

EX PARTE OR LATE FILED
DUPLICATE FILE COPY ORIGINAL



295 North Maple Ave.
Basking Ridge, NJ 07920

May 15, 1998

Ms. Magalie Roman Salas
Secretary
Federal Communications Commission
1919 M. St., NW, Room 222
Washington, D.C. 20554

RECEIVED

MAY 15 1998

FEDERAL COMMUNICATIONS COMMISSION
OFFICE OF THE SECRETARY

RE: Ex Parte Presentation – Proxy Cost Models
CC Docket No. 96-45

Dear Ms. Salas:

The attached sheets provide the computer source code for PNR's algorithms for placing surrogate data points both on the boundary of the Census Block in which they are found, and alternatively, along the roads of a specified collection of Tiger road types within that Census Block.

The Census Block surrogating method is the one currently used by PNR in developing surrogate geocode points for use in its Spatial Clustering Module. The Road surrogating method is the alternative described in AT&T's ex parte letter from Michael Lieberman to the Commission on March 2, 1998. It is also the method that generated certain of the empirical results filed by AT&T with the Commission on April 3, 1998.

Two copies of this Notice are being submitted to the Secretary of the FCC in accordance with Section 1.1206(a)(2) of the Commission's rules.

Sincerely,

Richard N. Clarke /ha

Richard N. Clarke

Attachments

cc: Donald Stockdale
Charles Keller
Sheryl Todd

Brad Wimmer
Robert Loube
Natalie Wales

No. of Copies rec'd 041
List A B C D E

CENSUS BLOCK SURROGATING SOURCE CODE

The following is the code to block surrogating. The input is the same as described in the Road Surrogating document.

The other input is the standard state/block boundary files that come with Mapinfo.

RECEIVED

MAY 15 1998

FEDERAL COMMUNICATIONS COMMISSION
OFFICE OF THE SECRETARY

```
'-----
'-----
'--
'--   Block Surrogating
'--
'-----
'-----

Include "MAPBASIC.DEF"

Declare Sub OkHandler
Declare Sub GetStateDlg

global OKProceed as Integer

Declare Sub MainLoop
Declare Sub FillBlock(WC as String, Block as String, NewObj as Object,
AddPoints as Integer, nResLines as Integer, rWeight as float, bWeight as
float, slb as float, emp as float, ctx as float)

Global RunString as String
Global StateName as String

dim TStart, TEnd as Integer
TStart = Timer()

'-----
'-----
'-----

'Register Table "C:\Projects\surrogate\nv_proj.DBF" TYPE DBF Charset
"WindowsLatin1" Into "C:\Projects\surrogate\nv_proj.TAB"
'Open Table "C:\Projects\surrogate\nv_proj.TAB" Interactive

Call GetStateDlg
if OKProceed then
    Close All

    RunString = "open table "" + StateName + "bk"" as Blocks"
    Run Command RunString

    RunString = "Register Table "" + StateName + "_proj.DBF"" TYPE
DBF Charset ""WindowsLatin1"" Into "" + StateName + "_proj.TAB""
    Run Command RunString

    RunString = "open table "" + StateName + "_proj"" as Projected"
    Run Command RunString
```

CENSUS BLOCK SURROGATING SOURCE CODE

```
RunString = "Create Table Missing (WC char(8), Block char(15),
n_res_sur Integer, n_bus_sur Integer, n_rl_ps Decimal(17,8), n_bl_ps
Decimal(17,8), n_slb_ps Decimal(17,8), n_emp_ps Decimal(17,8), n_ctx_ps
Decimal(17,8)) File "" + StateName + "_Miss" Type dbf"
Run Command RunString
RunString = "Create Table Points (WC char(8), Long Decimal(11,6),
Lat Decimal(11,6), nLines float, Block char(15), Type char(1), n_SLFirms
float, n_Emps float, n_Centrex float) File "" + StateName + "_Sur" Type
dbf"
Run Command RunString

'Open Window Message ' open Message window
'Set Window Message
'Font ("Helv", 1, 10, BLUE)
'Position (3, 2) ' place in upper left
'Width 5.0 ' make window 3" wide
'Height 6.0 ' make window 1" high
'Print "Surrogate " + StateName

call MainLoop
'note "Ready to Save"
'Commit Table Points As "\\moocow\surrrogates\ca_Sur"
Commit Table Points
Commit Table Missing

TEnd = Timer() - TStart
note TEnd + " Seconds to execute " + StateName + " (tiger 90)"
end if

'-----
'-----
'-----

Sub MainLoop

Dim ThisObj as Object
Dim Block, WC as String
dim Counts as Integer
dim nLines, nResLines, nBusLines as Integer

Counts=0
Do While Not EOT(Projected)
Counts = Counts + 1

'if Counts = 60 then
'NOTE "TEST EXIT FROM LOOP"
'Exit Do
'end if

'Print "WC "+Str$(Counts)
fetch next from Projected

block = Projected.cb
WC = Projected.Clli
'Print "Wire Center " + STR$(Counts) + " " + WC + " Block " + Block
```

CENSUS BLOCK SURROGATING SOURCE CODE

```

        select * From Blocks where Blocks.block = block into
TempTable
        Fetch First From TempTable

        if TableInfo(TempTable, TAB_INFO_NROWS) <> 0 AND
Str$(TempTable.obj) <> "" Then
            ThisObj = TempTable.obj

            nLines = 0
            nResLines = 0
            nBusLines = 0
            if Projected.n_Res_Sur > 0 then
                nResLines = Projected.n_Res_Sur
            end if

            if Projected.n_Bus_Sur > 0 then
                nBusLines = Projected.n_Bus_Sur
            end if

            nLines = nResLines+nBusLines
            dim rWeight, bWeight, slb, emp, ctx as float
            rWeight = Projected.N_rl_ps
            bWeight = Projected.N_bl_ps
            slb = Projected.N_slb_ps
            emp = Projected.N_emp_ps
            ctx = Projected.N_ctx_ps
            if nLines > 0 then
                call FillBlock(WC, Block, ThisObj, nLines,
nResLines, rWeight, bWeight, slb, emp, ctx)
            end if
        else
            Print "      Missed " + WC + " " + Block + " " +
Str$(nLines)

            Dim nResSur, nBusSur as Integer
            Dim nRlPs, nBlPs, nSlbPs, nEmpPs, nCtxPs as Float

            nResSur = Projected.n_Res_Sur
            nBusSur = Projected.N_bus_sur
            nRlPs = Projected.N_rl_ps
            nBlPs = Projected.N_bl_ps
            nSlbPs = Projected.N_slb_ps
            nEmpPs = Projected.N_emp_ps
            nCtxPs = Projected.N_ctx_ps

            Insert Into Missing Values(WC, Block, nResSur, nBusSur,
nRlPs, nBlPs, nSlbPs, nEmpPs, nCtxPs)
            end if
        Loop
End Sub

'-----
'-----
'-----
```

CENSUS BLOCK SURROGATING SOURCE CODE

```
Sub FillBlock(WC as String, Block as String, NewObj as Object, AddPoints
as Integer, nResLines as Integer, rWeight as float, bWeight as float, slb
as float, emp as float, ctx as float)
  Dim PerimLen As Float
  Dim Item as Integer
  Dim x, y, PrevX, PrevY, xInter, yInter As Float
  Dim Offset, CurDist as Float

  Dim nPoly, nNode As Integer
  nPoly = ObjectInfo(NewObj, OBJ_INFO_NPOLYGONS)
  nNode = ObjectInfo(NewObj, OBJ_INFO_NPOLYGONS+1)

  PerimLen = 0
  x = ObjectNodeX(NewObj, 1, 1) ' read longitude
  y = ObjectNodeY(NewObj, 1, 1) ' read latitude
  For Item = 2 to nNode
    PrevX = x
    PrevY = y
    x = ObjectNodeX(NewObj, 1, Item) ' read longitude
    y = ObjectNodeY(NewObj, 1, Item) ' read latitude
    PerimLen = PerimLen + Distance(x,y, PrevX, PrevY, "mi")
  Next

  Offset = PerimLen / AddPoints

  DIM LineObj, CircObj, Intersect as Object

  CurDist = Offset
  x = ObjectNodeX(NewObj, 1, 1) ' read longitude
  y = ObjectNodeY(NewObj, 1, 1) ' read latitude

  if (nResLines > 0) then
    Insert Into Points Values(WC, x, y, RWeight, Block, "S", 0,
0, 0)
  else
    Insert Into Points Values(WC, x, y, BWeight, Block, "C", slb,
emp, ctx)
  end if

  Dim Count as Integer
  Count = 1

  For Item = 2 to nNode
    PrevX = x
    PrevY = y
    x = ObjectNodeX(NewObj, 1, Item) ' read longitude
    y = ObjectNodeY(NewObj, 1, Item) ' read latitude

    CurDist = CurDist - Distance(x,y, PrevX, PrevY, "mi")

    while CurDist <= 0
      LineObj = CreateLine(x,y, PrevX, PrevY)
      CircObj = CreateCircle(X, Y, -CurDist)
      Intersect = IntersectNodes(LineObj, CircObj,
INCL_CROSSINGS)
```

CENSUS BLOCK SURROGATING SOURCE CODE

```

        nNode = ObjectInfo(Intersect, OBJ_INFO_NPNTS)
        if nNode <> 0 Then
            xInter = ObjectNodeX(Intersect, 1, 1) ' read
longitude
            yInter = ObjectNodeY(Intersect, 1, 1) ' read
latitude

            Count = Count + 1
            if Count > nResLines then
                Insert Into Points Values(WC, xInter,
yInter, BWeight, Block, "C", slb, emp, ctx)
            else
                Insert Into Points Values(WC, xInter,
yInter, RWeight, Block, "S", 0, 0, 0)
            end if
        end if

        CurDist = CurDist + Offset
    WEnd
Next
End Sub

Sub GetStateDlg
Dialog
    Title "Surrogate Point Distribution Tiger 90"
    Width 270

    Control StaticText
        Title "State (2 digit name):"
        Position 10, 10
    Control EditText
        Value StateName
        Into StateName
        Position 75, 8
        Width 15

    Control OKButton
        Title "Run"
        Position 120, 5
        Calling okhandler

    Control CancelButton
        Position 120, 23
End Sub
'-----
'-----
'-----

Sub OKHandler
    OKProceed = 1
End Sub
```

ROAD SURROGATING SOURCE CODE

The Road Surrogating process is done in FoxPro so the inputs are all table based.

There are two table structures at the core of the process. Assuming Florida:

FL_proj - is the same inputs used in the Block boundary surrogating. The fields are:

CLLI	- The 8 digit CLLI code
CB	- the 15 digit Block code
N_Res_Sur	- # of residential surrogate points needed
N_Bus_Sur	- # of firm surrogate points needed
N_rl_ps	- # of lines per surrogate residence
N_bl_ps	- # of lines per surrogate business
N_Slb_ps	- rate of single line business
N_Emp_ps	- average # of employees
N_ctx_ps	- rate of centrex

RSeg_FL????? - where ????? is the state/county code. This file has been built from the Tiger

files and contains the following fields:

Type	- Tiger Road type
Hits	- the number of times the segment hits the block boundary, 2 means the road is directly on the border.
BlockCode	- the block code without state/county prefix
Len	- Length of the segment
BegX	- Begin longitude
BegY	- Begin latitude
EndX	- End longitude
EndY	- End latitude

This file is generated directly from the road and block boundary TIGER files. It is summary used to speed up the surrogating process.

The following is the FoxPro code that takes the 2 inputs and generates the surrogate point file.

```
=====
=====
=====
```

```
StartTime = Time()

clear

cd d:\att_alm\roadsur

Set Talk Off
set safety off

close table all

use struct_sur in 0
copy struct to fl_Sur
use fl_Sur Alias Surrogates
```

ROAD SURROGATING SOURCE CODE

```
USE fl_proj.dbf Alias StateProj IN 0

SELECT *;
  FROM fl_proj Stateproj;
  ORDER BY Stateproj.cb;
  INTO table Projection

Select StateProj
use
Select Projection
nBlocks = RecCount()
Go Top

County = ""

private i, j
For i = 1 to nBlocks
  AddPoints = Projection.N_Res_Sur + Projection.N_Bus_Sur

  if AddPoints > 0
    if County <> Left(Projection.CB, 5)
      if len(County) > 0
        Select RoadSegs
        Use
      EndIf

      County = Left(Projection.CB, 5)
      USE fl\RSeg_fl&County..dbf Alias RoadSegs IN 0
    EndIf

    Select RoadSegs.*, Weights.weight FROM RoadSegs INNER JOIN
roadsur!weights ;
    ON RoadSegs.hits = Weights.hits ;
    Where BlockCode = PadR(alltrim(SubStr(Projection.cb,6,10)),
10, '0') ;
    Into Cursor Segments

    TotDistance = 0
    nSegs = RecCount()
    Go Top

    for j = 1 to nSegs
      TotDistance = TotDistance + Segments.Len *
Segments.Weight
      Skip
    EndFor

    Offset = TotDistance / AddPoints
    CurDist = Offset /2

    Count = 0
    for j = 1 to nSegs
      Go j in Segments
      CurDist = CurDist - Segments.Len * Segments.Weight

      X = Segments.BegX
```

ROAD SURROGATING SOURCE CODE

```

        Y = Segments.BegY
        Do While CurDist <= 0
            if Segments.BegX = Segments.EndX
                Y = Segments.EndY + ((Segments.EndY -
Segments.BegY) * (CurDist / Segments.Len)) / Segments.Weight
            else
                if Segments.BegY = Segments.EndY
                    X = Segments.EndX + ((Segments.EndX -
Segments.BegX) * (CurDist / Segments.Len)) / Segments.Weight
                else
                    Y = Segments.EndY + ((Segments.EndY -
Segments.BegY) * (CurDist / Segments.Len)) / Segments.Weight
                    X = Segments.EndX + ((Segments.EndX -
Segments.BegX) * (CurDist / Segments.Len)) / Segments.Weight
                EndIf
            EndIf

            Count = Count + 1

            if Count > Projection.n_Res_Sur
                Inp_nLines = Projection.n_bl_ps
                INP_Type = "C"
                INP_SLFirms = Projection.n_slb_ps
                INP_Emps = Projection.n_emp_ps
                INP_Centrex = Projection.n_ctx_ps
            else
                Inp_nLines = Projection.n_rl_ps
                INP_Type = "S"
                INP_SLFirms = 0
                INP_Emps = 0
                INP_Centrex = 0
            endif

            Select Surrogates
            APPEND BLANK
            REPLACE WC with Projection.Clli, ;
                Long with X, ;
                Lat with Y, ;
                NLines with Inp_nLines, ;
                Block with Projection.CB, ;
                Type with Inp_Type, ;
                n_SlFirms with Inp_SLFirms, ;
                n_Emps with Inp_Emps, ;
                n_Centrex with Inp_Centrex

            CurDist = CurDist + Offset
        EndDo
    EndFor
EndIf

    Skip in Projection
EndFor

use projection
use
Erase projection.dbf
```

ROAD SURROGATING SOURCE CODE

```
? StartTime + " TO " + Time()  
messagebox("FINISHED " + StartTime + " TO " + Time())  
EndProc
```

ROAD SURROGATING.txt

PNR and Associates
All rights reserved

Page 4 of 4