EX PARTE OR LATE FILED

Sprint

Pete Sywenki
Director, Federal Regulatory Relations

Law & External Affairs
1850 M Street, NW, Suite 1100
Washington, DC 20036
Voice 202 828 7452
Fax 202 296 3469
pete.n.sywenki@mail.sprint.com

RECEIVED EX PARTE

DEC - 3 1998

FEDERAL COMMUNICATIONS COMMISSION
OFFICE OF THE SECRETARY

December 3, 1998

Ms. Magalie Roman Salas
Secretary - Federal Communications Commission
1919 M Street, N.W. Room 222
Washington, D.C. 20554

RE: CC Docket Nos. 96-45 and 97-160

Dear Ms. Salas,

Yesterday, representatives of the Benchmark Cost Proxy Model (BCPM) joint sponsors met with members of the FCC's Common Carrier Bureau with regard to the above referenced proceedings. Representing the BCPM joint sponsors were Jim Stegeman of INDETEC, Ken Cartmell, Peter Copeland, and Glen Brown of USWest, and myself for Sprint. The CCB was represented by Bryan Clopton, Bill Sharkey, Katie King, Craig Brown, and Richard Smith.

The purpose of the meeting was to discuss the BCPM sponsor's findings from our ongoing review of the FCC's synthesized cost proxy model. In addition, we discussed an alternative source of customer location data that we have obtained from Stopwatch Maps, which we have been using to conduct our analyses. The materials attached to this letter were provided and formed the basis of our discussions. Included in these materials are findings from our review and a detailed description of the Stopwatch Maps customer location input data.

The original and three copies of this notice are being submitted to the Secretary of the FCC in accordance with Section 1.1206(b)(1) of the Commission's rules. If there are any questions, please call.

Sincerely,

Pete N. Sywenki

Pete Sywenki

Attachment

cc: C. Brown K. King R. Smith
B. Clopton R. Loube B. Sharkey

The Hybrid Cost Proxy Model
Code Review

By
The BCPM Sponsors
December 2, 1998

Introduction

The BCPM sponsors have undertaken a comprehensive review the HCPM. The review has been divided into 2 main study areas. The first is to produce comparative output data between the HCPM and BCPM on North Dakota, Washington, Nevada, Florida and Georgia. These state runs should enable sponsors to gain an understanding of the relevance and sensitivity of various model input parameters. Preliminary output is summarized in HCPM review.xls.

The second study area is a detailed code review. The code review utilized source code released by the FCC and examined under a compiler or within Visual Basic, as appropriate. This purpose of this document is to describe the interim results of the code review process.

Study Methodology

State runs were generated on Pentium 400 MHz computers with 64-128 MB of RAM and 6 Gigabyte hard-drives. Smaller states were run under Windows 98. Larger states required an upgrade to Windows NT 4.0.

Code analysis was performed on Windows 95/98 and 3.1x computers. The compiler and debugger, Turbo Pascal for Windows 1.5 was run in its native 3.1x environment or under Win 95 with a manufacturer supplied patch.

Results

Our code analysis shows the HCPM continues to have some problems with code design and program logic.

Code design

Probably the most significant challenges in our code review are the design of the code and the environment in which the HCPM was developed. The model continues to be a collection of Pascal and Visual Basic executables that are not structured in a logical or consistent manner. Auditing the progress of calculations is hampered by the collection of modules and logic steps used. Further, variables are not named consistently or uniformly.

Development environment

The Pascal compiler and debugger selected for development are no longer actively supported by the manufacturer. This could potentially impact the developers ability to maintenance and improve the HCPM.

Inprise, the manufacturer of Turbo Pascal 1.5 for Windows, released a patch to allow users to debug in a Win95 environment. It is unknown whether the manufacturer will support the development environment under future versions of Windows. Although it is possible to run the model in a Win95 environment, the model is still constrained to an extremely small amount of memory accessible in a 16 bit application. The memory limitation of the compiler places extreme limitations on our ability to augment the code with audit and tracer routines. In many portions of the code, adding a new code line will force the compiler to abort.

With the FCC's release of a new model, we were forced to review and redo our analysis. Although we were told this new release only contained minor changes, we discovered an addition of at least 40 subroutines, new file structure and a new interface. This only serves to illustrate the increasing importance of version control and traceability within future code releases. Version control with detailed information on changes and modifications not only saves time and money, but it will produce an audit trail to assist users understanding how and why output results change.

Interface

With the addition of the new Visual Basic control interface, HCPM took a significant step forward in usability. Unfortunately, the interface limits the user's access to many batch commands available within the code. The user cannot generate auditing output (-xv switch on FEEDDIST) nor use the full flexibility of batch commands.

The user is not aware of many of the steps the code is going through. As the HCPM selects a clustering approach and potentially re-clusters, the user is not informed. Also, if the data exceeds hard-coded constraints within clustering (number of raster points per cluster), no indication is given that the program will modify the selected raster size to reduce the number of raster points used. The interface should provide this type of information. Further the documentation provides no references to these hardcoded constraints.

Finally, when running the HCPM in Windows 95, the DOS window that is opened during unzip, is not automatically closed.

Program Logic

As we discovered questionable areas of the code, we attempted to excerpt code sections and attach them to issue statements. Copies of them are included as an appendix.

The text below provides summaries of our findings.

Clustering procedure

Clustering customers who are equidistant between two centroids may put the clustering algorithm into an endless loop. This can occur under any of the three clustering approaches.

The clustering algorithms use different line limits. The divisive algorithm uses lines * fill factor. The other algorithms use raw line counts.

Clustering sometimes uses hard coded variables rather than recognizing the users input values. Raster size is a potential example.

When a set of clusters is optimized by noise reduction, the process potentially terminates due to exceeding a time constraint. This could cause the HCPM to output different results on machines with different processing speeds.

The quadrant assignment formula in Clusintf is incorrect when switch and cluster points are on the same vertical axis.

Feeder and Distribution

There are clusters with no investments for SAIs. The code seems to indicate that an SAI investment will only be registered when the choice of technology is analog. There is no SAI investment under T1 or fiber technologies although placement of an SAI is indicated in the .OUT file.

HCPM does not track single business lines, rather it uses a hardcoded value of 0.10 to report out single business lines.

HCPM undercounts total business lines since it subtracts specials from the PNR supplied value, which represents total non-special business lines.

NID and Drop investments are a consistent value and are assigned equally based on number of lots. NID and Drop investments do not seem to vary based on customer or structure type. For example, a 50 unit apartment building will have the same NID and Drop investment as a single unit house.

The code seems to assign a minimum of 3 conduits within the feeder plant.

HCPM develops feeder costs for the entire wirecenter and then assigns feeder costs equally to all lines within the wirecenter.

The number of lots may be understated due to the Round function. If the value of lines/lot is >.01 and <.49 the value is rounded down. The number of lots is not trued up against the line counts per wire center.

Weighted density used for distribution distorts cluster density by factors ranging from 1 to 100. In other words, all the clusters in a wirecenter will have the similar densities even though one cluster may have a density of 10 while another has a density of 1000. Interface should allow the user to turn off the weighting.

Structure

In looking up costs for manholes in normal terrain, softrock values are used.

Soil slope factor does not appear to be applied correctly.

The water table depth adjustment is not made.

The determination of hardrock, softrock and normal structure appears to be incorrect.

Manhole costs for softrock structure and normal structure are reversed.

Reallocation of structure percentages is not done properly. The percentages are only recalculated if they add to less than the total.

Processing

The HCPM continues to have problems processing large states on the WIN95 platform. The model may terminate or cause a GPF.

The HCPM sometimes drops a wirecenter during processing. We can find no consistent explanation for this

HAI Logic

The HAI platform as it existed in HAI5.0a has been extensively reviewed on the public record. We would encourage the use of this extensive research. However, the use of the HAI with the HCPM has not been investigated nor have the changes to the HAI that were made for the synthesized model.

We have just begun our efforts in this review. The text below provides a summary of our initial findings.

Inputs

Sharing in HAI needs to be set to 1. Otherwise sharing will be double counted with inputs from HCPM.

HCPM cable and structure inputs do not flow to Transport inputs of HAI.

Expense Logic

The attached document that was filed in Texas, overviews some of the coding problems in the Expense module. We have not verified that these errors still exist in the updated HCPM/HAI logic. However, they need to be reviewed.

Appendix 1
HCPM Code
Representative Issue Statements

Issue:

Soil slope factor not applied properly.

Potential Impact:

Structure influenced calculations may be incorrect.

HCPM Code in question:

from structure_cost_fn():

```
 if (MinSlope < MinSlopeTrigger) and (MaxSlope > MaxSlopeTrigger) then
    begin
       ugd_structure := ugd_structure*CombSlopeFactor;
       bur_structure := bur_structure*CombSlopeFactor;
       aer_structure := aer_structure*CombSlopeFactor;
       manhole_cost  := manhole_cost*CombSlopeFactor;
    end
    else
    if (MinSlope < MinSlopeTrigger) then
    begin
       ugd_structure := ugd_structure*MinSlopeFactor;
       bur_structure := bur_structure*MinSlopeFactor;
       aer_structure := aer_structure*MinSlopeFactor;
       manhole_cost  := manhole_cost*MinSlopeFactor;
    end
    else
    if (MaxSlope > MaxSlopeTrigger) then
    begin
       ugd_structure := ugd_structure*MaxSlopeFactor;
       bur_structure := bur_structure*MaxSlopeFactor;
       aer_structure := aer_structure*MaxSlopeFactor;
       manhole_cost  := manhole_cost*MaxSlopeFactor;
    end;
```

Code should read:

```
        If (MinSlope > MinSlopeTrigger) then
                {minimum soil slope beyond limits}
                If (MaxSlope > MaxSlopeTrigger) then
                        {maximum soil slope beyond limits}
                        begin
                        ugd_structure := ugd_structure*CombSlopeFactor;
                        bur_structure := bur_structure*CombSlopeFactor;
                        aer_structure := aer_structure*CombSlopeFactor;
                        manhole_cost  := manhole_cost*CombSlopeFactor;
                        end
                else
                        {maximum soil slope within limits}
                        begin
                        ugd_structure := ugd_structure*MinSlopeFactor;
                        bur_structure := bur_structure*MinSlopeFactor;
                        aer_structure := aer_structure*MinSlopeFactor;
                        manhole_cost  := manhole_cost*MinSlopeFactor;
```

```
            end
else
        {minimum soil slope within limits}
        if (MaxSlope > MaxSlopeTrigger) then
        begin
            {maximum soil slope beyond limits}
        ugd_structure := ugd_structure*MaxSlopeFactor;
        bur_structure := bur_structure*MaxSlopeFactor;
        aer_structure := aer_structure*MaxSlopeFactor;
        manhole_cost  := manhole_cost*MaxSlopeFactor;
        end
```

Issue:

Water table depth adjustment not made.

Potential Impact:

There is currently no adjustment made if the water table depth is less than the trench depth.

Calculation should look like:

```
if (WaterTb <= CriticalDepth) then
        begin
        ugd_structure := ugd_structure*WaterFactor;
        bur_structure := bur_structure*WaterFactor;
        aer_structure := aer_structure*WaterFactor;
        manhole_cost  := manhole_cost*WaterFactor;
        end
```

Issue:

Determination of Hardrock, Softrock and Normal structure appears to be incorrect.

Potential Impact:

Calculations dependent upon structure may be incorrect.

HCPM Code in question-

```
if (depth_to_bedrock < critical_depth) and (hardness='HARD') then { use hard rock values}
      .
      .
if ( (depth_to_bedrock >= critical_depth) and (soil_texture_indicator=1) )
or ( (depth_to_bedrock <  critical_depth) and (hardness <> 'HARD') ) then { use soft values }
      .
      .
else  { use normal values }
```

The comparable formula from Excel (Loop.xls) is

```
=IF(AND(DepthToBedrock<=NormalUGBuriedCover,RockHardness="HARD"),1,IF(AND(
DepthToBedrock>NormalUGBuriedCover,SurfaceIndicator=0),3,2))
```

where 1 = Hardrock, 2 = Softrock, 3 = Normal. The Pascal code should be restated as follows:

```
if (depth_to_bedrock < critical_depth) and (hardness='HARD') {use hardrock values}
 .
 .
else
        if ((depth_to_bedrock <= critical_depth) and (soil_texture_indicator=0)) {normal
values}
         .
         .
        else
        {softrock values}
```

Issue:

Manhole costs for SoftRock Structure and Normal Structure are reversed.

Potential Impact:

Calculations dependent on structure determinations may be incorrect.

HCPM Code in question:

```
if ( (depth_to_bedrock >= critical_depth) and (soil_texture_indicator=1) )
or ( (depth_to_bedrock <  critical_depth) and (hardness <> 'HARD') ) then { use soft values }
begin
    if feeder_indicator=1 then
    begin
        ugd_structure := ugd_share*SoftRockStruc[zone]^.FeedUgd;
        bur_structure := bur_share*SoftRockStruc[zone]^.FeedBur;
        aer_structure := aer_share*SoftRockStruc[zone]^.FeedAer;
    end

    else
    begin
        ugd_structure := ugd_share*SoftRockStruc[zone]^.DistUgd;
        bur_structure := bur_share*SoftRockStruc[zone]^.DistBur;
        aer_structure := aer_share*SoftRockStruc[zone]^.DistAer;
    end;

    if feeder_indicator=1 then
    NumberOfDucts := round( copper_lines/feed_copper_cable_capacity + half ) +
                round( fiber_lines/fiber_cable_capacity + half) + 1
    else
    NumberOfDucts := round( copper_lines/dist_copper_cable_capacity + half ) + 1;
    if NumberOfDucts < 2 then NumberOfDucts := 2;

    ManholeSpacing := ManholeSpac[zone]^.ManholeSpacing;

    i := NumManholeSizes;
    repeat
        i := i-1;
        if NumberOfDucts >= ManholeCost[i]^.DuctCap
        then manhole_cost := ManholeCost[i]^.NormalCost/ManholeSpacing; { manhole cost
per foot for underground}
    until NumberOfDucts >= ManholeCost[i]^.DuctCap;

    if NumberOfDucts > ManholeCost[NumManholeSizes-1]^.DuctCap
    then manhole_cost := manhole_cost +
        ManholeCost[NumManholeSizes]^.NormalCost*(NumberOfDucts -
ManholeCost[NumManholeSizes-1]^.DuctCap);
end
else  { use normal values }
begin
    if feeder_indicator=1 then
    begin
        ugd_structure := ugd_share*NormalStruc[zone]^.FeedUgd;
        bur_structure := bur_share*NormalStruc[zone]^.FeedBur;
        aer_structure := aer_share*NormalStruc[zone]^.FeedAer;
```

```
        end

        else
        begin
            ugd_structure := ugd_share*NormalStruc[zone]^.DistUgd;
            bur_structure := bur_share*NormalStruc[zone]^.DistBur;
            aer_structure := aer_share*NormalStruc[zone]^.DistAer;
        end;

        if feeder_indicator=1 then
        NumberOfDucts := round( copper_lines/feed_copper_cable_capacity + half ) +
                    round( fiber_lines/fiber_cable_capacity + half ) + 1
        else
        NumberOfDucts := round( copper_lines/dist_copper_cable_capacity + half ) + 1;
        if NumberOfDucts < 2 then NumberOfDucts := 2;

        ManholeSpacing := ManholeSpac[zone]^.ManholeSpacing;

        i := NumManholeSizes;
        repeat
            i := i-1;
            if NumberOfDucts >= ManholeCost[i]^.DuctCap
            then manhole_cost := ManholeCost[i]^.SoftCost/ManholeSpacing; { manhole cost per
foot for underground}
        until NumberOfDucts >= ManholeCost[i]^.DuctCap;

        if NumberOfDucts > ManholeCost[NumManholeSizes-1]^.DuctCap
        then manhole_cost := manhole_cost +
            ManholeCost[NumManholeSizes]^.SoftCost*(NumberOfDucts -
ManholeCost[NumManholeSizes-1]^.DuctCap);
    end;
```

Issue:

Reallocation of structure percentages is not done properly. The percentages are only recalculated if they add to less than the total.

HCPM Code in question:

```
free_pct := one - pct_ugd - pct_bur - pct_aer;
if free_pct < zero then free_pct := zero;
```

(note: code line split for clarity)

if free_pct < zero then structure costs have been over allocated and no correction is made. Corrections to allocation should be made in all cases.

Suggestions:

1. Input corrections should be done when a file is read. The reallocation of structure percentages should be done in either globals.pas or incorporated into a separate module. This has the benefit that the corrections are made only once and not once for every microgrid.

2. Since Terrain and Density data are only kept at cluster level, the calculation of factors should be done at that level also. There are a couple of possible ways this could be done -

a. Calculate a density index and keep it with the cluster record. This would eliminate the need for lookups every-time it is used.

b. Revise the cluster record to include the structure type (hard, soft, normal) and a composite factor that would include sharing, water table, and soil slope adjustments. The structure cost would then be easily calculated, without repeating these lookups for every microgrid.

These calculations could be done in the clustering portion of the HCPM and written out as part of the *.CLU file.

Issue:
HCPM seems not to share the 4 fibers on a small DLC system with other systems on the same route.

Potential Impact:
This could lead to an overstatement of the total fibers needed.

HCPM Code in question:

Source: Printout.Pas:

```
cabcost :=
            feed_cable_cost( (n2016+n672+n96+n24)*4.0/FiberFillFactor, density, fiber, uc, bc,
ac, uf, bf, af,
                  pct_ugd, pct_bur, pct_aer );
```

Issue:

When running HCPM in Windows 95, the DOS window that is opened when running PKZIP/PKUNZIP is not closed automatically when the zipping/unzipping process is complete.

Cause:

In Windows 95, when a DOS application writes text to the screen, the DOS window does not get closed automatically. In order for the DOS window to close automatically, the PKZIP/PKUNZIP commands must be called in such a way as to prevent any text from being written to the screen.

Solution:

One solution is to use a DOS batch file to call PKZIP/PKUNZIP and pipe the messages to the DOS nul device. The contents of the batch file would be as follows:

```
@echo off
%1 %2 %3 %4 %5 %6 > nul
```

the VB code to call PKZIP/PKUNZIP would be as follows:

myFile = **gbatpath** & " " & ghcpmPath & "\pkunzip.exe -o " & gdataPath & "\" & state & ".zip @infiles.lst"

ExecApp (myFile)

Where ***gbatpath*** = the path and filename of the DOS file created above.

The above assignment to MyFile will be resolved as followed:
(assuming: gbatpath = "c:\hcpm\runzip.bat", ghcpmpath = "c:\hcpm", gdatapath = "c:\hcpmdata" and state = "co"

MyFile = c:\hcpm\runzip.bat c:\hcpm\pkunzip.exe –o c:\hcpmdata\co.zip @infiles.lst

This will call the DOS batch file "runzip.bat", which will be resolved as follows:

```
@echo off
c:\hcpm\pkunzip.exe –o c:\hcpmdata\co.zip @infiles.lst > nul
```

This will call pkunzip, and pipe any screen messages to the NUL device.

Issue:
HCPM seems to develop feeder cost for the wirecenter and then assign the feeder cost EQUALLY to all lines within the wirecenter.

Potential Impact:
If the determination is made that the subsidy unit should be below the wirecenter, the current code will produce erroneous results. If the model is to be used for UNE's the results could be erroneous.

HCPM Code in question:

Source: Printout.Pas:

```
FeedAllocation*feed_ugd_cable:4:2,',',
            FeedAllocation*feed_bur_cable:4:2,',',
            FeedAllocation*feed_aer_cable:4:2,',',
            FeedAllocation*feed_ugd_fiber:4:2,',',
            FeedAllocation*feed_bur_fiber:4:2,',',
            FeedAllocation*feed_aer_fiber:4:2,',',
            FeedAllocation*0.1*(feed_ugd_structure):4:2,',', { feeder conduit }
            FeedAllocation*FeedManholeCost:4:2,',',         { feeder manhole }
            FeedAllocation*0.45*(feed_ugd_structure):4:2,',', { copper feeder plcmt }
            FeedAllocation*0.45*(feed_ugd_structure):4:2,',', { fiber feeder plcmt }
            FeedAllocation*0.5*feed_bur_structure:4:2,',', { copper buried plcmt }
            FeedAllocation*0.5*feed_bur_structure:4:2,',', { fiber buried plcmt }
            FeedAllocation*feed_aer_structure:4:2,',',
```

Issue:
HCPM seems to develop single business lines as a single factor multiplied against Total Business lines

Potential Impact:
The determination of subsidy and some of the loop cost could be incorrect. If subsidy is for only single business lines, a better estimation should occur. This better estimation would also improve the development of the loop cost (drops, nids, terminals).

HCPM Code in question:

Source: Printout.Pas:

```
{ single line business lines } SA_array^[i]^.BusLines*0.1:1:0,',',
```

Issue:

SAI investments are only populated when the feeder technology selected is analog. Neither T1 nor fiber technology seem to have any associated SAI investments.

HCPM code in question:

:

```
technology := copper26;

if ( c24 < c26 )
or (feeder_distance + SA_array^[i]^.MaxDistance > copper_gauge_xover)
then technology := copper24;

if ( ( ct1 < min( c24,c26 ) ) )
or (feeder_distance + SA_array^[i]^.MaxDistance > max_copper_distance)
or (feeder_distance > copper_t1_xover)
then technology := t_1;

if ( cf < min( min(c24,c26), ct1 ) )
or (feeder_distance > t1_fiber_xover)
then technology := fiber;

SA_array^[i]^.feeder_technology := technology;

if technology = fiber then
begin
   SA_array^[i]^.fiber_terminal_cost :=
   fiber_terminal_cost_fn(SA_array^[i]^.lines/FillFactor,feeder_distance,SA_array^[i]^.c
                  n2016,n672,n96,n24,pct_ugd,pct_bur,pct_aer);

   SA_array^[i]^.n2016 := n2016;
   SA_array^[i]^.n672 := n672;
   SA_array^[i]^.n96 := n96;
   SA_array^[i]^.n24 := n24;
end
else if technology = t_1 then
begin
   SA_array^[i]^.t1_terminal_cost := t1_terminal_cost_fn(SA_array^[i]^.lines/FillFactor,
   SA_array^[i]^.nc96 := n96;
   SA_array^[i]^.nc24 := n24;
   n2016 := 0;
   n672 := 0;
end
else                                   { technology is analog }
   SA_array^[i]^.interface_cost := tmp3;
```

The array SA_array^[i]^.interface_cost populates each cluster record under the SAI Investment field.

Issue:

NID and Drop investments are developed on a per lot basis. The same NID and Drop costs are used for each lot, whether the lot has 50 lines or 1.

Potential Impact:

NID investments may be distorted due to the assumption of one Drop and NID per lot. If we assume the IN files represent the number of housing units at a geocoded location, the code sets lots = number of records on IN file. This means that a lot with a 50 unit apartment building or 50 line Business location gets the same Drop and NID cost as the single family house. On the other hand, if we assume that the IN file represents a unique record for each housing unit then the code would install 50 drops and 50 NIDs for the 50 unit apartment building. Under either assumption, the NID and Drop costs are incorrect.

HCPM Code in question:

```
drop_cost := total_lots*drop_length*cost_per_drop_kf;
drop_feet := total_lots*drop_length;

{ Finally, calculate cost of nids for this microgrid }

MG_nid_cost := nid_cost*total_lots;
```

Appendix 2
HCPM
Open Issues List

Number	Issue	Date Added	Source	Subgroup	[illegible]	[illegible]	
11	Clustering customers may put program into an endless loop both Agglomerative and nearest neighbor. May occur in Divisive	11/13/98		Cluster			
12	Quadrant assignment formula is CLUSTINF is incorrect when switch and cluster point are on the same vertical axis.	11/18/98		Cluster			
29	Clustering algorithms utilize different line limits. The Divisive algorithm uses lines multiplied by the line fill factor. The other two clustering methods use straight-line counts. It is not apparent to the user that this is happening.	10/1/98		Cluster			
30	The clustering methods are not consistent. In some cases the constraints are hard-coded even though there are user inputs.	10/1/98		Cluster			
31	No automatic approach to picking the best clustering approach.	10/1/98		Cluster			
32	When a set of clusters is optimized by noise reduction, the process potentially terminates due to exceeding a time constraint. Could cause different answers on different machines.	10/1/98		Cluster			
34	The limit on the number of rasters is hardcoded to 3000 for divisive clustering. If the number of populated rasters in a wirecenter exceed this, the size of the rasters is increased in units of 500 until the number of populated rasters is less than 3000.	10/1/98		Cluster			
35	In agglomerative clustering, the line constraint does not include the	10/1/98		Cluster			

Number	Issue	Date Added	Source	Subcategory	Referred to Document	Date Resolved	[illegible]
	application of the fill factor.						
36	In the nearest neighbor clustering, the line constraint is hard coded to 1800.	10/1/98		Cluster			
1	System design-loose collection of executables	11/2/98		Code			
2	Full Flexibility of model built into batch commands is not available on VB interface	11/18/98		Code			
8	Some algorithms are based on rule of thumb-can't determine the validity.	11/2/98		Code			
14	Turbo Pascal Compiler is no longer supported by manufacturer	11/15/98		Code			
18	Plot button on interface doesn't work.	11/16/98		Code			
24	Code is repetitive. Layout could be simplified and allow easier review of the code.	11/18/98		Code			
25	Turbo Pascal Code is maxed out. We were unable to add ANY auditing code without rewriting the entire module.	11/18/98		Code			
27	Model appears to drop wirecenters in its processing. One run will have it in it, the next may not.	11/18/98		Code			
41	No code to allow the truing up of line counts to actual wirecenter line counts.	10/1/98		Code			
45	When running HCPM in Windows 95, the dos window that is opened is not closed automatically when zip/unzip process is complete	11/20/98		Code			
16	There are clusters which have no investments for SAIs or DLCs. The program appears to only capture SAI investment for Clusters served via copper (non T1 or DLC).	11/16/98		Feeddist			

Number	Issue	Date Added	Source	Subgroup	Relevant Document	[illegible]	[illegible]
17	Weighted density used for distribution distorts cluster density by factors ranging from 1 to 100. In other words, all the clusters in a wirecenter will have the similar densities even though one cluster may have a density of 10 while another has a density of 1000. Interface should allow the user to turn off the weighting.	11/16/98		Feeddist			
21	Model does not track Single business lines. Uses a hardcoded value of 10% to report out single lines	11/18/98		Feeddist			
22	NID and Drop values are fairly constant no matter the customer type (Res, Bus) or the number of lines to the lot.	11/18/98		Feeddist			
26	Appears to be that there is a minimum of 3 conduits put in for the feeder plant.	11/18/98		Feeddist			
43	HCPM seems to develop feeder cost for the wirecenter and then assigns feeder cost equally to all lines within wirecenter	11/20/98		Feeddist			
44	HCPM seems not to share the 4 fibers on a small DLC system with other systems on the same route.	11/20/98		Feeddist			Assumption of feeder modeling in documentation.
5	Numerous user inputs are ignored	11/2/98		Inputs			
6	Inputs are not fully defined	11/2/98		Inputs			
9	Stability of source (PNR) data- wirecenter boundaries/surrogates	11/3/98		Inputs			
10	Sensitivity of HCPM to input changes is not fully understood	11/3/98		Inputs			
13	Effect of customer location data is not fully understood	11/4/98		Inputs			
37	Why is T1 technology used inside of a	10/1/98		Inputs			

Number	Issue	Date Added	Source	[illegible]	[illegible]	[illegible]	[illegible]
	cluster.						
38	All inputs should represent loaded EFI'd values	11/18/98		Inputs			
39	T1 investment does not have any recognition of the need for repeaters	11/18/98		Inputs			
28	Business lines are undercounted. GIS inputs are non-special line counts. Model develops specials as a ratio and then subtracts the specials from the input Business lines.	11/18/98		Linecount			
23	Lots are understated due to the use of the Round function. Anytime the value of Lines/LinesperHousehold is greater than x.01 and less than x.49, the lots are rounded down. Should instead add .49 to the formula	11/18/98		Lot			
40	Lot calculation is not correct. Detailed explanation will follow.	11/18/98		Lot			
3	Initial clustering is not correct. Has potential of making route distance off by 2*raster length	11/2/98		Resolved		Corrected by FCC in newest release of 11/16/98	
19	In Clustintf, when running in batch mode, the state data (states.txt) is not loaded properly. When looking up the state, the target 'h^' is used. There is not a match so the values of take_rate, SpecialAccessRate, lines_per_house are those read from feeddist.prm. This impacts the density. When running a single state, the proper values are used. This problem does not exist in the feeddist program.	11/16/98		Resolved		Corrected by FCC in newest release of 11/16/98	

Number	Issue	Date Added	Source	Subject Area	Referring Document	Date Resolved	[illegible]
15	HCPM produced different results (investments) when Clust, Clust+inf and Feeddist are run versus Run all modules for HAI button.	11/17/98		Resolved		Corrected by FCC in newest release of 11/16/98	
33	The terrain data could be improved. First, the source could be modified to provide it by Census Block. Second, the code uses the terrain of the point closest to the Cluster centroid as the terrain for the entire centroid. This could be improved to recognize the terrain of the entire cluster.	10/1/98		Structure			
42	In looking up cost for Manholes in normal terrain, Softrock values are used.			Structure			
46	Soil slope factor does not appear to be applied correctly	11/20/98		Structure			
47	Water table depth adjustment not made	11/20/98		Structure			
48	Determination of Hardrock, softrock and Normal structure appears to be incorrect	11/20/98		Structure			
49	Manhole costs for Softrock structure and Normal structure are reversed	11/20/98		Structure			
50	Reallocation of structure percentages is not done properly. The percentages are only recalculated if they add to less than the total.	11/20/98		Structure			
4	Output file content not documented. Nor is HCPM output available with new model.	11/18/98		Unsure			
7	Intermediate workings of model are not understood	11/2/98		Unsure			

Appendix 3
HAI Expense Module Review
GTE review from Texas public record

Errors found in the Expense Module in HAI 5.0a

Subhendu Roy
GTE
September 22, 1998

A comparison of Wire center and Density zone runs with the Staff proposed 30 changes for GTE TX showed that the overall per line cost for all lines from the Wire center runs were lower than with Density zone runs. We would have expected the two costs to be the same provided the per line costs were calculated using the same line counts for weighting in both cases.

The HAI model produces costs by UNEs and then combines them to generate per line costs for USF sizing. So the first point of examination was whether the weighting methodology was the same in both modules. It was found that in the Density zone run for USF there were a number of options for weighting the density zone costs. The default weighting was by total switched lines. There was no option for weighting by all lines (both switched and non-switched). We could however get the total lines by density zone from another worksheet in the module and use that to calculate per line cost. For the Wire center runs no overall per line cost for GTE TX was readily produced and we got only per line cost by each wire center. We could however calculate an overall per line cost using the line counts for each wire center. When we used total lines (both switched and non-switched) for weighting in both cases we still found that the difference in the per line costs in the two modules persisted.

The differences observed in per line costs must therefore lie in the UNE costs used to calculate the per line cost. On examination of all the UNEs it was found that the difference in cost occurred because of differences in 3 UNE costs, namely, Feeder, Distribution and Operator Systems. So the reason for the differences in these 3 UNEs were examined in greater detail. The direct costs for all UNEs consisted of two elements, the capital cost and the expenses. Both were derived using investment and therefore the first step in the comparison was whether these were different in the two modules. In the next step we checked whether there were any differences in the costs even when the investments were the same.

In the comparison between the Density zone and Wire center modules we look at the formulae contained in the expense module for the two cases. Thus the Density zone module refers to the file R50a_expense_density.xls and the Wire center module refers to the file R50a_expense_wirecenter.xls. Both these files are contained in the subdirectory HM50/Modules.

Feeder

On comparison it was found that all the elements other than the buried and underground structure had the same investments in both modules. The difference in the buried and underground structure investment arose because in the Density Zone the investment already incorporated structure sharing while the Wire Center file showed the numbers without sharing and the sharing in that case was accounted for later while calculating costs. However that was unlikely to cause any difference so long as the sharing was done similarly. All the differences were therefore likely to be in the calculation of costs using these investments.

The following errors and differences were found in the formulae in the Density Zone and Wire Center file for the various components of cost.

Manhole Costs

<u>Error 1</u>

The formulae in the two modules for calculating costs were as follows

Density Zone; File: R50a_expense_density.xls; Tab: Feeder

Cell B13 =IF(Inputs!$G67>0.5,[redacted]'Investment Input'!$Q12*(1/((1/Inputs!$G67)-1)))
Cell B31 = B13*((1-(Inputs!$K$35-TRUNC(Inputs!$K$35)))*INDEX(CCCFact,1,TRUNC(Inputs!$K$35))+(Inputs!$K$35-TRUNC(Inputs!$K$35))*INDEX(CCCFact,1,1+TRUNC(Inputs!$K$35)))
Cell B48 = B13*'96 Actuals'!$F$51

So the total cost is given by
= IF(Inputs!$G67>0.5,[redacted],'Investment Input'!$Q12* (1/((1/Inputs!$G67)-1)))*((1-(Inputs!$K$35-TRUNC(Inputs!$K$35)))*INDEX(CCCFact,1,TRUNC(Inputs!$K$35))+(Inputs!$K$35-TRUNC(Inputs!$K$35))*INDEX(CCCFact,1,1+TRUNC(Inputs!$K$35)))
+ IF(Inputs!$G67>0.5,[redacted],'Investment Input'!$Q12*(1/((1/Inputs!$G67)-1)))* '96 Actuals'!$F$51

Wire Center; File: R50a_expense_wirecenter.xls; Tab: Investment Input

Cell DI3 = ((IF(Inputs!$G$70>0.5,[redacted]P3*(1/((1/Inputs!$G$70)-1))))*((1-(Inputs!$K$35-TRUNC(Inputs!$K$35)))*INDEX(CCCFact,1,TRUNC(Inputs!$K$35))+(Inputs!$K$35-TRUNC(Inputs!$K$35))*INDEX(CCCFact,1,1+TRUNC(Inputs!$K$35))))+((IF(Inputs!$G$70>0.5,[redacted],P3*(1/((1/Inputs!$G$70)-1))))*'96 Actuals'!$F$51)

The formulae in the two modules might appear complex but they were basically the same other than one crucial difference in the first line (See shaded portion). What it was saying is that if underground sharing (Inputs!$G67 for the Density zone or Inputs!$G$70 for the Wire center) was > 0.5 then in Density Zone it would pick up the manhole investment ('Investment Input'!$Q12) but for Wire Center it would pick up only the value 1. As a result hardly any manhole investment was included in Wire Center whenever underground structure sharing was > 0.5. That the model would produce identical results if all underground structure sharing was < 0.5 was further confirmed by running a scenario with underground structure sharing < 0.5.

Feeder Structure

In the case of feeder structures two different types of errors were present for underground and buried cable and associated structure costs.

Error 2

In the case of underground costs the formulae for calculating expenses in the two modules were as follows

Density Zone; File: R50a_expense_density.xls; Tab: Feeder

Cell B41 = B6*96 Actuals'!$F$45
Cell B44 = B9*'96 Actuals'!$F$45
Cell B49 = B14*96 Actuals'!$F$51
Cell B50 = B15*96 Actuals'!$F$51

So the total expenses are
=(B6+B9)*'96 Actuals'!$F$45 [redacted]

Wire Center; File: R50a_expense_wirecenter.xls; Tab: Investment Input

Cell DE3 = (I3+L3)*'96 Actuals'!$F$45

The difference was that while in both modules expenses associated with underground copper and fiber were accounted for, the Wire Center module did not include the expenses involved with underground structure investment corresponding to (B14+B15) in the Density zone file.

Error 3

In the case of buried structure the formulae for calculating expenses associated with structures in the two modules were as follows

Density Zone; File: R50a_expense_density.xls; Tab: Feeder

Cell B42 = B7*'96 Actuals'!$F$46
Cell B45 = B10*'96 Actuals'!$F$46
Cell B51 = B16*'96 Actuals'!$F$46
Cell B52 = B17*'96 Actuals'!$F$46

So the total cost was

=(B7+B10+B16+B17)*'96 Actuals'!$F$51

Wire Center; File: R50a_expense_wirecenter.xls; Tab: Investment Input

Cell DB3 = ((J3+M3)*'96 Actuals'!$F$46)+((S3+T3)*'96 Actuals'!$F$46*[redacted])

As already discussed the Buried Structure Investment in Density Zone already incorporated structure sharing as mentioned earlier. In case of Wire Center the factor Inputs!H70 was used to account for sharing (It was actually sharing for the density zone 650-850 about which we discuss later). However the cell was not fixed using a $ sign (e.g. Inputs!$H$70) as was done in other calculations. The formula mentioned was for the first wire center in the worksheet and as we go down to the subsequent wire center investments (S4+T4 and so on) the calculation picked up Inputs!H71 and so on and since the Input table had no entries after Inputs!H74 the wire centers from 6th row onwards got multiplied by 0 and did not pick up any expenses.

<u>Difference 1</u>

The remaining problem stemmed from the methodology for assigning structure sharing in case of Wire Center calculations for both underground and buried. As already mentioned the structure sharing in Density Zone was incorporated while calculating the investments. In the case of Wire Center this was included while calculating costs for each wire center by using the sharing percentage for the density zone 650-850. This was presumably assigned because it was the middle density zone among the 9 density zones and might be treated as the average for all wire centers irrespective of their actual density zones. As a result the WC and DZ results continued to be different even after the errors pointed out earlier were corrected.

The difference in Feeder costs thus occur from three errors in the manhole costs, in the underground structure expense, and in the buried structure expense. In addition there is a difference due to the manner in which the structure sharing is accounted for in both underground and buried.

Distribution

In the case of distribution the difference in cost arose from the underground trenching costs and buried placement costs. In both cases a problem similar to that in feeder discussed earlier was found. In addition there was an error in the underground trenching formula.

<u>Error 4</u>

In case of underground trenching cost the formulae were

Density Zone; File: R50a_expense_density.xls; Tab: Distribution

Cell B25 = B11*((1-(Inputs!$K$35-TRUNC(Inputs!$K$35)))*INDEX(CCCFact,1,TRUNC(Inputs!$K$35))+(Inputs!$K$35-TRUNC(Inputs!$K$35))*INDEX(CCCFact,1,1+TRUNC(Inputs!$K$35)))
Cell B38 = B11*'96 Actuals'!$F$51

So the total cost was
=B11*((1-(Inputs!$K$35-TRUNC(Inputs!$K$35)))*INDEX(CCCFact,1,TRUNC(Inputs!$K$35))+(Inputs!$K$35-TRUNC(Inputs!$K$35))*INDEX(CCCFact,1,1+TRUNC(Inputs!$K$35))) + B11*'96 Actuals'!$F$51

Wire Center; File: R50a_expense_wirecenter.xls; Tab: Investment Input

Cell CN3 = (Z3*[redacted])*((1-(Inputs!$K$35-TRUNC(Inputs!$K$35)))*INDEX(CCCFact,1,TRUNC(Inputs!$K$35))+(Inputs!$K$35-TRUNC(Inputs!$K$35))*INDEX(CCCFact,1,1+TRUNC(Inputs!$K$35)))+([redacted]

The problem regarding assigning the structure sharing (Inputs!$E$70) for the middle density zone 650-850 to all wire centers irrespective of their density was similar to the feeder calculations. However there was an additional error since while calculating expenses in Wire Center the sharing was not incorporated in the last term.

In case of buried placement cost the formulae were

Density Zone; File: R50a_expense_density.xls; Tab: Distribution

Cell B23 = B9*((1-(Inputs!$K$31-TRUNC(Inputs!$K$31)))*INDEX(CCCFact,1,TRUNC(Inputs!$K$31))+(Inputs!$K$31-TRUNC(Inputs!$K$31))*INDEX(CCCFact,1,1+TRUNC(Inputs!$K$31)))
Cell B36 = B9*'96 Actuals'!$F$46

So the total cost was
= B9*((1-(Inputs!$K$31-TRUNC(Inputs!$K$31)))*INDEX(CCCFact,1,TRUNC(Inputs!$K$31))+(Inputs!$K$31-TRUNC(Inputs!$K$31))*INDEX(CCCFact,1,1+TRUNC(Inputs!$K$31))) + B9*'96 Actuals'!$F$46

Wire Center; File: R50a_expense_wirecenter.xls; Tab: Investment Input

Cell CS3 = (AA3*Inputs!$D$70)*((1-(Inputs!$K$31-TRUNC(Inputs!$K$31)))*INDEX(CCCFact,1,TRUNC(Inputs!$K$31))+(Inputs!$K$31-TRUNC(Inputs!$K$31))*INDEX(CCCFact,1,1+TRUNC(Inputs!$K$31)))+(AA3[redacted]*'96 Actuals'!$F$46)

Again the problem regarding assigning the structure sharing (Inputs!$E$70) for the middle density zone 650-850 to all wire centers irrespective of their density was similar to the feeder calculations.

Operator Systems

<u>Error 5</u>

Density Zone; File: R50a_expense_density.xls; Tab: Operator
Cell B16: Inputs!K38

Wire Center; File: R50a_expense_wirecenter.xls; Tab: Investment Input
Cell EW3: Inputs!K38

The final error was found in the Operator system costs and that was in OS trunk costs. The formulae used in both cases at first glance appeared to be the same. However the difference arose since in the formula one of the inputs (Inputs!K38: average nonmetallic cable life) used in the determination of OS trunk costs was calculated using actual cable investment computed within the model. In the process an error was committed in the derivation of the input since the Density Zone calculation was made after incorporating structure sharing while the Wire Center calculation was made without accounting for structure sharing.

Surrogate Points Input Data to HCPM

Stopwatch Maps has generated customer location files suitable for input to the HCPM Model, using data that is available per Census Block from both public and private sources. Specifically, those sources are:

- US Census Bureau 1990 housing unit statistics, estimated to the 1997 level using additional Census Bureau figures for each county (as in BCPM, which estimated to the 1995 level)
- Residential units-in-structure from US Census Bureau STF3A data, packaged by Claritas, as in BCPM (the full distribution data is available only at the Census Block *Group* level, and is applied to each Census Block of the group)
- Business firms and business lines from PNR & Associates, as in BCPM
- Wire center boundaries and switch locations from the BLR Wire Center Premium Package, as in BCPM (note that, in this package, a wire center is composed of an integral set of Census Blocks)
- Roads from US Census Bureau's TIGER95 files (we would use TIGER97, but only five of the six CDs are released as of this date)
- Terrain from the Stopwatch Maps *Terrain by Census Block* copyrighted product, derived from STATSGO data, which provides a set of terrain data common to each Census Block

The above source data is used to generate a set of discrete *customer points*, for each wire center of each state, in the form appropriate for entry into HCPM's clustering program.

In this paper we first identify the form and content of the data that is delivered. We then address some considerations in the development of that data.

Data Delivered

The data is delivered on a set of CDs. Each CD contains the data for one or more states, each state's files compressed into a .ZIP file. When unZIPped, that data is in the ASCII comma-separated-variable file format prescribed by HCPM and described in this paper.

Each ZIPped file on CD, containing all the data for a state, has a name of the form ***xx*_IN.ZIP**, where *xx* is the standard two-character state abbreviation.

Within each ZIPped file is the set of HCPM input files for that state, one input file per wire center. Each HCPM input file has a name of the form ***xxxxxxxx*.IN**, where *xxxxxxxx* is the 8-

character CLLI of the wire center. In addition, there is a file **INFILES.LST,** containing a simple name list of all the .IN files, as required for HCPM processing.

Each file is a comma-separated ASCII text file, *not* using quotation marks to bound character fields (except where the field data contains a comma, which – in this data – occurs only in the Company Name field). The HCPM documention prescribes the form and content of that file. We first reprint section 6.2 of the HCPM document, which describes that file:

6.2 Data Inputs for CLUSTER: <FILENAME>.IN

The input file, <FILENAME>.IN, is a comma delimited ascii file. Its first line contains either the word "BLOCK" or the word "HOUSEHOLD" to identify the data aggregation level. The second and fourth lines are header lines. The third line contains the wirecenter's CILLI code, the latitude and longitude of its switch, the latitude and longitude of its central point, and the name of the company that provides it service. Starting on the fifth line, there is a record for each block or household. That record contains the following data:

CBNum	State FIPS + County FIPS + Tract No. + Block No.
Lon	Longitude of record's central point
Lat	Latitude of record's central point
HH	Number of households or number of residential lines
Bus	Number of business lines
Bedrock	Bedrock depth
Hardness	Rock type
Soil	Soil texture
WaterTbl	Water table depth
MinSlope	Minimum slope
MaxSlope	Maximum slope
Area	Area in thousands of square kilometers. 0 if household level data.

An example of the first few lines of input file "WHMRMDWM.IN" follows:

```
HOUSEHOLD
Wc_code, SwX, SwY, CenX, CenY, Company
WHMRMDWM,-76.46065,39.373451,-76.4427,39.380646,BELL ATLANTIC - MARYLAND INC - MD
CBNum,Lon,Lat,HH,Bus,Bedrock,Hardness,Soil,WaterTbl,MinSlope,MaxSlope,Area
24005411302101,-76.447148416,39.377513532,1,0,57.37,,SIL,4.24,4.21,11.36,0
24005411302101,-76.453734918,39.373976651,0,8,57.37,,SIL,4.24,4.21,11.36,0
```

The first four records of each of these files have a format different from all succeeding records in the file. We describe all 5 record types here.

Record 1

The data produced here is a set of discrete points. Therefore, the first record always contains the characters **HOUSEHOLD.**

Record 2

The second record contains the field names for the content of the record that follows. These names are the character string:

Wc_code,SwX,SwY,CenX,CenY,Company

Record 3

The third record contains the field data for the wire center itself. The fields are:

- **Wc_code**: The 8-character CLLI code of the wire center. It is necessarily the same as the name of the file itself. This name is taken from the *CLLI* field of the BLR wire center source data.

- **SwX**: Longitude of the main switch of the wire center. This is derived from the V & H coordinates of the switch in the BLR product (fields *Wc_v* and *Wc_h* respectively). It is expressed in degrees with 6 fractional digits.

- **SwY**: Latitude of the main switch of the wire center. This is derived from the V & H coordinates of the switch in the BLR product (fields *Wc_v* and *Wc_h* respectively). It is expressed in degrees with 6 fractional digits.

- **CenX**: Longitude of the centroid of the wire center's area. This is taken directly from the *Lon* field for the wire center in the BLR product. It is expressed in degrees with 6 fractional digits.

- **CenY**: Latitude of the centroid of the wire center's area. This is taken directly from the *Lat* field for the wire center in the BLR product. It is expressed in degrees with 6 fractional digits.

- **Company**: Name of the operating company. This is a field of as many as 60 characters, taken directly from the *Lecname* field for the wire center in the BLR product.

Record 4

The fourth record contains the field names for the content of all the records that follow. These names are the character string:

CBNum,Lon,Lat,HH,Bus,Bedrock,Hardness,Soil,WaterTbl,MinSlope,MaxSlope,Area

All Other Records

Each remaining record for a wire center represents a *customer location point*. These are residential and business customer points that – in this set of data – are *randomly generated* along all valid roads (interior as well as exterior) of a Census Block. Details of that generation of points appear in the next major section of this paper.

The points appear in Census Block order. Within a Census Block, the points appear in *no specified order* (which is the natural result of random location generation). Each point represents either a *residential structure* or a *business firm*. Each residential structure contains a certain number of housing units. Each business firm has a certain number of telephone lines. These are allocated from the source data (see the next major section for details) and indicated in these records.

These are the fields of a customer location record:

- **CBNum**: Census Block FIPS code, always 14 or 15 characters
- **Lon**: Longitude of the customer location point, expressed in degrees with 6 fractional digits (six fractional digits represents precision to *within 4 inches*, which is a precision finer than any true accuracy that can be achieved)
- **Lat**: Latitude of the customer location point, expressed in degrees with 6 fractional digits
- **HH**: Number of *housing units* in the structure represented by the point ... zero if a business location point
- **Bus**: Number of business lines for the business firm located by the point ... zero if a residential location point
- **Bedrock**: Minimum depth to bedrock for this Census Block (taken from the Stopwatch Maps product), expressed in inches as a whole number
- **Hardness**: Rock hardness for this Census Block (taken from the Stopwatch Maps product) ... **HARD** or **SOFT** or empty for normal hardness
- **Soil**: Soil surface texture for this Census Block (taken from the Stopwatch Maps product) ... See attached table for possible values
- **WaterTbl**: Minimum water table depth for this Census Block (taken from the Stopwatch Maps product), expressed in feet with 2 fractional digits
- **MinSlope**: Minimum slope for this Census Block (taken from the Stopwatch Maps product), expressed in degrees with 2 fractional digits
- **MaxSlope**: Maximum slope for this Census Block (taken from the Stopwatch Maps product), expressed in degrees with 2 fractional digits
- **Area**: As prescribed by HCPM for discrete point data, this field is always 0 (zero)

Considerations

There are several considerations in the generation of this data that should be called out. The remainder of this paper addresses those considerations.

Road Types Used

Just as in BCPM, we limit the roads along which we generate customer points to those we consider people may live and work along (eliminating, for example, limited access highways and their on/off ramps, but not eliminating the service roads that run beside many of them). Specifically, the *CFCCs (Census Feature Classification Codes)* of the road types, and whether we use or ignore each, are summarized here:

- **A1*x***: Limited access highway segments … *Ignored*
- **A2*x***: Primary roads without limited access … *Used* except for *A22, A23, A26,* and *A27*, in tunnels or underpasses, which are *ignored*
- **A3*x***: Secondary roads without limited access … *Used* except for *A32, A33, A36,* and *A37*, in tunnels or underpasses, which are *ignored*
- **A4*x***: Neighborhood roads without limited access … *Used* except for *A42, A43, A46,* and *A47*, in tunnels or underpasses, which are *ignored*
- **A5*x***: Vehicular trail (not usable by cars) … *Ignored*
- **A61**: Cul-de-sac … *Used*
- **A62**: Traffic circle … *Used*
- **A63**: Access ramp … *Ignored*
- **A64**: Service road … *Used*
- **A65**: Ferry crossing … *Ignored*
- **A7*x***: Alleys, driveways, and walkways … *Ignored*

Allocation Along Roads

The customer points are allocated randomly along all roads (of the types used) of each Census Block. Each customer point is set back 50 feet (perpendicularly) from the road segment along which it is generated, on whatever side of the road it is generated. For *boundary* roads, only the side of the road that belongs to the subject Census Block is used (if there is another Census Block on the other side of the road, as almost always there is, the other side is used in connection with that other Census Block.) For interior roads, both sides are used.

The distance along each road against which random offsets are allocated is really a "side-of-the-road" distance. Boundary roads have only a single side (in this Census Block), while interior roads have two sides. Thus, if a Census Block has – just for example – one mile of boundary roads and one mile of interior roads, the random allocation of points to road locations for *this* Census Block is *twice* as likely to place a point along those interior roads (because they have two sides) as along the boundary roads (which have only one side in *this* Census Block).

We point out also that the *distance* along a road against which random offsets are allocated is the true curved length of that road.

Random Number Distribution

We generate pseudo-random numbers, for the purpose of allocation, that will have – over a very large number of random generations – a uniform (an "unbiased") distribution. That does not mean that in any Census Block the distribution will be uniform (it will not, it will be random), but it does mean that the random allocations will not be *deliberately* clustered in a Census Block.

Census Blocks with No Roads

Although these are uncommon, we encounter some Census Blocks with one or more customers but *no* roads of the eligible types. For example, in Maryland, restricting ourselves to the types of roads listed above, 1/6th of 1% of the *populated* Census Blocks (99 Census Blocks) have no roads of the eligible types. When this occurs, we must have a basis under which to randomly allocate customer points.

In this circumstance, we generate the appropriate number of points randomly along *other* road types or, if somehow even those are not present, along the periphery of the Census Block. In every case, these are set back 50 feet.

Housing Units vs. Households

Please note that in this data *housing units* (not households) are the units of residential allocation. A point record may represent one or more *housing units* (the number is the the value in the 4th field of the record).

Granularity of Terrain Data

Note that the terrain data we use is averaged for each *Census Block*. This is finer than what HCPM currently uses. The terrain data that appears in current HCPM examples is *less* granular … It is averaged across a whole Census Block *Group*.

We would note further that this terrain data is copyrighted by Stopwatch Maps. If and when this data is to be *widely* distributed, we wish to discuss procedures for the protection of our data.

Soil Surface Texture Values

The following are the soil surface texture values used. These typically appear individually, but two *may* appear together in the form *xxxx-xxxx*, where each *xxxx* is one of these values:

Abbr	*Meaning*	*Abbr*	*Meaning*
BY	Bouldery	CBA	Angular Cobbly
BYV	Very Bouldery	CBV	Very Cobbly
BYX	Extremely Bouldery	CBX	Extremely Cobbly
C	Clay	CE	Coprogenous Earth
CB	Cobbly	CEM	Cemented

CIND	Cinders
CL	Clay Loam
CN	Channery
CNV	Very Channery
CNX	Extremely Channery
COS	Coarse Sand
COSL	Coarse Sandy Loam
CR	Cherty
CRC	Coarse Cherty
CRV	Very Cherty
CRX	Extremely Cherty
DE	Diotomaceous Earth
FB	Fibric Material
FL	Flaggy
FLV	Very Flaggy
FLX	Extremely Flaggy
FRAG	Fragmental Material
FS	Fine Sand
FSL	Fine Sandy Loam
G	Gravel
GR	Gravelly
GRC	Coarse Gravelly
GRF	Fine Gravelly
GRV	Very Gravelly
GRX	Extremely Gravelly
GYP	Gypsiferous Material
HM	Hemic Material
ICE	Ice or Frozen Soil
IND	Indurated
L	Loam
LS	Loamy Sand
LVFS	Loamy Very Fine Sand
MARL	Marl
MK	Mucky
MPT	Mucky Peat
MUCK	Muck
PEAT	Peat
PT	Peaty
RB	Rubbly
S	Sand
SC	Sandy Clay
SCL	Sandy Clay Loam
SG	Sand and Gravel
SH	Shaly
SHV	Very Shaly
SHX	Extremely Shaly
SI	Silty
SIC	Silty Clay
SICL	Silty Clay Loam
SIL	Silt Loam
SL	Sandy Loam
SP	Sapric Material
SR	Stratified
ST	Stony
STV	Very Stony
STX	Extremely Stony
SY	Slaty
SYV	Very Slaty
SYX	Extremely Slaty
UNK	Unknown
UWB	Unweathered Bedrock
VAR	Variable
VFS	Very Fine Sand
VFSL	Very Fine Sandy Loam
WB	Weathered Bedrock