_WW,_USECNT[_WW], PRINTO=2

ENDLOOP

ENDPHASE

ENDRUN
```

HIGHWAY PATH

1. HPMAT00A.S

```
; Do not change filenames or add or remove FILEI/FILEO statements
using an editor. Use Cube/Application Manager.
RUN PGM=MATRIX MSG='Converts FSUTMS TCARDS to Voyager Turn
Penalty Format'
FILEO PRINTO[1] = "D:\FSUTMS\D4\TCRPM\CUBE\TCARDS.PRN"
FILEI RECI = "{DATADIR}\TCARDS.PEN",
 A=1-6, B=7-12,C=13-18, TPEN=25-28;,
;         fld=1-6,7-12,13-18,25-28

if (RI.tpen<>0) PRINT form=6.0,list=RI.A,RI.B,RI.C,"
1",RI.TPEN/100(8.2), PRINTO=1
if (RI.tpen=0) PRINT form=6.0,list=RI.A,RI.B,RI.C," 1    -1",
PRINTO=1

ENDRUN
```

2. HPMAT00C.S

```
; Do not change filenames or add or remove FILEI/FILEO statements
using an editor. Use Cube/Application Manager.
RUN PGM=MATRIX MSG='Extract Terminal Times from ZDATA4B'
FILEO PRINTO[1] = "D:\FSUTMS\D4\TCRPM\CUBE\EXTERNALTT.TXT"
FILEI RECI = "D:\FSUTMS\D4\TCRPM\CUBE\zdata4b.{year}{alt}";,
; fld=1,5-8,88-89
card=substr(reci,1,1)
taz=substr(reci,5,4)
termtime=substr(reci,88,2)
if (card='4') print csv=t,list=taz,termtime, printo=1
ENDRUN
```

3. HPHWY00A.S

```
; Do not change filenames or add or remove FILEI/FILEO statements
using an editor. Use Cube/Application Manager.
RUN PGM=HIGHWAY PRNFILE="D:\FSUTMS\D4\TCRPM\CUBE\HPHWY00A.PRN"
MSG='Build Free Flow Skims'
FILEI LOOKUPI[2] = "D:\FSUTMS\D4\TCRPM\CUBE\ACTIVECENTROIDS.CSV"
FILEI NETI = "D:\FSUTMS\D4\TCRPM\CUBE\UNLOADED BASE.NET"
FILEI LOOKUPI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\EXTERNALTT.TXT"
FILEO MATO[1] =
"D:\FSUTMS\D4\TCRPM\CUBE\FHSKIMS.{Alt}{YEAR}.MAT",
  MO=1-7,30
NAME=COST,DISTANCE,TIME1,TIME2,SVCTIME,SVCMINUTES,TOLLTIMEEQ,TERM
INALTIME, dec=8*3
FILEI TURNPENI = "D:\FSUTMS\D4\TCRPM\CUBE\TCARDS.PRN"
LOOKUP NAME=ACTIVEC, LOOKUP[1]=1, RESULT=2, LOOKUPI=2
ZONEMSG=100
ARRAY TERM=99,ATYPE={ZONESA}
; cbd
```

```
term[11]={term11}
term[12]={term12}
term[13]={term13}
; cbd fringe
term[21]={term21}
; other urban
term[31]={term31}
term[32]={term32}
term[33]={term33}
term[34]={term34}
; obd
term[41]={term41}
term[42]={term42}
term[43]={term43}
; rural
term[51]={term51}
term[52]={term52}

PROCESS PHASE=LINKREAD
       IF (A<={ZONESA}) ATYPE[A]=LI.AREA_TYPE ; ASSIGNS AN AREA
TYPE TO EACH ZONE BASED ON LAST ENTRY (LIKE TTPREP)
; use time for impedance unless toll present, then convert toll +
service time to time equivalent
       if (li.facility_type=99)

lw.linkcost=(li.cartoll*{ctoll}*60)+li.svcminutes+(li.svcseconds/
60)
       else
          lw.linkcost=li.time
       endif
;       PRINT CSV=T, LIST=A,B,LW.LINKCOST
       if (li.facility_type=80-89) addtogroup=1
ENDPROCESS


PROCESS PHASE=ILOOP
ACTIVE=ACTIVEC(1,I)
IF (ACTIVE=1)
; first, skim the network based on travel time, LOV
      PATH=LW.LINKCOST,peni=1,excludegroup=1,
      MW[01]=PATHTRACE(LI.CARTOLL),NOACCESS=999.99,dec=2,
      MW[02]=PATHTRACE(LI.DISTANCE),NOACCESS=999.99,dec=2,
      MW[03]=PATHTRACE(LI.TIME),NOACCESS=999.99,dec=3,
      MW[05]=PATHTRACE(LI.SVCMINUTES),NOACCESS=999.99,dec=2,
      MW[06]=PATHTRACE(LI.SVCSECONDS),NOACCESS=999.99,dec=2
; next, skim the network based on travel time, HOV
      PATH=LW.LINKCOST,peni=1,
      MW[11]=PATHTRACE(LI.CARTOLL),NOACCESS=999.99,dec=2,
      MW[12]=PATHTRACE(LI.DISTANCE),NOACCESS=999.99,dec=2,
      mw[13]=PATHTRACE(LI.TIME),NOACCESS=999.99,dec=3,
      MW[15]=PATHTRACE(LI.SVCMINUTES),NOACCESS=999.99,dec=2,
      MW[16]=PATHTRACE(LI.SVCSECONDS),NOACCESS=999.99,dec=2
; next, convert the toll skims to travel time equivalent
      MW[07]=(MW[01]*{CTOLL}*60)+MW[05]+(MW[06]/60)
      MW[17]=(MW[11]*{CTOLL}*60)+MW[15]+(MW[16]/60)
; next, create a terminal time matrix
```

```
lookup name=extt, lookup[1]=1, result=2,
interpolate=n,fail[1]=0,fail[2]=0, lookupi=1
JLOOP
      MW[04]=MW[3]       MW[14]=mw[13]
      if (i<={zonesi}&j<={zonesi}); internal to internal
        mw[30]=(term[atype[i]]+term[atype[j]])
      endif
      if (i>{zonesi}&j<={zonesi}) ; external to internal
        ttime=extt(1,i)
        mw[30]=(ttime+term[atype[j]])
      endif
      if (i<={zonesi}&j>{zonesi}) ; internal to external
        ttime=extt(1,j)
        mw[30]=(term[atype[i]]+ttime)
      endif
      if (i>{zonesi}&j>{zonesi}) ; external to external
        ttimei=extt(1,i)
        ttimej=extt(1,j)
        mw[30]=(ttimei+ttimej)
      endif
ENDJLOOP

; lastly, treat intrazonals
  MW[01][i] = 0  ; Intrazonal time HOV
  MW[02][i] = lowest(2,2) * 0.25  ; Intrazonal time HOV
  MW[03][i] = lowest(3,2) * 0.25  ; Intrazonal time HOV
  MW[04][i] = lowest(4,2) * 0.25  ; Intrazonal time HOV
  MW[05][i] = 0  ; Intrazonal time HOV
  MW[06][i] = 0  ; Intrazonal time HOV
  MW[07][i] = lowest(7,2) * 0.25  ; Intrazonal time HOV
  MW[11][i] = 0  ; Intrazonal time HOV
  MW[12][i] = lowest(12,2) * 0.25  ; Intrazonal time HOV
  MW[13][i] = lowest(13,2) * 0.25  ; Intrazonal time HOV
  MW[14][i] = lowest(14,2) * 0.25  ; Intrazonal time HOV
  MW[15][i] = 0  ; Intrazonal time HOV
  MW[16][i] = 0  ; Intrazonal time HOV
  MW[17][i] = lowest(17,2) * 0.25  ; Intrazonal time HOV
LIST = "AREA TYPE FOR ZONE",I(5.0)," IS",ATYPE[I](3.0)

ENDIF

ENDPHASE

ENDRUN
```

4. HPMAT00B.C

```
; Do not change filenames or add or remove FILEI/FILEO statements
using an editor. Use Cube/Application Manager.
RUN PGM=MATRIX PRNFILE="D:\FSUTMS\D4\TCRPM\CUBE\HPMAT00A.PRN"
MSG='Compile Skims for GM and Mode'
FILEI LOOKUPI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\ACTIVECENTROIDS.CSV"
FILEO MATO[2] = "D:\FSUTMS\D4\TCRPM\CUBE\FHSKIMS.MAT",
MO=11-14,NAME=COST,DISTANCE,TIME,TIME2,FORMAT=TRANPLAN
FILEO MATO[1] = "D:\FSUTMS\D4\TCRPM\CUBE\FHSKIMTT.TEM",
MO=1-5,NAME=COST,DISTANCE,TIME,TIME2,TIME1WTT,FORMAT=TRANPLAN
```

```
FILEI MATI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\FHSKIMS.{Alt}{YEAR}.MAT"
LOOKUP NAME=ACTIVEC, LOOKUP[1]=1, RESULT=2, LOOKUPI=1

ZONEMSG=100
ACTIVEI=ACTIVEC(1,I)
IF (ACTIVEI=1)

jloop
; EQUIVALENT OF WHAT GOES INTO GM
MW[1]=MI.1.COST*100
MW[2]=MI.1.DISTANCE*100
MW[3]=MI.1.TIME1*100
MW[4]=(MI.1.TIME1+MI.1.TOLLTIMEEQ+MI.1.TERMINALTIME)*100
MW[5]=(MI.1.TIME1+MI.1.TERMINALTIME)*100
; EQUIVALENT OF WHAT GOES INT0 MODE CHOICE LOV
MW[11]=MI.1.COST*100
MW[12]=MI.1.DISTANCE*100
MW[13]=MI.1.TIME1*100
MW[14]=MI.1.TIME1*100
endjloop
endif
ENDRUN
```

TRIP DISTRIBUTION

1. DTMAT00A.S

```
; Do not change filenames or add or remove FILEI/FILEO statements
using an editor. Use Cube/Application Manager.
RUN PGM=MATRIX PRNFILE="D:\FSUTMS\D4\TCRPM\CUBE\DTMAT00A.PRN"
MSG='Builds Public School Matrix'
FILEO MATO[1] =
"D:\FSUTMS\D4\TCRPM\CUBE\PSCHOOL.{ALT}{YEAR}.MAT",
MO=1,NAME=PUBLICSCHOOL
FILEI MATI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\PBSCH.{ALT}{YEAR}",
 PATTERN=IJM:V, FIELDS=1-5,6-10,11-13,14-20

PAR ZONEMSG=100 ZONES={ZONESA}

MW[1]=MI.1.1
ENDRUN
```

2. DTMAT00C.S

```
; Do not change filenames or add or remove FILEI/FILEO statements
using an editor. Use Cube/Application Manager.
RUN PGM=MATRIX PRNFILE="D:\FSUTMS\D4\TCRPM\CUBE\DTMAT00B.PRN"
MSG='Convert Gamma Parameters for Lookup Function'
FILEI RECI = "{DATADIR}\GAMMA_PARAM.DBF"
FILEO PRINTO[1] = "D:\FSUTMS\D4\TCRPM\CUBE\GAMMA.CSV"

print csv=t,
,list=ri.purpose(2.0),ri.a(15.4),ri.b(15.4),ri.c(15.4), printo=1


ENDRUN
```

3. DTMAT00E.S

```
; Do not change filenames or add or remove FILEI/FILEO statements
using an editor. Use Cube/Application Manager.
RUN PGM=MATRIX PRNFILE="D:\FSUTMS\D4\TCRPM\CUBE\DTMAT00D.PRN"
MSG='Put Gamma Values in Matrix'
FILEO MATO[2] = "D:\FSUTMS\D4\TCRPM\CUBE\KFACTOR.MAT",
 MO=41, NAME=KFACTOR
FILEI LOOKUPI[2] = "D:\FSUTMS\D4\TCRPM\CUBE\ACTIVECENTROIDS.CSV"
FILEI MATI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\FHSKIMTT.TEM"
FILEO MATO[1] = "D:\FSUTMS\D4\TCRPM\CUBE\GAMMA.MAT",
 MO=21-
31,NAME=GAMMA1,GAMMA2,GAMMA3,GAMMA4,GAMMA5,GAMMA6,GAMMA7,GAMMA8,G
AMMA9,GAMMA10,GAMMA11
FILEI LOOKUPI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\GAMMA.CSV"
PAR ZONEMSG=100
lookup, name=gamma, lookup[1]=1, result=2, lookup[2]=1, result=3,
lookup[3]=1, result=4, LOOKUPI=1
lookup, name=activec, lookup[1]=1, result=2,LOOKUPI=2
active=activec(1,i)
```

```
if (active=1)
mw[51]=mi.1.5/100
  LOOP PRP=1,11
      a=GAMMA(1,PRP)
      b=GAMMA(2,PRP)
      c=GAMMA(3,PRP)
      GSKIM=PRP+20 ; Output Gamma Skim
      JLOOP
          MW[41]=1
          if (MW[51]>0) mw[GSKIM]=(a*MW[51]^(-b))*exp((-
c)*MW[51]) ; PUT GAMMA MATRICES IN MW[21]-MW[24]
          if (i=1001&j=1010) PRINT CSV=T,
LIST=I,J,MW[51],A,B,C,MW[GSKIM]
          if(i>700 & j>700) MW[41]=0
ENDJLOOP
  ENDLOOP
endif
ENDRUN
```

4. DTDST00A.S

```
; Do not change filenames or add or remove FILEI/FILEO statements
using an editor. Use Cube/Application Manager.
RUN PGM=DISTRIBUTION
PRNFILE="D:\FSUTMS\D4\TCRPM\CUBE\DISTRIB.PRN" MSG='PRE-
Distribution of 11 Trip Purposes'
FILEI ZDATI[1] = "{SCENARIO_DIR}\PANDA2.DBF"
FILEI MATI[3] = "D:\FSUTMS\D4\TCRPM\CUBE\KFACTOR.MAT"
FILEI MATI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\GAMMA.MAT"
FILEI MATI[2] = "D:\FSUTMS\D4\TCRPM\CUBE\FHSKIMTT.TEM"
FILEI LOOKUPI[2] = "D:\FSUTMS\D4\TCRPM\CUBE\ACTIVECENTROIDS.CSV"
FILEI LOOKUPI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\GAMMA.CSV"
FILEO MATO[1] = "D:\FSUTMS\D4\TCRPM\CUBE\PTRIPS1.MAT",
 MO=1-
11,NAME=HBW,HBSH,HBSR,HBSCPRV,HBO,NHBW,NHBO,AIRPORT,TRUCK4TIRE,TR
KSNGLUNIT,TRACTORTRAILER, dec=11*6
PAR ZONEMSG=100 MAXITERS={atiter}, MAXRMSE=1

 LOOKUP NAME=GAMMA,; Gamma Function Parameters
    LOOKUP[1]=1, RESULT=2,           ; CONSTANT
    LOOKUP[2]=1, RESULT=3,           ; Alpha VALUE
    LOOKUP[3]=1, RESULT=4,           ; BETA VALUE
    LookupI=1
LOOKUP NAME=ACTIVE, LOOKUP[1]=1, RESULT=2, LOOKUPI=2
SETPA P[1]=ZI.1.P1, A[1]=ZI.1.A1
SETPA P[2]=ZI.1.P2, A[2]=ZI.1.A2
SETPA P[3]=ZI.1.P3, A[3]=ZI.1.A3
SETPA P[4]=ZI.1.P4, A[4]=ZI.1.A4
SETPA P[5]=ZI.1.P5, A[5]=ZI.1.A5
SETPA P[6]=ZI.1.P6, A[6]=ZI.1.A6
SETPA P[7]=ZI.1.P7, A[7]=ZI.1.A7
SETPA P[8]=ZI.1.P7, A[8]=ZI.1.A7 ; airport is not working in
TCRPM trip gen
SETPA P[9]=ZI.1.P9, A[9]=ZI.1.A9
SETPA P[10]=ZI.1.P10, A[10]=ZI.1.A10
SETPA P[11]=ZI.1.P11, A[11]=ZI.1.A11
```

```
MW[50]=MI.2.5/100
MW[21]=MI.1.1
MW[22]=MI.1.2
MW[23]=MI.1.3
MW[24]=MI.1.4
MW[25]=MI.1.5
MW[26]=MI.1.6
MW[27]=MI.1.7
MW[28]=MI.1.8
MW[29]=MI.1.9
MW[30]=MI.1.10
MW[31]=MI.1.11
MW[41]=MI.3.1
ACTIVEC=ACTIVE(1,I)
IF (ACTIVEC=1) ;ONLY FOR ACTIVE CENTROIDS

LOOP PURP=1,11
   PAF=0
   MW[PURP] = A[PURP] * MW[PURP+20] * MW[41]
   ATTRSUM=ROWSUM(PURP)
   IF (ATTRSUM>0) PAF=P[PURP]/ATTRSUM
   MW[PURP]=PAF * MW[PURP]
ENDLOOP


FREQUENCY  BASEMW=50,VALUEMW=1, RANGE=0-80-1,TITLE='HBW TRAVEL
TIME FREQUENCY'
FREQUENCY  BASEMW=50,VALUEMW=2, RANGE=0-80-1,TITLE='HBSH TRAVEL
TIME FREQUENCY'
FREQUENCY  BASEMW=50,VALUEMW=3, RANGE=0-80-1,TITLE='HBSR TRAVEL
TIME FREQUENCY'
FREQUENCY  BASEMW=50,VALUEMW=4, RANGE=0-80-1,TITLE='HBSC-PRIVATE
TRAVEL TIME FREQUENCY'
FREQUENCY  BASEMW=50,VALUEMW=5, RANGE=0-80-1,TITLE='HBO TRAVEL
TIME FREQUENCY'
FREQUENCY  BASEMW=50,VALUEMW=6, RANGE=0-80-1,TITLE='NHBW TRAVEL
TIME FREQUENCY'
FREQUENCY  BASEMW=50,VALUEMW=7, RANGE=0-80-1,TITLE='NHBO TRAVEL
TIME FREQUENCY'
;FREQUENCY  BASEMW=50,VALUEMW=8, RANGE=0-80-1,TITLE='AIRPORT
TRAVEL TIME FREQUENCY'
FREQUENCY  BASEMW=50,VALUEMW=9, RANGE=0-80-1,TITLE='FOUR-TIRE
TRUCK TRAVEL TIME FREQUENCY'
FREQUENCY  BASEMW=50,VALUEMW=10, RANGE=0-80-1,TITLE='SINGLE UNIT
TRUCK TRAVEL TIME FREQUENCY'
FREQUENCY  BASEMW=50,VALUEMW=11, RANGE=0-80-1,TITLE='COMBINATION
UNIT TRUCK TRAVEL TIME FREQUENCY'

MW[8]=0
ENDIF
ENDRUN
```

PRE-ASSIGNMENT

1. PLMAT00A.S

```
; Do not change filenames or add or remove FILEI/FILEO statements
using an editor. Use Cube/Application Manager.
RUN PGM=MATRIX PRNFILE="D:\FSUTMS\D4\TCRPM\CUBE\PLMAT00A.PRN"
MSG='Assemble Person & Vehicle Trips, Bucket Round for TRANPLAN
Mode Choice'
FILEI MATI[3] = "{SCENARIO_DIR}\EETAB.{ALT}{YEAR}.MAT"
FILEI LOOKUPI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\ACTIVECENTROIDS.CSV"
FILEO MATO[1] = "D:\FSUTMS\D4\TCRPM\CUBE\HTTABTEM.MAT",
 MO=14,1-
13,NAME=TOTALTRIPS,HBW,HBSH,HBSR,HBSCALL,HBO,NHBW,NHBO,AIRPORT,TR
UCK4TIRE,TRKSNGLUNIT,TRACTORTRAILER,PUBLICSCHOOL,EETRIPS
FILEI MATI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\PTRIPS1.MAT"
FILEI MATI[2] = "D:\FSUTMS\D4\TCRPM\CUBE\PSCHOOL.{ALT}{YEAR}.MAT"
PAR ZONEMSG=100
lookup, name=activec, lookup[1]=1, result=2, lookupi=1
active=activec(1,i)
if (active=1)
MW[1]=(MI.1.1+MI.1.1.T)*{AOFAC1}*0.5
MW[2]=(MI.1.2+MI.1.2.T)*{AOFAC2}*0.5
MW[3]=(MI.1.3+MI.1.3.T)*{AOFAC3}*0.5
MW[4]=(MI.1.4+MI.1.4.T)*{AOFAC4}*0.5
MW[5]=(MI.1.5+MI.1.5.T)*{AOFAC5}*0.5
MW[6]=(MI.1.6+MI.1.6.T)*{AOFAC6}*0.5
MW[7]=(MI.1.7+MI.1.7.T)*{AOFAC7}*0.5
MW[8]=(MI.1.8+MI.1.8.T)*{AOFAC8}*0.5
MW[9]=(MI.1.9+MI.1.9.T)*{AOFAC9}*0.5
MW[10]=(MI.1.10+MI.1.10.T)*{AOFAC10}*0.5
MW[11]=(MI.1.11+MI.1.11.T)*{AOFAC11}*0.5
MW[12]=(MI.2.1+MI.2.1.T)*{AOFAC4}*0.5
MW[13]=(MI.3.1+MI.3.1.T)*0.5

MW[14]=MW[1]+MW[2]+MW[3]+MW[4]+MW[5]+MW[6]+MW[7]+MW[8]+MW[9]+MW[1
0]+MW[11]+MW[12]+MW[13]

ENDIF
ENDRUN
```

2. PLHWY00A.S

```
; Do not change filenames or add or remove FILEI/FILEO statements
using an editor. Use Cube/Application Manager.
RUN PGM=HIGHWAY PRNFILE="D:\FSUTMS\D4\TCRPM\CUBE\PLHWY00A.PRN"
MSG='Runs Preassignment'
FILEI TURNPENI = "D:\FSUTMS\D4\TCRPM\CUBE\TCARDS.PRN"
FILEO PRINTO[1] = "D:\FSUTMS\D4\TCRPM\CUBE\PRETOLL.PRN"
FILEI MATI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\HTTABTEM.MAT"
FILEI NETI = "D:\FSUTMS\D4\TCRPM\CUBE\UNLOADED BASE.NET"
FILEO NETO = "D:\FSUTMS\D4\TCRPM\CUBE\PRELOAD.NET"
PAR ZONEMSG=100 MAXITERS={iter} GAP={EPSILON2}
ARRAY TOLLREVENUE=99 TOLLVMT=99
TURNS N=1-99999
```

```
ZONEMSG=100
CTOLL={CTOLL}
; look up deceleration rate based on approach speed
LOOKUP,
 INTERPOLATE=Y, LIST=Y, NAME=DECEL,
        LOOKUP[1]=1,RESULT=2,
R = '30 4',
    '70 6.2'

PROCESS PHASE=LINKREAD
T0=li.TIME
      LW.FFTIME=LI.TIME
      T0=LI.TIME
      C=(LI.CAP_HR/li.confac)*li.uroadfactor
      IF (LI.TMODE=8|LI.TMODE=5|LI.BPREXPONENT=0) ADDTOGROUP=9
; classify links based on presence/absence of tolls
      LINKCLASS=1 ; no toll
      if (li.cartoll>0)
       LINKCLASS=2 ; with toll
       LW.TOLLTIME=({CTOLL}*LI.CARTOLL)*60
       T0=({CTOLL}*LI.CARTOLL)*60 + LI.SVCMINUTES +
LI.SVCSECONDS/60
       if (iteration=0)
        LW.ARRIVR=(V/LI.UROADFACTOR)*LI.CONFAC/LI.PLZALNSMAX  ;
hourly volume per toll lane ie. arrival rate in vehicles per hour
        LW.SERVT=LI.SVCMINUTES+(LI.SVCSECONDS/60)           ;
Plaza lane service time in minutes per vehicle
        IF (LI.TOLLTYPE=1)
           LW.SERVR=(1/LW.SERVT)*60                          ;
Plaza lane service rate in vehicle per hour
        ELSE
           LW.SERVR=100
        ENDIF
        PRINT LIST='ARRIVR= ',LW.ARRIVR
        PRINT LIST='SERVT= ',LW.SERVT
        PRINT LIST='SERVR= ',LW.SERVR
        if (LW.ARRIVR>=LW.SERVR) LW.ARRIVR=0.99*LW.SERVR    ;
prevent infinite or negative queue
       endif

      endif

      if (li.TOLL_ACC>0)
       LINKCLASS=3 ; Toll Plaza Acceleration link
       T0= T0 + (LI.FREEFLOWSPEED/{ACCELRATE})/60
      endif

      if (li.TOLL_DEC>0)
       LINKCLASS=4 ; Toll Plaza Deceleration link
       T0 = T0 + (LI.FREEFLOWSPEED/DECEL(1,LI.FREEFLOWSPEED))/60
      endif
if (li.toll>maxplzno) maxplzno=li.toll
      IF (LI.FACILITY_TYPE=49) LINKCLASS=5
ENDPHASE

;=================================================================
=========================
```

```
;=== ILOOP (ASSIGNMENT) PHASE ===
;=================================================================
=========================
PHASE=ILOOP
  PATHLOAD PATH=TIME, VOL[1]=MI.1.1, PENI=1, EXCLUDEGROUP=9
ENDPHASE

;=================================================================
=========================
;=== ADJUST PHASE (WITH REPORTING)===
;=================================================================
=========================
; User Specified Functions
; Delay functions, one for congested travel time, the other for
toll links convert to time equivalent
PHASE=ADJUST
 if (time>0) LW.CGSTSPEED=(LI.DISTANCE/TIME)*60
 if (LW.ARRIVR>=LW.SERVR) LW.ARRIVR=0.99*LW.SERVR    ; prevent
infinite or negative queue
 if (li.cartoll>0)
   LW.ARRIVR=(V/LI.UROADFACTOR)*LI.CONFAC/LI.PLZALNSMAX  ; hourly
volume per toll lane ie. arrival rate in vehicles per hour
   LW.SERVT=LI.SVCMINUTES+(LI.SVCSECONDS/60)           ; Plaza
lane service time in minutes per vehicle
   IF (LI.TOLLTYPE=1)
          LW.SERVR=(1/LW.SERVT)*60        ; Plaza lane service
rate in vehicle per hour
          if (LW.ARRIVR>=LW.SERVR) LW.ARRIVR=0.99*LW.SERVR    ;
prevent infinite or negative queue
   ELSE
          LW.SERVR=100
          LW.ARRIVR=1
   ENDIF
print list="****",LI.PLAZADESC,"****"
PRINT LIST='ARRIVR= ',LW.ARRIVR
PRINT LIST='SERVT= ',LW.SERVT
PRINT LIST='SERVR= ',LW.SERVR
PRINT LIST='TOLLTIME= ',TIME
endif
if (lw.servr-lw.arrivr=0.0) lw.servr=lw.servr+0.01
; FUNCTION
TC[1]=LI.TIME*(1+LI.BPRCOEFFICIENT*(MIN(V/C,{VCMAX}))^LI.BPREXPON
ENT) ; congested time for non-toll links
 FUNCTION   TC[2]=(1/(LW.SERVR-LW.ARRIVR))*60 +
{CTOLL}*LI.CARTOLL*60 ; congested time for toll links
 FUNCTION
TC[3]=LI.TIME*(1+LI.BPRCOEFFICIENT*(MIN(V/C,{VCMAX}))^LI.BPREXPON
ENT) + (LW.CGSTSPEED/{ACCELRATE})/60; congested time toll
acceleration links
 FUNCTION
TC[4]=LI.TIME*(1+LI.BPRCOEFFICIENT*(MIN(V/C,{VCMAX}))^LI.BPREXPON
ENT) + (LW.CGSTSPEED/DECEL(1,LW.CGSTSPEED))/60; congested time
toll deceleration links
 FUNCTION   TC[5]=LI.TIME; congested time toll deceleration links
IF (LI.FACILITY_TYPE=90-99)
TOLLVMT[ITERATION]=TOLLVMT[ITERATION]+(LI.DISTANCE*V)
if (linkno=1)
```

```
   print,
        printo=1,
list=" "
   print,
        printo=1,
list=" "
   print,
        printo=1,
list="                        ITERATION ",ITERATION(2.0),"
SUMMARY REPORT ON TOLL LINKS -- CTOLL VALUE =",ctoll(8.7)
   print,
        printo=1,
list=" "
   print,
        printo=1,
list="                                                          No.
ServTim  % Heavy            Delay   Toll     Plan."

   print,
        printo=1,
list="  Class  Type  ..Nodes..    ......... Name .........  Lanes
Toll   (mm:ss)   Trucks   Volume   (Min)   (Min)     Rev.
Util."
   print,
        printo=1,
list=" "
endif
IF (li.toll>0) print,
        printo=1,
list=li.toll(7.0),li.tolltype(5.0),li.a(6.0),li.b(6.0),"
",li.plazadesc(25),li.plzalnsmin(7.0),"
$",LI.CARTOLL(5.2),,li.svcminutes(6.0),":",li.svcseconds(2.0),li.
pcttrucks*100(9.2),v(10.0c),time-
LW.TOLLTIME(8.2),LW.TOLLTIME(8.2C)," $",LI.CARTOLL*V(7.0C)
IF (LI.TOLL>0)
TOLLREVENUE[ITERATION]=TOLLREVENUE[ITERATION]+(Li.CARTOLL*V)
IF (LINKNO=NUMLINKS) PRINT,
        printo=1,
LIST="     *-----------------------
********************************** TOTAL TOLL REVENUE FOR THIS
ITERATION IS    $",TOLLREVENUE[ITERATION](10.0C)
IF (LINKNO=NUMLINKS) PRINT,
        printo=1,
LIST="     *-TOLL REVENUE SUMMARY-
**********************************  TOLL FACILITY VMT FOR THIS
ITERATION IS     ",TOLLVMT[ITERATION](10.0C)
IF (LINKNO=NUMLINKS) PRINT,
        printo=1,
LIST="     *-----------------------******************************
AVERAGE REVENUE PER VMT FOR THIS ITERATION IS
$",TOLLREVENUE[ITERATION]/TOLLVMT[ITERATION](6.5C)

ENDRUN
```

3. PLNET00A.S

```
; Do not change filenames or add or remove FILEI/FILEO statements
using an editor. Use Cube/Application Manager.
RUN PGM=NETWORK PRNFILE="D:\FSUTMS\D4\TCRPM\CUBE\PLNET00A.PRN"
MSG='Adds Transit Speeds'
FILEO PRINTO[1] = "D:\FSUTMS\D4\TCRPM\CUBE\TSPDCG.PRN"
FILEI LOOKUPI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\TSPDS.CSV"
FILEO NETO = "D:\FSUTMS\D4\TCRPM\CUBE\PRELOADS.NET"
FILEI LINKI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\PRELOAD.NET"

;==================
  if (time_1>0)
    _AUTOSPEED=60*DISTANCE/TIME_1
  else
    _AUTOSPEED=15
  ENDIF

  LOOKUP name=curve,
     lookup[1]=1, result=16,
     lookup[2]=2, result=17,
     lookup[3]=3, result=18,
     lookup[4]=4, result=19,
     lookup[5]=5, result=20,
     lookup[6]=6, result=21,
     lookup[7]=7, result=22,
     lookup[8]=8, result=23,
     lookup[9]=9, result=24,
     lookup[10]=10, result=25,
     lookup[11]=11, result=26,
     lookup[12]=12, result=27,
     lookup[13]=13, result=28,
     lookup[14]=14, result=29,
     lookup[15]=15, result=30,
     interpolate=y, list=y, lookupi=1

read,
        FILE = "D:\FSUTMS\D4\TCRPM\CUBE\CURVES.TXT"

     ; APPLY DEFAULT CURVES
     IF (_AUTOSPEED>0)
       _TSPD1=CURVE(_CURVEMODE1,_AUTOSPEED)
       _TSPD4=CURVE(_CURVEMODE4,_AUTOSPEED)
       _TSPD5=CURVE(_CURVEMODE5,_AUTOSPEED)
       _TSPD6=CURVE(_CURVEMODE6,_AUTOSPEED)
       _TSPD8=CURVE(_CURVEMODE8,_AUTOSPEED)
       _TSPD12=CURVE(_CURVEMODE12,_AUTOSPEED)

       LBTIME=60*DISTANCE/_TSPD4
       EBTIME=60*DISTANCE/_TSPD6
       RLTIME=60*DISTANCE/_TSPD8
       SHTIME=60*DISTANCE/_TSPD12

     ENDIF
     ; OVERRIDES FROM NETWORK AND YET MISSING VALUES
     IF (TTIME>0&TMODE= 4) LBTIME=TTIME
```

```
     IF (TTIME>0&TMODE= 6) EBTIME=TTIME
     IF (TTIME>0&TMODE= 8) RLTIME=TTIME
     IF (TTIME>0&TMODE=12) SHTIME=TTIME
     IF (LBTIME=0) LBTIME=99
     IF (EBTIME=0) EBTIME=99
     IF (SHTIME=0) SHTIME=99
     _WKSPEED=60*DISTANCE/LBTIME
     _LBSPEED=60*DISTANCE/LBTIME
     _EBSPEED=60*DISTANCE/EBTIME
     _RLSPEED=60*DISTANCE/RLTIME
     _SHSPEED=60*DISTANCE/SHTIME

; WALK CONTROLS
     WALKSPEED=CURVE(_CURVEMODE1,2.5)
     WALKTIME=60*DISTANCE/WALKSPEED
     IF (FACILITY_TYPE=10-19,70-99)
        WALKDISTANCE=9.99
        WALKTIME=9.99
     ENDIF

PRINT LIST="A=",A(4.0)," B=",B(4.0),
           "  SPEEDS-  Auto=",_AUTOSPEED(4.1),
           " Walk=",WALKSPEED(4.1),
           " Local Bus=",_LBSPEED(4.1),
           " Express Bus=",_EBSPEED(4.1),
           " Shuttle Bus=",_SHSPEED(4.1),
           " Rail=",_RLSPEED(4.1),
                           PRINTO=1
;==================

ENDRUN
```

4. PLHWY00B.S

```
; Do not change filenames or add or remove FILEI/FILEO statements
using an editor. Use Cube/Application Manager.
RUN PGM=HIGHWAY PRNFILE="D:\FSUTMS\D4\TCRPM\CUBE\PLHWY00B.PRN"
MSG='Build Congested Flow Skims (LOV & HOV)'
FILEI LOOKUPI[2] = "D:\FSUTMS\D4\TCRPM\CUBE\ACTIVECENTROIDS.CSV"
FILEI LOOKUPI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\EXTERNALTT.TXT"
FILEI TURNPENI = "D:\FSUTMS\D4\TCRPM\CUBE\TCARDS.PRN"
FILEI NETI = "D:\FSUTMS\D4\TCRPM\CUBE\PRELOADS.NET"
FILEO MATO[1] = "D:\FSUTMS\D4\TCRPM\CUBE\RHSKIMS.MAT",
  MO=1-7,30
NAME=COST,DISTANCE,TIME1,TIME2,SVCTIME,SVCMINUTES,TOLLTIMEEQ,TERM
INALTIME

ZONEMSG=100
ARRAY TERM=99,ATYPE={ZONESA}
; cbd
term[11]={term11}
term[12]={term12}
term[13]={term13}
; cbd fringe
term[21]={term21}
; other urban
```

```
term[31]={term31}
term[32]={term32}
term[33]={term33}
term[34]={term34}
; obd
term[41]={term41}
term[42]={term42}
term[43]={term43}
; rural
term[51]={term51}
term[52]={term52}

PROCESS PHASE=LINKREAD
       IF (A<={ZONESA}) ATYPE[A]=LI.AREA_TYPE ; ASSIGNS AN AREA
TYPE TO EACH ZONE BASED ON LAST ENTRY (LIKE TTPREP)
; use time for impedance unless toll present, then convert toll +
service time to time equivalent
       if (li.facility_type=99)

lw.linkcost=(li.cartoll*{ctoll}*60)+li.svcminutes+(li.svcseconds/
60)
       else
          lw.linkcost=li.time_1
       endif
       if (li.facility_type=80-89) addtogroup=1
ENDPROCESS


PROCESS PHASE=ILOOP
lookup, name=activec, lookup[1]=1, result=2, lookupi=2
active=activec(1,i)
if (active=1)

; first, skim the network based on travel time, LOV
      PATH=LW.LINKCOST,peni=1,excludegroup=1,
      MW[01]=PATHTRACE(LI.CARTOLL),noaccess=999.99,dec=2,
      MW[02]=PATHTRACE(LI.DISTANCE),noaccess=999.99,dec=2,
      MW[03]=PATHTRACE(LI.TIME_1),noaccess=999.99,dec=2,
      MW[04]=PATHTRACE(LI.TIME_1),noaccess=999.99,dec=2,
      MW[05]=PATHTRACE(LI.SVCMINUTES),noaccess=999.99,dec=2,
      MW[06]=PATHTRACE(LI.SVCSECONDS),noaccess=999.99,dec=2
; next, convert the toll skims to travel time equivalent
      MW[07]=(MW[01]*{CTOLL}*60)+MW[05]+(MW[06]/60)
; next, create a terminal time matrix
lookup name=extt, lookup[1]=1, result=2,
interpolate=n,fail[1]=0,fail[2]=0, lookupi=1
JLOOP
      if (i<={zonesi}&j<={zonesi}); internal to internal
        mw[30]=(term[atype[i]]+term[atype[j]])
      endif
      if (i>{zonesi}&j<={zonesi}) ; external to internal
        ttime=extt(1,i)
        mw[30]=(ttime+term[atype[j]])
      endif
      if (i<={zonesi}&j>{zonesi}) ; internal to external
        ttime=extt(1,j)
        mw[30]=(term[atype[i]]+ttime)
```

```
      endif
      if (i>{zonesi}&j>{zonesi}) ; external to external
        ttimei=extt(1,i)
        ttimej=extt(1,j)
        mw[30]=(ttimei+ttimej)
      endif
ENDJLOOP

; lastly, treat intrazonals
  MW[01][i] = 0  ; Intrazonal time HOV
  MW[02][i] = ROWMIN(2)/2  ; Intrazonal time HOV
  MW[03][i] = ROWMIN(3)/2  ; Intrazonal time HOV
  MW[04][i] = ROWMIN(4)/2  ; Intrazonal time HOV
  MW[05][i] = 0  ; Intrazonal time HOV
  MW[06][i] = 0  ; Intrazonal time HOV
  MW[07][i] = ROWMIN(7)/2  ; Intrazonal time HOV
LIST = "AREA TYPE FOR ZONE",I(3.0)," IS",ATYPE[I](3.0)
endif
ENDPHASE

ENDRUN
```

5. PLMAT00C.S

```
; Do not change filenames or add or remove FILEI/FILEO statements
using an editor. Use Cube/Application Manager.
RUN PGM=MATRIX PRNFILE="D:\FSUTMS\D4\TCRPM\CUBE\PLMAT00C.PRN"
MSG='Compile Skims for GM and Mode'
FILEI LOOKUPI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\ACTIVECENTROIDS.CSV"
FILEI MATI[3] = "D:\FSUTMS\D4\TCRPM\CUBE\FHSKIMS.MAT"
FILEI MATI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\RHSKIMS.MAT"
FILEO MATO[1] = "D:\FSUTMS\D4\TCRPM\CUBE\RHSKIMTT.{ALT}{YEAR}",
MO=1-4,NAME=COST,DISTANCE,TIME,TIME2,FORMAT=TRANPLAN
lookup, name=activec, lookup[1]=1, result=2, lookupi=1
active=activec(1,i)
if (active=1)
ZONEMSG=100
; EQUIVALENT OF WHAT GOES INTO GM
MW[1]=MI.1.COST*100
MW[2]=MI.1.DISTANCE*100
MW[3]=MI.1.TIME1*100
MW[4]=(MI.1.TIME1+mi.1.TOLLTIMEEQ+mi.1.TERMINALTIME)*100
MW[5]=(MI.1.TIME1+mi.1.TOLLTIMEEQ)*100

; EQUIVALENT OF WHAT GOES INT0 MODE CHOICE LOV
MW[11]=MI.1.COST*100
MW[12]=MI.1.DISTANCE*100
MW[13]=MI.1.TIME1*100
MW[14]=MI.1.TIME1*100


MW[41]=MI.1.TIME2*100     ; HSKIMS.{ALT}{YEAR}, CONGESTED NO HOV
Facilities
MW[42]=MI.1.DISTANCE*100
MW[43]=MI.1.COST*100
MW[44]=MI.03.004
```

```
MW[45]=MI.03.002
MW[46]=MI.03.001


loop ww=41
  mw[ww][i]=0
endloop

  mintime1=rowmin(41)
  mintime2=rowmin(44)

jloop ; sets intrazonal distance to 1/2 the distance to the
nearest zone by time
  if (mw[41]=mintime1) mw[42][i]=mw[42][j]/2
  if (mw[44]=mintime1) mw[45][i]=mw[45][j]/2
endjloop

   mw[41][i]=mintime1/2
   mw[44][i]=mintime2/2

if (mw[41][i]>200) mw[41][i]=200
if (mw[44][i]>200) mw[44][i]=200

endif

ENDRUN
```

6. PLMAT00J.S

```
; Do not change filenames or add or remove FILEI/FILEO statements
using an editor. Use Cube/Application Manager.
RUN PGM=MATRIX PRNFILE="D:\FSUTMS\D4\TCRPM\CUBE\PLMAT00J.PRN"
MSG='Put Gamma Values in Matrix'
FILEI LOOKUPI[2] = "D:\FSUTMS\D4\TCRPM\CUBE\ACTIVECENTROIDS.CSV"
FILEI LOOKUPI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\GAMMA.CSV"
FILEI MATI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\RHSKIMTT.{ALT}{YEAR}"
FILEO MATO[1] = "D:\FSUTMS\D4\TCRPM\CUBE\PLMAT00B.MAT",
 MO=21-
31,NAME=GAMMA1,GAMMA2,GAMMA3,GAMMA4,GAMMA5,GAMMA6,GAMMA7,GAMMA8,G
AMMA9,GAMMA10,GAMMA11
PAR ZONEMSG=100
lookup, name=gamma, lookup[1]=1, result=2, lookup[2]=1, result=3,
lookup[3]=1, result=4, LOOKUPI=1
lookup, name=activec, lookup[1]=1, result=2,LOOKUPI=2
mw[51]=mi.1.4/100
active=activec(1,i)
if (active=1)
  LOOP PRP=1,11
      a=GAMMA(1,PRP)
      b=GAMMA(2,PRP)
      c=GAMMA(3,PRP)
      GSKIM=PRP+20 ; Output Gamma Skim
      JLOOP
          if (MW[51]>0) mw[GSKIM]=(a*MW[51]^(-b))*exp((-
c)*MW[51]) ; PUT GAMMA MATRICES IN MW[21]-MW[24]
```

```
         if (i=1001&j=1010) PRINT CSV=T,
LIST=I,J,MW[51],A,B,C,MW[GSKIM]
      ENDJLOOP
  ENDLOOP
endif
ENDRUN
```

7. PLDST00A.S

```
; Do not change filenames or add or remove FILEI/FILEO statements
using an editor. Use Cube/Application Manager.
RUN PGM=DISTRIBUTION
PRNFILE="D:\FSUTMS\D4\TCRPM\CUBE\PLDST00A.PRN"
FILEI ZDATI[1] = "{SCENARIO_DIR}\PANDA2.DBF"
FILEI MATI[3] = "D:\FSUTMS\D4\TCRPM\CUBE\FHSKIMS.MAT"
FILEI MATI[2] = "D:\FSUTMS\D4\TCRPM\CUBE\KFACTOR.MAT"
FILEI LOOKUPI[2] = "D:\FSUTMS\D4\TCRPM\CUBE\ACTIVECENTROIDS.CSV"
FILEI LOOKUPI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\GAMMA.CSV"
FILEI MATI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\PLMAT00B.MAT"
FILEO MATO[1] = "D:\FSUTMS\D4\TCRPM\CUBE\PLDST00A.MAT",
 MO=1-
11,NAME=HBW,HBSH,HBSR,HBSCPRV,HBO,NHBW,NHBO,AIRPORT,TRUCK4TIRE,TR
KSNGLUNIT,TRACTORTRAILER, dec=11*6
PAR ZONEMSG=100 MAXITERS={atiter}, MAXRMSE=1

 LOOKUP NAME=GAMMA,; Gamma Function Parameters
    LOOKUP[1]=1, RESULT=2,           ; CONSTANT
    LOOKUP[2]=1, RESULT=3,           ; Alpha VALUE
    LOOKUP[3]=1, RESULT=4,           ; BETA VALUE
    LookupI=1
LOOKUP NAME=ACTIVE, LOOKUP[1]=1, RESULT=2, LOOKUPI=2
SETPA P[1]=ZI.1.P1, A[1]=ZI.1.A1
SETPA P[2]=ZI.1.P2, A[2]=ZI.1.A2
SETPA P[3]=ZI.1.P3, A[3]=ZI.1.A3
SETPA P[4]=ZI.1.P4, A[4]=ZI.1.A4
SETPA P[5]=ZI.1.P5, A[5]=ZI.1.A5
SETPA P[6]=ZI.1.P6, A[6]=ZI.1.A6
SETPA P[7]=ZI.1.P7, A[7]=ZI.1.A7
SETPA P[8]=ZI.1.P7, A[8]=ZI.1.A7 ; airport is not working in
TCRPM trip gen
SETPA P[9]=ZI.1.P9, A[9]=ZI.1.A9
SETPA P[10]=ZI.1.P10, A[10]=ZI.1.A10
SETPA P[11]=ZI.1.P11, A[11]=ZI.1.A11

MW[21]=MI.1.1
MW[22]=MI.1.2
MW[23]=MI.1.3
MW[24]=MI.1.4
MW[25]=MI.1.5
MW[26]=MI.1.6
MW[27]=MI.1.7
MW[28]=MI.1.8
MW[29]=MI.1.9
MW[30]=MI.1.10
MW[31]=MI.1.11
MW[41]=MI.2.1
```

```
MW[50]=MI.3.4/100

ACTIVEC=ACTIVE(1,I)
IF (ACTIVEC=1) ;ONLY FOR ACTIVE CENTROIDS

LOOP PURP=1,11
   PAF=0
   MW[PURP] = A[PURP] * MW[PURP+20] * MW[41]
   ATTRSUM=ROWSUM(PURP)
   IF (ATTRSUM>0) PAF=P[PURP]/ATTRSUM
   MW[PURP]=PAF * MW[PURP]
ENDLOOP

FREQUENCY  BASEMW=50,VALUEMW=1, RANGE=0-80-1,TITLE='HBW TRAVEL
TIME FREQUENCY'
FREQUENCY  BASEMW=50,VALUEMW=2, RANGE=0-80-1,TITLE='HBSH TRAVEL
TIME FREQUENCY'
FREQUENCY  BASEMW=50,VALUEMW=3, RANGE=0-80-1,TITLE='HBSR TRAVEL
TIME FREQUENCY'
FREQUENCY  BASEMW=50,VALUEMW=4, RANGE=0-80-1,TITLE='HBSC-PRIVATE
TRAVEL TIME FREQUENCY'
FREQUENCY  BASEMW=50,VALUEMW=5, RANGE=0-80-1,TITLE='HBO TRAVEL
TIME FREQUENCY'
FREQUENCY  BASEMW=50,VALUEMW=6, RANGE=0-80-1,TITLE='NHBW TRAVEL
TIME FREQUENCY'
FREQUENCY  BASEMW=50,VALUEMW=7, RANGE=0-80-1,TITLE='NHBO TRAVEL
TIME FREQUENCY'
;FREQUENCY  BASEMW=50,VALUEMW=8, RANGE=0-80-1,TITLE='AIRPORT
TRAVEL TIME FREQUENCY'
FREQUENCY  BASEMW=50,VALUEMW=9, RANGE=0-80-1,TITLE='FOUR-TIRE
TRUCK TRAVEL TIME FREQUENCY'
FREQUENCY  BASEMW=50,VALUEMW=10, RANGE=0-80-1,TITLE='SINGLE UNIT
TRUCK TRAVEL TIME FREQUENCY'
FREQUENCY  BASEMW=50,VALUEMW=11, RANGE=0-80-1,TITLE='COMBINATION
UNIT TRUCK TRAVEL TIME FREQUENCY'

MW[8]=0
ENDIF
ENDRUN
```

8. PLMAT00D.S

```
; Do not change filenames or add or remove FILEI/FILEO statements
using an editor. Use Cube/Application Manager.
RUN PGM=MATRIX PRNFILE="D:\FSUTMS\D4\TCRPM\CUBE\PLMAT00D.PRN"
MSG='Develops Weighted Trip Table'
FILEI MATI[4] = "D:\FSUTMS\D4\TCRPM\CUBE\FHSKIMS.MAT"
FILEI LOOKUPI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\ACTIVECENTROIDS.CSV"
FILEO MATO[1] = "D:\FSUTMS\D4\TCRPM\CUBE\WTDTRIP.MAT",
 mo=1-11,
name=hbwsk,hbshsk,hbsrsk,hbscsk,hbosk,nhbwsk,nhbosk,airportsk,sut
rucksk,mutrucksk,trktrlsk
FILEI MATI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\PTRIPS1.MAT"
FILEI MATI[2] = "D:\FSUTMS\D4\TCRPM\CUBE\PLDST00A.MAT"

MW[50]=MI.4.4/100
```

```
lookup, name=activec, lookup[1]=1, result=2, lookupi=1
active=activec(1,i)
if (active=1)
par zonemsg=100
jloop
MW[1]=MI.1.1*{WFHBW}  +MI.2.1*{WRHBW}
MW[2]=MI.1.2*{WFHBSHP}+MI.2.2*{WRHBSHP}
MW[3]=MI.1.3*{WFHBSR} +MI.2.3*{WRHBSR}
MW[4]=MI.1.4*{WFHBSCH}+MI.2.4*{WRHBSCH}
MW[5]=MI.1.5*{WFHBO}  +MI.2.5*{WRHBO}
MW[6]=MI.1.6*{WFNHBW} +MI.2.6*{WRNHBW}
MW[7]=MI.1.7*{WFNHBO} +MI.2.7*{WRNHBO}
MW[8]=MI.1.8*{WFARPRT}+MI.2.8*{WRARPRT}
MW[9]=MI.1.9*{WFTRK}  +MI.2.9*{WRTRK}
MW[10]=MI.1.10*{WFTRK}  +MI.2.10*{WRTRK}
MW[11]=MI.1.11*{WFTRK}  +MI.2.11*{WRTRK}
endjloop
endif


FREQUENCY  BASEMW=50,VALUEMW=1, RANGE=0-80-1,TITLE='HBW TRAVEL
TIME FREQUENCY'
FREQUENCY  BASEMW=50,VALUEMW=2, RANGE=0-80-1,TITLE='HBSH TRAVEL
TIME FREQUENCY'
FREQUENCY  BASEMW=50,VALUEMW=3, RANGE=0-80-1,TITLE='HBSR TRAVEL
TIME FREQUENCY'
FREQUENCY  BASEMW=50,VALUEMW=4, RANGE=0-80-1,TITLE='HBSC-PRIVATE
TRAVEL TIME FREQUENCY'
FREQUENCY  BASEMW=50,VALUEMW=5, RANGE=0-80-1,TITLE='HBO TRAVEL
TIME FREQUENCY'
FREQUENCY  BASEMW=50,VALUEMW=6, RANGE=0-80-1,TITLE='NHBW TRAVEL
TIME FREQUENCY'
FREQUENCY  BASEMW=50,VALUEMW=7, RANGE=0-80-1,TITLE='NHBO TRAVEL
TIME FREQUENCY'
;FREQUENCY  BASEMW=50,VALUEMW=8, RANGE=0-80-1,TITLE='AIRPORT
TRAVEL TIME FREQUENCY'
FREQUENCY  BASEMW=50,VALUEMW=9, RANGE=0-80-1,TITLE='FOUR-TIRE
TRUCK TRAVEL TIME FREQUENCY'
FREQUENCY  BASEMW=50,VALUEMW=10, RANGE=0-80-1,TITLE='SINGLE UNIT
TRUCK TRAVEL TIME FREQUENCY'
FREQUENCY  BASEMW=50,VALUEMW=11, RANGE=0-80-1,TITLE='COMBINATION
UNIT TRUCK TRAVEL TIME FREQUENCY'

ENDRUN
```

9. PLMAT00H.S

```
; Do not change filenames or add or remove FILEI/FILEO statements
using an editor. Use Cube/Application Manager.
RUN PGM=MATRIX PRNFILE="D:\FSUTMS\D4\TCRPM\CUBE\PLMAT00H.PRN"
MSG='Create Vehicle Hours Calculation'
FILEI LOOKUPI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\ACTIVECENTROIDS.CSV"
FILEI MATI[2] = "D:\FSUTMS\D4\TCRPM\CUBE\RHSKIMTT.{ALT}{YEAR}"
FILEI MATI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\WTDTRIP.MAT"

PAR ZONEMSG=100
```

```
lookup, name=activec, lookup[1]=1, result=2, lookupi=1
active=activec(1,i)
if (active=1)

MW[21]=MI.2.4/6000
MW[22]=MI.2.4/6000
MW[23]=MI.2.4/6000
MW[24]=MI.2.4/6000
MW[25]=MI.2.4/6000
MW[26]=MI.2.4/6000
MW[27]=MI.2.4/6000
MW[28]=MI.2.4/6000
MW[29]=MI.2.4/6000

MW[1]=MI.1.1*MW[21]
MW[2]=MI.1.2*MW[22]
MW[3]=MI.1.3*MW[23]
MW[4]=MI.1.4*MW[24]
MW[5]=MI.1.5*MW[25]
MW[6]=MI.1.6*MW[26]
MW[7]=MI.1.7*MW[27]
MW[8]=MI.1.8*MW[28]
MW[9]=MI.1.9*MW[29]
MW[10]=MI.1.10*MW[29]
MW[11]=MI.1.11*MW[29]
endif

ENDRUN
```

10. PLMAT00E.S

```
; Do not change filenames or add or remove FILEI/FILEO statements
using an editor. Use Cube/Application Manager.
RUN PGM=MATRIX PRNFILE="D:\FSUTMS\D4\TCRPM\CUBE\PLMAT00F.PRN"
MSG='Assemble Person & Vehicle Trips for Mode Choice'
FILEI LOOKUPI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\ACTIVECENTROIDS.CSV"
FILEI MATI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\WTDTRIP.MAT"
FILEO MATO[3] = "D:\FSUTMS\D4\TCRPM\CUBE\APORTVEH.MAT",
 mo=12, name=airportvehicles
FILEO MATO[2] = "D:\FSUTMS\D4\TCRPM\CUBE\MODEIN.TEM",
 MO=21-23,NAME=HBW,HBO,NHB,FORMAT=TRANPLAN
FILEO MATO[1] = "D:\FSUTMS\D4\TCRPM\CUBE\PTRIPS11CG.MAT",
 MO=1-
11,NAME=HBW,HBSH,HBSR,HBSCPR,HBO,NHBW,NHBO,AIRPORT,TRUCK4TIRE,TRK
SNGLUNIT,TRACTORTRAILER
PAR ZONEMSG=100
lookup, name=activec, lookup[1]=1, result=2, lookupi=1
active=activec(1,i)
if (active=1)

ARRAY BUCKET=12
JLOOP
  IF (I=1&J=1)
    BUCKET[1]=0.5
    BUCKET[2]=0.5
    BUCKET[3]=0.5
```

```
  BUCKET[4]=0.5
  BUCKET[5]=0.5
  BUCKET[6]=0.5
  BUCKET[7]=0.5
  BUCKET[8]=0.5
  BUCKET[9]=0.5
  BUCKET[10]=0.5
  BUCKET[11]=0.5
ENDIF
HBW=MI.1.1+BUCKET[1]
MW[1]=INT(HBW)
BUCKET[1]=HBW-MW[1]

HBSH=MI.1.2+BUCKET[2]
MW[2]=INT(HBSH)
BUCKET[2]=HBSH-MW[2]

HBSR=MI.1.3+BUCKET[3]
MW[3]=INT(HBSR)
BUCKET[3]=HBSR-MW[3]

HBSC=MI.1.4+BUCKET[4]
MW[4]=INT(HBSC)
BUCKET[4]=HBSC-MW[4]

HBO=MI.1.5+BUCKET[5]
MW[5]=INT(HBO)
BUCKET[5]=HBO-MW[5]

NHBW=MI.1.6+BUCKET[6]
MW[6]=INT(NHBW)
BUCKET[6]=NHBW-MW[6]

NHBO=MI.1.7+BUCKET[7]
MW[7]=INT(NHBO)
BUCKET[7]=NHBO-MW[7]

AIR=MI.1.8+BUCKET[8]
MW[8]=INT(AIR)
BUCKET[8]=AIR-MW[8]

TCA=MI.1.9+BUCKET[9]
MW[9]=INT(TCA)
BUCKET[9]=TCA-MW[9]

TCB=MI.1.10+BUCKET[10]
MW[10]=INT(TCB)
BUCKET[10]=TCB-MW[10]

TCC=MI.1.11+BUCKET[11]
MW[11]=INT(TCC)
BUCKET[11]=TCC-MW[11]

ARV=MI.1.8*{AOFAC8}+BUCKET[12]
MW[12]=INT(ARV)
BUCKET[12]=ARV-MW[12]
```

```
ENDJLOOP
endif
; PREPARE TABLES FOR MODE CHOICE
MW[21]=MW[1]
MW[22]=MW[2]+MW[3]+MW[4]+MW[5]
MW[23]=MW[6]+MW[7]

ENDRUN
```

TRANSIT NETWORK

(Scripts relevant to current modes have only been shown here)

1. PTMAT00A.S

```
; Do not change filenames or add or remove FILEI/FILEO statements
using an editor. Use Cube/Application Manager.
RUN PGM=MATRIX MSG='Builds Connector Criteria'
FILEO PRINTO[7] = "D:\FSUTMS\D4\TCRPM\CUBE\STATNO.TXT"
FILEO PRINTO[6] = "D:\FSUTMS\D4\TCRPM\CUBE\SIDEWALKSMD.TXT"
FILEO PRINTO[5] = "D:\FSUTMS\D4\TCRPM\CUBE\SIDEWALKSAM.TXT"
FILEO PRINTO[4] = "D:\FSUTMS\D4\TCRPM\CUBE\STATIONMD.TXT"
FILEO PRINTO[3] = "D:\FSUTMS\D4\TCRPM\CUBE\STATIONAM.TXT"
FILEI LOOKUPI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\NODES.CSV"

FILEI RECI = "{DATADIR}\STATDATA.{YEAR}{ALT}",
 STATNO=1-4, STATNODE=6-10,STATZONE=13-16, SVCAREA=19-22,
PARKINGSPACES=24-28, AMCOST=29-34, MDCOST=35-40,
 TERMPNR=41-46, TERMKNR=47-52, AMUSEFLAG=55, DESCRIPTION=57-
79,farezone=81-82;,
;         fld=1-4,6-10,13-16,19-22,24-28,29-34,35-40,41-46,47-
52,54,55,57-79,81-82
par zones={zonesa}
ZONESA1={ZONESA}+1
ARRAY STATSTOP=99999 STATNUMB=99999, statspaces=99999,
PNRTERM=99999, KNRTERM=99999, nrz=99999,fzone=99999,
amcst=99999, mdcst=99999, statzn=99999
VALUEOFTIME=15.00 ; DOLLARS PER HOUR USED TO ADD TIME EQUIVALENT
OF PARKING COST TO AUTO CONNECTORS
w=' ; '
;STATNAME=SUBSTR(RECI,57,23)
PRINT LIST=RI.DESCRIPTION
; ALLOW PT TO IDENTIFY STATION NUMBER VS. STATION NODE NUMBER
STATSTOP[RI.STATNODE]=1, STATNUMB[RI.STATNODE]=RI.STATNO,
STATSPACES[RI.STATNODE]=RI.PARKINGSPACES,
PNRTERM[RI.STATNODE]=RI.TERMPNR, KNRTERM[RI.STATNODE]=RI.TERMKNR,
fzone[RI.STATNODE]=RI.FAREZONE
statzn[ri.statnode]=ri.statzone
amcst[ri.statnode]=ri.amcost
mdcst[ri.statnode]=ri.mdcost
; input record counter
IF (RI.AMUSEFLAG=1)
 PRINT
form=5.0,list="GENERATE,COST=(li.distance),",w,RI.DESCRIPTION,"\n
MINCOST=255*1.0,MAXCOST=255*",ri.svcarea(5.2L),
   ",EXTRACTCOST=(li.TIME_1),LIST=F,","\n
DIRECTION=1,NTLEGMODE=2,FROMNODE=1-
{ZONESA},TONODE=",RI.STATNODE(5.0L),PRINTO=3
 PRINT
form=5.0,list="GENERATE,COST=(LW.WALKDISTANCE),",W,RI.DESCRIPTION
,"\n
EXTRACTCOST=(LW.WALKTIME),MAXCOST=255*{MAXD},LIST=F,NTLEGMODE=3,O
NEWAY=F,",
      "\n  FROMNODE=",RI.STATNODE,", TONODE=",ZONESA1(4.0L),"-
99999",PRINTO=5
```

```
endif
mduseflag=ri.amuseflag
IF (mduseflag=1)
 PRINT
form=5.0,list="GENERATE,COST=(li.distance),",w,RI.DESCRIPTION,"\n
MINCOST=255*1.0,MAXCOST=255*",ri.svcarea(5.2L),
   ",EXTRACTCOST=(li.TIME),LIST=F,","\n
DIRECTION=1,NTLEGMODE=2,FROMNODE=1-
{ZONESA},TONODE=",RI.STATNODE(5.0L),PRINTO=4
 PRINT
form=5.0,list="GENERATE,COST=(LW.WALKDISTANCE),",W,RI.DESCRIPTION
,"\n
EXTRACTCOST=(LW.WALKTIME),MAXCOST=255*{MAXD},LIST=F,NTLEGMODE=3,O
NEWAY=F,",
       "\n  FROMNODE=",RI.STATNODE,", TONODE=",ZONESA1(4.0L),"-
99999",PRINTO=6
endif

; put nodes, x and y coordinates into memory for lookup nearest
TAZ question
lookup lookupi=1,name=netcoord, lookup[1]=1, result=2,
lookup[2]=1, result=3, fail=0

workstat=ri.statnode
workstatx=netcoord(1,workstat,0)
workstaty=netcoord(2,workstat,0)
mindist=999.99
loop ww=1,zones
  zx=netcoord(1,ww,0)
  zy=netcoord(2,ww,0)
  if (ww!=workstat) dist=sqrt((workstatx-zx)^2+(workstaty-
zy)^2)/{units}
  if (dist<mindist) mindist=dist, nearestzone=ww
;  print form=9.0 list=workstat,workstatx,workstaty,ww,
zx,zy,dist(6.2),mindist(6.2),nearestzone
endloop
nrz[ri.statnode]=nearestzone

; NOTE: i is an internal program flag that indicates if the RECI
input file is still being read, 1 if still reading, 0 if complete
if (i=0)
  loop STN=1,99999
   if (statstop[stn]>0) print
list=stn(6.0),statnumb[stn](6.0),STATSPACES[stn](6.0),PNRTERM[stn
](6.0),KNRTERM[stn](6.0),nrz[stn](8.0),fzone[stn],
statzn[stn](7.0),amcst[stn](8.0),mdcst[stn](8.0),
" station, statnumber,
spaces,pnrterm,knrterm,nearestzone(distance),farezone,nearestzone
(statdata input),ampnrcost,mdpnrcost",   printo=7
  endloop
endif


ENDRUN
```

2. PTPTR00A.S

```
; Do not change filenames or add or remove FILEI/FILEO statements
using an editor. Use Cube/Application Manager.
RUN PGM=PUBLIC TRANSPORT
PRNFILE="D:\FSUTMS\D4\TCRPM\CUBE\PTPTR00A.PRN" MSG='Peak with
Walk Access'
FILEI LOOKUPI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\ACTIVECENTROIDS.CSV"
FILEO LINKO = "D:\FSUTMS\D4\TCRPM\CUBE\TWALKAM.DBF"
FILEI NETI = "D:\FSUTMS\D4\TCRPM\CUBE\PRELOADS.NET"
FILEO ROUTEO[4] = "D:\FSUTMS\D4\TCRPM\CUBE\TROUTEAM4.RTE",
 REPORTI=1-{ZONESA}, REPORTJ={CBDZONE}
FILEO ROUTEO[3] = "D:\FSUTMS\D4\TCRPM\CUBE\TROUTEAM3.RTE",
 REPORTI=1-{ZONESA}, REPORTJ={CBDZONE}
FILEO ROUTEO[2] = "D:\FSUTMS\D4\TCRPM\CUBE\TROUTEAM2.RTE",
 REPORTI=1-{ZONESA}, REPORTJ={CBDZONE}
FILEO MATO[4] = "D:\FSUTMS\D4\TCRPM\CUBE\TSKIMAM4.MAT",
 MO=1-
13,NAME=TIMEM1,TIMEM2,TIMEM3,TIMEM4_12,TIMEM5,TIMEM6,TIMEM7,TIMEM
8,TRANSFERS,FIRSTWAIT,SECONDWAIT,TOTALTIME,FARE
FILEO MATO[3] = "D:\FSUTMS\D4\TCRPM\CUBE\TSKIMAM3.MAT",
 MO=1-
13,NAME=TIMEM1,TIMEM2,TIMEM3,TIMEM4_12,TIMEM5,TIMEM6,TIMEM7,TIMEM
8,TRANSFERS,FIRSTWAIT,SECONDWAIT,TOTALTIME,FARE
FILEO MATO[2] = "D:\FSUTMS\D4\TCRPM\CUBE\TSKIMAM2.MAT",
 MO=1-
13,NAME=TIMEM1,TIMEM2,TIMEM3,TIMEM4_12,TIMEM5,TIMEM6,TIMEM7,TIMEM
8,TRANSFERS,FIRSTWAIT,SECONDWAIT,TOTALTIME,FARE
FILEI FACTORI[4] = "{datadir}\TNETAM4.FAC"
FILEI FACTORI[3] = "{datadir}\TNETAM3.FAC"
FILEI FACTORI[2] = "{datadir}\TNETAM2.FAC"
FILEI FACTORI[1] = "{datadir}\TNETAM1.FAC"
FILEO ROUTEO[1] = "D:\FSUTMS\D4\TCRPM\CUBE\TROUTEAM1.RTE",
 REPORTI=1-{ZONESA}, REPORTJ={CBDZONE}
FILEI LINEI[1] = "{DATADIR}\TROUTE.LIN"
FILEI FAREI = "{DATADIR}\BASEFARES.FAR"
FILEI SYSTEMI = "{DATADIR}\SYSTEM.PTS"
FILEO MATO[1] = "D:\FSUTMS\D4\TCRPM\CUBE\TSKIMAM1.MAT",
 MO=1-
13,NAME=TIMEM1,TIMEM2,TIMEM3,TIMEM4_12,TIMEM5,TIMEM6,TIMEM7,TIMEM
8,TRANSFERS,FIRSTWAIT,SECONDWAIT,TOTALTIME,FARE
FILEI TURNPENI = "D:\FSUTMS\D4\TCRPM\CUBE\TCARDS.PRN"
FILEO PRINTO[1] = "D:\FSUTMS\D4\TCRPM\CUBE\PTPTR00C.PRN"
FILEO NETO = "D:\FSUTMS\D4\TCRPM\CUBE\TNETAMWALK.NET"
FILEO REPORTO = "D:\FSUTMS\D4\TCRPM\CUBE\TNETAMWALK.PRN"
; OVERALL PARAMETERS OF RUN
PARAMETERS USERCLASSES=1-4,FARE=N,MAPSCALE={UNITS},
           NOROUTEERRS=999999, NOROUTEMSGS=999999,
           TRANTIME=LW.TRANTIME,
           TRANTIME[4]=LI.LBTIME,
           TRANTIME[5]=LI.RLTIME,
           TRANTIME[6]=LI.EBTIME,
           TRANTIME[8]=LI.RLTIME,
           TRANTIME[12]=LI.SHTIME
REPORT LINES=T
```

```
PROCESS PHASE=LINKREAD
 LW.TRANTIME=LI.LBTIME
 LW.WALKTIME=LI.WALKTIME
 LW.WALKDISTANCE=LI.DISTANCE
ENDPROCESS

PROCESS PHASE=DATAPREP
; WALK ACCESS
  GENERATE, COST=(LW.WALKTIME),MAXCOST=4*24,LIST=T,NTLEGMODE =
1,MAXNTLEGS=4,
            DIRECTION=1,  FROMNODE=1-{zonesa}, TONODE=944-99999
; WALK EGRESS
  GENERATE, COST=(LW.WALKTIME),MAXCOST=4*24,LIST=T,NTLEGMODE =
101,MAXNTLEGS=4,
            DIRECTION=2,  FROMNODE=1-{zonesa}, TONODE=944-99999
; SIDEWALKS
READ,
        FILE = "D:\FSUTMS\D4\TCRPM\CUBE\SIDEWALKSAM.TXT"


ENDPROCESS

PROCESS PHASE=SKIMIJ
lookup, name=activec, lookup[1]=1, result=2,LOOKUPI=1
active=activec(1,i)
if (active=1)

  MW[1]=TIMEA(0,101)+TIMEA(0,1)
  MW[2]=TIMEA(0,2)
  MW[3]=TIMEA(0,3)
  MW[4]=TIMEA(0,4)+TIMEA(0,12)
  MW[5]=TIMEA(0,5)
  MW[6]=TIMEA(0,6)
  MW[7]=TIMEA(0,7)
  MW[8]=TIMEA(0,8)
  MW[9]=BRDINGS(0,TMODES)-1
  MW[10]=IWAITA(0)
  MW[11]=XWAITA(0)
  MW[12]=TIMEA(0,ALLMODES)+MW[10]+MW[11]
  MW[13]=FAREA(0,ALLMODES)
   IF (MW[9]<0) MW[9]=0
endif
ENDPROCESS

ENDRUN
```

3. PTMAT00F.S

```
; Do not change filenames or add or remove FILEI/FILEO statements
using an editor. Use Cube/Application Manager.
RUN PGM=MATRIX PRNFILE="D:\FSUTMS\D4\TCRPM\CUBE\PTMAT00F.PRN"
MSG='Generate CODWAM.{ALT}{YEAR}'
FILEO PRINTO[1] = "D:\FSUTMS\D4\TCRPM\CUBE\CODWAM.{ALT}{YEAR}"
FILEI RECI = "D:\FSUTMS\D4\TCRPM\CUBE\TWALKAM.DBF"
```

```
ARRAY COST={ZONESA}, MINCOST={ZONESA},MAXCOST={ZONESA},
CONNECTORCNT={ZONESA}

; CULL OUT THE ACCESS LINKS
ANODE=RI.A
IF (ANODE=1-{ZONESA}&RI.MODE=1)
  CST=RI.TIME
  CONNECTORCNT[ANODE]=CONNECTORCNT[ANODE]+1
  IF (CST>MAXCOST[ANODE]) MAXCOST[ANODE]=CST
  IF (CONNECTORCNT[ANODE]=1) MINCOST[ANODE]=CST
  IF (CST<MINCOST[ANODE]) MINCOST[ANODE]=CST
ENDIF

; GENERATE CODWAM FILE
  IF (I=0)
       LOOP WW=1,{ZONESA}
         IF (MINCOST[WW]=0) MINCOST[WW]=999.9
         MINCOST[WW]=MINCOST[WW]*10
         PRINT
LIST=WW(5.0),MINCOST[WW](5.0),CONNECTORCNT[WW](5.0), PRINTO=1
       ENDLOOP
  ENDIF


ENDRUN
```

4. PTPTR00C.S

```
; Do not change filenames or add or remove FILEI/FILEO statements
using an editor. Use Cube/Application Manager.
RUN PGM=PUBLIC TRANSPORT
PRNFILE="D:\FSUTMS\D4\TCRPM\CUBE\PTPTR00E.PRN" MSG='Peak with
Auto Access'
FILEI LOOKUPI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\ACTIVECENTROIDS.CSV"
FILEI NETI = "D:\FSUTMS\D4\TCRPM\CUBE\PRELOADS.NET"
FILEI FACTORI[3] = "{DATADIR}\TNETAM7.FAC"
FILEI FACTORI[2] = "{DATADIR}\TNETAM6.FAC"
FILEI FACTORI[1] = "{DATADIR}\TNETAM5.FAC"
FILEO MATO[2] = "D:\FSUTMS\D4\TCRPM\CUBE\TSKIMAM6.MAT",
 MO=1-
13,NAME=TIMEM1,TIMEM2,TIMEM3,TIMEM4_12,TIMEM5,TIMEM6,TIMEM7,TIMEM
8,TRANSFERS,FIRSTWAIT,SECONDWAIT,TOTALTIME,FARE
FILEO MATO[1] = "D:\FSUTMS\D4\TCRPM\CUBE\TSKIMAM5.MAT",
 MO=1-
13,NAME=TIMEM1,TIMEM2,TIMEM3,TIMEM4_12,TIMEM5,TIMEM6,TIMEM7,TIMEM
8,TRANSFERS,FIRSTWAIT,SECONDWAIT,TOTALTIME,FARE
FILEO MATO[3] = "D:\FSUTMS\D4\TCRPM\CUBE\TSKIMAM7.MAT",
 MO=1-
13,NAME=TIMEM1,TIMEM2,TIMEM3,TIMEM4_12,TIMEM5,TIMEM6,TIMEM7,TIMEM
8,TRANSFERS,FIRSTWAIT,SECONDWAIT,TOTALTIME,FARE
FILEO ROUTEO[3] = "D:\FSUTMS\D4\TCRPM\CUBE\TROUTEAM7.RTE",
 REPORTI=1-{ZONESA}, REPORTJ={CBDZONE}
FILEO ROUTEO[2] = "D:\FSUTMS\D4\TCRPM\CUBE\TROUTEAM6.RTE",
 REPORTI=1-{ZONESA}, REPORTJ={CBDZONE}
FILEO ROUTEO[1] = "D:\FSUTMS\D4\TCRPM\CUBE\TROUTEAM5.RTE",
 REPORTI=1-{ZONESA}, REPORTJ={CBDZONE}
```

```
FILEI TURNPENI = "D:\FSUTMS\D4\TCRPM\CUBE\TCARDS.PRN"
FILEI LINEI[1] = "{DATADIR}\TROUTE.LIN"
FILEI FAREI = "{DATADIR}\BASEFARES.FAR"
FILEI SYSTEMI = "{DATADIR}\SYSTEM.PTS"
FILEO PRINTO[1] = "D:\FSUTMS\D4\TCRPM\CUBE\PTPTR00G.PRN"
FILEO NETO = "D:\FSUTMS\D4\TCRPM\CUBE\TNETAMAUTO.NET"
FILEO REPORTO = "D:\FSUTMS\D4\TCRPM\CUBE\TNETAMAUTO.PRN"

PARAMETERS USERCLASSES=1-3,FARE=N,MAPSCALE={UNITS},
           NOROUTEERRS=999999, NOROUTEMSGS=999999,
           TRANTIME=LW.TRANTIME,
           TRANTIME[4]=LI.LBTIME,
           TRANTIME[5]=LI.RLTIME,
           TRANTIME[6]=LI.EBTIME,
           TRANTIME[8]=LI.RLTIME,
           TRANTIME[12]=LI.SHTIME
REPORT LINES=T
PROCESS PHASE=LINKREAD
 LW.TRANTIME=LI.LBTIME
 LW.WALKTIME=LI.WALKTIME
 LW.WALKDISTANCE=LI.DISTANCE
ENDPROCESS


PROCESS PHASE=DATAPREP
; WALK EGRESS
  GENERATE,
COST=(LW.WALKTIME),MAXCOST=4*24,LIST=T,NTLEGMODE=101,MAXNTLEGS=4,
            DIRECTION=2,  FROMNODE=1-{zonesa}, TONODE=944-99999
; SIDEWALKS
READ,
        FILE = "D:\FSUTMS\D4\TCRPM\CUBE\SIDEWALKSAM.TXT"

; AUTO CONNECTORS
READ,
        FILE = "D:\FSUTMS\D4\TCRPM\CUBE\STATIONAM.TXT"

ENDPROCESS

PROCESS PHASE=SKIMIJ
lookup, name=activec, lookup[1]=1, result=2,LOOKUPI=1
active=activec(1,i)
if (active=1)


  MW[1]=TIMEA(0,101)+TIMEA(0,1)
  MW[2]=TIMEA(0,2)
  MW[3]=TIMEA(0,3)
  MW[4]=TIMEA(0,4)+TIMEA(0,12)
  MW[5]=TIMEA(0,5)
  MW[6]=TIMEA(0,6)
  MW[7]=TIMEA(0,7)
  MW[8]=TIMEA(0,8)
  MW[9]=BRDINGS(0,TMODES)-1
  MW[10]=IWAITA(0)
  MW[11]=XWAITA(0)
  MW[12]=TIMEA(0,ALLMODES)+MW[10]+MW[11]
```

```
  MW[13]=FAREA(0,ALLMODES)
   IF (MW[9]<0) MW[9]=0
  if (userclass=1)
  rs6=rowsum(6)
  if (rs6>0) r6flag=1
  log prefix=tnetamauto var=r6flag
  elseif (userclass=2)
  rs7=rowsum(7)
  if (rs7>0) r7flag=1
  log prefix=tnetamauto var=r7flag
  elseif (userclass=4)
  rs8=rowsum(8)
  if (rs8>0) r8flag=1
  log prefix=tnetamauto var=r8flag
  endif

endif

ENDPROCESS


ENDRUN
```

5. PTPTR00E.C

```
; Do not change filenames or add or remove FILEI/FILEO statements
using an editor. Use Cube/Application Manager.
RUN PGM=PUBLIC TRANSPORT
PRNFILE="D:\FSUTMS\D4\TCRPM\CUBE\PTPTR00J.PRN" MSG='Off-peak with
Walk Access'
FILEI LOOKUPI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\ACTIVECENTROIDS.CSV"
FILEO LINKO = "D:\FSUTMS\D4\TCRPM\CUBE\TWALKMD.DBF"
FILEI NETI = "D:\FSUTMS\D4\TCRPM\CUBE\UNLOADED BASE.NET"
FILEI TURNPENI = "D:\FSUTMS\D4\TCRPM\CUBE\TCARDS.PRN"
FILEO ROUTEO[4] = "D:\FSUTMS\D4\TCRPM\CUBE\TROUTEMD4.RTE",
 REPORTI=1-{ZONESA}, REPORTJ={CBDZONE}
FILEO ROUTEO[3] = "D:\FSUTMS\D4\TCRPM\CUBE\TROUTEMD3.RTE",
 REPORTI=1-{ZONESA}, REPORTJ={CBDZONE}
FILEO ROUTEO[2] = "D:\FSUTMS\D4\TCRPM\CUBE\TROUTEMD2.RTE",
 REPORTI=1-{ZONESA}, REPORTJ={CBDZONE}
FILEO MATO[4] = "D:\FSUTMS\D4\TCRPM\CUBE\TSKIMMD4.MAT",
 MO=1-
13,NAME=TIMEM1,TIMEM2,TIMEM3,TIMEM4_12,TIMEM5,TIMEM6,TIMEM7,TIMEM
8,TRANSFERS,FIRSTWAIT,SECONDWAIT,TOTALTIME,FARE
FILEO MATO[3] = "D:\FSUTMS\D4\TCRPM\CUBE\TSKIMMD3.MAT",
 MO=1-
13,NAME=TIMEM1,TIMEM2,TIMEM3,TIMEM4_12,TIMEM5,TIMEM6,TIMEM7,TIMEM
8,TRANSFERS,FIRSTWAIT,SECONDWAIT,TOTALTIME,FARE
FILEO MATO[2] = "D:\FSUTMS\D4\TCRPM\CUBE\TSKIMMD2.MAT",
 MO=1-
13,NAME=TIMEM1,TIMEM2,TIMEM3,TIMEM4_12,TIMEM5,TIMEM6,TIMEM7,TIMEM
8,TRANSFERS,FIRSTWAIT,SECONDWAIT,TOTALTIME,FARE
FILEI FACTORI[4] = "{datadir}\TNETMD4.FAC"
FILEI FACTORI[3] = "{datadir}\TNETMD3.FAC"
FILEI FACTORI[2] = "{datadir}\TNETMD2.FAC"
FILEI FACTORI[1] = "{datadir}\TNETMD1.FAC"
```

```
FILEO ROUTEO[1] = "D:\FSUTMS\D4\TCRPM\CUBE\TROUTEMD1.RTE",
 REPORTI=1-{ZONESA}, REPORTJ={CBDZONE}
FILEI LINEI[1] = "{DATADIR}\TROUTE.LIN"
FILEI FAREI = "{DATADIR}\BASEFARES.FAR"
FILEI SYSTEMI = "{DATADIR}\SYSTEM.PTS"
FILEO MATO[1] = "D:\FSUTMS\D4\TCRPM\CUBE\TSKIMMD1.MAT",
 MO=1-
13,NAME=TIMEM1,TIMEM2,TIMEM3,TIMEM4_12,TIMEM5,TIMEM6,TIMEM7,TIMEM
8,TRANSFERS,FIRSTWAIT,SECONDWAIT,TOTALTIME,FARE
FILEO PRINTO[1] = "D:\FSUTMS\D4\TCRPM\CUBE\PTPTR00L.PRN"
FILEO NETO = "D:\FSUTMS\D4\TCRPM\CUBE\TNETMDWALK.NET"
FILEO REPORTO = "D:\FSUTMS\D4\TCRPM\CUBE\TNETMDWALK.PRN"
; OVERALL PARAMETERS OF RUN
PARAMETERS USERCLASSES=1-4,FARE=N,MAPSCALE={UNITS},
           NOROUTEERRS=999999, NOROUTEMSGS=999999,
           TRANTIME=LW.TRANTIME,
           TRANTIME[4]=LI.LBTIME,
           TRANTIME[5]=LI.RLTIME,
           TRANTIME[6]=LI.EBTIME,
           TRANTIME[8]=LI.RLTIME,
           TRANTIME[12]=LI.SHTIME
REPORT LINES=T
PROCESS PHASE=LINKREAD
 LW.TRANTIME=LI.LBTIME
 LW.WALKTIME=LI.WALKTIME
 LW.WALKDISTANCE=LI.DISTANCE
ENDPROCESS

PROCESS PHASE=DATAPREP
; WALK ACCESS
  GENERATE, COST=(LW.WALKTIME),MAXCOST=4*24,LIST=T,NTLEGMODE =
1,MAXNTLEGS=4,
            DIRECTION=1,  FROMNODE=1-{zonesa}, TONODE=944-99999
; WALK EGRESS
  GENERATE, COST=(LW.WALKTIME),MAXCOST=4*24,LIST=T,NTLEGMODE =
101,MAXNTLEGS=4,
            DIRECTION=2,  FROMNODE=1-{zonesa}, TONODE=944-99999
;; SIDEWALKS
READ,
        FILE = "D:\FSUTMS\D4\TCRPM\CUBE\SIDEWALKSMD.TXT"


ENDPROCESS

PROCESS PHASE=SKIMIJ
lookup, name=activec, lookup[1]=1, result=2,LOOKUPI=1
active=activec(1,i)
if (active=1)


  MW[1]=TIMEA(0,101)+TIMEA(0,1)
  MW[2]=TIMEA(0,2)
  MW[3]=TIMEA(0,3)
  MW[4]=TIMEA(0,4)+TIMEA(0,12)
  MW[5]=TIMEA(0,5)
  MW[6]=TIMEA(0,6)
  MW[7]=TIMEA(0,7)
```

```
  MW[8]=TIMEA(0,8)
  MW[9]=BRDINGS(0,TMODES)-1
  MW[10]=IWAITA(0)
  MW[11]=XWAITA(0)
  MW[12]=TIMEA(0,ALLMODES)+MW[10]+MW[11]
  MW[13]=FAREA(0,ALLMODES)
   IF (MW[9]<0) MW[9]=0
  if (userclass=2)
  rs6=rowsum(6)
  if (rs6>0) r6flag=1
  log prefix=tnetmdwalk var=r6flag
  elseif (userclass=3)
  rs7=rowsum(7)
  if (rs7>0) r7flag=1
  log prefix=tnetmdwalk var=r7flag
  elseif (userclass=4)
  rs8=rowsum(8)
  if (rs8>0) r8flag=1
  log prefix=tnetmdwalk var=r8flag
  endif
endif
ENDPROCESS

ENDRUN
```

6. PTPTR00F.S

```
; Do not change filenames or add or remove FILEI/FILEO statements
using an editor. Use Cube/Application Manager.
RUN PGM=PUBLIC TRANSPORT
PRNFILE="D:\FSUTMS\D4\TCRPM\CUBE\PTPTR00M.PRN" MSG='Off-peak with
Auto Access'
FILEI LOOKUPI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\ACTIVECENTROIDS.CSV"
FILEI NETI = "D:\FSUTMS\D4\TCRPM\CUBE\UNLOADED BASE.NET"
FILEI TURNPENI = "D:\FSUTMS\D4\TCRPM\CUBE\TCARDS.PRN"
FILEI FACTORI[3] = "{DATADIR}\TNETMD7.FAC"
FILEI FACTORI[2] = "{DATADIR}\TNETMD6.FAC"
FILEI FACTORI[1] = "{DATADIR}\TNETMD5.FAC"
FILEO MATO[2] = "D:\FSUTMS\D4\TCRPM\CUBE\TSKIMMD6.MAT",
 MO=1-
13,NAME=TIMEM1,TIMEM2,TIMEM3,TIMEM4_12,TIMEM5,TIMEM6,TIMEM7,TIMEM
8,TRANSFERS,FIRSTWAIT,SECONDWAIT,TOTALTIME,FARE
FILEO MATO[1] = "D:\FSUTMS\D4\TCRPM\CUBE\TSKIMMD5.MAT",
 MO=1-
13,NAME=TIMEM1,TIMEM2,TIMEM3,TIMEM4_12,TIMEM5,TIMEM6,TIMEM7,TIMEM
8,TRANSFERS,FIRSTWAIT,SECONDWAIT,TOTALTIME,FARE
FILEO MATO[3] = "D:\FSUTMS\D4\TCRPM\CUBE\TSKIMMD7.MAT",
 MO=1-
13,NAME=TIMEM1,TIMEM2,TIMEM3,TIMEM4_12,TIMEM5,TIMEM6,TIMEM7,TIMEM
8,TRANSFERS,FIRSTWAIT,SECONDWAIT,TOTALTIME,FARE
FILEO ROUTEO[3] = "D:\FSUTMS\D4\TCRPM\CUBE\TROUTEMD7.RTE",
 REPORTI=1-{ZONESA}, REPORTJ={CBDZONE}
FILEO ROUTEO[2] = "D:\FSUTMS\D4\TCRPM\CUBE\TROUTEMD6.RTE",
 REPORTI=1-{ZONESA}, REPORTJ={CBDZONE}
FILEO ROUTEO[1] = "D:\FSUTMS\D4\TCRPM\CUBE\TROUTEMD5.RTE",
 REPORTI=1-{ZONESA}, REPORTJ={CBDZONE}
```

```
FILEI LINEI[1] = "{DATADIR}\TROUTE.LIN"
FILEI FAREI = "{DATADIR}\BASEFARES.FAR"
FILEI SYSTEMI = "{DATADIR}\SYSTEM.PTS"
FILEO PRINTO[1] = "D:\FSUTMS\D4\TCRPM\CUBE\PTPTR00O.PRN"
FILEO NETO = "D:\FSUTMS\D4\TCRPM\CUBE\TNETMDAUTO.NET"
FILEO REPORTO = "D:\FSUTMS\D4\TCRPM\CUBE\TNETMDAUTO.PRN"

; OVERALL PARAMETERS OF RUN
PARAMETERS USERCLASSES=1-3,FARE=N,MAPSCALE={UNITS},
           NOROUTEERRS=999999, NOROUTEMSGS=999999,
           TRANTIME=LW.TRANTIME,
           TRANTIME[4]=LI.LBTIME,
           TRANTIME[5]=LI.RLTIME,
           TRANTIME[6]=LI.EBTIME,
           TRANTIME[8]=LI.RLTIME,
           TRANTIME[12]=LI.SHTIME
REPORT LINES=T
PROCESS PHASE=LINKREAD
 LW.TRANTIME=LI.LBTIME
 LW.WALKTIME=LI.WALKTIME
 LW.WALKDISTANCE=LI.DISTANCE
ENDPROCESS


PROCESS PHASE=DATAPREP
; WALK EGRESS
  GENERATE,
COST=(LW.WALKTIME),MAXCOST=4*24,LIST=T,NTLEGMODE=101,MAXNTLEGS=4,
            DIRECTION=2,  FROMNODE=1-{zonesa}, TONODE=944-99999
; SIDEWALKS
READ,
        FILE = "D:\FSUTMS\D4\TCRPM\CUBE\SIDEWALKSMD.TXT"

; AUTO CONNECTORS
READ,
        FILE = "D:\FSUTMS\D4\TCRPM\CUBE\STATIONMD.TXT"

ENDPROCESS

PROCESS PHASE=SKIMIJ
lookup, name=activec, lookup[1]=1, result=2,LOOKUPI=1
active=activec(1,i)
if (active=1)


  MW[1]=TIMEA(0,101)+TIMEA(0,1)
  MW[2]=TIMEA(0,2)
  MW[3]=TIMEA(0,3)
  MW[4]=TIMEA(0,4)+TIMEA(0,12)
  MW[5]=TIMEA(0,5)
  MW[6]=TIMEA(0,6)
  MW[7]=TIMEA(0,7)
  MW[8]=TIMEA(0,8)
  MW[9]=BRDINGS(0,TMODES)-1
  MW[10]=IWAITA(0)
  MW[11]=XWAITA(0)
  MW[12]=TIMEA(0,ALLMODES)+MW[10]+MW[11]
```

```
  MW[13]=FAREA(0,ALLMODES)
   IF (MW[9]<0) MW[9]=0
  if (userclass=1)
  rs6=rowsum(6)
  if (rs6>0) r6flag=1
  log prefix=tnetmdauto var=r6flag
  elseif (userclass=2)
  rs7=rowsum(7)
  if (rs7>0) r7flag=1
  log prefix=tnetmdauto var=r7flag
  elseif (userclass=4)
  rs8=rowsum(8)
  if (rs8>0) r8flag=1
  log prefix=tnetmdauto var=r8flag
  endif
endif
ENDPROCESS


ENDRUN
```

PRE-MODE

1. 00MAT00I.S

```
; Treasure Coast Pre-Mode Choice Model to separate out Non-
Motorized Trips
; Coefficients and Constants are obtained from Survey
; Do not change filenames or add or remove FILEI/FILEO statements
using an editor. Use Cube/Application Manager.
RUN PGM=MATRIX
  ZONES=731
FILEI MATI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\FHSKIMS.MAT"
FILEI ZDATI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\SOC.DBF",
 Z=TAZ
FILEO MATO[1] = "D:\FSUTMS\D4\TCRPM\CUBE\TEST1.MAT",
 MO=1,17,18,27,28,37,38,47,48,57,58,67,68,  NAME= DISTANCE,
MOT_HBW, NONMOT_HBW, MOT_NHBW, NONMOT_NHBW,
    MOT_NHBO, NONMOT_NHBO,MOT_HBSH, NONMOT_HBSH,MOT_HBSR,
NONMOT_HBSR,MOT_HBO, NONMOT_HBO

COMP K_HBW=   3.88          ; CONSTANT FOR HBW
COMP K_NHBW=  1.907         ; CONSTANT FOR NHBW
COMP K_NHBO=  3.920         ; CONSTANT FOR NHBO
COMP K_HBSHOP=3.950         ; CONSTANT FOR HBSHOP
COMP K_HBSR=  0.987         ; CONSTANT FOR HBSR
COMP K_HBO=   2.305         ; CONSTANT FOR HBO

COMP AK_HBW=  0.238         ; HBW DISTANCE COEFF
COMP BK_HBW=  -0.0001       ; HBW TOTAL EMP COEFF within 1 mile
of attraction
COMP CK_HBW=  -2.12         ; COEFF FOR HBW IF NO. OF CAR IS LESS
THAN NO. OF WORKER

COMP AK_NHBW=  0.701         ; NHBW DISTANCE COEFF
COMP BK_NHBW= -0.00011       ; NHBW TOTAL EMP COEFF within 1 mile
of attraction

COMP AK_NHBO=  0.166         ; NHBO DISTANCE COEFF
COMP BK_NHBO= -0.00004       ; NHBO TOTAL EMP COEFF within 1 mile
of attraction

COMP AK_HBSH=  0.29          ; HBSHOP DISTANCE COEFF
COMP BK_HBSH= -0.00031       ; HBSHOP COMM EMP COEFF within 1
mile of attraction
COMP CK_HBSH= -4.91          ; HBSHOP COEFF FOR NO CAR IN HH
COMP DK_HBSH= -1.33          ; HBSHOP COEFF FOR NO. OF CAR LESS
THAN NO. OF WORKER

COMP AK_HBSR=  0.26          ; HBSR DISTANCE COEFF
COMP BK_HBSR= -0.00012       ; HBSR COMM EMP COEFF within 1 mile
of attraction
COMP CK_HBSR= -3.00          ; HBSR COEFF FOR NO CAR IN HH
COMP DK_HBSR= -0.626         ; HBSR COEFF FOR NO. OF CAR LESS
THAN NO. OF WORKER
```

```
COMP AK_HBO=  0.494          ; HBO DISTANCE COEFF
COMP BK_HBO= -0.00033        ; HBO COMM EMP COEFF within 1 mile
of attraction
COMP CK_HBO= -1.950          ; HBO COEFF FOR NO CAR IN HH
COMP DK_HBO= -0.917          ; HBO COEFF FOR NO. OF CAR LESS THAN
NO. OF WORKER

  MW[1]=MI.1.2/100           ; DISTANCE
; HBW
  JLOOP
      IF (ZI.1.CAR<ZI.1.WORKER)
           MW[2][J]=
K_HBW+AK_HBW*MW[1][J]+BK_HBW*ZI.1.TOTEMP[J]+CK_HBW*1.0
      ELSE
           MW[2][J]=
K_HBW+AK_HBW*MW[1][J]+BK_HBW*ZI.1.TOTEMP[J]+CK_HBW*0.0
      ENDIF
  ENDJLOOP
  MW[3]= EXP(MW[2])                          ; MOTORIZED
  MW[4]= EXP(0)                              ; NON MOTORIZED
      MW[17] = 1000*MW[3]/(MW[3]+MW[4])      ; MOTORIZED HBW
SHARE
      MW[18] = 1000-MW[17]                   ; NON-MOTORIZED HBW
SHARE
;NHBW
  JLOOP
     MW[22][J]= K_NHBW+AK_NHBW*MW[1][J]+BK_NHBW*ZI.1.TOTEMP[J]
  ENDJLOOP
  MW[23]= EXP(MW[22])                        ; MOTORIZED
  MW[24]= EXP(0)                             ; NON MOTORIZED
  MW[27] = 1000*MW[23]/(MW[23]+MW[24])       ; MOTORIZED NHBW
SHARE
  MW[28] = 1000-MW[27]                       ; NON-MOTORIZED NHBW
SHARE

 ;NHBO
  JLOOP
     MW[32][J]= K_NHBO+AK_NHBO*MW[1][J]+BK_NHBO*ZI.1.TOTEMP[J]
  ENDJLOOP
  MW[33]= EXP(MW[32])                        ; MOTORIZED
  MW[34]= EXP(0)                             ; NON MOTORIZED
  MW[37] = 1000*MW[33]/(MW[33]+MW[34])       ; MOTORIZED NHBO
SHARE
  MW[38] = 1000-MW[37]                       ; NON-MOTORIZED NHBO
SHARE

; HBShop
  JLOOP
      IF (ZI.1.CAR<ZI.1.WORKER)
          IF (ZI.1.CAR=0.0)
             MW[42][J]=
K_HBSHOP+AK_HBSH*MW[1][J]+CK_HBSH*1.0+DK_HBSH*1.0+BK_HBSH*ZI.1.CO
MMEMP[J]
          ELSE
             MW[42][J]=
K_HBSHOP+AK_HBSH*MW[1][J]+CK_HBSH*0.0+DK_HBSH*1.0+BK_HBSH*ZI.1.CO
MMEMP[J]
```

```
          ENDIF
      ELSE
          IF (ZI.1.CAR=0.0)
             MW[42][J]=
K_HBSHOP+AK_HBSH*MW[1][J]+CK_HBSH*1.0+DK_HBSH*0.0+BK_HBSH*ZI.1.CO
MMEMP[J]
          ELSE
             MW[42][J]=
K_HBSHOP+AK_HBSH*MW[1][J]+CK_HBSH*0.0+DK_HBSH*0.0+BK_HBSH*ZI.1.CO
MMEMP[J]
          ENDIF
      ENDIF
  ENDJLOOP
  MW[43]= EXP(MW[42])                        ; MOTORIZED
  MW[44]= EXP(0)                             ; NON MOTORIZED
  MW[47] = 1000*MW[43]/(MW[43]+MW[44])       ; MOTORIZED HBShop
SHARE
  MW[48] = 1000-MW[47]                       ; NON-MOTORIZED
HBShop SHARE

; HBSR
  JLOOP
      IF (ZI.1.CAR<ZI.1.WORKER)
          IF (ZI.1.CAR=0.0)
             MW[52][J]= K_HBSR+AK_HBSR*MW[1][J]+CK_HBSR*1.0
+DK_HBSR*1.0 +BK_HBSR*ZI.1.COMMEMP[J]
          ELSE
             MW[52][J]= K_HBSR+AK_HBSR*MW[1][J]+CK_HBSR*0.0
+DK_HBSR*1.0 +BK_HBSR*ZI.1.COMMEMP[J]
          ENDIF
      ELSE
          IF (ZI.1.CAR=0.0)
             MW[52][J]= K_HBSR+AK_HBSR*MW[1][J]+CK_HBSR*1.0
+DK_HBSR*0.0 +BK_HBSR*ZI.1.COMMEMP[J]
          ELSE
             MW[52][J]= K_HBSR+AK_HBSR*MW[1][J]+CK_HBSR*0.0
+DK_HBSR*0.0 +BK_HBSR*ZI.1.COMMEMP[J]
          ENDIF
      ENDIF
  ENDJLOOP
  MW[53]= EXP(MW[52])                        ; MOTORIZED
  MW[54]= EXP(0)                             ; NON MOTORIZED
  MW[57] = 1000*MW[53]/(MW[53]+MW[54])       ; MOTORIZED HBSR
SHARE
  MW[58] = 1000-MW[57]                       ; NON-MOTORIZED HBSR
SHARE

; HBO
  JLOOP
      IF (ZI.1.CAR<ZI.1.WORKER)
          IF (ZI.1.CAR=0.0)
             MW[62][J]= K_HBO+AK_HBO*MW[1][J]+CK_HBO*1.0
+DK_HBO*1.0 +BK_HBO*ZI.1.COMMEMP[J]
          ELSE
             MW[62][J]= K_HBO+AK_HBO*MW[1][J]+CK_HBO*0.0
+DK_HBO*1.0 +BK_HBO*ZI.1.COMMEMP[J]
          ENDIF
```

```
      ELSE
          IF (ZI.1.CAR=0.0)
             MW[62][J]= K_HBO+AK_HBO*MW[1][J]+CK_HBO*1.0
+DK_HBO*0.0 +BK_HBO*ZI.1.COMMEMP[J]
          ELSE
             MW[62][J]= K_HBO+AK_HBO*MW[1][J]+CK_HBO*0.0
+DK_HBO*0.0 +BK_HBO*ZI.1.COMMEMP[J]
          ENDIF
      ENDIF
  ENDJLOOP
  MW[63]= EXP(MW[62])                          ; MOTORIZED
  MW[64]= EXP(0)                               ; NON MOTORIZED
  MW[67] = 1000*MW[63]/(MW[63]+MW[64])         ; MOTORIZED HBO SHARE
  MW[68] = 1000-MW[67]                         ; NON-MOTORIZED HBO
SHARE

ENDRUN
```

2. 00MAT00J.S

```
; Do not change filenames or add or remove FILEI/FILEO statements
using an editor. Use Cube/Application Manager.
RUN PGM=MATRIX
FILEI MATI[2] = "D:\FSUTMS\D4\TCRPM\CUBE\PTRIPS11CG.MAT"
FILEI MATI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\TEST1.MAT"
FILEO MATO[1] = "D:\FSUTMS\D4\TCRPM\CUBE\MOT_NONMOT.MAT",
        MO=1-12, NAME=MOT_HBW, NONMOT_HBW, MOT_HBSH, NONMOT_HBSH,
MOT_HBSR, NONMOT_HBSR,
                      MOT_HBO, NONMOT_HBO, MOT_NHBW, NONMOT_NHBW,
MOT_NHBO, NONMOT_NHBO
   MW[1]= MI.1.2*MI.2.1/1000     ; MOT_HBW
   MW[2]= MI.2.1-MW[1]
   MW[3]= MI.1.8*MI.2.2/1000     ; MOT_HBSH
   MW[4]= MI.2.2-MW[3]
   MW[5]= MI.1.10*MI.2.3/1000    ; MOT_HBSR
   MW[6]= MI.2.3-MW[5]
   MW[7]= MI.1.12*MI.2.5/1000    ; MOT_HBO
   MW[8]= MI.2.5-MW[7]
   MW[9]= MI.1.4*MI.2.6/1000     ; MOT_NHBW
   MW[10]= MI.2.6-MW[9]
   MW[11]= MI.1.6*MI.2.7/1000    ; MOT_NHBO
   MW[12]= MI.2.7-MW[11]

ENDRUN
```

3. 00MAT00K.S

```
; Do not change filenames or add or remove FILEI/FILEO statements
using an editor. Use Cube/Application Manager.
RUN PGM=MATRIX
FILEI MATI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\FHSKIMS.MAT"
FILEO MATO[1] = "D:\FSUTMS\D4\TCRPM\CUBE\TEMP1.MAT",
 MO=1, Name=TEST
MW[1]=1
```

```
ENDRUN
```

4. 00MAT00L.S

```
; Do not change filenames or add or remove FILEI/FILEO statements
using an editor. Use Cube/Application Manager.
RUN PGM=MATRIX
FILEI ZDATI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\HB_SCH.DBF",
 Z=TAZ
FILEI MATI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\TEMP1.MAT"
FILEO MATO[1] = "D:\FSUTMS\D4\TCRPM\CUBE\TEMP2.MAT",
 MO=2-7, NAME= WALKBIKE, DA,SR2,SR3,PUBBUS,SCHBUS
    MW[1]=MI.1.1

  JLOOP
     MW[2][J]= 100*ZI.1.WALKBIKE
     MW[3][J]= 100*ZI.1.DA
     MW[4][J]= 100*ZI.1.SR2
     MW[5][J]= 100*ZI.1.SR3
     MW[6][J]= 100*ZI.1.PUBBUS
     MW[7][J]= 100*ZI.1.SCHBUS
  ENDJLOOP

ENDRUN
```

5. 00MAT00M.S

```
; Do not change filenames or add or remove FILEI/FILEO statements
using an editor. Use Cube/Application Manager.
RUN PGM=MATRIX
FILEI MATI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\TEMP2.MAT"
FILEO MATO[1] = "D:\FSUTMS\D4\TCRPM\CUBE\FACTOR.MAT",
    MO=1-6, NAME= WALKBIKE, DA,SR2,SR3,PUBBUS,SCHBUS
    RENUMBER,
        FILE = "D:\FSUTMS\D4\TCRPM1\CUBE\CORRESP.CSV",
ZONES=731,MISSINGZI=M,MISSINGZO=M
    MW[1]=MI.1.1
    MW[2]=MI.1.2
    MW[3]=MI.1.3
    MW[4]=MI.1.4
    MW[5]=MI.1.5
    MW[6]=MI.1.6

ENDRUN
```

6. 00MAT00O.S

```
; Do not change filenames or add or remove FILEI/FILEO statements
using an editor. Use Cube/Application Manager.
RUN PGM=MATRIX
FILEI MATI[3] = "D:\FSUTMS\D4\TCRPM\CUBE\PTRIPS11CG.MAT"
FILEI MATI[2] = "D:\FSUTMS\D4\TCRPM\CUBE\HBSC_MOT_NOTMOT.MAT"
FILEI MATI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\MOT_NONMOT.MAT"
FILEO MATO[1] = "D:\FSUTMS\D4\TCRPM\CUBE\PTRIPS11.MAT",
```

```
  MO=1-11, NAME= HBW, HBSH, HBSR, HBSCPRV, HBO, NHBW, NHBO,
AIRPORT, TRUCK4TIRE,
                                TRKSNGLUNIT, TRACTORTRAILER
    MW[1]=MI.1.1
    MW[2]=MI.1.3
    MW[3]=MI.1.5
    MW[4]=MI.2.2
    MW[5]=MI.1.7
    MW[6]=MI.1.9
    MW[7]=MI.1.11
    MW[8]=MI.3.8
    MW[9]=MI.3.9
    MW[10]=MI.3.10
    MW[11]=MI.3.11

ENDRUN
```

7. 00MAT00N.S

```
; Do not change filenames or add or remove FILEI/FILEO statements
using an editor. Use Cube/Application Manager.
RUN PGM=MATRIX
FILEI MATI[2] = "D:\FSUTMS\D4\TCRPM\CUBE\PTRIPS11CG.MAT"
FILEI MATI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\FACTOR.MAT"
FILEO MATO[1] = "D:\FSUTMS\D4\TCRPM\CUBE\HBSC_MOT_NOTMOT.MAT",
    MO=1-6, NAME= NONMOT_HBSC, MOT_HBSC, DA_SCH, SR2_SCH,
SR3_SCH, WALK_LOCAL_BUS
    MW[1]=MI.1.1*MI.2.4/100
    MW[2]=MI.2.4-MW[1]
    MW[3]=MI.1.2*MI.2.4/100      ; DA HBSC
    MW[4]=MI.1.3*MI.2.4/100      ; SR2 HBSC
    MW[5]=MI.1.4*MI.2.4/100      ; SR3+ HBSC
    MW[6]=MI.1.5*MI.2.4/100      ; Public Bus, Assume Walk to
local bus for school
; School bus trips are not included

ENDRUN
```

MODE

1. 01MAT00C.S

```
; TREASURE COAST HBW MC
; Do not change filenames or add or remove FILEI/FILEO statements
using an editor. Use Cube/Application Manager.
RUN PGM=MATRIX
FILEI MATI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\TSKIMAM1.MAT"
FILEI MATI[2] = "D:\FSUTMS\D4\TCRPM\CUBE\TSKIMAM2.MAT"
FILEI MATI[3] = "D:\FSUTMS\D4\TCRPM\CUBE\TSKIMAM3.MAT"
FILEI MATI[4] = "D:\FSUTMS\D4\TCRPM\CUBE\TSKIMAM4.MAT"
FILEI MATI[5] = "D:\FSUTMS\D4\TCRPM\CUBE\TSKIMAM5.MAT"
FILEI MATI[6] = "D:\FSUTMS\D4\TCRPM\CUBE\TSKIMAM6.MAT"
FILEI MATI[7] = "D:\FSUTMS\D4\TCRPM\CUBE\TSKIMAM7.MAT"
FILEI MATI[8] = "D:\FSUTMS\D4\TCRPM\CUBE\RHSKIMS.MAT"
FILEI MATI[9] = "D:\FSUTMS\D4\TCRPM\CUBE\PTRIPS11.MAT"
FILEI ZDATI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\PARK.DBF",
 Z=TAZ
FILEI ZDATI[2] = "D:\FSUTMS\D4\TCRPM\CUBE\HHCAR.DBF",
 Z=TAZ
FILEO MATO[1] = "D:\FSUTMS\D4\TCRPM\CUBE\MC_HBW.MAT",
 MO=161-166, DEC=3,3,3,3,3,3, NAME=
DA,SR2,SR3,WALK_LOCAL,WALK_PREM,AUTO_PREM
FILEO MATO[2] = "D:\FSUTMS\D4\TCRPM\CUBE\HBW_TEST.MAT",
 MO=197-199, DEC=3,3,3, NAME=0CARHH, 1CARHH, 2CARHH
FILEO MATO[3] = "D:\FSUTMS\D4\TCRPM\CUBE\ZEROCAR_HBW.MAT",
 MO=201-206, DEC=3,3,3,3,3,3
FILEO MATO[4] = "D:\FSUTMS\D4\TCRPM\CUBE\ONECAR_HBW.MAT",
 MO=211-216, DEC=3,3,3,3,3,3
FILEO MATO[5] = "D:\FSUTMS\D4\TCRPM\CUBE\TWOCAR_HBW.MAT",
 MO=221-226, DEC=3,3,3,3,3,3

;*********************************************
; TREASURE COAST MODE CHOICE MODELS, HBW
;*********************************************
 AUTOCOST  = 15.0            ; auto operating cost in cents per
mile
 K_HBW_DA  = 0.0             ; MODAL CONSTANT FOR HBW DA
 K_HBW_DA_0C  = 0.0          ; CONSTANT FOR HBW DA ZERO CAR  (WAS
0.0)
 K_HBW_DA_1C  = 0.0          ; CONSTANT FOR HBW DA ONE CAR
 K_HBW_DA_2C  = 0.0          ; CONSTANT FOR HBW DA TWO+ CAR


 K_HBW_SR2 = -0.5542          ; MODAL CONSTANT FOR HBW SR2 (WAS -
0.4542)
 K_HBW_SR2_0C = -1.17          ; CONSTANT FOR HBW SR2, ZERO CAR
 K_HBW_SR2_1C = -1.21          ; CONSTANT FOR HBW SR2, ONE CAR
 K_HBW_SR2_2C = -2.02          ; CONSTANT FOR HBW SR2, TWO+ CAR

 K_HBW_SR3 = -0.7199           ; MODAL CONSTANT FOR HBW SR3+ (WAS
-0.6199)
 K_HBW_SR3_0C = -2.77          ; CONSTANT FOR HBW SR3+ ZERO CAR
 K_HBW_SR3_1C = -2.805          ; CONSTANT FOR HBW SR3+ 1 CAR
```

```
 K_HBW_SR3_2C = -2.95          ; CONSTANT FOR HBW SR3+ 2+ CAR


 K_HBW_WLB = -2.1837           ; CONSTANT FOR HBW WALK TO LOCAL
BUS
 K_HBW_WLB_0C = 4.04           ; CONSTANT FOR HBW WALK TO LOCAL
BUS, 0 CAR
 K_HBW_WLB_1C = 1.63          ; CONSTANT FOR HBW WALK TO LOCAL
BUS, 1 CAR
 K_HBW_WLB_2C = -0.45          ; CONSTANT FOR HBW WALK TO LOCAL
BUS, 2+ CAR
;
*****************************************************************
*********
 K_HBW_WEB = -999            ; CONSTANT FOR HBW WALK TO EXPRESS
BUS
 K_HBW_WEB_0C = -999         ; CONSTANT FOR HBW WALK TO EXPRESS
BUS, 0 CAR
 K_HBW_WEB_1C = -999         ; CONSTANT FOR HBW WALK TO EXPRESS
BUS, 1 CAR
 K_HBW_WEB_2C = -999         ; CONSTANT FOR HBW WALK TO EXPRESS
BUS, 2+ CAR

 K_HBW_AEB = -999            ; CONSTANT FOR HBW AUTO TO EXPRESS
BUS
 K_HBW_AEB_0C = -999         ; CONSTANT FOR HBW AUTO TO EXPRESS
BUS, 0 CAR
 K_HBW_AEB_1C = -999         ; CONSTANT FOR HBW AUTO TO EXPRESS
BUS, 1 CAR
 K_HBW_AEB_2C = -999         ; CONSTANT FOR HBW AUTO TO EXPRESS
BUS, 2+ CAR

 AK_HBW_DA  = 0.00        ; WALK TIME COEFFICIENT FOR HBW DA
 AK_HBW_SR2 = 0.00        ; WALK TIME COEFFICIENT FOR HBW SR2
 AK_HBW_SR3 = 0.00        ; WALK TIME COEFFICIENT FOR HBW SR3+
 AK_HBW_WLB = -0.045      ; WALK TIME COEFFICIENT FOR HBW WALK TO
LOCAL BUS
 AK_HBW_WEB = -0.045      ; WALK TIME COEFFICIENT FOR HBW WALK TO
EXPRESS BUS
 AK_HBW_AEB = -0.02       ; AUTO ACCESS time COEFFICIENT FOR HBW
AUTO TO EXPRESS BUS

 BK_HBW_DA  = -0.045         ; WAIT/Terminal TIME COEFFICIENT FOR
HBW DA
 BK_HBW_SR2 = -0.045         ; WAIT/terminal TIME COEFFICIENT FOR
HBW SR2
 BK_HBW_SR3 = -0.045         ; WAIT/Terminal TIME COEFFICIENT FOR
HBW SR3+
 BK_HBW_WLB1 = -0.045     ; First WAIT TIME COEFFICIENT FOR HBW
WALK TO LOCAL BUS, wait<7 min
 BK_HBW_WLB2 = -0.023     ; First WAIT TIME COEFFICIENT FOR HBW
WALK TO LOCAL BUS, wait>7 min
 BK_HBW_WEB1 = -0.045     ; First WAIT TIME COEFFICIENT FOR HBW
Walk TO express BUS, wait<7 min
 BK_HBW_WEB2 = -0.023     ; First WAIT TIME COEFFICIENT FOR HBW
Walk TO express BUS, wait>7 min
```

```
 BK_HBW_AEB1 = -0.045     ; First WAIT TIME COEFFICIENT FOR HBW
AUTO TO LOCAL BUS, wait<7 min
 BK_HBW_AEB2 = -0.023     ; First WAIT TIME COEFFICIENT FOR HBW
AUTO TO LOCAL BUS, wait>7 min

 CK_HBW_DA  = -0.02      ; IVTT TIME COEFFICIENT FOR HBW DA
 CK_HBW_SR2 = -0.02      ; IVTT TIME COEFFICIENT FOR HBW SR2
 CK_HBW_SR3 = -0.02      ; IVTT TIME COEFFICIENT FOR HBW SR3+
 CK_HBW_WLB = -0.02      ; IVTT TIME COEFFICIENT FOR HBW WALK TO
LOCAL BUS
 CK_HBW_WEB = -0.02      ; IVTT TIME COEFFICIENT FOR HBW WALK TO
EXPRESS BUS
 CK_HBW_AEB = -0.02      ; IVTT TIME COEFFICIENT FOR HBW AUTO TO
EXPRESS BUS

 DK_HBW_DA  = -0.0032       ; PARKING COST COEFFICIENT FOR HBW DA
 DK_HBW_SR2 = -0.0032       ; PARKING COST COEFFICIENT FOR HBW
SR2
 DK_HBW_SR3 = -0.0032       ; PARKING COST COEFFICIENT FOR HBW
SR3+
 DK_HBW_WLB = 0.0       ; PARKING COST COEFFICIENT FOR HBW WALK
TO LOCAL BUS
 DK_HBW_WEB = 0.0       ; PARKING COST COEFFICIENT FOR HBW WALK
TO EXPRESS BUS
 DK_HBW_AEB = -0.0032       ; PARKING COST COEFFICIENT FOR HBW
AUTO TO EXPRESS BUS

 EK_HBW_DA  = -0.0025       ; HIGHWAY COST COEFFICIENT FOR HBW DA
 EK_HBW_SR2 = -0.0025       ; HIGHWAY COST COEFFICIENT FOR HBW
SR2
 EK_HBW_SR3 = -0.0025       ; HIGHWAY COST COEFFICIENT FOR HBW
SR3+
 EK_HBW_WLB = -0.0032       ; FARE COEFFICIENT FOR HBW WALK TO
LOCAL BUS
 EK_HBW_WEB = -0.0032       ; FARE COEFFICIENT FOR HBW WALK TO
EXPRESS BUS
 EK_HBW_AEB = -0.0032       ; HIGHWAY COST COEFFICIENT FOR HBW
AUTO TO EXPRESS BUS
 FK_HBW_XLB = -0.045        ; TRABSFER COEFF, HBW

  MW[1]=MI.8.8     ; AUTO TERMINAL TIME
  MW[2]=MI.8.7     ; TOLL TIME
  MW[3]=MI.8.3     ; AUTO IVTT
  MW[4]=MI.8.2     ; AUTO DISTANCE
  MW[5]=MI.8.7+MI.8.3   ; TOTAL AUTO TIME

  MW[6]= MI.1.1+MI.1.2                        ; WALK ACCESS TIME
, WALK TO LOCAL
  MW[7]= MI.1.10                              ; FIRST WAIT for
walk to local
  MW[8]= MI.1.11                              ; XFER WAIT FOR
WALK TO LOCAL
  MW[9]= MI.1.4                               ; IVTT, WALK TO
LOCAL
  MW[10]= MI.1.13                             ; WALK TO LOCAL
FARE
```

```
  MW[16]= MI.2.1+MI.2.2                          ; WALK ACCESS TIME
, WALK TO EXPRESS BUS
  MW[17]= MI.2.10                                ; FIRST WAIT FOR
WALK TO EXPRESS
  MW[18]= MI.2.11                                ; XFER WAIT FOR
WALK TO EXPRESS
  MW[19]= MI.2.4                                 ; IVTT, WALK TO
EXPRESS
  MW[20]= MI.2.13                                ; WALK TO EXPRESS
FARE

  MW[26]= MI.5.1+MI.5.2                          ; AUTO ACCESS TIME
, AUTO TO EXPRESS BUS
  MW[27]= MI.5.10                                ; FIRST WAIT FOR
AUTO TO EXPRESS
  MW[28]= MI.5.11                                ; XFER WAIT FOR
AUTO TO EXPRESS
  MW[29]= MI.5.4                                 ; IVTT, AUTO TO
EXPRESS
  MW[30]= MI.5.13                                ; AUTO TO EXPRESS
FARE


MW[41]=AK_HBW_DA*0.0+BK_HBW_DA*MW[1]+CK_HBW_DA*MW[5]+DK_HBW_DA*ZI
.1.PARK[J]+EK_HBW_DA*AUTOCOST*MW[4]+K_HBW_DA+K_HBW_DA_0C
; DA UTILITY, 0 CAR HH

MW[42]=AK_HBW_DA*0.0+BK_HBW_DA*MW[1]+CK_HBW_DA*MW[5]+DK_HBW_DA*ZI
.1.PARK[J]+EK_HBW_DA*AUTOCOST*MW[4]+K_HBW_DA+K_HBW_DA_1C
; DA UTILITY, 1 CAR HH

MW[43]=AK_HBW_DA*0.0+BK_HBW_DA*MW[1]+CK_HBW_DA*MW[5]+DK_HBW_DA*ZI
.1.PARK[J]+EK_HBW_DA*AUTOCOST*MW[4]+K_HBW_DA+K_HBW_DA_2C
; DA UTILITY, 2+ CAR HH


MW[51]=AK_HBW_SR2*0.0+BK_HBW_SR2*MW[1]+CK_HBW_SR2*MW[5]+DK_HBW_SR
2*ZI.1.PARK[J]+EK_HBW_SR2*AUTOCOST*MW[4]+K_HBW_SR2+K_HBW_SR2_0C
; SR2 UTILITY, 0 CAR

MW[52]=AK_HBW_SR2*0.0+BK_HBW_SR2*MW[1]+CK_HBW_SR2*MW[5]+DK_HBW_SR
2*ZI.1.PARK[J]+EK_HBW_SR2*AUTOCOST*MW[4]+K_HBW_SR2+K_HBW_SR2_1C
; SR2 UTILITY, 1 CAR

MW[53]=AK_HBW_SR2*0.0+BK_HBW_SR2*MW[1]+CK_HBW_SR2*MW[5]+DK_HBW_SR
2*ZI.1.PARK[J]+EK_HBW_SR2*AUTOCOST*MW[4]+K_HBW_SR2+K_HBW_SR2_2C
; SR2 UTILITY, 2 CAR


MW[61]=AK_HBW_SR3*0.0+BK_HBW_SR3*MW[1]+CK_HBW_SR3*MW[5]+DK_HBW_SR
3*ZI.1.PARK[J]+EK_HBW_SR3*AUTOCOST*MW[4]+K_HBW_SR3+K_HBW_SR3_0C
; SR2 UTILITY, 0 CAR

MW[62]=AK_HBW_SR3*0.0+BK_HBW_SR3*MW[1]+CK_HBW_SR3*MW[5]+DK_HBW_SR
3*ZI.1.PARK[J]+EK_HBW_SR3*AUTOCOST*MW[4]+K_HBW_SR3+K_HBW_SR3_1C
; SR3 UTILITY, 1 CAR
```

```
MW[63]=AK_HBW_SR3*0.0+BK_HBW_SR3*MW[1]+CK_HBW_SR3*MW[5]+DK_HBW_SR
3*ZI.1.PARK[J]+EK_HBW_SR3*AUTOCOST*MW[4]+K_HBW_SR3+K_HBW_SR3_2C
; SR3 UTILITY, 2 CAR

  JLOOP INCLUDE = 1-731
    IF (MW[7][J]>0 && MW[7][J]<7.0 )            ; IWAIT <7.0
MINUTE

MW[71]=AK_HBW_WLB*MW[6]+BK_HBW_WLB1*MW[7]+CK_HBW_WLB*MW[9]+DK_HBW
_WLB*ZI.1.PARK[J]+EK_HBW_WLB*MW[10]*100+FK_HBW_XLB*MW[8]+K_HBW_WL
B+K_HBW_WLB_0C         ; WALK TO LOCAL BUS UTILITY, 0 CAR

MW[72]=AK_HBW_WLB*MW[6]+BK_HBW_WLB1*MW[7]+CK_HBW_WLB*MW[9]+DK_HBW
_WLB*ZI.1.PARK[J]+EK_HBW_WLB*MW[10]*100+FK_HBW_XLB*MW[8]+K_HBW_WL
B+K_HBW_WLB_1C         ; WALK TO LOCAL BUS UTILITY, 1 CAR

MW[73]=AK_HBW_WLB*MW[6]+BK_HBW_WLB1*MW[7]+CK_HBW_WLB*MW[9]+DK_HBW
_WLB*ZI.1.PARK[J]+EK_HBW_WLB*MW[10]*100+FK_HBW_XLB*MW[8]+K_HBW_WL
B+K_HBW_WLB_2C         ; WALK TO LOCAL BUS UTILITY, 2+ CAR
    ELSE IF (MW[7][J]>=7.0)                     ; IWAIT > 7.0
MINUTE

MW[71]=AK_HBW_WLB*MW[6]+BK_HBW_WLB2*MW[7]+CK_HBW_WLB*MW[9]+DK_HBW
_WLB*ZI.1.PARK[J]+EK_HBW_WLB*MW[10]*100+FK_HBW_XLB*MW[8]+K_HBW_WL
B+K_HBW_WLB_0C         ; WALK TO LOCAL BUS UTILITY, 0 CAR

MW[72]=AK_HBW_WLB*MW[6]+BK_HBW_WLB2*MW[7]+CK_HBW_WLB*MW[9]+DK_HBW
_WLB*ZI.1.PARK[J]+EK_HBW_WLB*MW[10]*100+FK_HBW_XLB*MW[8]+K_HBW_WL
B+K_HBW_WLB_1C         ; WALK TO LOCAL BUS UTILITY, 1 CAR

MW[73]=AK_HBW_WLB*MW[6]+BK_HBW_WLB2*MW[7]+CK_HBW_WLB*MW[9]+DK_HBW
_WLB*ZI.1.PARK[J]+EK_HBW_WLB*MW[10]*100+FK_HBW_XLB*MW[8]+K_HBW_WL
B+K_HBW_WLB_2C         ; WALK TO LOCAL BUS UTILITY, 2+ CAR
    ENDIF
  ENDJLOOP

  JLOOP INCLUDE = 1-731
    IF (MW[17][J]>0 && MW[17][J]<7.0 )            ; IWAIT <7.0
MINUTE

MW[81]=AK_HBW_WEB*MW[16]+BK_HBW_WEB1*MW[17]+CK_HBW_WEB*MW[19]+DK_
HBW_WEB*ZI.1.PARK[J]+EK_HBW_WEB*MW[20]*100+FK_HBW_XLB*MW[18]+K_HB
W_WEB+K_HBW_WEB_0C         ; WALK TO EXPRESS BUS UTILITY, 0 CAR

MW[82]=AK_HBW_WEB*MW[16]+BK_HBW_WEB1*MW[17]+CK_HBW_WEB*MW[19]+DK_
HBW_WEB*ZI.1.PARK[J]+EK_HBW_WEB*MW[20]*100+FK_HBW_XLB*MW[18]+K_HB
W_WEB+K_HBW_WEB_1C         ; WALK TO EXPRESS BUS UTILITY, 1 CAR

MW[83]=AK_HBW_WEB*MW[16]+BK_HBW_WEB1*MW[17]+CK_HBW_WEB*MW[19]+DK_
HBW_WEB*ZI.1.PARK[J]+EK_HBW_WEB*MW[20]*100+FK_HBW_XLB*MW[18]+K_HB
W_WEB+K_HBW_WEB_2C         ; WALK TO EXPRESS BUS UTILITY, 2+ CAR
    ELSE IF (MW[17][J]>=7.0)                     ; IWAIT > 7.0
MINUTE

MW[81]=AK_HBW_WEB*MW[16]+BK_HBW_WEB2*MW[17]+CK_HBW_WEB*MW[19]+DK_
```

```
HBW_WEB*ZI.1.PARK[J]+EK_HBW_WEB*MW[20]*100+FK_HBW_XLB*MW[18]+K_HB
W_WEB+K_HBW_WEB_0C          ; WALK TO EXPRESS BUS UTILITY, 0 CAR

MW[82]=AK_HBW_WEB*MW[16]+BK_HBW_WEB2*MW[17]+CK_HBW_WEB*MW[19]+DK_
HBW_WEB*ZI.1.PARK[J]+EK_HBW_WEB*MW[20]*100+FK_HBW_XLB*MW[18]+K_HB
W_WEB+K_HBW_WEB_1C          ; WALK TO EXPRESS BUS UTILITY, 1 CAR

MW[83]=AK_HBW_WEB*MW[16]+BK_HBW_WEB2*MW[17]+CK_HBW_WEB*MW[19]+DK_
HBW_WEB*ZI.1.PARK[J]+EK_HBW_WEB*MW[20]*100+FK_HBW_XLB*MW[18]+K_HB
W_WEB+K_HBW_WEB_2C          ; WALK TO EXPRESS BUS UTILITY, 2+ CAR
    ENDIF
  ENDJLOOP

  JLOOP INCLUDE = 1-731
    IF (MW[27][J]>0 && MW[27][J]<7.0 )            ; IWAIT <7.0
MINUTE

MW[91]=AK_HBW_AEB*MW[26]+BK_HBW_AEB1*MW[27]+CK_HBW_AEB*MW[29]+DK_
HBW_AEB*ZI.1.PARK[J]+EK_HBW_AEB*MW[30]*100+FK_HBW_XLB*MW[28]+K_HB
W_AEB+K_HBW_AEB_0C          ; AUTO TO EXPRESS BUS UTILITY, 0 CAR

MW[92]=AK_HBW_AEB*MW[26]+BK_HBW_AEB1*MW[27]+CK_HBW_AEB*MW[29]+DK_
HBW_AEB*ZI.1.PARK[J]+EK_HBW_AEB*MW[30]*100+FK_HBW_XLB*MW[28]+K_HB
W_AEB+K_HBW_AEB_1C          ; AUTO TO EXPRESS BUS UTILITY, 1 CAR

MW[93]=AK_HBW_AEB*MW[26]+BK_HBW_AEB1*MW[27]+CK_HBW_AEB*MW[29]+DK_
HBW_AEB*ZI.1.PARK[J]+EK_HBW_AEB*MW[30]*100+FK_HBW_XLB*MW[28]+K_HB
W_AEB+K_HBW_AEB_2C          ; AUTO TO EXPRESS BUS UTILITY, 2+ CAR
    ELSE IF (MW[27][J]>=7.0)                      ; IWAIT > 7.0
MINUTE

MW[91]=AK_HBW_AEB*MW[26]+BK_HBW_AEB2*MW[27]+CK_HBW_AEB*MW[29]+DK_
HBW_AEB*ZI.1.PARK[J]+EK_HBW_AEB*MW[30]*100+FK_HBW_XLB*MW[28]+K_HB
W_AEB+K_HBW_AEB_0C          ; AUTO TO EXPRESS BUS UTILITY, 0 CAR

MW[92]=AK_HBW_AEB*MW[26]+BK_HBW_AEB2*MW[27]+CK_HBW_AEB*MW[29]+DK_
HBW_AEB*ZI.1.PARK[J]+EK_HBW_AEB*MW[30]*100+FK_HBW_XLB*MW[28]+K_HB
W_AEB+K_HBW_AEB_1C          ; AUTO TO EXPRESS BUS UTILITY, 1 CAR

MW[93]=AK_HBW_AEB*MW[26]+BK_HBW_AEB2*MW[27]+CK_HBW_AEB*MW[29]+DK_
HBW_AEB*ZI.1.PARK[J]+EK_HBW_AEB*MW[30]*100+FK_HBW_XLB*MW[28]+K_HB
W_AEB+K_HBW_AEB_2C          ; AUTO TO EXPRESS BUS UTILITY, 2+ CAR
    ENDIF
  ENDJLOOP

; DO EXPONENTIAL OF UTILITIES

MW[101] = EXP(MW[41])*100000   ; DA 0 CAR
MW[102] = EXP(MW[42])*100000   ; DA 1 CAR
MW[103] = EXP(MW[43])*100000   ; DA 2 CAR

MW[111] = EXP(MW[51])*100000   ; SR2 0 CAR
MW[112] = EXP(MW[52])*100000   ; SR2 1 CAR
MW[113] = EXP(MW[53])*100000   ; SR2 2 CAR

MW[121] = EXP(MW[61])*100000   ; SR3 0 CAR
MW[122] = EXP(MW[62])*100000   ; SR3 1 CAR
```

```
MW[123] = EXP(MW[63])*100000   ; SR3 2 CAR

JLOOP INCLUDE = 1-731
  IF (MW[9][J]>0)                   ; WALK TO LOCAL IVTT >0.0
    MW[131] = EXP(MW[71])*100000   ; WALK TO LOCAL BUS 0 CAR
    MW[132] = EXP(MW[72])*100000   ; WALK TO LOCAL BUS 1 CAR
    MW[133] = EXP(MW[73])*100000   ; WALK TO LOCAL BUS 2+ CAR
  ENDIF
ENDJLOOP

JLOOP INCLUDE = 1-731
  IF (MW[19][J]>0)                  ; WALK TO PREMIUM IVTT >0.0
    MW[141] = EXP(MW[81])*100000   ; WALK TO express BUS 0 CAR
    MW[142] = EXP(MW[82])*100000   ; WALK TO EXPRESS BUS 1 CAR
    MW[143] = EXP(MW[83])*100000   ; WALK TO EXPRESS BUS 2+ CAR
  ENDIF
ENDJLOOP

JLOOP INCLUDE = 1-731
  IF (MW[29][J]>0)                  ; AUTO TO PREMIUM IVTT >0.0
    MW[151] = EXP(MW[91])*100000   ; Auto TO express BUS 0 CAR
    MW[152] = EXP(MW[92])*100000   ; Auto TO EXPRESS BUS 1 CAR
    MW[153] = EXP(MW[93])*100000   ; Auto TO EXPRESS BUS 2+ CAR
  ENDIF
ENDJLOOP

; SHARE FOR ZERO CAR HH
MW[171] =
100000*MW[101]/(MW[101]+MW[111]+MW[121]+MW[131]+MW[141]+MW[151])
; DA SHARE for 0 Car
MW[172] =
100000*MW[111]/(MW[101]+MW[111]+MW[121]+MW[131]+MW[141]+MW[151])
; SR2 SHARE for 0 Car
MW[173] =
100000*MW[121]/(MW[101]+MW[111]+MW[121]+MW[131]+MW[141]+MW[151])
; SR3 SHARE for 0 Car
MW[174] =
100000*MW[131]/(MW[101]+MW[111]+MW[121]+MW[131]+MW[141]+MW[151])
; Walk to Local Share for 0 Car
MW[175] =
100000*MW[141]/(MW[101]+MW[111]+MW[121]+MW[131]+MW[141]+MW[151])
; Walk to premium Share for 0 Car
MW[176] =
100000*MW[151]/(MW[101]+MW[111]+MW[121]+MW[131]+MW[141]+MW[151])
; Auto to premium Share for 0 Car

; SHARE FOR ONE CAR HH
MW[181] =
100000*MW[102]/(MW[102]+MW[112]+MW[122]+MW[132]+MW[142]+MW[152])
; DA SHARE for 1 Car

MW[182] =
100000*MW[112]/(MW[102]+MW[112]+MW[122]+MW[132]+MW[142]+MW[152])
; SR2 SHARE for 1 Car
MW[183] =
100000*MW[122]/(MW[102]+MW[112]+MW[122]+MW[132]+MW[142]+MW[152])
; SR3 SHARE for 1 Car
```

```
MW[184] =
100000*MW[132]/(MW[102]+MW[112]+MW[122]+MW[132]+MW[142]+MW[152])
; Walk to Local Share for 1 Car
MW[185] =
100000*MW[142]/(MW[102]+MW[112]+MW[122]+MW[132]+MW[142]+MW[152])
; Walk to premium Share for 1 Car
MW[186] =
100000*MW[152]/(MW[102]+MW[112]+MW[122]+MW[132]+MW[142]+MW[152])
; Auto to premium Share for 1 Car

; SHARE FOR TWO+ CAR HH
MW[191] =
100000*MW[103]/(MW[103]+MW[113]+MW[123]+MW[133]+MW[143]+MW[153])
; DA SHARE for 2+ Car
MW[192] =
100000*MW[113]/(MW[103]+MW[113]+MW[123]+MW[133]+MW[143]+MW[153])
; SR2 SHARE for 2+ Car
MW[193] =
100000*MW[123]/(MW[103]+MW[113]+MW[123]+MW[133]+MW[143]+MW[153])
; SR3 SHARE for 2+ Car
MW[194] =
100000*MW[133]/(MW[103]+MW[113]+MW[123]+MW[133]+MW[143]+MW[153])
; Walk to Local Share for 2+ Car
MW[195] =
100000*MW[143]/(MW[103]+MW[113]+MW[123]+MW[133]+MW[143]+MW[153])
; Walk to premium Share for 2+ Car
MW[196] =
100000*MW[153]/(MW[103]+MW[113]+MW[123]+MW[133]+MW[143]+MW[153])
; Auto to premium Share for 2+ Car

MW[197]=MI.9.1*ZI.2.CAR0
MW[198]=MI.9.1*ZI.2.CAR1
MW[199]=MI.9.1-MW[197]-MW[198]
MW[200]=MW[197]+MW[198]

MW[134] = MW[171]*MW[197]                ;DA TRIPS FOR ZERO CAR
MW[135] = MW[172]*MW[197]                ;SR2 TRIPS FOR ZERO CAR
MW[136] = MW[173]*MW[197]                ;SR3+ TRIPS FOR ZERO CAR
MW[137] = MW[174]*MW[197]                ;WALK TO LOCAL TRIPS FOR
ZERO CAR
MW[138] = MW[175]*MW[197]                ;WALK_PREM TRIPS FOR
ZERO CAR
MW[139] = MW[176]*MW[197]                ;DRV_PREM TRIPS FOR ZERO
CAR

MW[144] = MW[181]*MW[198]                ;DA TRIPS FOR ONE CAR
MW[145] = MW[182]*MW[198]                ;SR2 TRIPS FOR ONE CAR
MW[146] = MW[183]*MW[198]                ;SR3+ TRIPS FOR ONE CAR
MW[147] = MW[184]*MW[198]                ;WALK TO LOCAL TRIPS FOR
ONE CAR
MW[148] = MW[185]*MW[198]                ;WALK_PREM TRIPS FOR ONE
CAR
MW[149] = MW[186]*MW[198]                ;DRV_PREM TRIPS FOR ONE
CAR

MW[154] = MW[191]*MW[199]                ;DA TRIPS FOR TWO+ CAR
MW[155] = MW[192]*MW[199]                ;SR2 TRIPS FOR TWO+ CAR
```

```
MW[156] = MW[193]*MW[199]                ;SR3+ TRIPS FOR TWO+ CAR
MW[157] = MW[194]*MW[199]                ;WALK TO LOCAL TRIPS FOR
TWO+ CAR
MW[158] = MW[195]*MW[199]                ;WALK_PREM TRIPS FOR
TWO+ CAR
MW[159] = MW[196]*MW[199]                ;DRV_PREM TRIPS FOR TWO+
CAR

; FINAL TRIPS

MW[161]=(MW[134]+MW[144]+MW[154])/100000              ; DA TRIPS
MW[162]=(MW[135]+MW[145]+MW[155])/100000              ; SR2 TRIPS
MW[163]=(MW[136]+MW[146]+MW[156])/100000              ; SR3+
TRIPS
MW[164]=(MW[137]+MW[147]+MW[157])/100000              ; WALK TO
LOCAL BUS TRIPS
MW[165]=(MW[138]+MW[148]+MW[158])/100000              ; WALK TO
PREMIUM BUS TRIPS
MW[166]=(MW[139]+MW[149]+MW[159])/100000              ; DRIVE TO
PREMIUM BUS TRIPS

; zero car HH
MW[201]=MW[134]/100000
MW[202]=MW[135]/100000
MW[203]=MW[136]/100000
MW[204]=MW[137]/100000
MW[205]=MW[138]/100000
MW[206]=MW[139]/100000
; 1 car HH
MW[211]=MW[144]/100000
MW[212]=MW[145]/100000
MW[213]=MW[146]/100000
MW[214]=MW[147]/100000
MW[215]=MW[148]/100000
MW[216]=MW[149]/100000
; Two+ car HH
MW[221]=MW[154]/100000
MW[222]=MW[155]/100000
MW[223]=MW[156]/100000
MW[224]=MW[157]/100000
MW[225]=MW[158]/100000
MW[226]=MW[159]/100000

ENDRUN
```

2. 01MAT00D.S

```
; TREASURE COAST HBNW MODE CHOICE MODEL
; Do not change filenames or add or remove FILEI/FILEO statements
using an editor. Use Cube/Application Manager.
RUN PGM=MATRIX
FILEI MATI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\TSKIMMD1.MAT"
FILEI MATI[2] = "D:\FSUTMS\D4\TCRPM\CUBE\TSKIMMD2.MAT"
FILEI MATI[3] = "D:\FSUTMS\D4\TCRPM\CUBE\TSKIMMD3.MAT"
FILEI MATI[4] = "D:\FSUTMS\D4\TCRPM\CUBE\TSKIMMD4.MAT"
FILEI MATI[5] = "D:\FSUTMS\D4\TCRPM\CUBE\TSKIMMD5.MAT"
FILEI MATI[6] = "D:\FSUTMS\D4\TCRPM\CUBE\TSKIMMD6.MAT"
FILEI MATI[7] = "D:\FSUTMS\D4\TCRPM\CUBE\TSKIMMD7.MAT"
FILEI MATI[8] = "D:\FSUTMS\D4\TCRPM\CUBE\FHSKIMS.T00.MAT"
FILEI MATI[9] = "D:\FSUTMS\D4\TCRPM\CUBE\PTRIPS11.MAT"
FILEI ZDATI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\PARK.DBF",
 Z=TAZ
FILEI ZDATI[2] = "D:\FSUTMS\D4\TCRPM\CUBE\HHCAR.DBF",
 Z=TAZ
FILEO MATO[1] = "D:\FSUTMS\D4\TCRPM\CUBE\MC_HBNW.MAT",
 MO=161-166, DEC=3,3,3,3,3,3, NAME=
DA,SR2,SR3,WALK_LOCAL,WALK_PREM,AUTO_PREM
FILEO MATO[2] = "D:\FSUTMS\D4\TCRPM\CUBE\HBNW_TEST.MAT",
 MO=197-199, DEC=3,3,3, NAME=0CARHH, 1CARHH, 2CARHH
FILEO MATO[3] = "D:\FSUTMS\D4\TCRPM\CUBE\ZEROCAR_HBNW.MAT",
 MO=201-206, DEC=3,3,3,3,3,3
FILEO MATO[4] = "D:\FSUTMS\D4\TCRPM\CUBE\ONECAR_HBNW.MAT",
 MO=211-216, DEC=3,3,3,3,3,3
FILEO MATO[5] = "D:\FSUTMS\D4\TCRPM\CUBE\TWOCAR_HBNW.MAT",
 MO=221-226, DEC=3,3,3,3,3,3

;**********************************************
; TREASURE COAST MODE CHOICE MODELSM, HBNW
;**********************************************
 AUTOCOST  = 15.0            ; auto operating cost in cents
 K_HBNW_DA  = 0.0             ; MODAL CONSTANT FOR HBNW DA
 K_HBNW_DA_0C  = 0.0          ; CONSTANT FOR HBNW DA ZERO CAR
(WAS 0)
 K_HBNW_DA_1C  = 0.0          ; CONSTANT FOR HBNW DA ONE CAR
(WAS 0)
 K_HBNW_DA_2C  = 0.0          ; CONSTANT FOR HBNW DA TWO+ CAR
(WAS 0)


 K_HBNW_SR2 = -0.4542         ; MODAL CONSTANT FOR HBNW SR2
 K_HBNW_SR2_0C = 0.375          ; CONSTANT FOR HBNW SR2, ZERO CAR
 K_HBNW_SR2_1C = 0.377          ; CONSTANT FOR HBNW SR2, ONE CAR
 K_HBNW_SR2_2C = 0.255          ; CONSTANT FOR HBNW SR2, TWO+ CAR

 K_HBNW_SR3 = -0.6199         ; MODAL CONSTANT FOR HBNW SR3+
 K_HBNW_SR3_0C = 0.005          ; CONSTANT FOR HBNW SR3+ ZERO CAR
 K_HBNW_SR3_1C = -0.014          ; CONSTANT FOR HBNW SR3+ 1 CAR
 K_HBNW_SR3_2C = -0.06           ; CONSTANT FOR HBNW SR3+ 2+ CAR

 K_HBNW_WLB = -2.1837         ; CONSTANT FOR HBNW WALK TO LOCAL
BUS
```

```
 K_HBNW_WLB_0C = 2.32      ; CONSTANT FOR HBNW WALK TO LOCAL BUS,
0 CAR
 K_HBNW_WLB_1C = 0.75      ; CONSTANT FOR HBNW WALK TO LOCAL BUS,
1 CAR
 K_HBNW_WLB_2C = -0.72        ; CONSTANT FOR HBNW WALK TO LOCAL
BUS, 2+ CAR

 K_HBNW_WEB = -999   ; -999   ; CONSTANT FOR HBNW WALK TO EXPRESS
BUS
 K_HBNW_WEB_0C = -999   ; -999   ; CONSTANT FOR HBNW WALK TO
EXPRESS BUS, 0 CAR
 K_HBNW_WEB_1C = -999   ; -999   ; CONSTANT FOR HBNW WALK TO
EXPRESS BUS, 1 CAR
 K_HBNW_WEB_2C = -999   ; -999   ; CONSTANT FOR HBNW WALK TO
EXPRESS BUS, 2+ CAR

 K_HBNW_AEB = -999  ; -999    ; CONSTANT FOR HBNW AUTO TO EXPRESS
BUS
 K_HBNW_AEB_0C = -999  ; -999    ; CONSTANT FOR HBNW AUTO TO
EXPRESS BUS, 0 CAR
 K_HBNW_AEB_1C = -999  ; -999    ; CONSTANT FOR HBNW AUTO TO
EXPRESS BUS, 1 CAR
 K_HBNW_AEB_2C = -999  ; -999    ; CONSTANT FOR HBNW AUTO TO
EXPRESS BUS, 2+ CAR

 AK_HBNW_DA  = 0.00        ; WALK TIME COEFFICIENT FOR HBNW DA
 AK_HBNW_SR2 = 0.00        ; WALK TIME COEFFICIENT FOR HBNW SR2
 AK_HBNW_SR3 = 0.00        ; WALK TIME COEFFICIENT FOR HBNW SR3+
 AK_HBNW_WLB = -0.035      ; WALK TIME COEFFICIENT FOR HBNW WALK
TO LOCAL BUS
 AK_HBNW_WEB = -0.035      ; WALK TIME COEFFICIENT FOR HBNW WALK
TO EXPRESS BUS
 AK_HBNW_AEB = -0.015      ; AUTO ACCESS time COEFFICIENT FOR
HBNW AUTO TO EXPRESS BUS

 BK_HBNW_DA  = -0.035         ; WAIT/Terminal TIME COEFFICIENT
FOR HBNW DA
 BK_HBNW_SR2 = -0.035         ; WAIT/terminal TIME COEFFICIENT
FOR HBNW SR2
 BK_HBNW_SR3 = -0.035         ; WAIT/Terminal TIME COEFFICIENT
FOR HBNW SR3+
 BK_HBNW_WLB1 = -0.035     ; First WAIT TIME COEFFICIENT FOR HBNW
WALK TO LOCAL BUS, wait<7 min
 BK_HBNW_WLB2 = -0.035     ; First WAIT TIME COEFFICIENT FOR HBNW
WALK TO LOCAL BUS, wait>7 min
 BK_HBNW_WEB1 = -0.035     ; First WAIT TIME COEFFICIENT FOR HBNW
Walk TO express BUS, wait<7 min
 BK_HBNW_WEB2 = -0.035     ; First WAIT TIME COEFFICIENT FOR HBNW
Walk TO express BUS, wait>7 min
 BK_HBNW_AEB1 = -0.035     ; First WAIT TIME COEFFICIENT FOR HBNW
AUTO TO LOCAL BUS, wait<7 min
 BK_HBNW_AEB2 = -0.035     ; First WAIT TIME COEFFICIENT FOR HBNW
AUTO TO LOCAL BUS, wait>7 min

 CK_HBNW_DA  = -0.015      ; IVTT TIME COEFFICIENT FOR HBNW DA
 CK_HBNW_SR2 = -0.015      ; IVTT TIME COEFFICIENT FOR HBNW SR2
 CK_HBNW_SR3 = -0.015      ; IVTT TIME COEFFICIENT FOR HBNW SR3+
```

```
 CK_HBNW_WLB = -0.015      ; IVTT TIME COEFFICIENT FOR HBNW WALK
TO LOCAL BUS
 CK_HBNW_WEB = -0.015      ; IVTT TIME COEFFICIENT FOR HBNW WALK
TO EXPRESS BUS
 CK_HBNW_AEB = -0.015      ; IVTT TIME COEFFICIENT FOR HBNW AUTO
TO EXPRESS BUS

 DK_HBNW_DA  = -0.0048       ; PARKING COST COEFFICIENT FOR HBNW
DA
 DK_HBNW_SR2 = -0.0048       ; PARKING COST COEFFICIENT FOR HBNW
SR2
 DK_HBNW_SR3 = -0.0048       ; PARKING COST COEFFICIENT FOR HBNW
SR3+
 DK_HBNW_WLB = 0.0       ; PARKING COST COEFFICIENT FOR HBNW WALK
TO LOCAL BUS
 DK_HBNW_WEB = 0.0       ; PARKING COST COEFFICIENT FOR HBNW WALK
TO EXPRESS BUS
 DK_HBNW_AEB = -0.0048       ; PARKING COST COEFFICIENT FOR HBNW
AUTO TO EXPRESS BUS

 EK_HBNW_DA  = -0.0048       ; HIGHWAY COST COEFFICIENT FOR HBNW
DA
 EK_HBNW_SR2 = -0.0048       ; HIGHWAY COST COEFFICIENT FOR HBNW
SR2
 EK_HBNW_SR3 = -0.0048       ; HIGHWAY COST COEFFICIENT FOR HBNW
SR3+
 EK_HBNW_WLB = -0.0048       ; FARE COEFFICIENT FOR HBNW WALK TO
LOCAL BUS
 EK_HBNW_WEB = -0.0048       ; FARE COEFFICIENT FOR HBNW WALK TO
EXPRESS BUS
 EK_HBNW_AEB = -0.0048       ; FARE COEFFICIENT FOR HBNW AUTO TO
EXPRESS BUS
 FK_HBNW_XLB = -0.035        ; TRANSFER COEFF, HBNW

  MW[1]=MI.8.8     ; AUTO TERMINAL TIME
  MW[2]=MI.8.7     ; TOLL TIME
  MW[3]=MI.8.3     ; AUTO IVTT
  MW[4]=MI.8.2     ; AUTO DISTANCE
  MW[5]=MI.8.7+MI.8.3   ; TOTAL AUTO TIME

  MW[6]= MI.1.1+MI.1.2                        ; WALK ACCESS TIME
, WALK TO LOCAL
  MW[7]= MI.1.10                              ; FIRST WAIT for
walk to local
  MW[8]= MI.1.11                              ; XFER WAIT FOR
WALK TO LOCAL
  MW[9]= MI.1.4                               ; IVTT, WALK TO
LOCAL
  MW[10]= MI.1.13                             ; WALK TO LOCAL
FARE

  MW[16]= MI.2.1+MI.2.2                       ; WALK ACCESS TIME
, WALK TO EXPRESS BUS
  MW[17]= MI.2.10                             ; FIRST WAIT FOR
WALK TO EXPRESS
  MW[18]= MI.2.11                             ; XFER WAIT FOR
WALK TO EXPRESS
```

```
  MW[19]= MI.2.4                                  ; IVTT, WALK TO
EXPRESS
  MW[20]= MI.2.13                                 ; WALK TO EXPRESS
FARE

  MW[26]= MI.5.1+MI.5.2                           ; AUTO ACCESS TIME
, AUTO TO EXPRESS BUS
  MW[27]= MI.5.10                                 ; FIRST WAIT FOR
AUTO TO EXPRESS
  MW[28]= MI.5.11                                 ; XFER WAIT FOR
AUTO TO EXPRESS
  MW[29]= MI.5.4                                  ; IVTT, AUTO TO
EXPRESS
  MW[30]= MI.5.13                                 ; AUTO TO EXPRESS
FARE


MW[41]=AK_HBNW_DA*0.0+BK_HBNW_DA*MW[1]+CK_HBNW_DA*MW[5]+DK_HBNW_D
A*ZI.1.PARK[J]+EK_HBNW_DA*AUTOCOST*MW[4]+K_HBNW_DA+K_HBNW_DA_0C
; DA UTILITY, 0 CAR HH

MW[42]=AK_HBNW_DA*0.0+BK_HBNW_DA*MW[1]+CK_HBNW_DA*MW[5]+DK_HBNW_D
A*ZI.1.PARK[J]+EK_HBNW_DA*AUTOCOST*MW[4]+K_HBNW_DA+K_HBNW_DA_1C
; DA UTILITY, 1 CAR HH

MW[43]=AK_HBNW_DA*0.0+BK_HBNW_DA*MW[1]+CK_HBNW_DA*MW[5]+DK_HBNW_D
A*ZI.1.PARK[J]+EK_HBNW_DA*AUTOCOST*MW[4]+K_HBNW_DA+K_HBNW_DA_2C
; DA UTILITY, 2+ CAR HH


MW[51]=AK_HBNW_SR2*0.0+BK_HBNW_SR2*MW[1]+CK_HBNW_SR2*MW[5]+DK_HBN
W_SR2*ZI.1.PARK[J]+EK_HBNW_SR2*AUTOCOST*MW[4]+K_HBNW_SR2+K_HBNW_S
R2_0C        ; SR2 UTILITY, 0 CAR

MW[52]=AK_HBNW_SR2*0.0+BK_HBNW_SR2*MW[1]+CK_HBNW_SR2*MW[5]+DK_HBN
W_SR2*ZI.1.PARK[J]+EK_HBNW_SR2*AUTOCOST*MW[4]+K_HBNW_SR2+K_HBNW_S
R2_1C        ; SR2 UTILITY, 1 CAR

MW[53]=AK_HBNW_SR2*0.0+BK_HBNW_SR2*MW[1]+CK_HBNW_SR2*MW[5]+DK_HBN
W_SR2*ZI.1.PARK[J]+EK_HBNW_SR2*AUTOCOST*MW[4]+K_HBNW_SR2+K_HBNW_S
R2_2C        ; SR2 UTILITY, 2 CAR


MW[61]=AK_HBNW_SR3*0.0+BK_HBNW_SR3*MW[1]+CK_HBNW_SR3*MW[5]+DK_HBN
W_SR3*ZI.1.PARK[J]+EK_HBNW_SR3*AUTOCOST*MW[4]+K_HBNW_SR3+K_HBNW_S
R3_0C        ; SR2 UTILITY, 0 CAR

MW[62]=AK_HBNW_SR3*0.0+BK_HBNW_SR3*MW[1]+CK_HBNW_SR3*MW[5]+DK_HBN
W_SR3*ZI.1.PARK[J]+EK_HBNW_SR3*AUTOCOST*MW[4]+K_HBNW_SR3+K_HBNW_S
R3_1C        ; SR3 UTILITY, 1 CAR

MW[63]=AK_HBNW_SR3*0.0+BK_HBNW_SR3*MW[1]+CK_HBNW_SR3*MW[5]+DK_HBN
W_SR3*ZI.1.PARK[J]+EK_HBNW_SR3*AUTOCOST*MW[4]+K_HBNW_SR3+K_HBNW_S
R3_2C        ; SR3 UTILITY, 2 CAR

  JLOOP INCLUDE = 1-731
```

```
    IF (MW[7][J]>0 && MW[7][J]<7.0 )            ; IWAIT <7.0
MINUTE

MW[71]=AK_HBNW_WLB*MW[6]+BK_HBNW_WLB1*MW[7]+CK_HBNW_WLB*MW[9]+DK_
HBNW_WLB*ZI.1.PARK[J]+EK_HBNW_WLB*MW[10]*100+FK_HBNW_XLB*MW[8]+K_
HBNW_WLB+K_HBNW_WLB_0C         ; WALK TO LOCAL BUS UTILITY, 0 CAR

MW[72]=AK_HBNW_WLB*MW[6]+BK_HBNW_WLB1*MW[7]+CK_HBNW_WLB*MW[9]+DK_
HBNW_WLB*ZI.1.PARK[J]+EK_HBNW_WLB*MW[10]*100+FK_HBNW_XLB*MW[8]+K_
HBNW_WLB+K_HBNW_WLB_1C         ; WALK TO LOCAL BUS UTILITY, 1 CAR

MW[73]=AK_HBNW_WLB*MW[6]+BK_HBNW_WLB1*MW[7]+CK_HBNW_WLB*MW[9]+DK_
HBNW_WLB*ZI.1.PARK[J]+EK_HBNW_WLB*MW[10]*100+FK_HBNW_XLB*MW[8]+K_
HBNW_WLB+K_HBNW_WLB_2C         ; WALK TO LOCAL BUS UTILITY, 2+
CAR
    ELSE IF (MW[7][J]>=7.0)                      ; IWAIT > 7.0
MINUTE

MW[71]=AK_HBNW_WLB*MW[6]+BK_HBNW_WLB2*MW[7]+CK_HBNW_WLB*MW[9]+DK_
HBNW_WLB*ZI.1.PARK[J]+EK_HBNW_WLB*MW[10]*100+FK_HBNW_XLB*MW[8]+K_
HBNW_WLB+K_HBNW_WLB_0C         ; WALK TO LOCAL BUS UTILITY, 0 CAR

MW[72]=AK_HBNW_WLB*MW[6]+BK_HBNW_WLB2*MW[7]+CK_HBNW_WLB*MW[9]+DK_
HBNW_WLB*ZI.1.PARK[J]+EK_HBNW_WLB*MW[10]*100+FK_HBNW_XLB*MW[8]+K_
HBNW_WLB+K_HBNW_WLB_1C         ; WALK TO LOCAL BUS UTILITY, 1 CAR

MW[73]=AK_HBNW_WLB*MW[6]+BK_HBNW_WLB2*MW[7]+CK_HBNW_WLB*MW[9]+DK_
HBNW_WLB*ZI.1.PARK[J]+EK_HBNW_WLB*MW[10]*100+FK_HBNW_XLB*MW[8]+K_
HBNW_WLB+K_HBNW_WLB_2C         ; WALK TO LOCAL BUS UTILITY, 2+
CAR
    ENDIF
  ENDJLOOP

  JLOOP INCLUDE = 1-731
    IF (MW[17][J]>0 && MW[17][J]<7.0 )            ; IWAIT <7.0
MINUTE

MW[81]=AK_HBNW_WEB*MW[16]+BK_HBNW_WEB1*MW[17]+CK_HBNW_WEB*MW[19]+
DK_HBNW_WEB*ZI.1.PARK[J]+EK_HBNW_WEB*MW[20]*100+FK_HBNW_XLB*MW[18
]+K_HBNW_WEB+K_HBNW_WEB_0C         ; WALK TO EXPRESS BUS UTILITY,
0 CAR

MW[82]=AK_HBNW_WEB*MW[16]+BK_HBNW_WEB1*MW[17]+CK_HBNW_WEB*MW[19]+
DK_HBNW_WEB*ZI.1.PARK[J]+EK_HBNW_WEB*MW[20]*100+FK_HBNW_XLB*MW[18
]+K_HBNW_WEB+K_HBNW_WEB_1C         ; WALK TO EXPRESS BUS UTILITY,
1 CAR

MW[83]=AK_HBNW_WEB*MW[16]+BK_HBNW_WEB1*MW[17]+CK_HBNW_WEB*MW[19]+
DK_HBNW_WEB*ZI.1.PARK[J]+EK_HBNW_WEB*MW[20]*100+FK_HBNW_XLB*MW[18
]+K_HBNW_WEB+K_HBNW_WEB_2C         ; WALK TO EXPRESS BUS UTILITY,
2+ CAR
    ELSE IF (MW[17][J]>=7.0)                      ; IWAIT > 7.0
MINUTE

MW[81]=AK_HBNW_WEB*MW[16]+BK_HBNW_WEB2*MW[17]+CK_HBNW_WEB*MW[19]+
DK_HBNW_WEB*ZI.1.PARK[J]+EK_HBNW_WEB*MW[20]*100+FK_HBNW_XLB*MW[18
```

```
]+K_HBNW_WEB+K_HBNW_WEB_0C         ; WALK TO EXPRESS BUS UTILITY,
0 CAR

MW[82]=AK_HBNW_WEB*MW[16]+BK_HBNW_WEB2*MW[17]+CK_HBNW_WEB*MW[19]+
DK_HBNW_WEB*ZI.1.PARK[J]+EK_HBNW_WEB*MW[20]*100+FK_HBNW_XLB*MW[18
]+K_HBNW_WEB+K_HBNW_WEB_1C         ; WALK TO EXPRESS BUS UTILITY,
1 CAR

MW[83]=AK_HBNW_WEB*MW[16]+BK_HBNW_WEB2*MW[17]+CK_HBNW_WEB*MW[19]+
DK_HBNW_WEB*ZI.1.PARK[J]+EK_HBNW_WEB*MW[20]*100+FK_HBNW_XLB*MW[18
]+K_HBNW_WEB+K_HBNW_WEB_2C         ; WALK TO EXPRESS BUS UTILITY,
2+ CAR
    ENDIF
  ENDJLOOP

  JLOOP INCLUDE = 1-731
    IF (MW[27][J]>0 && MW[27][J]<7.0 )            ; IWAIT <7.0
MINUTE

MW[91]=AK_HBNW_AEB*MW[26]+BK_HBNW_AEB1*MW[27]+CK_HBNW_AEB*MW[29]+
DK_HBNW_AEB*ZI.1.PARK[J]+EK_HBNW_AEB*MW[30]*100+FK_HBNW_XLB*MW[28
]+K_HBNW_AEB+K_HBNW_AEB_0C         ; AUTO TO EXPRESS BUS UTILITY,
0 CAR

MW[92]=AK_HBNW_AEB*MW[26]+BK_HBNW_AEB1*MW[27]+CK_HBNW_AEB*MW[29]+
DK_HBNW_AEB*ZI.1.PARK[J]+EK_HBNW_AEB*MW[30]*100+FK_HBNW_XLB*MW[28
]+K_HBNW_AEB+K_HBNW_AEB_1C         ; AUTO TO EXPRESS BUS UTILITY,
1 CAR

MW[93]=AK_HBNW_AEB*MW[26]+BK_HBNW_AEB1*MW[27]+CK_HBNW_AEB*MW[29]+
DK_HBNW_AEB*ZI.1.PARK[J]+EK_HBNW_AEB*MW[30]*100+FK_HBNW_XLB*MW[28
]+K_HBNW_AEB+K_HBNW_AEB_2C         ; AUTO TO EXPRESS BUS UTILITY,
2+ CAR
    ELSE IF (MW[27][J]>=7.0)                      ; IWAIT > 7.0
MINUTE

MW[91]=AK_HBNW_AEB*MW[26]+BK_HBNW_AEB2*MW[27]+CK_HBNW_AEB*MW[29]+
DK_HBNW_AEB*ZI.1.PARK[J]+EK_HBNW_AEB*MW[30]*100+FK_HBNW_XLB*MW[28
]+K_HBNW_AEB+K_HBNW_AEB_0C         ; AUTO TO EXPRESS BUS UTILITY,
0 CAR

MW[92]=AK_HBNW_AEB*MW[26]+BK_HBNW_AEB2*MW[27]+CK_HBNW_AEB*MW[29]+
DK_HBNW_AEB*ZI.1.PARK[J]+EK_HBNW_AEB*MW[30]*100+FK_HBNW_XLB*MW[28
]+K_HBNW_AEB+K_HBNW_AEB_1C         ; AUTO TO EXPRESS BUS UTILITY,
1 CAR

MW[93]=AK_HBNW_AEB*MW[26]+BK_HBNW_AEB2*MW[27]+CK_HBNW_AEB*MW[29]+
DK_HBNW_AEB*ZI.1.PARK[J]+EK_HBNW_AEB*MW[30]*100+FK_HBNW_XLB*MW[28
]+K_HBNW_AEB+K_HBNW_AEB_2C         ; AUTO TO EXPRESS BUS UTILITY,
2+ CAR
    ENDIF
  ENDJLOOP

; DO EXPONENTIAL OF UTILITIES

MW[101] = EXP(MW[41])*100000   ; DA 0 CAR
MW[102] = EXP(MW[42])*100000   ; DA 1 CAR
```

```
MW[103] = EXP(MW[43])*100000   ; DA 2 CAR

MW[111] = EXP(MW[51])*100000   ; SR2 0 CAR
MW[112] = EXP(MW[52])*100000   ; SR2 1 CAR
MW[113] = EXP(MW[53])*100000   ; SR2 2 CAR

MW[121] = EXP(MW[61])*100000   ; SR3 0 CAR
MW[122] = EXP(MW[62])*100000   ; SR3 1 CAR
MW[123] = EXP(MW[63])*100000   ; SR3 2 CAR
JLOOP INCLUDE = 1-731
  IF (MW[9][J]>0)                  ; WALK TO LOCAL IVTT >0.0
   MW[131] = EXP(MW[71])*100000   ; WALK TO LOCAL BUS 0 CAR
   MW[132] = EXP(MW[72])*100000   ; WALK TO LOCAL BUS 1 CAR
   MW[133] = EXP(MW[73])*100000   ; WALK TO LOCAL BUS 2+ CAR
  ENDIF
ENDJLOOP

JLOOP INCLUDE = 1-731
  IF (MW[19][J]>0)                 ; WALK TO PREMIUM IVTT >0.0
    MW[141] = EXP(MW[81])*100000   ; WALK TO express BUS 0 CAR
    MW[142] = EXP(MW[82])*100000   ; WALK TO EXPRESS BUS 1 CAR
    MW[143] = EXP(MW[83])*100000   ; WALK TO EXPRESS BUS 2+ CAR
  ENDIF
ENDJLOOP

JLOOP INCLUDE = 1-731
  IF (MW[29][J]>0)                 ; AUTO TO PREMIUM IVTT >0.0
    MW[151] = EXP(MW[91])*100000   ; Auto TO express BUS 0 CAR
    MW[152] = EXP(MW[92])*100000   ; Auto TO EXPRESS BUS 1 CAR
    MW[153] = EXP(MW[93])*100000   ; Auto TO EXPRESS BUS 2+ CAR
  ENDIF
ENDJLOOP

; SHARE FOR ZERO CAR HH
MW[171] =
100000*MW[101]/(MW[101]+MW[111]+MW[121]+MW[131]+MW[141]+MW[151])
; DA SHARE for 0 Car
MW[172] =
100000*MW[111]/(MW[101]+MW[111]+MW[121]+MW[131]+MW[141]+MW[151])
; SR2 SHARE for 0 Car
MW[173] =
100000*MW[121]/(MW[101]+MW[111]+MW[121]+MW[131]+MW[141]+MW[151])
; SR3 SHARE for 0 Car
MW[174] =
100000*MW[131]/(MW[101]+MW[111]+MW[121]+MW[131]+MW[141]+MW[151])
; Walk to Local Share for 0 Car
MW[175] =
100000*MW[141]/(MW[101]+MW[111]+MW[121]+MW[131]+MW[141]+MW[151])
; Walk to premium Share for 0 Car

MW[176] =
100000*MW[151]/(MW[101]+MW[111]+MW[121]+MW[131]+MW[141]+MW[151])
; Auto to premium Share for 0 Car

; SHARE FOR ONE CAR HH
```

```
MW[181] =
100000*MW[102]/(MW[102]+MW[112]+MW[122]+MW[132]+MW[142]+MW[152])
; DA SHARE for 1 Car
MW[182] =
100000*MW[112]/(MW[102]+MW[112]+MW[122]+MW[132]+MW[142]+MW[152])
; SR2 SHARE for 1 Car
MW[183] =
100000*MW[122]/(MW[102]+MW[112]+MW[122]+MW[132]+MW[142]+MW[152])
; SR3 SHARE for 1 Car
MW[184] =
100000*MW[132]/(MW[102]+MW[112]+MW[122]+MW[132]+MW[142]+MW[152])
; Walk to Local Share for 1 Car
MW[185] =
100000*MW[142]/(MW[102]+MW[112]+MW[122]+MW[132]+MW[142]+MW[152])
; Walk to premium Share for 1 Car
MW[186] =
100000*MW[152]/(MW[102]+MW[112]+MW[122]+MW[132]+MW[142]+MW[152])
; Auto to premium Share for 1 Car

; SHARE FOR TWO+ CAR HH
MW[191] =
100000*MW[103]/(MW[103]+MW[113]+MW[123]+MW[133]+MW[143]+MW[153])
; DA SHARE for 2+ Car
MW[192] =
100000*MW[113]/(MW[103]+MW[113]+MW[123]+MW[133]+MW[143]+MW[153])
; SR2 SHARE for 2+ Car
MW[193] =
100000*MW[123]/(MW[103]+MW[113]+MW[123]+MW[133]+MW[143]+MW[153])
; SR3 SHARE for 2+ Car
MW[194] =
100000*MW[133]/(MW[103]+MW[113]+MW[123]+MW[133]+MW[143]+MW[153])
; Walk to Local Share for 2+ Car
MW[195] =
100000*MW[143]/(MW[103]+MW[113]+MW[123]+MW[133]+MW[143]+MW[153])
; Walk to premium Share for 2+ Car
MW[196] =
100000*MW[153]/(MW[103]+MW[113]+MW[123]+MW[133]+MW[143]+MW[153])
; Auto to premium Share for 2+ Car

MW[197]=(MI.9.2+MI.9.3+MI.9.5)*ZI.2.CAR0
MW[198]=(MI.9.2+MI.9.3+MI.9.5)*ZI.2.CAR1
MW[199]=MI.9.2+MI.9.3+MI.9.5-MW[197]-MW[198]
MW[200]=MW[197]+MW[198]

MW[134] = MW[171]*MW[197]                ;DA TRIPS FOR ZERO CAR
MW[135] = MW[172]*MW[197]                ;SR2 TRIPS FOR ZERO CAR
MW[136] = MW[173]*MW[197]                ;SR3+ TRIPS FOR ZERO CAR
MW[137] = MW[174]*MW[197]                ;WALK TO LOCAL TRIPS FOR
ZERO CAR
MW[138] = MW[175]*MW[197]                ;WALK_PREM TRIPS FOR
ZERO CAR
MW[139] = MW[176]*MW[197]                ;DRV_PREM TRIPS FOR ZERO
CAR

MW[144] = MW[181]*MW[198]                ;DA TRIPS FOR ONE CAR
MW[145] = MW[182]*MW[198]                ;SR2 TRIPS FOR ONE CAR
MW[146] = MW[183]*MW[198]                ;SR3+ TRIPS FOR ONE CAR
```

```
MW[147] = MW[184]*MW[198]                ;WALK TO LOCAL TRIPS FOR
ONE CAR
MW[148] = MW[185]*MW[198]                ;WALK_PREM TRIPS FOR ONE
CAR
MW[149] = MW[186]*MW[198]                ;DRV_PREM TRIPS FOR ONE
CAR

MW[154] = MW[191]*MW[199]                ;DA TRIPS FOR TWO+ CAR
MW[155] = MW[192]*MW[199]                ;SR2 TRIPS FOR TWO+ CAR
MW[156] = MW[193]*MW[199]                ;SR3+ TRIPS FOR TWO+ CAR
MW[157] = MW[194]*MW[199]                ;WALK TO LOCAL TRIPS FOR
TWO+ CAR
MW[158] = MW[195]*MW[199]                ;WALK_PREM TRIPS FOR
TWO+ CAR
MW[159] = MW[196]*MW[199]                ;DRV_PREM TRIPS FOR TWO+
CAR

; FINAL TRIPS

MW[161]=(MW[134]+MW[144]+MW[154])/100000              ; DA TRIPS
MW[162]=(MW[135]+MW[145]+MW[155])/100000              ; SR2 TRIPS
MW[163]=(MW[136]+MW[146]+MW[156])/100000              ; SR3+
TRIPS
MW[164]=(MW[137]+MW[147]+MW[157])/100000              ; WALK TO
LOCAL BUS TRIPS
MW[165]=(MW[138]+MW[148]+MW[158])/100000              ; WALK TO
PREMIUM BUS TRIPS
MW[166]=(MW[139]+MW[149]+MW[159])/100000              ; DRIVE TO
PREMIUM BUS TRIPS

; zero car HH
MW[201]=MW[134]/100000
MW[202]=MW[135]/100000
MW[203]=MW[136]/100000
MW[204]=MW[137]/100000
MW[205]=MW[138]/100000
MW[206]=MW[139]/100000
; 1 car HH
MW[211]=MW[144]/100000
MW[212]=MW[145]/100000
MW[213]=MW[146]/100000
MW[214]=MW[147]/100000
MW[215]=MW[148]/100000
MW[216]=MW[149]/100000
; Two+ car HH
MW[221]=MW[154]/100000
MW[222]=MW[155]/100000
MW[223]=MW[156]/100000
MW[224]=MW[157]/100000
MW[225]=MW[158]/100000
MW[226]=MW[159]/100000

ENDRUN
```

3. 01MAT00E.S

```
; Do not change filenames or add or remove FILEI/FILEO statements
using an editor. Use Cube/Application Manager.
RUN PGM=MATRIX
FILEI MATI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\TSKIMMD1.MAT"
FILEI MATI[2] = "D:\FSUTMS\D4\TCRPM\CUBE\TSKIMMD2.MAT"
FILEI MATI[3] = "D:\FSUTMS\D4\TCRPM\CUBE\TSKIMMD3.MAT"
FILEI MATI[4] = "D:\FSUTMS\D4\TCRPM\CUBE\TSKIMMD4.MAT"
FILEI MATI[5] = "D:\FSUTMS\D4\TCRPM\CUBE\TSKIMMD5.MAT"
FILEI MATI[6] = "D:\FSUTMS\D4\TCRPM\CUBE\TSKIMMD6.MAT"
FILEI MATI[7] = "D:\FSUTMS\D4\TCRPM\CUBE\TSKIMMD7.MAT"
FILEI MATI[8] = "D:\FSUTMS\D4\TCRPM\CUBE\FHSKIMS.T00.MAT"
FILEI MATI[9] = "D:\FSUTMS\D4\TCRPM\CUBE\PTRIPS11.MAT"
FILEI ZDATI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\PARK.DBF",
 Z=TAZ
FILEI ZDATI[2] = "D:\FSUTMS\D4\TCRPM\CUBE\HHCAR.DBF",
 Z=TAZ
FILEO MATO[1] = "D:\FSUTMS\D4\TCRPM\CUBE\MC_NHB.MAT",
 MO=161-166,DEC=3,3,3,3,3,3, NAME=
DA,SR2,SR3,WALK_LOCAL,WALK_PREM,AUTO_PREM
FILEO MATO[2] = "D:\FSUTMS\D4\TCRPM\CUBE\NHB_TEST.MAT",
 MO=200,199, DEC=3,3, NAME=1CARHH,2CARHH
FILEO MATO[3] = "D:\FSUTMS\D4\TCRPM\CUBE\ONECAR_NHB.MAT",
 MO=201-206,DEC=3,3,3,3,3,3
FILEO MATO[4] = "D:\FSUTMS\D4\TCRPM\CUBE\TWOCAR_NHB.MAT",
 MO=211-216,DEC=3,3,3,3,3,3

;*********************************************
; TREASURE COAST MODE CHOICE MODELSM, NHB
;*********************************************
 AUTOCOST  = 15.0            ; auto operating cost in cents
 K_NHB_DA  = 0.0             ; MODAL CONSTANT FOR NHB DA
 K_NHB_DA_0C  = 0.0          ; CONSTANT FOR NHB DA ZERO CAR
 K_NHB_DA_1C  = 0.0          ; CONSTANT FOR NHB DA ONE CAR
 K_NHB_DA_2C  = 0.0          ; CONSTANT FOR NHB DA TWO+ CAR


 K_NHB_SR2 = -0.4542         ; MODAL CONSTANT FOR NHB SR2
 K_NHB_SR2_0C = 0.411          ; CONSTANT FOR NHB SR2, ZERO CAR
 K_NHB_SR2_1C = 0.411          ; CONSTANT FOR NHB SR2, ONE CAR
 K_NHB_SR2_2C = 0.411          ; CONSTANT FOR NHB SR2, TWO+ CAR

 K_NHB_SR3 = -0.6199         ; MODAL CONSTANT FOR NHB SR3+
 K_NHB_SR3_0C = -0.477          ; CONSTANT FOR NHB SR3+ ZERO CAR
 K_NHB_SR3_1C = -0.477         ; CONSTANT FOR NHB SR3+ 1 CAR
 K_NHB_SR3_2C = -0.477          ; CONSTANT FOR NHB SR3+ 2+ CAR

 K_NHB_WLB = -2.1837         ; CONSTANT FOR NHB WALK TO LOCAL BUS
 K_NHB_WLB_0C = -0.89        ; CONSTANT FOR NHB WALK TO LOCAL
BUS, 0 CAR
 K_NHB_WLB_1C = -0.89        ; CONSTANT FOR NHB WALK TO LOCAL
BUS, 1 CAR
 K_NHB_WLB_2C = -0.89        ; CONSTANT FOR NHB WALK TO LOCAL
BUS, 2+ CAR9
```

```
 K_NHB_WEB = -999            ; CONSTANT FOR NHB WALK TO EXPRESS
BUS
 K_NHB_WEB_0C = -999         ; CONSTANT FOR NHB WALK TO EXPRESS
BUS, 0 CAR
 K_NHB_WEB_1C = -999         ; CONSTANT FOR NHB WALK TO EXPRESS
BUS, 1 CAR
 K_NHB_WEB_2C = -999         ; CONSTANT FOR NHB WALK TO EXPRESS
BUS, 2+ CAR

 K_NHB_AEB = -999  ;         ; CONSTANT FOR NHB AUTO TO EXPRESS
BUS
 K_NHB_AEB_0C = -999         ; CONSTANT FOR NHB AUTO TO EXPRESS
BUS, 0 CAR
 K_NHB_AEB_1C = -999         ; CONSTANT FOR NHB AUTO TO EXPRESS
BUS, 1 CAR
 K_NHB_AEB_2C = -999         ; CONSTANT FOR NHB AUTO TO EXPRESS
BUS, 2+ CAR

 AK_NHB_DA  = 0.00        ; WALK TIME COEFFICIENT FOR NHB DA
 AK_NHB_SR2 = 0.00        ; WALK TIME COEFFICIENT FOR NHB SR2
 AK_NHB_SR3 = 0.00        ; WALK TIME COEFFICIENT FOR NHB SR3+
 AK_NHB_WLB = -0.045      ; WALK TIME COEFFICIENT FOR NHB WALK TO
LOCAL BUS
 AK_NHB_WEB = -0.045      ; WALK TIME COEFFICIENT FOR NHB WALK TO
EXPRESS BUS
 AK_NHB_AEB = -0.018      ; AUTO ACCESS time COEFFICIENT FOR NHB
AUTO TO EXPRESS BUS

 BK_NHB_DA  = -0.045         ; WAIT/Terminal TIME COEFFICIENT FOR
NHB DA
 BK_NHB_SR2 = -0.045         ; WAIT/terminal TIME COEFFICIENT FOR
NHB SR2
 BK_NHB_SR3 = -0.045         ; WAIT/Terminal TIME COEFFICIENT FOR
NHB SR3+
 BK_NHB_WLB1 = -0.045     ; First WAIT TIME COEFFICIENT FOR NHB
WALK TO LOCAL BUS, wait<7 min
 BK_NHB_WLB2 = -0.045     ; First WAIT TIME COEFFICIENT FOR NHB
WALK TO LOCAL BUS, wait>7 min
 BK_NHB_WEB1 = -0.045     ; First WAIT TIME COEFFICIENT FOR NHB
Walk TO express BUS, wait<7 min
 BK_NHB_WEB2 = -0.045     ; First WAIT TIME COEFFICIENT FOR NHB
Walk TO express BUS, wait>7 min
 BK_NHB_AEB1 = -0.045     ; First WAIT TIME COEFFICIENT FOR NHB
AUTO TO LOCAL BUS, wait<7 min
 BK_NHB_AEB2 = -0.045     ; First WAIT TIME COEFFICIENT FOR NHB
AUTO TO LOCAL BUS, wait>7 min

 CK_NHB_DA  = -0.018      ; IVTT TIME COEFFICIENT FOR NHB DA
 CK_NHB_SR2 = -0.018      ; IVTT TIME COEFFICIENT FOR NHB SR2
 CK_NHB_SR3 = -0.018      ; IVTT TIME COEFFICIENT FOR NHB SR3+
 CK_NHB_WLB = -0.018      ; IVTT TIME COEFFICIENT FOR NHB WALK TO
LOCAL BUS
 CK_NHB_WEB = -0.018      ; IVTT TIME COEFFICIENT FOR NHB WALK TO
EXPRESS BUS
 CK_NHB_AEB = -0.018      ; IVTT TIME COEFFICIENT FOR NHB AUTO TO
EXPRESS BUS
```

```
 DK_NHB_DA  = -0.0048       ; PARKING COST COEFFICIENT FOR NHB DA
 DK_NHB_SR2 = -0.0048       ; PARKING COST COEFFICIENT FOR NHB
SR2
 DK_NHB_SR3 = -0.0048       ; PARKING COST COEFFICIENT FOR NHB
SR3+
 DK_NHB_WLB = 0.0       ; PARKING COST COEFFICIENT FOR NHB WALK
TO LOCAL BUS
 DK_NHB_WEB = 0.0       ; PARKING COST COEFFICIENT FOR NHB WALK
TO EXPRESS BUS
 DK_NHB_AEB = -0.0048       ; PARKING COST COEFFICIENT FOR NHB
AUTO TO EXPRESS BUS

 EK_NHB_DA  = -0.0048       ; HIGHWAY COST COEFFICIENT FOR NHB DA
 EK_NHB_SR2 = -0.0048       ; HIGHWAY COST COEFFICIENT FOR NHB
SR2
 EK_NHB_SR3 = -0.0048       ; HIGHWAY COST COEFFICIENT FOR NHB
SR3+
 EK_NHB_WLB = -0.0048       ; FARE COEFFICIENT FOR NHB WALK TO
LOCAL BUS
 EK_NHB_WEB = -0.0048       ; FARE COEFFICIENT FOR NHB WALK TO
EXPRESS BUS
 EK_NHB_AEB = -0.0048       ; FARE COEFFICIENT FOR NHB AUTO TO
EXPRESS BUS
 FK_NHB_XLB = -0.045        ; TRANSFER COEFF, NHB

  MW[1]=MI.8.8     ; AUTO TERMINAL TIME
  MW[2]=MI.8.7     ; TOLL TIME
  MW[3]=MI.8.3     ; AUTO IVTT
  MW[4]=MI.8.2     ; AUTO DISTANCE
  MW[5]=MI.8.7+MI.8.3   ; TOTAL AUTO TIME

  MW[6]= MI.1.1+MI.1.2                          ; WALK ACCESS TIME
, WALK TO LOCAL
  MW[7]= MI.1.10                                ; FIRST WAIT for
walk to local
  MW[8]= MI.1.11                                ; XFER WAIT FOR
WALK TO LOCAL
  MW[9]= MI.1.4                                 ; IVTT, WALK TO
LOCAL
  MW[10]= MI.1.13                               ; WALK TO LOCAL
FARE

  MW[16]= MI.2.1+MI.2.2                         ; WALK ACCESS TIME
, WALK TO EXPRESS BUS
  MW[17]= MI.2.10                               ; FIRST WAIT FOR
WALK TO EXPRESS
  MW[18]= MI.2.11                               ; XFER WAIT FOR
WALK TO EXPRESS
  MW[19]= MI.2.4                                ; IVTT, WALK TO
EXPRESS
  MW[20]= MI.2.13                               ; WALK TO EXPRESS
FARE

  MW[26]= MI.5.1+MI.5.2                         ; AUTO ACCESS TIME
, AUTO TO EXPRESS BUS
  MW[27]= MI.5.10                               ; FIRST WAIT FOR
AUTO TO EXPRESS
```

```
  MW[28]= MI.5.11                                ; XFER WAIT FOR
AUTO TO EXPRESS
  MW[29]= MI.5.4                                 ; IVTT, AUTO TO
EXPRESS
  MW[30]= MI.5.13                                ; AUTO TO EXPRESS
FARE


MW[41]=AK_NHB_DA*0.0+BK_NHB_DA*MW[1]+CK_NHB_DA*MW[5]+DK_NHB_DA*ZI
.1.PARK[J]+EK_NHB_DA*AUTOCOST*MW[4]+K_NHB_DA+K_NHB_DA_0C
; DA UTILITY, 0 CAR HH

MW[42]=AK_NHB_DA*0.0+BK_NHB_DA*MW[1]+CK_NHB_DA*MW[5]+DK_NHB_DA*ZI
.1.PARK[J]+EK_NHB_DA*AUTOCOST*MW[4]+K_NHB_DA+K_NHB_DA_1C
; DA UTILITY, 1 CAR HH

MW[43]=AK_NHB_DA*0.0+BK_NHB_DA*MW[1]+CK_NHB_DA*MW[5]+DK_NHB_DA*ZI
.1.PARK[J]+EK_NHB_DA*AUTOCOST*MW[4]+K_NHB_DA+K_NHB_DA_2C
; DA UTILITY, 2+ CAR HH


MW[51]=AK_NHB_SR2*0.0+BK_NHB_SR2*MW[1]+CK_NHB_SR2*MW[5]+DK_NHB_SR
2*ZI.1.PARK[J]+EK_NHB_SR2*AUTOCOST*MW[4]+K_NHB_SR2+K_NHB_SR2_0C
; SR2 UTILITY, 0 CAR

MW[52]=AK_NHB_SR2*0.0+BK_NHB_SR2*MW[1]+CK_NHB_SR2*MW[5]+DK_NHB_SR
2*ZI.1.PARK[J]+EK_NHB_SR2*AUTOCOST*MW[4]+K_NHB_SR2+K_NHB_SR2_1C
; SR2 UTILITY, 1 CAR

MW[53]=AK_NHB_SR2*0.0+BK_NHB_SR2*MW[1]+CK_NHB_SR2*MW[5]+DK_NHB_SR
2*ZI.1.PARK[J]+EK_NHB_SR2*AUTOCOST*MW[4]+K_NHB_SR2+K_NHB_SR2_2C
; SR2 UTILITY, 2 CAR


MW[61]=AK_NHB_SR3*0.0+BK_NHB_SR3*MW[1]+CK_NHB_SR3*MW[5]+DK_NHB_SR
3*ZI.1.PARK[J]+EK_NHB_SR3*AUTOCOST*MW[4]+K_NHB_SR3+K_NHB_SR3_0C
; SR2 UTILITY, 0 CAR

MW[62]=AK_NHB_SR3*0.0+BK_NHB_SR3*MW[1]+CK_NHB_SR3*MW[5]+DK_NHB_SR
3*ZI.1.PARK[J]+EK_NHB_SR3*AUTOCOST*MW[4]+K_NHB_SR3+K_NHB_SR3_1C
; SR3 UTILITY, 1 CAR

MW[63]=AK_NHB_SR3*0.0+BK_NHB_SR3*MW[1]+CK_NHB_SR3*MW[5]+DK_NHB_SR
3*ZI.1.PARK[J]+EK_NHB_SR3*AUTOCOST*MW[4]+K_NHB_SR3+K_NHB_SR3_2C
; SR3 UTILITY, 2 CAR

  JLOOP INCLUDE = 1-731
    IF (MW[7][J]>0 && MW[7][J]<7.0 )             ; IWAIT <7.0
MINUTE

MW[71]=AK_NHB_WLB*MW[6]+BK_NHB_WLB1*MW[7]+CK_NHB_WLB*MW[9]+DK_NHB
_WLB*ZI.1.PARK[J]+EK_NHB_WLB*MW[10]*100+FK_NHB_XLB*MW[8]+K_NHB_WL
B+K_NHB_WLB_0C         ; WALK TO LOCAL BUS UTILITY, 0 CAR

MW[72]=AK_NHB_WLB*MW[6]+BK_NHB_WLB1*MW[7]+CK_NHB_WLB*MW[9]+DK_NHB
_WLB*ZI.1.PARK[J]+EK_NHB_WLB*MW[10]*100+FK_NHB_XLB*MW[8]+K_NHB_WL
B+K_NHB_WLB_1C         ; WALK TO LOCAL BUS UTILITY, 1 CAR
```

```
MW[73]=AK_NHB_WLB*MW[6]+BK_NHB_WLB1*MW[7]+CK_NHB_WLB*MW[9]+DK_NHB
_WLB*ZI.1.PARK[J]+EK_NHB_WLB*MW[10]*100+FK_NHB_XLB*MW[8]+K_NHB_WL
B+K_NHB_WLB_2C          ; WALK TO LOCAL BUS UTILITY, 2+ CAR
    ELSE IF (MW[7][J]>=7.0)                       ; IWAIT > 7.0
MINUTE

MW[71]=AK_NHB_WLB*MW[6]+BK_NHB_WLB2*MW[7]+CK_NHB_WLB*MW[9]+DK_NHB
_WLB*ZI.1.PARK[J]+EK_NHB_WLB*MW[10]*100+FK_NHB_XLB*MW[8]+K_NHB_WL
B+K_NHB_WLB_0C          ; WALK TO LOCAL BUS UTILITY, 0 CAR

MW[72]=AK_NHB_WLB*MW[6]+BK_NHB_WLB2*MW[7]+CK_NHB_WLB*MW[9]+DK_NHB
_WLB*ZI.1.PARK[J]+EK_NHB_WLB*MW[10]*100+FK_NHB_XLB*MW[8]+K_NHB_WL
B+K_NHB_WLB_1C          ; WALK TO LOCAL BUS UTILITY, 1 CAR

MW[73]=AK_NHB_WLB*MW[6]+BK_NHB_WLB2*MW[7]+CK_NHB_WLB*MW[9]+DK_NHB
_WLB*ZI.1.PARK[J]+EK_NHB_WLB*MW[10]*100+FK_NHB_XLB*MW[8]+K_NHB_WL
B+K_NHB_WLB_2C          ; WALK TO LOCAL BUS UTILITY, 2+ CAR
    ENDIF
  ENDJLOOP

  JLOOP INCLUDE = 1-731
    IF (MW[17][J]>0 && MW[17][J]<7.0 )            ; IWAIT <7.0
MINUTE

MW[81]=AK_NHB_WEB*MW[16]+BK_NHB_WEB1*MW[17]+CK_NHB_WEB*MW[19]+DK_
NHB_WEB*ZI.1.PARK[J]+EK_NHB_WEB*MW[20]*100+FK_NHB_XLB*MW[18]+K_NH
B_WEB+K_NHB_WEB_0C          ; WALK TO EXPRESS BUS UTILITY, 0 CAR

MW[82]=AK_NHB_WEB*MW[16]+BK_NHB_WEB1*MW[17]+CK_NHB_WEB*MW[19]+DK_
NHB_WEB*ZI.1.PARK[J]+EK_NHB_WEB*MW[20]*100+FK_NHB_XLB*MW[18]+K_NH
B_WEB+K_NHB_WEB_1C          ; WALK TO EXPRESS BUS UTILITY, 1 CAR

MW[83]=AK_NHB_WEB*MW[16]+BK_NHB_WEB1*MW[17]+CK_NHB_WEB*MW[19]+DK_
NHB_WEB*ZI.1.PARK[J]+EK_NHB_WEB*MW[20]*100+FK_NHB_XLB*MW[18]+K_NH
B_WEB+K_NHB_WEB_2C          ; WALK TO EXPRESS BUS UTILITY, 2+ CAR
    ELSE IF (MW[17][J]>=7.0)                      ; IWAIT > 7.0
MINUTE

MW[81]=AK_NHB_WEB*MW[16]+BK_NHB_WEB2*MW[17]+CK_NHB_WEB*MW[19]+DK_
NHB_WEB*ZI.1.PARK[J]+EK_NHB_WEB*MW[20]*100+FK_NHB_XLB*MW[18]+K_NH
B_WEB+K_NHB_WEB_0C          ; WALK TO EXPRESS BUS UTILITY, 0 CAR

MW[82]=AK_NHB_WEB*MW[16]+BK_NHB_WEB2*MW[17]+CK_NHB_WEB*MW[19]+DK_
NHB_WEB*ZI.1.PARK[J]+EK_NHB_WEB*MW[20]*100+FK_NHB_XLB*MW[18]+K_NH
B_WEB+K_NHB_WEB_1C          ; WALK TO EXPRESS BUS UTILITY, 1 CAR

MW[83]=AK_NHB_WEB*MW[16]+BK_NHB_WEB2*MW[17]+CK_NHB_WEB*MW[19]+DK_
NHB_WEB*ZI.1.PARK[J]+EK_NHB_WEB*MW[20]*100+FK_NHB_XLB*MW[18]+K_NH
B_WEB+K_NHB_WEB_2C          ; WALK TO EXPRESS BUS UTILITY, 2+ CAR
    ENDIF
  ENDJLOOP

  JLOOP INCLUDE = 1-731
    IF (MW[27][J]>0 && MW[27][J]<7.0 )            ; IWAIT <7.0
MINUTE
```

```
MW[91]=AK_NHB_AEB*MW[26]+BK_NHB_AEB1*MW[27]+CK_NHB_AEB*MW[29]+DK_
NHB_AEB*ZI.1.PARK[J]+EK_NHB_AEB*MW[30]*100+FK_NHB_XLB*MW[28]+K_NH
B_AEB+K_NHB_AEB_0C         ; AUTO TO EXPRESS BUS UTILITY, 0 CAR

MW[92]=AK_NHB_AEB*MW[26]+BK_NHB_AEB1*MW[27]+CK_NHB_AEB*MW[29]+DK_
NHB_AEB*ZI.1.PARK[J]+EK_NHB_AEB*MW[30]*100+FK_NHB_XLB*MW[28]+K_NH
B_AEB+K_NHB_AEB_1C         ; AUTO TO EXPRESS BUS UTILITY, 1 CAR

MW[93]=AK_NHB_AEB*MW[26]+BK_NHB_AEB1*MW[27]+CK_NHB_AEB*MW[29]+DK_
NHB_AEB*ZI.1.PARK[J]+EK_NHB_AEB*MW[30]*100+FK_NHB_XLB*MW[28]+K_NH
B_AEB+K_NHB_AEB_2C         ; AUTO TO EXPRESS BUS UTILITY, 2+ CAR
    ELSE IF (MW[27][J]>=7.0)                      ; IWAIT > 7.0
MINUTE

MW[91]=AK_NHB_AEB*MW[26]+BK_NHB_AEB2*MW[27]+CK_NHB_AEB*MW[29]+DK_
NHB_AEB*ZI.1.PARK[J]+EK_NHB_AEB*MW[30]*100+FK_NHB_XLB*MW[28]+K_NH
B_AEB+K_NHB_AEB_0C         ; AUTO TO EXPRESS BUS UTILITY, 0 CAR

MW[92]=AK_NHB_AEB*MW[26]+BK_NHB_AEB2*MW[27]+CK_NHB_AEB*MW[29]+DK_
NHB_AEB*ZI.1.PARK[J]+EK_NHB_AEB*MW[30]*100+FK_NHB_XLB*MW[28]+K_NH
B_AEB+K_NHB_AEB_1C         ; AUTO TO EXPRESS BUS UTILITY, 1 CAR

MW[93]=AK_NHB_AEB*MW[26]+BK_NHB_AEB2*MW[27]+CK_NHB_AEB*MW[29]+DK_
NHB_AEB*ZI.1.PARK[J]+EK_NHB_AEB*MW[30]*100+FK_NHB_XLB*MW[28]+K_NH
B_AEB+K_NHB_AEB_2C         ; AUTO TO EXPRESS BUS UTILITY, 2+ CAR
    ENDIF
  ENDJLOOP

; DO EXPONENTIAL OF UTILITIES

MW[101] = EXP(MW[41])*100000   ; DA 0 CAR
MW[102] = EXP(MW[42])*100000   ; DA 1 CAR
MW[103] = EXP(MW[43])*100000   ; DA 2 CAR

MW[111] = EXP(MW[51])*100000   ; SR2 0 CAR
MW[112] = EXP(MW[52])*100000   ; SR2 1 CAR
MW[113] = EXP(MW[53])*100000   ; SR2 2 CAR

MW[121] = EXP(MW[61])*100000   ; SR3 0 CAR
MW[122] = EXP(MW[62])*100000   ; SR3 1 CAR
MW[123] = EXP(MW[63])*100000   ; SR3 2 CAR

JLOOP INCLUDE = 1-731
  IF (MW[9][J]>0)                  ; WALK TO LOCAL IVTT >0.0
   MW[131] = EXP(MW[71])*100000   ; WALK TO LOCAL BUS 0 CAR
   MW[132] = EXP(MW[72])*100000   ; WALK TO LOCAL BUS 1 CAR
   MW[133] = EXP(MW[73])*100000   ; WALK TO LOCAL BUS 2+ CAR
  ENDIF
ENDJLOOP

JLOOP INCLUDE = 1-731
  IF (MW[19][J]>0)                 ; WALK TO PREMIUM IVTT >0.0
    MW[141] = EXP(MW[81])*100000   ; WALK TO express BUS 0 CAR
    MW[142] = EXP(MW[82])*100000   ; WALK TO EXPRESS BUS 1 CAR
    MW[143] = EXP(MW[83])*100000   ; WALK TO EXPRESS BUS 2+ CAR
  ENDIF
```

```
ENDJLOOP

JLOOP INCLUDE = 1-731
  IF (MW[29][J]>0)                  ; AUTO TO PREMIUM IVTT >0.0
    MW[151] = EXP(MW[91])*100000   ; Auto TO express BUS 0 CAR
    MW[152] = EXP(MW[92])*100000   ; Auto TO EXPRESS BUS 1 CAR
    MW[153] = EXP(MW[93])*100000   ; Auto TO EXPRESS BUS 2+ CAR
  ENDIF
ENDJLOOP

; SHARE FOR ZERO CAR HH
MW[171] =
100000*MW[101]/(MW[101]+MW[111]+MW[121]+MW[131]+MW[141]+MW[151])
; DA SHARE for 0 Car
MW[172] =
100000*MW[111]/(MW[101]+MW[111]+MW[121]+MW[131]+MW[141]+MW[151])
; SR2 SHARE for 0 Car
MW[173] =
100000*MW[121]/(MW[101]+MW[111]+MW[121]+MW[131]+MW[141]+MW[151])
; SR3 SHARE for 0 Car
MW[174] =
100000*MW[131]/(MW[101]+MW[111]+MW[121]+MW[131]+MW[141]+MW[151])
; Walk to Local Share for 0 Car
MW[175] =
100000*MW[141]/(MW[101]+MW[111]+MW[121]+MW[131]+MW[141]+MW[151])
; Walk to premium Share for 0 Car
MW[176] =
100000*MW[151]/(MW[101]+MW[111]+MW[121]+MW[131]+MW[141]+MW[151])
; Auto to premium Share for 0 Car

; SHARE FOR ONE CAR HH
MW[181] =
100000*MW[102]/(MW[102]+MW[112]+MW[122]+MW[132]+MW[142]+MW[152])
; DA SHARE for 1 Car
MW[182] =
100000*MW[112]/(MW[102]+MW[112]+MW[122]+MW[132]+MW[142]+MW[152])
; SR2 SHARE for 1 Car
MW[183] =
100000*MW[122]/(MW[102]+MW[112]+MW[122]+MW[132]+MW[142]+MW[152])
; SR3 SHARE for 1 Car
MW[184] =
100000*MW[132]/(MW[102]+MW[112]+MW[122]+MW[132]+MW[142]+MW[152])
; Walk to Local Share for 1 Car
MW[185] =
100000*MW[142]/(MW[102]+MW[112]+MW[122]+MW[132]+MW[142]+MW[152])
; Walk to premium Share for 1 Car
MW[186] =
100000*MW[152]/(MW[102]+MW[112]+MW[122]+MW[132]+MW[142]+MW[152])
; Auto to premium Share for 1 Car

; SHARE FOR TWO+ CAR HH
MW[191] =
100000*MW[103]/(MW[103]+MW[113]+MW[123]+MW[133]+MW[143]+MW[153])
; DA SHARE for 2+ Car
MW[192] =
100000*MW[113]/(MW[103]+MW[113]+MW[123]+MW[133]+MW[143]+MW[153])
; SR2 SHARE for 2+ Car
```

```
MW[193] =
100000*MW[123]/(MW[103]+MW[113]+MW[123]+MW[133]+MW[143]+MW[153])
; SR3 SHARE for 2+ Car
MW[194] =
100000*MW[133]/(MW[103]+MW[113]+MW[123]+MW[133]+MW[143]+MW[153])
; Walk to Local Share for 2+ Car
MW[195] =
100000*MW[143]/(MW[103]+MW[113]+MW[123]+MW[133]+MW[143]+MW[153])
; Walk to premium Share for 2+ Car
MW[196] =
100000*MW[153]/(MW[103]+MW[113]+MW[123]+MW[133]+MW[143]+MW[153])
; Auto to premium Share for 2+ Car

MW[197]=(MI.9.6+MI.9.7)*ZI.2.CAR0
MW[198]=(MI.9.6+MI.9.7)*ZI.2.CAR1
MW[199]=MI.9.6+MI.9.7-MW[197]-MW[198]
MW[200]=MW[197]+MW[198]

MW[134] = MW[171]*MW[197]          ;DA TRIPS FOR ZERO CAR
MW[135] = MW[172]*MW[197]          ;SR2 TRIPS FOR ZERO CAR
MW[136] = MW[173]*MW[197]          ;SR3+ TRIPS FOR ZERO CAR
MW[137] = MW[174]*MW[197]          ;WALK TO LOCAL TRIPS FOR ZERO
CAR
MW[138] = MW[175]*MW[197]          ;WALK_PREM TRIPS FOR ZERO CAR
MW[139] = MW[176]*MW[197]          ;DRV_PREM TRIPS FOR ZERO CAR

MW[144] = MW[181]*MW[198]          ;DA TRIPS FOR ONE CAR
MW[145] = MW[182]*MW[198]          ;SR2 TRIPS FOR ONE CAR
MW[146] = MW[183]*MW[198]          ;SR3+ TRIPS FOR ONE CAR
MW[147] = MW[184]*MW[198]          ;WALK TO LOCAL TRIPS FOR ONE
CAR
MW[148] = MW[185]*MW[198]          ;WALK_PREM TRIPS FOR ONE CAR
MW[149] = MW[186]*MW[198]         ;DRV_PREM TRIPS FOR ONE CAR

MW[154] = MW[191]*MW[199]        ;DA TRIPS FOR TWO+ CAR
MW[155] = MW[192]*MW[199]        ;SR2 TRIPS FOR TWO+ CAR
MW[156] = MW[193]*MW[199]        ;SR3+ TRIPS FOR TWO+ CAR
MW[157] = MW[194]*MW[199]        ;WALK TO LOCAL TRIPS FOR TWO+ CAR
MW[158] = MW[195]*MW[199]        ;WALK_PREM TRIPS FOR TWO+ CAR
MW[159] = MW[196]*MW[199]        ;DRV_PREM TRIPS FOR TWO+ CAR

; FINAL TRIPS

MW[161]=(MW[134]+MW[144]+MW[154])/100000              ; DA TRIPS
MW[162]=(MW[135]+MW[145]+MW[155])/100000              ; SR2 TRIPS
MW[163]=(MW[136]+MW[146]+MW[156])/100000              ; SR3+
TRIPS
MW[164]=(MW[137]+MW[147]+MW[157])/100000              ; WALK TO
LOCAL BUS TRIPS
MW[165]=(MW[138]+MW[148]+MW[158])/100000              ; WALK TO
PREMIUM BUS TRIPS
MW[166]=(MW[139]+MW[149]+MW[159])/100000              ; DRIVE TO
PREMIUM BUS TRIPS

MW[201]=(MW[134]+MW[144])/100000
MW[202]=(MW[135]+MW[145])/100000
MW[203]=(MW[136]+MW[146])/100000
```

```
MW[204]=(MW[137]+MW[147])/100000
MW[205]=(MW[138]+MW[148])/100000
MW[206]=(MW[139]+MW[149])/100000

MW[211]=MW[154]/100000
MW[212]=MW[155]/100000
MW[213]=MW[156]/100000
MW[214]=MW[157]/100000
MW[215]=MW[158]/100000
MW[216]=MW[159]/100000

ENDRUN
```

4. 01MAT00H.S

```
; Do not change filenames or add or remove FILEI/FILEO statements
using an editor. Use Cube/Application Manager.
RUN PGM=MATRIX
PRNFILE="D:\FSUTMS\D4\TCRPM\CUBE\PREPARE_TABLE.PRN"
FILEO MATO[2] = "D:\FSUTMS\D4\TCRPM\CUBE\TRANSITTTAB.MAT",
  MO=11-16, NAME=AM_WLB,AM_WPM,AM_APM, MD_WLB, MD_WPM, MD_APM
FILEO MATO[1] = "D:\FSUTMS\D4\TCRPM\CUBE\HWYTTAB.MAT",
MO=1-9,
  NAME=TOTALTRIPS, DA, SR2, SR3, AIRPORT, TRUCK4TIRE,
TRKSNGLUNIT, TRACTORTRAILER, PUBLICSCHOOL,  DEC=3*S
FILEI MATI[6] = "D:\FSUTMS\D4\TCRPM\CUBE\HBSC_MOT_NOTMOT.MAT"
FILEI MATI[5] = "D:\FSUTMS\D4\TCRPM\CUBE\PSCHOOL.{ALT}{YEAR}.MAT"
FILEI MATI[4] = "{SCENARIO_DIR}\EETAB.{ALT}{YEAR}.MAT"
FILEI MATI[7] = "D:\FSUTMS\D4\TCRPM\CUBE\PTRIPS1.MAT"
FILEI MATI[3] = "D:\FSUTMS\D4\TCRPM\CUBE\MC_NHB.MAT"
FILEI MATI[2] = "D:\FSUTMS\D4\TCRPM\CUBE\MC_HBNW.MAT"
FILEI MATI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\MC_HBW.MAT"
MSG='BUILD OD VEHICLE TRIPS & PA TRANSIT TRIPS'
; DRIVE ALONE
MW[2]=((MI.1.1+MI.1.1.T+MI.2.1+MI.2.1.T+MI.3.1+MI.3.1.T)+MI.4.1+M
I.4.1.T+MI.6.3+MI.6.3.T)*0.5
; SR2
MW[3]=((MI.1.2+MI.1.2.T+MI.2.2+MI.2.2.T+MI.3.2+MI.3.2.T)+MI.6.4+M
I.6.4.T)*0.5/2.0
;SR3+
MW[4]=((MI.1.3+MI.1.3.T+MI.2.3+MI.2.3.T+MI.3.3+MI.3.3.T)+MI.6.5+M
I.6.5.T)*0.5/3.2
MW[5] = MI.7.8
MW[6] = MI.7.9
MW[7] = MI.7.10
MW[8] = MI.7.11
MW[9] = (MI.5.1+MI.5.1.T)*{AOFAC4}*0.5
MW[1] = MW[2]+MW[3]+MW[4]+MW[5]+MW[6]+MW[7]+MW[8]+MW[9]
; AM PEAK TRANSIT
MW[11]=MI.1.4                  ; WALK TO LOCAL
MW[12]=MI.1.5                  ; WALK TO PREMIUM
MW[13]=MI.1.6                  ; AUTO TO PREMIUM

; MD OFF-PEAK TRANSIT
MW[14]=MI.2.4+MI.3.4           ; WALK TO LOCAL
MW[15]=MI.2.5+MI.3.5           ; WALK TO PREMIUM
```

```
MW[16]=MI.2.6+MI.3.6          ; AUTO TO PREMIUM

ENDRUN
```

ASSIGNMENT

1. ASHWY00A.S

```
; Do not change filenames or add or remove FILEI/FILEO statements
using an editor. Use Cube/Application Manager.
RUN PGM=HIGHWAY PRNFILE="D:\FSUTMS\D4\TCRPM\CUBE\ASHWY00A.PRN"
MSG='Final Highway Assignment'
FILEO PATHO[1] = "D:\FSUTMS\D4\TCRPM\CUBE\LOADED.PTH"
FILEO PRINTO[1] = "D:\FSUTMS\D4\TCRPM\CUBE\TOLL FACILITY.PRN"
FILEO NETO = "D:\FSUTMS\D4\TCRPM\CUBE\LOADED.NET"
FILEI MATI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\HWYTTAB.MAT"
FILEI NETI = "D:\FSUTMS\D4\TCRPM\CUBE\UNLOADED BASE.NET"
FILEI TURNPENI = "D:\FSUTMS\D4\TCRPM\CUBE\TCARDS.PRN"

PAR ZONEMSG=100 MAXITERS={iter} GAP={EPSILON2}

ARRAY TOLLREVENUE=99 TOLLVMT=99
TURNS N=1-99999
ZONEMSG=100
CTOLL={CTOLL}

; look up deceleration rate based on approach speed
LOOKUP,
 INTERPOLATE=Y, LIST=Y, NAME=DECEL,
        LOOKUP[1]=1,RESULT=2,
R = '30 4',
    '70 6.2'

PROCESS PHASE=LINKREAD
; basics
T0=li.TIME
      LW.FFTIME=LI.TIME
      T0=LI.TIME
      C=(LI.CAP_HR/li.confac)*li.uroadfactor
;      IF (li.facility_type=80-89) ADDTOGROUP=1
      IF (LI.TMODE=8|LI.TMODE=5|LI.BPREXPONENT=0) ADDTOGROUP=9
; classify links based on presence/absence of tolls
      LINKCLASS=1 ; no toll
      if (li.cartoll>0)
       LINKCLASS=2 ; with toll
       LW.TOLLTIME=({CTOLL}*LI.CARTOLL)*60
       T0=({CTOLL}*LI.CARTOLL)*60 + LI.SVCMINUTES +
LI.SVCSECONDS/60
       if (iteration=0)
        LW.ARRIVR=(V/LI.UROADFACTOR)*LI.CONFAC/LI.PLZALNSMAX  ;
hourly volume per toll lane ie. arrival rate in vehicles per hour
        LW.SERVT=LI.SVCMINUTES+(LI.SVCSECONDS/60)           ;
Plaza lane service time in minutes per vehicle
        IF (LI.TOLLTYPE=1)
           LW.SERVR=(1/LW.SERVT)*60                          ;
Plaza lane service rate in vehicle per hour
        ELSE
           LW.SERVR=100
        ENDIF
```

```
        PRINT LIST='ARRIVR= ',LW.ARRIVR
        PRINT LIST='SERVT= ',LW.SERVT
        PRINT LIST='SERVR= ',LW.SERVR
        if (LW.ARRIVR>=LW.SERVR) LW.ARRIVR=0.99*LW.SERVR    ;
prevent infinite or negative queue
       endif

      endif

      if (li.TOLL_ACC>0)
       LINKCLASS=3 ; Toll Plaza Acceleration link
       T0= T0 + (LI.FREEFLOWSPEED/{ACCELRATE})/60
      endif

      if (li.TOLL_DEC>0)
       LINKCLASS=4 ; Toll Plaza Deceleration link
       T0 = T0 + (LI.FREEFLOWSPEED/DECEL(1,LI.FREEFLOWSPEED))/60
      endif
if (li.toll>maxplzno) maxplzno=li.toll
      IF (LI.FACILITY_TYPE=49) LINKCLASS=5
ENDPHASE

;=================================================================
=========================
;=== ILOOP (ASSIGNMENT) PHASE ===
;=================================================================
=========================
PHASE=ILOOP
  PATHLOAD PATH=COST, VOL[1]=MI.1.1,PENI=1,EXCLUDEGROUP=9,
patho=1,name='Assignment',allj=f,includecosts=f
ENDPHASE

;=================================================================
=========================
;=== ADJUST PHASE (WITH REPORTING)===
;=================================================================
=========================
; User Specified Functions
; Delay functions, one for congested travel time, the other for
toll links convert to time equivalent
PHASE=ADJUST
 if (time>0) LW.CGSTSPEED=(LI.DISTANCE/TIME)*60
 if (LW.ARRIVR>=LW.SERVR) LW.ARRIVR=0.99*LW.SERVR    ; prevent
infinite or negative queue
 if (li.cartoll>0)
   LW.ARRIVR=(V/LI.UROADFACTOR)*LI.CONFAC/LI.PLZALNSMAX  ; hourly
volume per toll lane ie. arrival rate in vehicles per hour
   LW.SERVT=LI.SVCMINUTES+(LI.SVCSECONDS/60)           ; Plaza
lane service time in minutes per vehicle
   IF (LI.TOLLTYPE=1)
          LW.SERVR=(1/LW.SERVT)*60        ; Plaza lane service
rate in vehicle per hour
          if (LW.ARRIVR>=LW.SERVR) LW.ARRIVR=0.99*LW.SERVR    ;
prevent infinite or negative queue
   ELSE
          LW.SERVR=100
          LW.ARRIVR=1
```

```
   ENDIF
print list="****",LI.PLAZADESC,"****"
PRINT LIST='ARRIVR= ',LW.ARRIVR
PRINT LIST='SERVT= ',LW.SERVT
PRINT LIST='SERVR= ',LW.SERVR
PRINT LIST='TOLLTIME= ',TIME
endif
if (lw.servr-lw.arrivr=0.0) lw.servr=lw.servr+0.01
 FUNCTION
TC[1]=LI.TIME*(1+LI.BPRCOEFFICIENT*(MIN(V/C,{VCMAX}))^LI.BPREXPON
ENT) ; congested time for non-toll links
 FUNCTION   TC[2]=(1/(LW.SERVR-LW.ARRIVR))*60 +
{CTOLL}*LI.CARTOLL*60 ; congested time for toll links
 FUNCTION
TC[3]=LI.TIME*(1+LI.BPRCOEFFICIENT*(MIN(V/C,{VCMAX}))^LI.BPREXPON
ENT) + (LW.CGSTSPEED/{ACCELRATE})/60; congested time toll
acceleration links
 FUNCTION
TC[4]=LI.TIME*(1+LI.BPRCOEFFICIENT*(MIN(V/C,{VCMAX}))^LI.BPREXPON
ENT) + (LW.CGSTSPEED/DECEL(1,LW.CGSTSPEED))/60; congested time
toll deceleration links
 FUNCTION   TC[5]=LI.TIME; congested time toll deceleration links
IF (LI.FACILITY_TYPE=90-99)
TOLLVMT[ITERATION]=TOLLVMT[ITERATION]+(LI.DISTANCE*V)
if (linkno=1)
   print,
        printo=1,
list=" "
   print,
        printo=1,
list=" "
   print,
        printo=1,
list="                        ITERATION ",ITERATION(2.0),"
SUMMARY REPORT ON TOLL LINKS -- CTOLL VALUE =",ctoll(8.7)
   print,
        printo=1,
list=" "
   print,
        printo=1,
list="                                                          No.
ServTim  % Heavy            Delay   Toll     Plan."
   print,
        printo=1,
list="  Class  Type  ..Nodes..    ......... Name .........  Lanes
Toll   (mm:ss)   Trucks   Volume   (Min)   (Min)     Rev.
Util."
   print,
        printo=1,
list=" "
endif
IF (li.toll>0) print,
        printo=1,
list=li.toll(7.0),li.tolltype(5.0),li.a(6.0),li.b(6.0),"
",li.plazadesc(25),li.plzalnsmin(7.0),"
$",LI.CARTOLL(5.2),,li.svcminutes(6.0),":",li.svcseconds(2.0),li.
```

```
pcttrucks*100(9.2),v(10.0c),time-
LW.TOLLTIME(8.2),LW.TOLLTIME(8.2C)," $",LI.CARTOLL*V(7.0C)
IF (LI.TOLL>0)
TOLLREVENUE[ITERATION]=TOLLREVENUE[ITERATION]+(Li.CARTOLL*V)
IF (LINKNO=NUMLINKS) PRINT,
        printo=1,
LIST="     *----------------------
********************************** TOTAL TOLL REVENUE FOR THIS
ITERATION IS    $",TOLLREVENUE[ITERATION](10.0C)
IF (LINKNO=NUMLINKS) PRINT,
        printo=1,
LIST="     *-TOLL REVENUE SUMMARY-
*********************************  TOLL FACILITY VMT FOR THIS
ITERATION IS     ",TOLLVMT[ITERATION](10.0C)
IF (LINKNO=NUMLINKS) PRINT,
        printo=1,
LIST="     *-----------------------****************************
AVERAGE REVENUE PER VMT FOR THIS ITERATION IS
$",TOLLREVENUE[ITERATION]/TOLLVMT[ITERATION](6.5C)

ENDRUN
```

2. ASPTR00A.S

```
; Do not change filenames or add or remove FILEI/FILEO statements
using an editor. Use Cube/Application Manager.
RUN PGM=PUBLIC TRANSPORT
PRNFILE="D:\FSUTMS\D4\TCRPM\CUBE\ASPTR00A.PRN" MSG='Peak Load
Walk to Transit'
FILEI MATI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\TRANSITTTAB.MAT"
FILEO REPORTO = "D:\FSUTMS\D4\TCRPM\CUBE\ASPTR00B.PRN"
FILEO NETO = "D:\FSUTMS\D4\TCRPM\CUBE\ASPTR00A.NET"
FILEI ROUTEI[2] = "D:\FSUTMS\D4\TCRPM\CUBE\TROUTEAM2.RTE"
FILEI ROUTEI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\TROUTEAM1.RTE"
FILEI NETI = "D:\FSUTMS\D4\TCRPM\CUBE\TNETAMWALK.NET"
FILEI TURNPENI = "D:\FSUTMS\D4\TCRPM\CUBE\TCARDS.PRN"

PARAMETERS NOROUTEMSGS=0, NOROUTEERRS=999999,
    TRIPSIJ[1]=MI.1.1,
    TRIPSIJ[2]=MI.1.2,
    TRIPSIJ[3]=0,
    TRIPSIJ[4]=0

REPORT LINES=T
ENDRUN
```

3. ASPTR00B.S

```
; Do not change filenames or add or remove FILEI/FILEO statements
using an editor. Use Cube/Application Manager.
RUN PGM=PUBLIC TRANSPORT
PRNFILE="D:\FSUTMS\D4\TCRPM\CUBE\ASPTR00C.PRN" MSG='Peak Load
Auto to Transit'
FILEI MATI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\TRANSITTTAB.MAT"
FILEO REPORTO = "D:\FSUTMS\D4\TCRPM\CUBE\ASPTR00D.PRN"
```

```
FILEO NETO = "D:\FSUTMS\D4\TCRPM\CUBE\ASPTR00A.NET"
FILEI ROUTEI[3] = "D:\FSUTMS\D4\TCRPM\CUBE\TROUTEAM7.RTE"
FILEI NETI = "D:\FSUTMS\D4\TCRPM\CUBE\TNETAMAUTO.NET"
FILEI TURNPENI = "D:\FSUTMS\D4\TCRPM\CUBE\TCARDS.PRN"
PARAMETERS NOROUTEMSGS=0, NOROUTEERRS=999999,
 TRIPSIJ[1]=MI.1.3,
 TRIPSIJ[2]=0,
 TRIPSIJ[3]=0,
 TRIPSIJ[4]=0

REPORT LINES=T
ENDRUN
```

4. ASPTR00C.S

```
; Do not change filenames or add or remove FILEI/FILEO statements
using an editor. Use Cube/Application Manager.
RUN PGM=PUBLIC TRANSPORT
PRNFILE="D:\FSUTMS\D4\TCRPM\CUBE\ASPTR00E.PRN" MSG='Off-peak Load
Walk to Transit'
FILEI MATI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\TRANSITTTAB.MAT"
FILEO REPORTO = "D:\FSUTMS\D4\TCRPM\CUBE\ASPTR00F.PRN"
FILEO NETO = "D:\FSUTMS\D4\TCRPM\CUBE\ASPTR00B.NET"
FILEI ROUTEI[2] = "D:\FSUTMS\D4\TCRPM\CUBE\TROUTEMD2.RTE"
FILEI ROUTEI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\TROUTEMD1.RTE"
FILEI NETI = "D:\FSUTMS\D4\TCRPM\CUBE\TNETMDWALK.NET"

PARAMETERS NOROUTEMSGS=0, NOROUTEERRS=999999,
    TRIPSIJ[1]=MI.1.4,
    TRIPSIJ[2]=MI.1.5,
    TRIPSIJ[3]=0,
    TRIPSIJ[4]=0
REPORT LINES=T
ENDRUN
```

5. ASPTR00D.S

```
; Do not change filenames or add or remove FILEI/FILEO statements
using an editor. Use Cube/Application Manager.
RUN PGM=PUBLIC TRANSPORT
PRNFILE="D:\FSUTMS\D4\TCRPM\CUBE\ASPTR00G.PRN" MSG='Off-peak Load
Auto to Transit'
FILEI MATI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\TRANSITTTAB.MAT"
FILEO REPORTO = "D:\FSUTMS\D4\TCRPM\CUBE\ASPTR00H.PRN"
FILEO NETO = "D:\FSUTMS\D4\TCRPM\CUBE\ASPTR00C.NET"
FILEI ROUTEI[3] = "D:\FSUTMS\D4\TCRPM\CUBE\TROUTEMD7.RTE"
FILEI NETI = "D:\FSUTMS\D4\TCRPM\CUBE\TNETMDAUTO.NET"

PARAMETERS NOROUTEMSGS=0, NOROUTEERRS=999999,
   TRIPSIJ[1]=0,
   TRIPSIJ[2]=0,
   TRIPSIJ[3]=MI.1.6
REPORT LINES=T
ENDRUN
```

EVALUATION

(Region-wide and county specific evaluation scripts are similar so have not been repeated for each county)

1. EVNET00B.S

```
; Do not change filenames or add or remove FILEI/FILEO statements
using an editor. Use Cube/Application Manager.
RUN PGM=NETWORK PRNFILE="D:\FSUTMS\D4\TCRPM\CUBE\RMSE-OUT.PRN"
FILEO LINKO = "D:\FSUTMS\D4\TCRPM\CUBE\LINKSWITHCOUNTS.DBF"
FILEO NETO = "D:\FSUTMS\D4\TCRPM\CUBE\TEMP.NET"
FILEO PRINTO[1] = "D:\FSUTMS\D4\TCRPM\CUBE\IRRMSE.PRN"
FILEI LINKI[1] = "D:\FSUTMS\D4\TCRPM\CUBE\LOADED.NET"

; ==========================================================
; LINKMERGE PHASE
; ==========================================================
PHASE=LINKMERGE
if(location==2) delete
if(location==3) delete
; DUMMAY VARIABLES FOR HEVALDBF
ZONE        =1
USECODE     =1
;LOCATION    =1
LANDUSE     =1
CCODE       =1
TOLL        =1
;

   ft=int(FACILITY_TYPE/10)
   at=int(AREA_TYPE/10)
   LNS=LI.1.LANES
   TOTAL_VOL=ROUND(V_1)
   count=COUNT
; initialize arrays and variables
   ARRAY _err=13, _cns=13, _cnt=13, _RGP=13, _volbyft=100,
_cntbyft=100
   ARRAY _volbyat=100, _cntbyat=100, _lnkbyft=100, _lnkbyat=100
   _group=(0.0*FACILITY_TYPE)
   _RGP[1]=1, _RGP[2]=5000, _RGP[3]=10000, _RGP[4]=20000,
_RGP[5]=30000, _RGP[6]=40000, _RGP[7]=50000,
   _RGP[8]=60000, _RGP[9]=70000, _RGP[10]=80000, _RGP[11]=90000,
_RGP[12]=100000, _RGP[13]=500000
   IF (A=1)
     LOOP _iter=1,13
           _err[_iter]=0,_cnt[_iter]=0,_cns[_iter]=0
     ENDLOOP
     LOOP _iter=1,99
          _volbyft[_iter]=0, _cntbyft[_iter]=0,
_lnkbyft[_iter]=0
           _volbyat[_iter]=0, _cntbyat[_iter]=0,
_lnkbyat[_iter]=0
     ENDLOOP
   ENDIF
;   FNAME='ROADNAME'
```

```
; establish some English sounding variables for the output
network
   links=1
   lanemiles=lns*distance
; calculate and compartmentalize
   IF(COUNT>0) VOLCNT=TOTAL_VOL/COUNT, NETDIFF=TOTAL_VOL-COUNT,
ABSDIFF=ABS(NETDIFF), ERRORSQ=NETDIFF^2,
PCTDIFF=100*NETDIFF/COUNT _group=1
   IF (COUNT>5000) _group=2
   IF (COUNT>10000) _group=3
   IF (COUNT>20000) _group=4
   IF (COUNT>30000) _group=5
   IF (COUNT>40000) _group=6
   IF (COUNT>50000) _group=7
   IF (COUNT>60000) _group=8
   IF (COUNT>70000) _group=9
   IF (COUNT>80000) _group=10
   IF (COUNT>90000) _group=11
   IF (COUNT>100000) _group=12
   IF (COUNT>0)
   _ERR[_group]=ERRORSQ+_ERR[_group],
_CNS[_group]=COUNT+_CNS[_group], _CNT[_group]=_CNT[_group]+1
   _ERR[13]=ERRORSQ+_ERR[13], _CNS[13]=COUNT+_CNS[13],
_CNT[13]=_CNT[13]+1
   ENDIF
IF (COUNT>0)
   _volbyft[ft]=_volbyft[ft]+TOTAL_VOL
   _cntbyft[ft]=_cntbyft[ft]+count
   _lnkbyft[ft]=_lnkbyft[ft]+1
   _volbyat[at]=_volbyat[at]+TOTAL_VOL
   _cntbyat[at]=_cntbyat[at]+count
   _lnkbyat[at]=_lnkbyat[at]+1
   _volbyft[100]=_volbyft[100]+TOTAL_VOL
   _cntbyft[100]=_cntbyft[100]+count
   _lnkbyft[100]=_lnkbyft[100]+1
   _volbyat[100]=_volbyat[100]+TOTAL_VOL
   _cntbyat[100]=_cntbyat[100]+count
   _lnkbyat[100]=_lnkbyat[100]+1
endif
CROSSTAB  VAR= LINKS LANEMILES, form=12.0c,
    row=FT, range=10-99-10,
    col=LNS, range=1-5-1,6-9
IF (COUNT>0)
CROSSTAB  VAR= TOTAL_VOL,COUNT, form=12.0c,
    row=FT, range=10-99-10,
    col=AT, range=10-59-10
ENDIF
;if (count=0) delete

ENDPHASE

; ==========================================================
; SUMMARY REPORTING
; ==========================================================


PHASE=SUMMARY
```

```
; a little loop to write out the Percent Root Mean Square Error

LOOP _iter=1,12
if (_iter=1) _limit='45 - 55'
if (_iter=2) _limit='35 - 45'
if (_iter=3) _limit='27 - 35'
if (_iter=4) _limit='24 - 27'
if (_iter=5) _limit='22 - 24'
if (_iter=6) _limit='20 - 22'
if (_iter=7) _limit='18 - 20'
if (_iter=8) _limit='17 - 18'
if (_iter=9) _limit='16 - 17'
if (_iter=10) _limit='15 - 16'
if (_iter=11) _limit='14 - 15'
if (_iter=12) _limit='LT 14  '

  if (_iter=1) print list="************************** ROOT MEAN
SQUARE ERROR SUMMARY ****************************", printo=1
  if (_cnt[_iter]>0) print,
 list="  Percent RMSE for Volume Group ",_iter(2.0c),"
",_RGP[_iter](7.0c),"-",_RGP[_iter+1](7.0c),":
",sqrt(_err[_iter]/(_cnt[_iter]-
1))/(_cns[_iter]/_cnt[_iter])*100(5.1),"%", " Allowable RMSE = ",
_limit,"%    N=",_cnt[_iter](5.0CL), PRINTO=1



;ASR
;LOOP _iter=1,12
;  if (_iter=1) print list="************************** ROOT MEAN
SQUARE ERROR SUMMARY ****************************", printo=1
;  if (_cnt[_iter]>0) print,
; list="  Percent RMSE for Volume Group ",_iter(2.0c),"
",_RGP[_iter](7.0c),"-",_RGP[_iter+1](7.0c),":
",sqrt(_err[_iter]/_cnt[_iter])/(_cns[_iter]/_cnt[_iter])*100(5.1
),"%,                      N=",_cnt[_iter](5.0CL), PRINTO=1

ENDLOOP

if (_iter=13) _limit='32 - 39'
 list="  Percent RMSE for Volume Group ",_iter(2.0c),"
",_RGP[1](7.0c),"-",_RGP[_iter](7.0c),":
",sqrt(_err[_iter]/(_cnt[_iter]-
1))/(_cns[_iter]/_cnt[_iter])*100(5.1),"%", " Allowable RMSE = ",
_limit,"%    N=",,_cnt[_iter](5.0CL), PRINTO=1

;_iter=13
; list="  Percent RMSE for Volume Group ",_iter(2.0c),"
",_RGP[1](7.0c),"-",_RGP[_iter](7.0c),":
",sqrt(_err[_iter]/_cnt[_iter])/(_cns[_iter]/_cnt[_iter])*100(5.1
),"%,                      N=",_cnt[_iter](5.0CL), PRINTO=1

; one for Vol/Cnt by FT
_iter=0
LOOP _iter=1,100
```

```
  if (_iter=1) print list="\n","\n ********************** VOLUME
AND COUNT SUMMARY BY FACILITY TYPE **********************",
PRINTO=1
if (_cntbyft[_iter]>0) print,
 list="Facility Type Summary for FT=",_iter(3.0c),
 " VOL=",_volbyft[_iter](11.0c),
 " CNT=",_cntbyft[_iter](11.0c),
 " VOL/CNT=",(_volbyft[_iter]/_cntbyft[_iter])(5.2c),
 "   N=",_lnkbyft[_iter](5.0c), PRINTO=1
ENDLOOP




; one for Vol/Cnt by AT
_iter=0
LOOP _iter=1,100
  if (_iter=1) print list="\n","\n ************************
VOLUME AND COUNT SUMMARY BY AREA TYPE ************************",
PRINTO=1
if (_cntbyat[_iter]>0) print,
 list="    Area Type Summary for AT=",_iter(3.0c),
 " VOL=",_volbyat[_iter](11.0c),
 " CNT=",_cntbyat[_iter](11.0c),
 " VOL/CNT=",(_volbyat[_iter]/_cntbyat[_iter])(5.2c),
 "   N=",_lnkbyat[_iter](5.0c), PRINTO=1
ENDLOOP


ENDPHASE


ENDRUN
```

2. EVPIL00B.S

```
; Do not change filenames or add or remove FILEI/FILEO statements
using an editor. Use Cube/Application Manager.
*{PATH1}\hevaldbf.exe
*{PATH1}\rmsedbf.exe
*copy /a heval.out irheval.out
*copy /a rmse.out irrmse.out
```