VOLUME XII – TPA
Scientific Notebook 432

TABLE OF CONTENTS - VOLUME XII (in process entries) RF **04/15/2005**

Pages

01/27/2005 *RF*

Computer and Software

As of February 2002, the pc in my office (called bubo) was:

Primary computer running WindowsNT 4.00.1381 is called bubo (Acer, x86 Family 6 Model 4 Stepping 2; AT compatible with 512 MBytes RAM).

Software :

- Adobe Acrobat & Distiller version 5.0
- Adobe Illustrator 8.0
- Excel 97 SR-2
- Lahey/Fujitsu Fortran 95 version 5.0
- Sigma Plot2000 version 6.00
- Word 97 SR-2
- Word Perfect version 10

UNIX: SUN (use uname –X on SUNs and uname -msR) as of March 2003
Spock is a SUN sparc Ultra 4 (4 cpu), 64-bit,
running SunOS version release 5.9 (Kernel ID = Generic_1112233-11)
Software:

- f77 SUN fortran Workshop Compilers Version 5.0
- Mathematica 5.0

03/25/2005 *RF*

Window PC was upgraded March 21, 2005 to Windows2000 operating system

- Adobe Acrobat & Distiller version 5.0
- Adobe Illustrator 10.0.3
- ArcView 3.2a
- Excel 2002 (10.4302.4219) SP-2 (Office XP)
- Lahey/Fujitsu Fortran 95 version 7.1
- Mathematica 4.2
- Sigma Plot2000 version 6.00
- Word 2002 (10.4302.4219) SP-2 (Office XP|)
- Word Perfect version 10

Paul Landis and Shannon contributed to this scientific notebook. They used

- Earth Vision 7.5 on the SunFire UNIX system
- ArcGIS Version 9.0 on a Windows computer

01/27/2005 *lf*

SEEPAGE ABSTRACTION

Collaborators: Keith Compton (NRC), Dick Codell (NRC)

Objectives

First, figure out the actual seepage abstraction by going through the TPA code. Second, figure out a way to modify the inputs to reflect current seepage information. Third, determine a new approach for the seepage abstraction. The third item is needed because the current implementation has fow and fmult as a function of time in an external file, and fwet is sampled. There is no correlation between these parameters, and no linkage to percolation rate approaching the drifts. This is a problem! TPA Version 3.2 seepage approach had these correlations, but no time (sic percolation) dependency.

Working Directories

spock: ~rfedors/TPA500f/*
./TPA500o/*

bubo: E:\TEF_kti\TPAstuff\Seepage_KeithCompton*
.\TPA-FluxToDriftTracking*

<u>TPA Seepage Algorithm – Description of Old Algorithm Base on TPA Code</u>

The old seepage abstraction had fow, fwet, and fmult as distributions in the tpa.inp file.

fow was the convergence/divergence of percolation above the drift

fmult included capillary diversion around the drift, along-wall seepage, and diversion by engineered barrier components (drip shield and waste package)

fwet was the portion of waste packages that get wet by seepage

In TPA 4.2 and later (including TPA 5.00f), the algorithm changed. Mean values of fow and fmult were entered into a new external file, and a new sampled parameter was created to handle the combined uncertainty of fow and fmult. Also, fmult now included only the diversion at the drift wall, because two new parameters were created to address the diversion at the drip shield and waste package.

==

02/07/2005 *RF*

TPA 4.0 had the 3 factors, fow, fmult, and fwet. Values for these were derived using the description provided in Appendix F of the TPA Version 4.0 documentation, which is the last version of a user manual and code description for the TPA code. For TPA 4.1, the seepage abstraction was modified. The new abstraction does not necessarily map directly to the old abstraction. Dick Codell provided the early abstraction, and Dick and Dave Esh (I believe) provided the modifications used in the current version. In this old approach, fow and fmult were sampled input parameters in the tpa.inp file.

Now, fow and fmult are brought in as time dependent factors (currently they do not change with time) in an external file. A single sampled parameter intends to replace the previous sampling of each fow and fmult. Note that this approach is not consistent with the old approach. Stated otherwise, the new [average fow * average fmult * new sampled factor] will not equal the old [sampled fow * sampled fmult]. The new sampled factor, WastePackageFlowMultificationFactor, is in the tpa.inp file. Furthermore, fmult no longer includes the in-drift diversion caused by the waste package. Two new factors take care of the drip shield diversion and waste package diversion (see releaset.f code).

Two power point file were developed, and are shown in Figure XII-1 below. The first shows a high-level schematic of the interaction of subroutines in TPA. The releaset.f code is where the factors are brought in; ebsrel.f calls the results of the releaset.f code; ebsrel.f gets the flow rate approaching the drift from the reflux3 module in nfenv.f.

The second power point slide has all the fluxes described in the context of diversion near and in the drift, and flow into the waste package. The figure was developed by someone else (a student) and revised by me based on what I saw in the TPA 5.0 code. Fwet is not included. Fwet is simply a uniform distribution between 0 and 1 as set in the tpa.inp file (SubAreaWetFraction). It does not seem to be correlated to fow, as recommended in Appendix F of TPA 4.0 user manual.

The other parameter that factors into seepage is the new threshold added by Osvaldo Pensado. A temperature threshold for seepage was added to the tpa.inp file; the SeepageThreshold[C] value is currently set to 100 C (above which, no seepage is allowed). I am lobbying to change this parameter from the constant to a sampled distribution between 105 and 125 C.

The most important aspect that could be included in a new seepage abstraction is a linkage of climate change (change in percolation) to changes in seepage fraction. The other thing to do is to re-assess the current values or ranges of the seepage parameters. This is maybe where the Hughson et al (2000) report may be useful.

The Hughson et al. (2000) report I referred to in an earlier email is on the LSN:
http://www.lsnnet.gov/docview.aspx?mode=1&lsn=NRC000003419&ic=1&im=0&sc=8&sm=0
In case that link doesn't work, just search for the title ANALYSIS OF NICHE STUDIES AND DEVELOPMENT OF BASES FOR TOTAL-SYSTEM PERFORMANCE ASSESSMENT SEEPAGE PARAMETERS completed in March 2000.

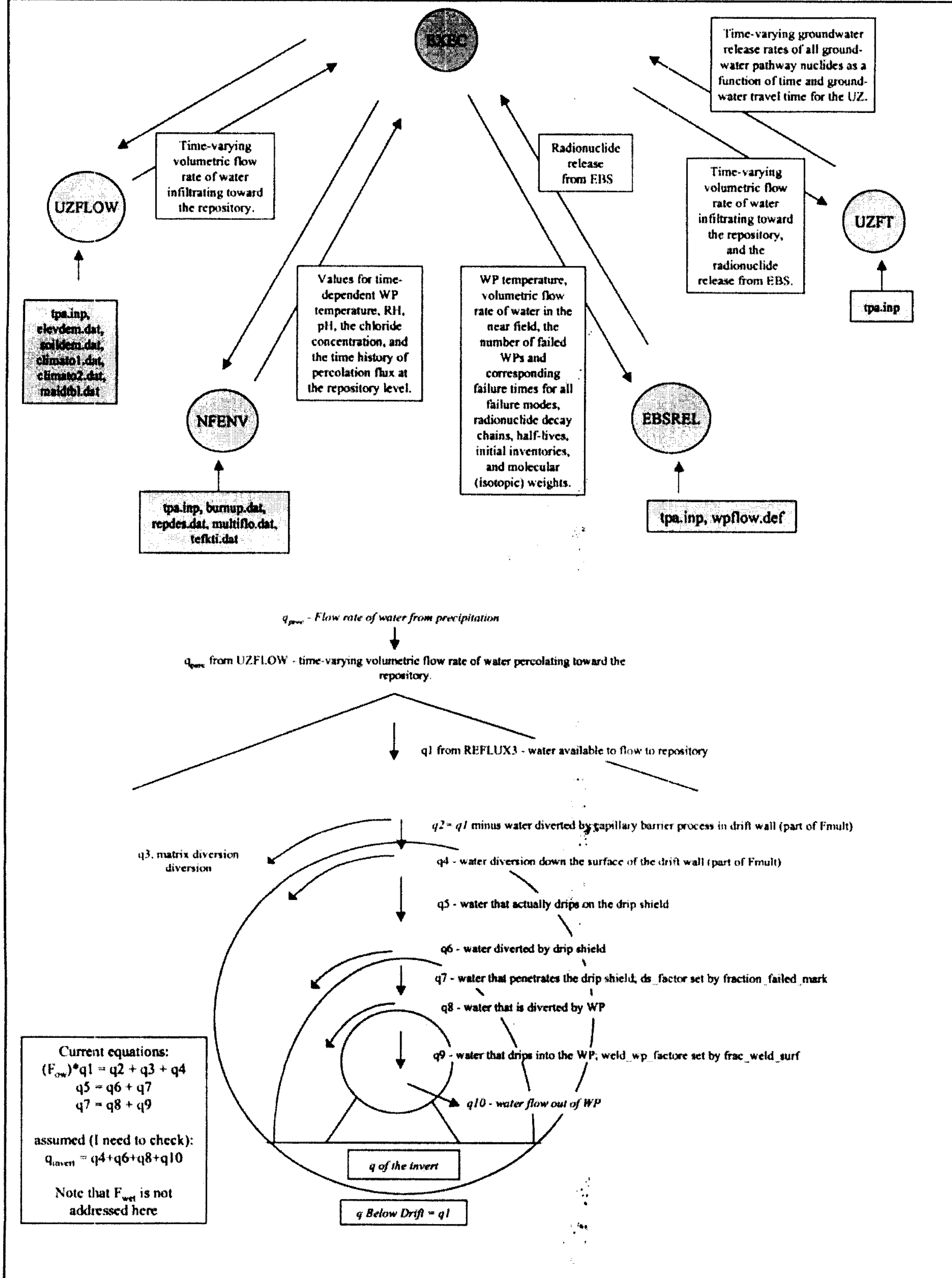


Figure XII-1 Flux tracking in TPA

Some more detail from the code are:

ebsrel.f brings in the factors for releaset.f to use in the equation:

drip = flowref * flowfactr * ctfow * ctfmult * ds_factor * weld_wp_factor

flowfactr: WastePackageFlowMultiplicationFactor in tpa.inp input file is lognormally distributed [3.15e-2, 1.05e+2], which is referred to as flowfractr in ebsrel.f and releaset.f

fow and fmult are read in as time dependent parameters from the external file wpflow.dat. Currently, these are set as 0.173205 and 0.044721 and do not change with time in TPA 5.00f.

ds_factor and weld_wp_factor modify the amount of water contacting the waste by accounting for performance of the drip shield and water missing holes in the waste package, or holes clogged up by corrosion minerals in cracks of the outer layer of the waste package.

The runtime file ebsflo.dat contains the sample flowfactr value for the realization, and the time dependent values of drip rate, fmult, and fow.

==

RF 3/3/05

Here is what we should be shooting for in the seepage abstraction:

Fow is correlated to Q_perc
Q_drift = Q_perc * (Fow + sampled noise) ! account for convergence/diverg
Q_ceiling = Q_drift * (Fmult_1 + noise) ! account for capillary diversion
Q_indrift = Q_ceiling * Fmult_2 ! account for along-wall seeps (film)
Q_WP = Q_ceiling * F_dswp ! account for drip shield & WP diversion
Fwet is correlated to (Fow + noise)

Note that the Fwet should be correlated to Fow, and that it should not be related to the seepage threshold. Correlating Fwet to Fow makes sense because of the "mass balance" concept of converging flow above the drift leading fewer preferential flow paths, thus fewer waste packages getting hit by seeps. Now, what should be sampled? We need to sample noise for Fow and Fmult1, and sample for Fmult2.

The basis for the F factors would include:

Fow correlation would be supported by catchment analysis of Hughson et al. (2000) by assuming different a relation between climate and method for estimating length scale (Cl-36, fault/fracture features, all fractures, calcite/opal occurence); Fmult_1 would be supported by Or et al. (2005); Fmult_2 is entirely sampled (Dick's old range?); Fwet correlation to (Fow + noise) supported by Hughson et al. (2000). What to do with the drip shield and waste package diversions is not entirely clear to me.

RF 3/24/05

Input for Revision of Seepage Model Parameters

Keith Compton and Dick Codell decided that the seepage model revision for March 2004 version of TPA code will include (i) add fwet to wpflow.def; (ii) make fow, fmult, and fwet in the external file dependent on a mean case percolation rate for the million-yr period. So, I need to provide percolation as a function of time and provide the change in fmult correlated to that percolation as a function of time. Keith and Dick will update the fow and fwet as a function of time.

Percolation As a Function of Time – Mean Case

Need to have a mean case percolation rate for a million-year simulation to help revise the wpflow.def file (fow and fmult). To do this, I will use Gary Walter's new climato2.dat file in a TPA 5.00o simulation (using tpameans.dat to get a mean case: run TPA code once, it automatically creates the mean case file tpameans.dat, use this to replace tpa.inp).

spock ; ~rfedors/TPA00o

Had to tweak Gary's new climato2.dat file to include two other considerations:

1. Because the TPA code will still be used to evaluate a performance period of 10,000 years, the climate for the first 10,000 years needs to be consistent with my revision from a few months ago (adjust climato2.dat so that precipitation tracks modern climate for the first 600 years, then monsoonal for 1400 years, and then glacial transition for the remainder of the 10,000-yr performance period
2. Make sure that the time=0 value of net infiltration for the entire repository in infiltper.res TPA output file returns the value specified for the ArealAverageMeanInfiltrationAtStart parameter in tpa.inp when the mode 1 net infiltration model is selected (which is basecase).

The revisions to Gary Walter's climato2.dat file are included below. Gary transmitted this new version of the million-year climato2.dat to Ron Janetzke to replace his previous transmittal. These revisions take care of the two problems noted above.

```
0    0    0
500  0    0
1000 0.364364174  0.364364174
2000 0.50775941   0.50775941
3000 0.586599682  0.586599682
4000 0.632253812  0.632253812
5000 0.655878738  0.655878738
6000 0.655878738  0.655878738
7000 0.655878738  0.655878738
8000 0.655878738  0.655878738
9000 0.655878738  0.655878738
10000       0.655878738 0.655878738
....
```

The resulting net infiltration estimates created using TPA 5.0.0o are shown in the next two figures below (Figures XII-2 and XII-3). The default number of time steps used to create the post-10,000yr data was changed from 100 to 400 step for this figure. The data was extracted out of infilper.res, which reports values for every 10th time step. (Note that 200 time steps are the default setting for the 10,000-yr performance period.

The resulting average for the 1,000,000 years is ~47 mm/yr.
bubo: D:\E_Drive\TEF_kti\TPAstuff\Seepage_KeithCompton\infilper.xls

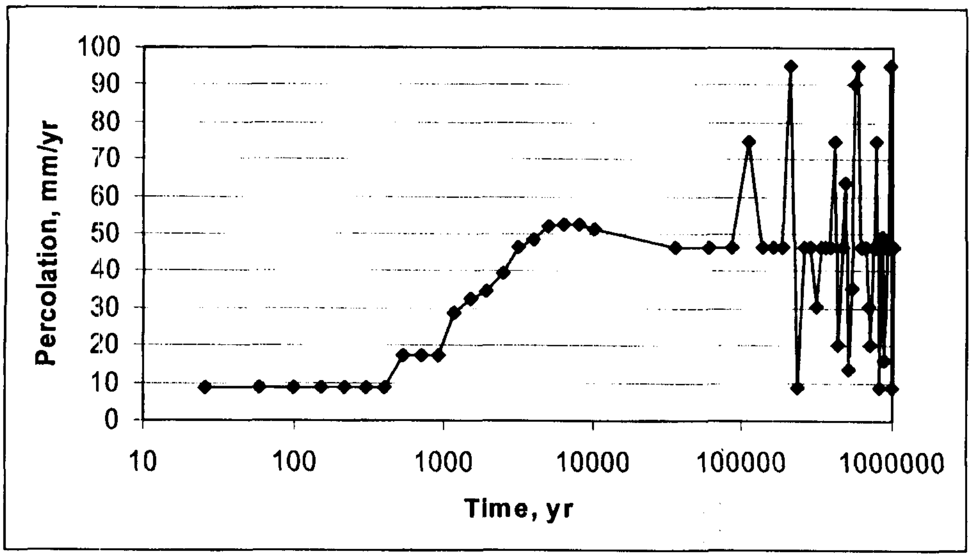


Figure XII-2. Percolation over time using a log10 scale to illustrate first 10,000 yrs.

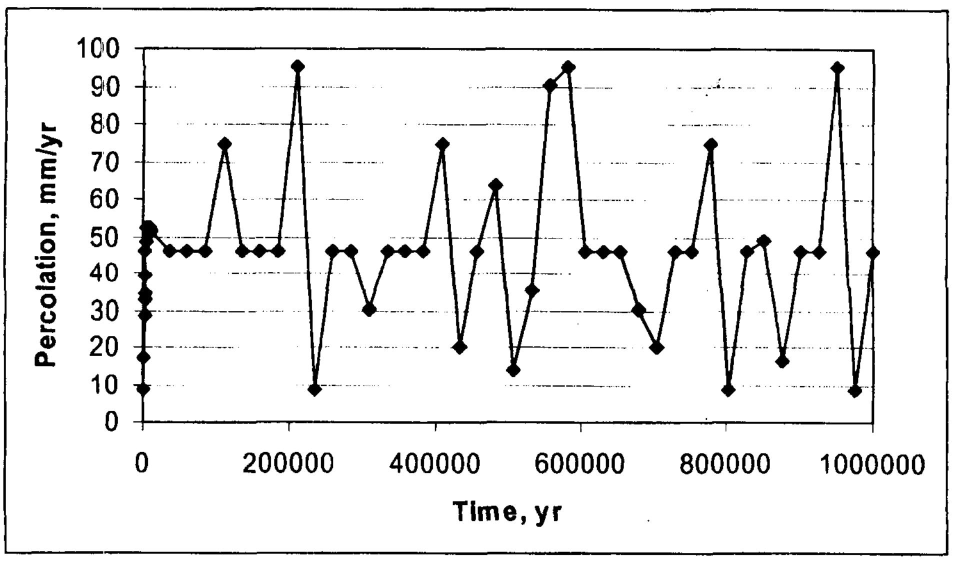


Figure XII-3. Percolation over time using an arithmetic scale.

The spreadsheet with these data and figures is
bubo: D:\E_Drive\TEF_kti\TPAstuff\Seepage_KeithCompton\percolation.xls

RF **04/04/05**

Infiltration Variability

Started with maidtab.dat from 2000, stored in
bubo: D:\E_Drive\Itym_Usage\Repository*
Took codes and maidtab.dat and copied them to
bubo: D:\E_Drive\TEF_kti\TPAstuff\Seepage_KeithCompton\InfiltrationVariability*
to modify and work with.

New subareas taken directly from TPA5.0.0o tpa.inp file.

Assuming a normal distribution for the spatially variable net infiltration, the fortran code called extract.f was used to calculate mean and standard deviation. The input files for the code are drifts.dat and maidtbl.dat. The file drifts.dat was derived from the tpa.inp file by picking points to outline the entire repository (LA design); note that a counter-clockwise order is needed. The maidtbl.dat file must contain a single set of net infiltration values. I used modern (precip=162/mm/yr, temperature=17 C) and two different future climates (precip=400 mm/yr with temperature= 7.3 C and 14 C) to represent a typical future climate (cooler and wetter). The files are named maidtbl_p162_T17.dat containing the precip=162 mm/yr and temperature=17 C. Similar names for other climates are used for the other file names. The 30-m pixel net infiltration data was used. Each maidtbl_*.dat file is copied to maidtbl.dat and the extract.exe executable is run. The output is repository.dat, which are renamed to connote the appropriate climate input.

These files are stored in
bubo: D:\E_Drive\TEF_kti\TPAstuff\Seepage_KeithCompton\InfiltrationVariability*

The fortran code extract.f is:
==========
```
        program extract
c script for determining if a point lies within the repository footprint,
c   or any other odd-shaped outline (rectangles still work).
c RFedors  June 14, 2000
c  bubo:J:\Itym-Usage\Repository\extract.f or bren:~/ITYM-Usage/extract.f
c Modified April 6, 2005 to for License Application Repository
c  bubo:   .\TEF_kti\TPAstuff\Seepage_KeithCompton\InfiltrationVariability
c23456789 123456789 123456789 123456789 123456789 123456789 123456789 12
      integer ioread, iowrit, mx, i, j, k
      integer ndrift, i_max, i_min, ict, lf_rt
      parameter (mx=1000,mxx=100000)
      real*8 ymax, ytop, xbot, xpos, ypos, sum, sum2
      real*8 avg, stdev, avg_t, stdev_t, xsegment, ax,ay
      real*8 drift(mx,2), segment(mx,2)
      real*8 array(mxx,3), repository(mxx), reposit(mxx,3)
      character*9 flagside(mx), junk
      character*60 header
      real*8 xllcorner, yllcorner, cellsize
c Set input and output unit numbers
      ioread = 7
      iowrit = 8
c Read in drift coords file, 1st line comment line, 2nd line # of points
c Counter-clockwise ordering needed.
```

```
c Account for repeated entry of first point as last entry.
c      open(unit = ioread, file = 'drift1.txt', form = 'formatted')
      open(unit = ioread, file = 'driftLA.txt', form = 'formatted')
      read(ioread,'(a60)') header
      read(ioread,'(i5)') ndrift
      do i = 1, ndrift
        read(ioread,'(2f10.2)') drift(i,1), drift(i,2)
      enddo
      close(ioread)
c  set up usage of drift coordinates; checking to right or left of segment;
c  find min and max y-coord, then assign left/right to line segments
      ymax = 0.d0
      ymin = 4.d10
      do i = 1, ndrift-1
        if(drift(i,2).ge.ymax) then
          ymax = drift(i,2)
          i_max = i
        endif
        if(drift(i,2).le.ymin) then
          ymin = drift(i,2)
          i_min = i
        endif
      enddo

      if(i_max.lt.i_min) then
          do i = 1, ndrift-1
            flagside(i) = 'left'
          enddo
          do i = i_max, i_min-1
            flagside(i) = 'right'
          enddo
        else
          do i = 1, ndrift-1
            flagside(i) = 'right'
          enddo
          do i = i_min, i_max-1
            flagside(i) = 'left'
          enddo
      endif
c calculate line segment equations going counter-clockwise;
c segment(i,1)=slope; segment(i,2)=intercept; for horizontal lines,
c set flagside to avoid checking either side of the segment and
c then set denominator to any number just to avoid blowout;
c for vertical lines (xbot=0), set numerator of slope to a small number.
      do i = 1, ndrift-1
        ytop = drift(i+1,2) - drift(i,2)
        xbot = drift(i+1,1) - drift(i,1)
        if(dabs(ytop).lt.1.d-9) then
            flagside(i)='neither'
            segment(i,1) = 1.d0
            segment(i,2) = 1.d0
          elseif(dabs(xbot).lt.1.d-10) then
            segment(i,1) = 1.d0
            segment(i,2) = 0.d0
          else
            segment(i,1) = ytop / xbot
            segment(i,2) = drift(i,2) - (segment(i,1)*drift(i,1))
```

```
        endif
      enddo
      do i = 1, ndrift-1
        print*, segment(i,1), segment(i,2)
      enddo
c read in DEM of infiltration; note that the coordinates of the
c southwest corner of the domain are given in the header, but the
c ordering of data is row-major starting from the northwest corner.
      open(unit = ioread, file = 'maidtbl.dat', form = 'formatted')
      do i = 1, 4
        read(ioread,'(a60)') header
      enddo
      read(ioread,'(a9,i10)') junk, ncols
      read(ioread,'(a9,i10)') junk, nrows
      read(ioread,'(a9,f16.5)') junk, xllcorner
      read(ioread,'(a9,f16.5)') junk, yllcorner
      read(ioread,'(a9,f15.5)') junk, cellsize
      do i = 1, 3
        read(ioread,'(a60)') header
      enddo
      print*, ncols, nrows, cellsize, xllcorner, yllcorner

      ypos = yllcorner + cellsize * dfloat(nrows-1)
      xpos = xllcorner
      k = 1
      do i = 1, nrows
        do j = 1, ncols
          read(ioread,'(e15.8)') array(k,3)
          array(k,1) = xpos
          array(k,2) = ypos
          xpos = xpos + cellsize
          k = k + 1
        enddo
        ypos = ypos - cellsize
        xpos = xllcorner
      enddo
      close(ioread)
c check to see if current position is within repository outline
      lf_rt = 0
      ict = 0
      do i = 1, nrows*ncols
        ay = array(i,2)
        ax = array(i,1)
        do m = 1, ndrift-1
          if(ay.le.drift(m,2).and.ay.gt.drift(m+1,2).or.
     &          ay.ge.drift(m,2).and.ay.lt.drift(m+1,2)) then
           xsegment = (array(i,2)-segment(m,2)) / segment(m,1)
           if(dabs(segment(m,2)).le.1.d-10) xsegment = drift(m,1)
           if(flagside(m).eq.'right'.and.ax.ge.xsegment) lf_rt= lf_rt+ 1
           if(flagside(m).eq.'left'.and.ax.le.xsegment) lf_rt= lf_rt + 1
          endif
          if(lf_rt.eq.2) then
               ict = ict + 1
               repository(ict) = array(i,3)
               reposit(ict,1) = array(i,1)
               reposit(ict,2) = array(i,2)
               reposit(ict,3) = array(i,3)
```

```
                 lf_rt = 0
              endif
            enddo
            lf_rt = 0
         enddo
         print*, ict, 'ict'
c writing out the array() matrix for digestion in arcinfo, which needs 3
columns
c         open(unit = iowrit, file = 'mainDrift.dat', form = 'formatted')
         open(unit = iowrit, file = 'arcview.dat', form = 'formatted')
         do i = 1, ict
            write(iowrit,'(2f12.2,e12.4)') reposit(i,1),
     &               reposit(i,2), reposit(i,3)
         enddo
          close(iowrit)
c statistics on repository cells
c         open(unit=iowrit, file='summary-main.dat', form='formatted')
         open(unit=iowrit, file='repository.dat', form='formatted')
         sum = 0.d0
         do i = 1, ict
            sum = sum + repository(i)
         enddo
         avg = sum / dfloat(ict)

         sum = 0.d0
         do i = 1, ict
            sum = sum + dabs(repository(i) - avg)
         enddo
         stdev = dsqrt(sum/dfloat(ict-1))

         sum = 0.d0
         do i = 1, ncols*nrows
            sum = sum + array(i,3)
         enddo
         avg_t = sum / dfloat(nrows*ncols)

         sum = 0.d0
         do i = 1, ncols*nrows
            sum = sum + dabs(array(i,3) - avg_t)
         enddo
         stdev_t = dsqrt(sum/dfloat(nrows*ncols-1))

         write(iowrit,*) 'Number in Repository = ', ict
         write(iowrit,*) 'Average = ', avg
         write(iowrit,*) 'Std Dev = ', stdev
         write(iowrit,*) 'Number in Modeling Domain = ', nrows*ncols
         write(iowrit,*) 'Average = ', avg_t
         write(iowrit,*) 'Std Dev = ', stdev_t

         stop
         end
===============
```

The net infiltration averages and standard deviations for the 5203 pixels in the repository are:

	Modern (P=162, T=17)	Precip=400 mm/yr, Temperature=7.3C	Precip=400 mm/yr, Temperature=14 C
Average, mm/yr	12.12	77.75	58.80
Standard Deviation	2.313	4.404	4.115

A visual representation of the spatial variability across the LA repository is provided in Figure XII-4 below. ArcView 3.2 was used, the ArcView project file is named itym.apr.

Bubo: E:\E_Drive\AVData\TPA\Infilt_March2005*

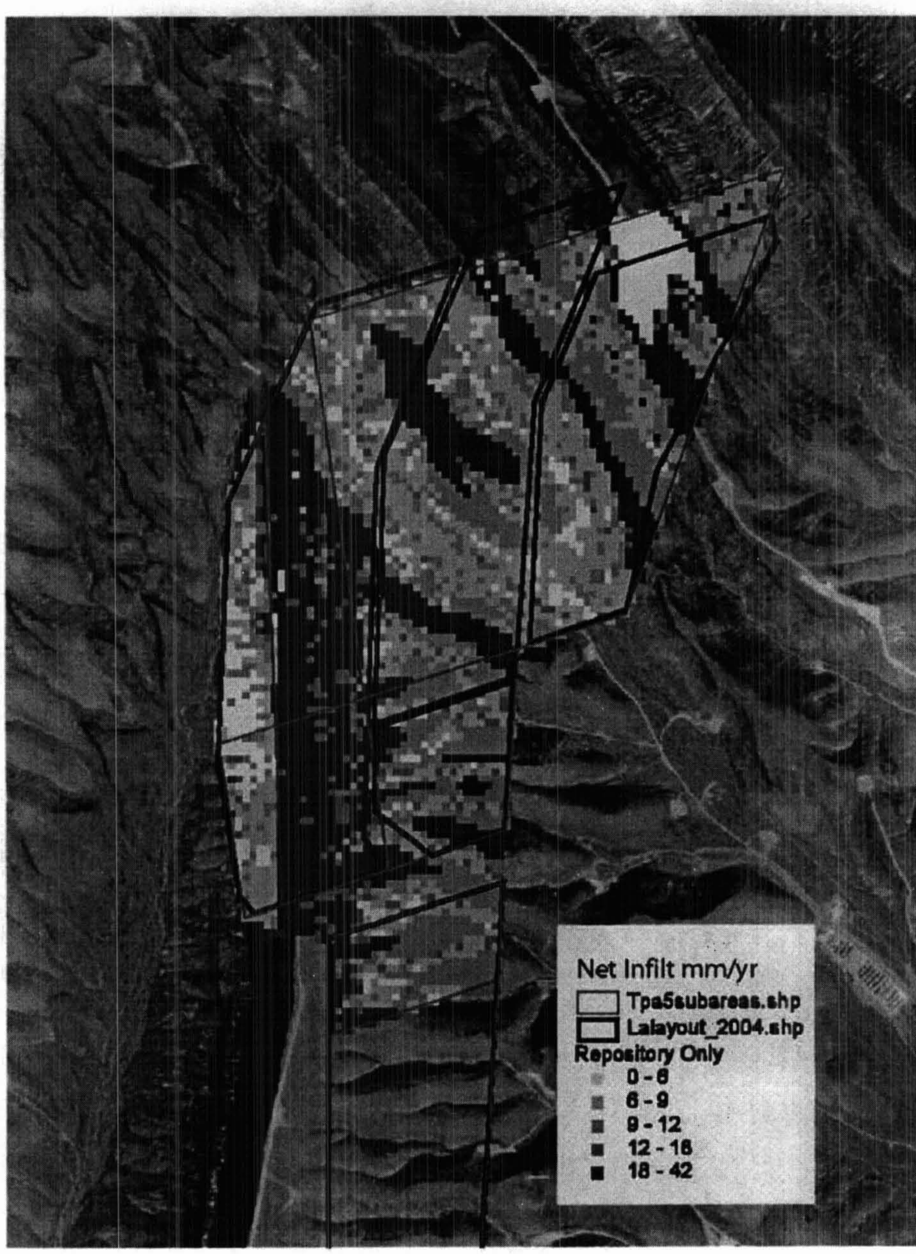


Figure XII- 4 Infiltration Variability Across Subareas

RF 4/8/05

Monte Carlo Simulations for Fmult

Monte Carlo simulations of capillary diversion and along-wall seepage to estimate Fmult

bubo: D:\E_Drive\TEF_kti\TPAstuff\Seepage_KeithCompton\mcarlo_fmult.mb
.\tsw35_sensitivity_March2005.xls (derived from Or et al. seepage paper)
modification in worksheet "VARIATION MATRIX Ksat"
.\percolation.xls (contains infilper.res and updated Fmult as function of percol.)
mcarlo_uncert_fmult.nb Mathematica 4.2 and 5.0

From tsw35_sensitivity.xls (see Or et al. 2005, spreadsheet for Figure 4 of that report), a range of coefficients for a sigmoidal fit to the seepage fraction curves was developed.
The expected curve led to the coefficients for the Or et al. (2005) Figure 4 fit.

y0=a/[1+exp{-(logQ-x0)/b}]

Expected case		Increased Ksat case	
0.98414	= a	0.941222	= a
0.260749	= b	0.173971	= b
0.626281	= x0	1.6478	= x0

A reasonable range for the along-wall seepage fraction for the same three coefficients

0.95 to 0.99
0.2 to 0.35
0.6 to 1.0

These coefficients were used as input to the Mathematica notebook mcarlo_uncert_fmult.nb. The diversion in the rock is directly from the fitted coefficients noted above. To represent the along-wall seepage, I chose another set of sigmoidal curves. To represent along-wall seepage (instead of dripping), I originally thought to use a y-intercept (nonzero, but around 0.10), a maximum (around 0.90) and a slope between these end points. The general shape of the along-wall seepage fraction versus water seeping across the drift wall is consistent with our expectations that along-wall seepage increases with percolation, dominates at low percolation, and is swamped by dripping at higher percolations.

The Mathematica script combines these two processes and then fits a sigmoidal curve to the result. Done in Monte Carlo fashion, a set of sigmoidal coefficients are developed and analyzed. The uncertainty was performed for the range of fraction expected for a typical future climate, i.e., a climate with about 50 mm/yr net infiltration.

Some results from separate monte Carlo runs of multiple realizations that show stability at 1000 realizations are:

	1000 real.	1000 real	1000 real	5000realizations mean	std dev
= a	0.910764	0.910758	0.910673	0.910757	0.021107
= b	0.209292	0.210994	0.211988	0.210392	0.02412
= x0	1.33349	1.35004	1.35982	1.35485	0.301468

The mcarlo_uncert_fmult.nb Mathematica notebook is on the following page.

```
<< Statistics`NonLinearFit`
<< Statistics`DescriptiveStatistics`
```

Monte Carleo 3-parameter sigmoidal equation: y = a / {1 + exp[(x0-logQ)/b] }, where y=seepage fraction, Q is percolation rate. Number of realizations is mreal.

```
x = {0.001, 0.01, 0.05, 0.1, 0.5, 1, 2, 3, 4, 5, 7.5, 10, 15, 20, 25, 30, 35, 40, 45, 50, 55, 60, 65, 70, 80, 90, 100, 125, 150, 175, 200,
   250, 300, 400, 500, 600, 700, 800, 900, 1000};
xlog10 = Log[10, x];
nperc = Length[x];
mreal = 1000

1000
```

Range of sigmoidal eqn coefficients to consider

```
a1 = Table[Random[Real, {0.97, 0.99}], {mreal}];
b1 = Table[Random[Real, {0.17, 0.26}], {mreal}];
x01 = Table[Random[Real, {0.62, 1.9}], {mreal}];
a2 = Table[Random[Real, {0.9, 0.97}], {mreal}];
b2 = Table[Random[Real, {0.2, 0.4}], {mreal}];
x02 = Table[Random[Real, {0.4, 1.0}], {mreal}];
```

The frac Table must be in correct format for input to NonlinearFit routine ==> {{x1,y1},{x2,y2},{xn,yn}}
To get into this matrix format, explicit referencing is needed (i.e., xlog10[[i]]). Combine (multiply) the two equations (one for capillary diversion, one for along-wall seepage) when generating the frac matrix. Use sigmoidal for both equations to simplify the calculations (avoids if statements for intercept, slope, max model for along wall seepage). Open up an empty array to store the results, use AppendTo instead of Append (the latter replaces??).

```
results = {};
Do[
  frac = Table[{xlog10[[i]], a1[[j]] * a2[[j]] / (1 + Exp[(x01[[j]] - xlog10[[i]]) / b1[[j]]]) / (1 + Exp[(x02[[j]] - xlog10[[i]]) / b2[[j]]])}, {i, 1, nperc} ];

  tempcoef =
   {atop, bmp, xtmp0} /.
    (BestFitParameters /. NonlinearRegress[frac, atop / (1 + Exp[(xtmp0 - xxxx) / bmp]), {xxxx}, {atop, bmp, xtmp0},
       RegressionReport -> BestFitParameters]);
  AppendTo[results, tempcoef];
  , {j, mreal}];
```

Capture statistics of coefficients, results[row,col], mreal => rows, different coefficients=>columns

```
acoef = {};
bbcoef = {};
x0coef = {};
Do[
  AppendTo[acoef, results[[k, 1]]];
     AppendTo[bbcoef, results[[k, 2]]];
     AppendTo[x0coef, results[[k, 3]]]
  , {k, mreal}];
```

Mean and Std Dev of a

```
Mean[acoef]
StandardDeviation[acoef]

0.912817

0.0204098
```

Mean and Std Dev of b

```
Mean[bbcoef]
StandardDeviation[bbcoef]

0.209747

0.0239327
```

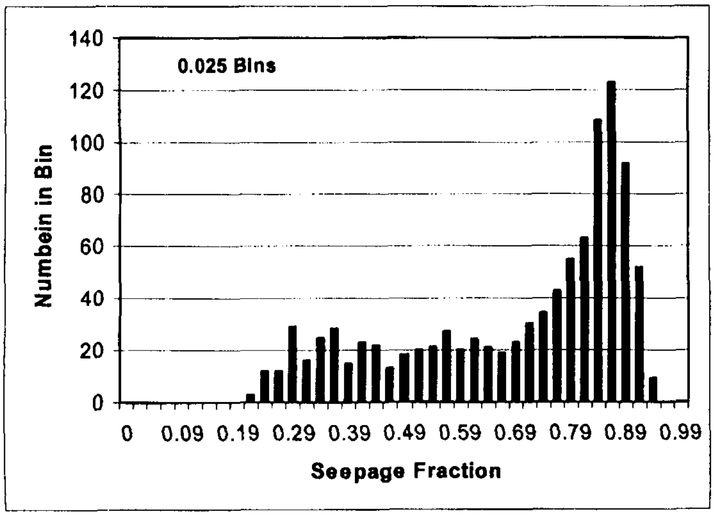
Mean and Std Dev of x0

```
Mean[x0coef]
StandardDeviation[x0coef]

1.36351

0.299429
```

Get estimate of uncertainty at typical percolation for future climate, ~50 mm/yr, and print to file to determine distribution type.

```
filebase = "D:/E_Drive/TEF_kti/TPAstuff/Seepage_KeithCompton/";
filename = "perc50.txt";
outfile = StringJoin[filebase, filename];
Export[outfile, acoef / (1 + Exp[-(Log[10, 50] - x0coef) / bbcoef]), "List"];
```

The output of the mcarlo_uncert_fmult.nb for 1000 realizations was imported into the Excel spreadsheet called perc50.xls. Figure XII-5 is the histogram of the distribution of results for a percolation rate of 50 mm/yr.



~~Figure XII-1~~ Fig XII-5 / RL 4/11/2008

From the mean case of the 1000 realizations, the mean values of the coefficients of the sigmoidal fit from the Mathematica notebook results were then used to develop the fmult values to put in the external TPA file wpflow.def. Keith Compton provided the fwet and fow values to the wpflow.def file. All of these seepage parameters are now a function of percolation rate, via the temporal variation of the mean case climate.
The calculation of the mean case fmult values over time was done in percolation.xls

Precip mm/yr	Fmult
0	0.11
26	0.11
58	0.11
99	0.11
150	0.11
215	0.11
297	0.11
400	0.11
530	0.34
693	0.34
900	0.34
1160	0.57
1487	0.62
1900	0.64
2420	0.69

3076	0.74
3903	0.76
4945	0.77
6258	0.78
7914	0.78
10000	0.77
34750	0.74
59500	0.74
84250	0.74
109000	0.84
133750	0.74
158500	0.74
183250	0.74
208000	0.87
232750	0.11
257500	0.74
282250	0.74
307000	0.59
331750	0.74
356500	0.74
381250	0.74
406000	0.84
430750	0.40
455500	0.74
480250	0.82
505000	0.24
529750	0.65
554500	0.86
579250	0.87
604000	0.74
628750	0.74
653500	0.74
678250	0.59
703000	0.40
727750	0.74
752500	0.74
777250	0.84
802000	0.11
826750	0.74
851500	0.76
876250	0.31
901000	0.74
925750	0.74
950500	0.87
975250	0.11
1000000	0.74

RF 3/24/05

INFILTRATION MAP - CHECK ON RIDGES NORTHEAST OF DRILL HOLE WASH

Objective

Need to check for possible errors in bedrock units at northeast extent of new LA subareas in TPA 5.00 code. This area looked suspicious, but had not been researched because the previous repository footprint (EDA-II design) did not reach that far north; the LA design repository footprint does include areas on the northeast trending ridges northeast of Drill Hole Wash.

Evaluation

While these looked like possible errors in ITYM and external files, I determined that the low net infiltration on Tonsil, Azreal, and Mile High Mesa ridgetops was due to the vitrophyre exposure (which is not present on Yucca Mountain crest in the repository footprint). Stuart Stothoff, who created these external files, had grouped the vitrophyres from the Tlva Canyon and the upper Topopah Springs into one grouping called "tc" or unit 8 in bunitdem.dat. The properties assigned to unit 8 are consistent with those of the "TC" unit of the Topopah Spring Tuff in Flint (1998; USGS Water-Resources Investigations Report 97-4243). The assignment of stratigraphic layers in Flint (1998) to the bunitdem.dat entries are shown in the figure below. The hydrologic properties in itym.dat, the input file for ITYM module use the short name from Flint (1998) except where aggregation of units occurred. Fifteen stratigraphic layers are given hydrological properties in itym.dat, but only 11 are present in bunitdem.dat; units 2, 10, 11, and 13 are not used.

Checked permeability values for CUC or unit 9. In Table 7 of Flint (1998), the CUC entry for geometric mean of measured hydraulic conductivity is
3.8e-08 m/s * [100 cm/m] * [1.019e-5 cm^2/(cm/s)] = 3.87e-11 cm^2
log10(3.87e-11 cm^2) = -10.41 which is the entry in itym.dat for unit 9 CUC

Loaded the maidtbl.dat, bunitdem.dat, soildem.dat, sunitdem.dat along with the new subareas for the LA repository footprint into ArcView 4.2. The external files were reformatted for input to ArcView (added 2 columns, easting and northing in meters) and renamed to *.txt files. The script used to reformat the TPA files to ArcView format is called dem.for, and is included below. Next, read the *.txt files into ArcView as tables (in the Project menu system), then Add as event theme in View menu system. I had to use ArcView on Petrel in the GIS lab because IMS forgot to install ArcView 3.2 on the new Windows machine put into my office last week while I was on travel. The paths will need to be changed in the itym.apr ArcView project file before the project will automatically load. The completed files were transfers from a location where Petrel could access them to the directory on my new Window machine (still called bubo):

Bubo: D:\E_Drive\AVData\TPA\Infilt_March2005* itym.apr, Bedrock View

Conclusion, net infiltration values on ridges northeast of Drill Hole Wash are not in error, they are expected because of the change in surficial bedrock unit (see Figure XII-6 (pixel assignments for bedrock).

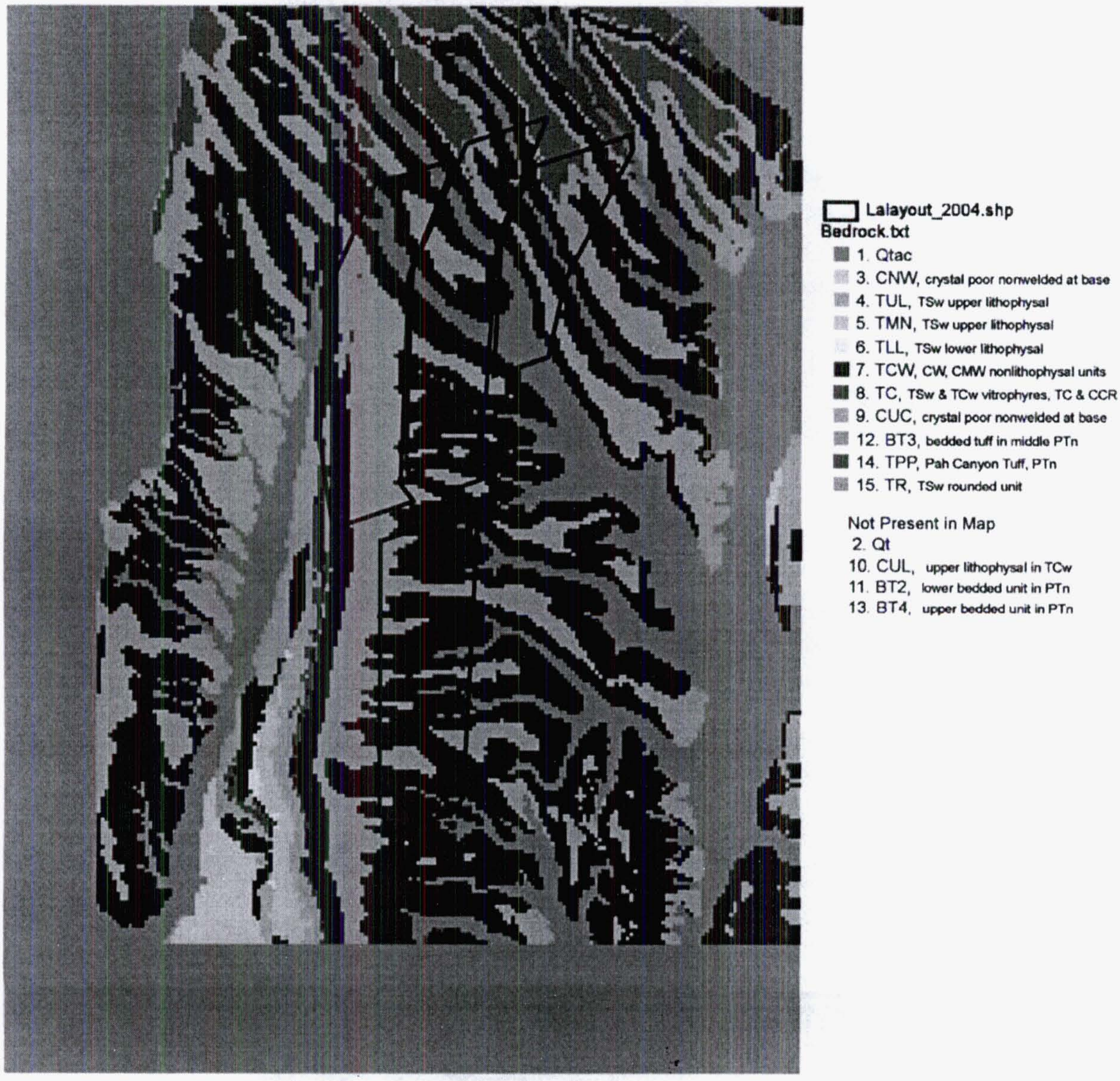


Figure XII-6 Pixel assignments for geology in bunitdem.dat

The old fortran script used to convert the TPA external files into a format that ArcView can read in as a table is called dem.for:

```
c     Last change:  RWF   6 Jun 2002    5:51 pm
        program dem
c Script reformats ITYM external data for input to ArcView in grid format
c
c RFedors  June 4, 2002
c
c23456789 123456789 123456789 123456789 123456789 123456789 123456789 12
      implicit  none
      integer ioread, iowrit, mxx, i, j, k, nrows, ncols
      parameter (mxx=200000)
```

```
      real*8 array(mxx,3),  xpos, ypos
      REAL*8 rxllcorner, ryllcorner, rcellsize
      integer xllcorner, yllcorner, cellsize
      character*12 file1, file2, fvar, junk
      character*60 header
      character*1 comment

c set input and output unit numbers
      ioread = 7
      iowrit = 8

c read in DEM of infiltration; note that the coordinates of the
c southwest corner of the domain are given in the header, but the
c ordering of data is row-major starting from the northwest corner.

      write(*,1010)
 1010  format(' enter input filename ')
      read(*,'(a12)') file1
      write(*,1013)
 1013  format(' enter output filename ')
      read(*,'(a12)') file2
      write(*,1016)
 1016  format(' enter dependent variable ')
      read(*,'(a12)') fvar

      open(unit = ioread, file = file1, status = 'unknown')

c Note that Stofhoff used 2 or 4 comment lines and flip-flops the
c order of listing NROWS and NCOLS
      k = 0
      do i = 1, 4
         read(ioread,'(a1,a60)') comment, header
         if(comment.ne."N")   k = k+1
      enddo
      rewind(ioread)

      do i = 1, k
          read(ioread,'(a60)') header
      enddo
      read(ioread,'(a5,i10)')  junk, nrows
      if(junk.eq."NROWS") then
         read(ioread,'(a5,i10)') junk, ncols
       else
         ncols = nrows
         read(ioread,'(a5,i10)') junk, nrows
      endif
      read(ioread,'(a9,i10)') junk, xllcorner
      read(ioread,'(a9,i10)') junk, yllcorner
      read(ioread,'(a9,i10)') junk, cellsize
      read(ioread,'(a60)') header
      print*, ncols, nrows, cellsize, xllcorner, yllcorner

      rxllcorner = dfloat(xllcorner)
      ryllcorner = dfloat(yllcorner)
      rcellsize = dfloat(cellsize)

      ypos = ryllcorner + rcellsize * dfloat(nrows-1)
      xpos = rxllcorner
      k = 1
      do i = 1, nrows
        do j = 1, ncols
          read(ioread,*) array(k,3)
          array(k,1) = xpos
          array(k,2) = ypos
          xpos = xpos + rcellsize
          k = k + 1
        enddo
        ypos = ypos - rcellsize
        xpos = rxllcorner
      enddo
      close(ioread)
```

```
c write out reformatted data including easting and northing locations
      open(unit=iowrit, file=file2, status='unknown', form='formatted')
c      open(unit=iowrit, file='maidtbl.txt', form='formatted')
      write(iowrit,1050) fvar
      do k = 1, nrows*ncols
         write(iowrit,1080)  array(k,1), array(k,2), array(k,3)
      enddo

 1050  format(' easting,', ' northing, ', a12)
 1080  format(e16.7,",",e16.7,",",e16.7)

      close(iowrit)

      stop
      end
```

end topic RF **3/24/05**

RF 3/30/05

UPDATING LAYER THICKNESSES FOR TPA VERSION 5.0.0 – LA REPOSITORY

Collaborators: **Shannon Colton** **(Vitric and zeolite thicknesses in Calico Hills)**
Paul Landis **(All other layer thicknesses)**

The TPA code requires UZ transport layer thicknesses for each subarea. In addition, a distribution representing uncertainty of vitric/zeolite spatial variation is needed. Shannon and Paul provided the following description of the source data and how the final thicknesses were derived. They generally followed the approach previously used by RFedors (SciNtbk #227), though improvements in the approach and updated data sources were incorporated into the analyses.

Files provided by Paul and Shannon are stored in:

bubo: D:\E_Drive\TEF_kti\TPAstuff\VitricThickness_Mar2005\slc_notebook\
.\VitricThickness_March2005\ByHorizon_MM3.1*

Support for New Subarea Delineation

The license application repository footprint was incorporated into TPA 5.0.0 and new subareas were developed. Winterle assembled data from and help delineate the subareas in a rational manner so as to avoid excessive smoothing or averaging (thus, better representing variability between subareas). Net infiltration (from TPA) and vitric (DOE) maps in ArcView format were provided to Winterle.

The infiltration and the zeolite were considered the two most important factors for delineating new subareas for the license application repository design to be used in TPA Version 5.0.0.

Net Infiltration map previously extracted from TPA 4.2 was modified and sent to Winterle. The color palette was manipulated to bring out the differences in net infiltration between ridges and valleys. The license application repository footprint was also added to the ArcView 3.2 project file

bubo: D:\E_Drive\AVData\TPA\Infilt_March2005\itym.apr Infiltration View

Figure XII-7 (Infiltration figure) below was provided to Winterle.

Shannon also extracted zeolite percentages for each layer of the Calico Hills hydrostratigraphic unit from the DOE Mineralogic Model Version 3.1 (MM3.1). Figure XII-8 through XII-13 contain the zeolite maps for each of the Calico Hills hydrostratigraphic horizons. The illustrator file were provided by Shannon and are stored in

bubo: D:\E_Drive\TEF_kti\TPAstuff\VitricThickness_Mar2005\ByHorizon_MM3.1*

Figure XII-7. Net infiltration map for subarea delineation.

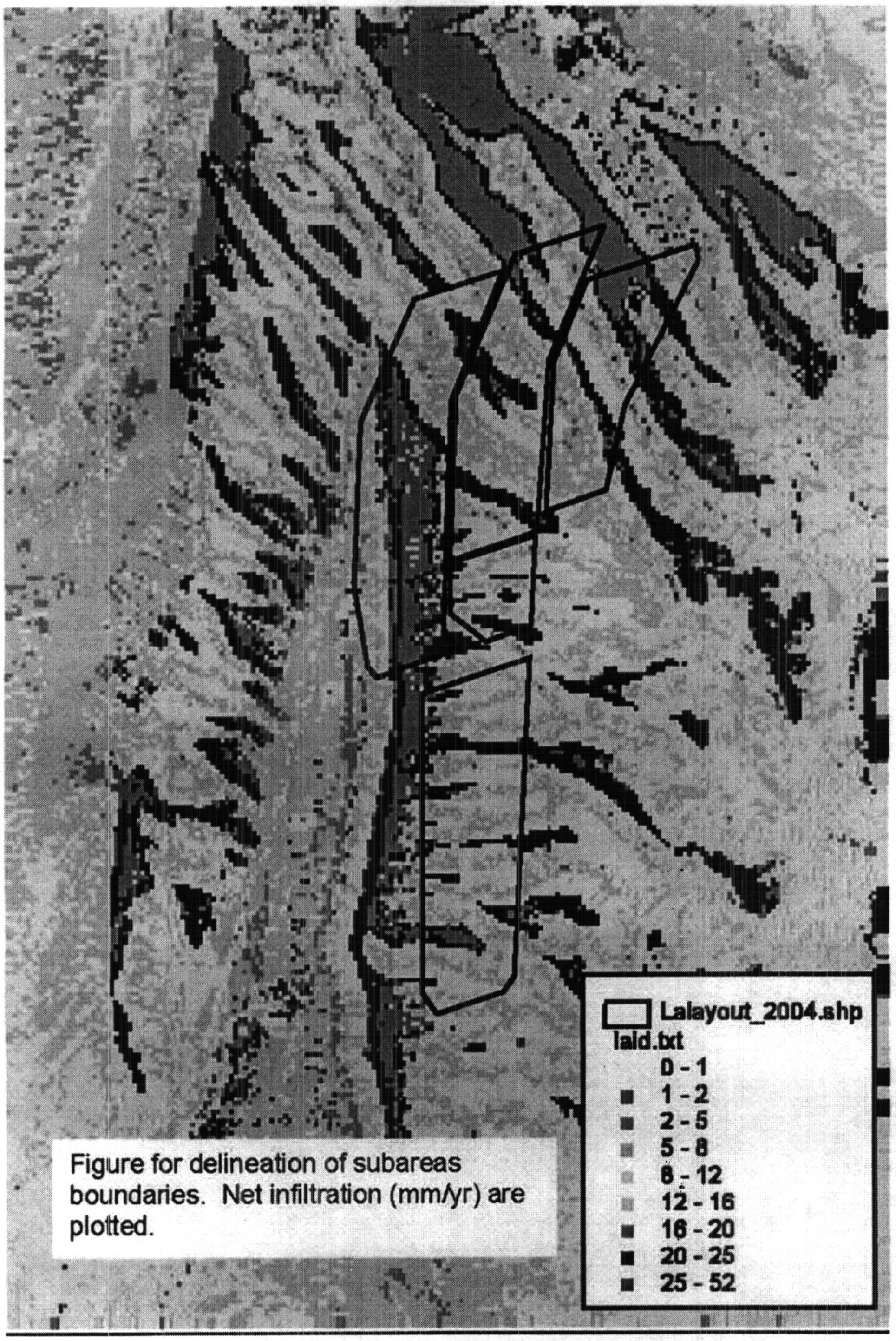


Figure XII-8. Zeolite percentage extracted from MM3.1 for Tptpv and bt1

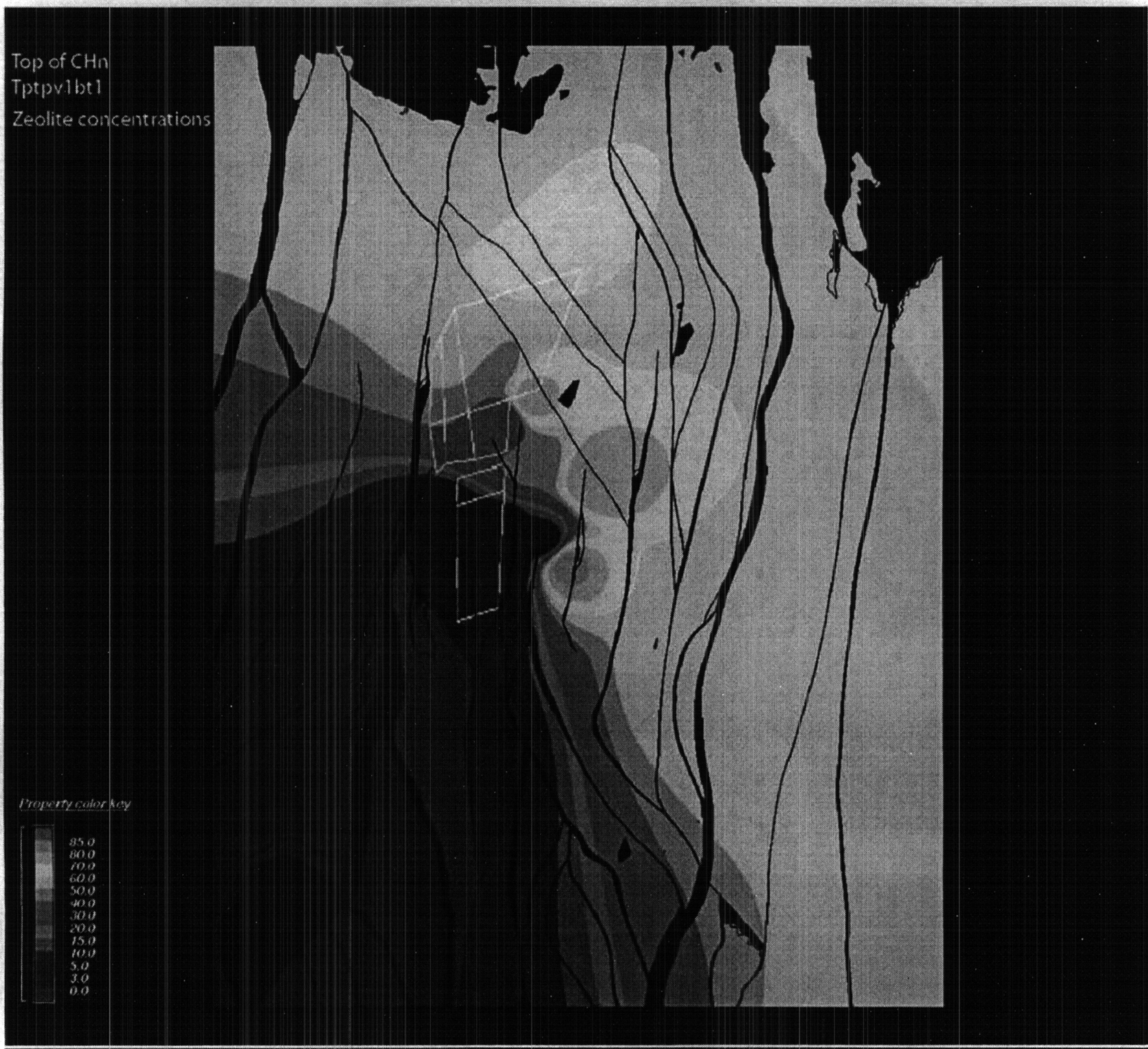


Figure XII-9. Zeolite percentage extracted from MM3.1 for Tac1 in CHn

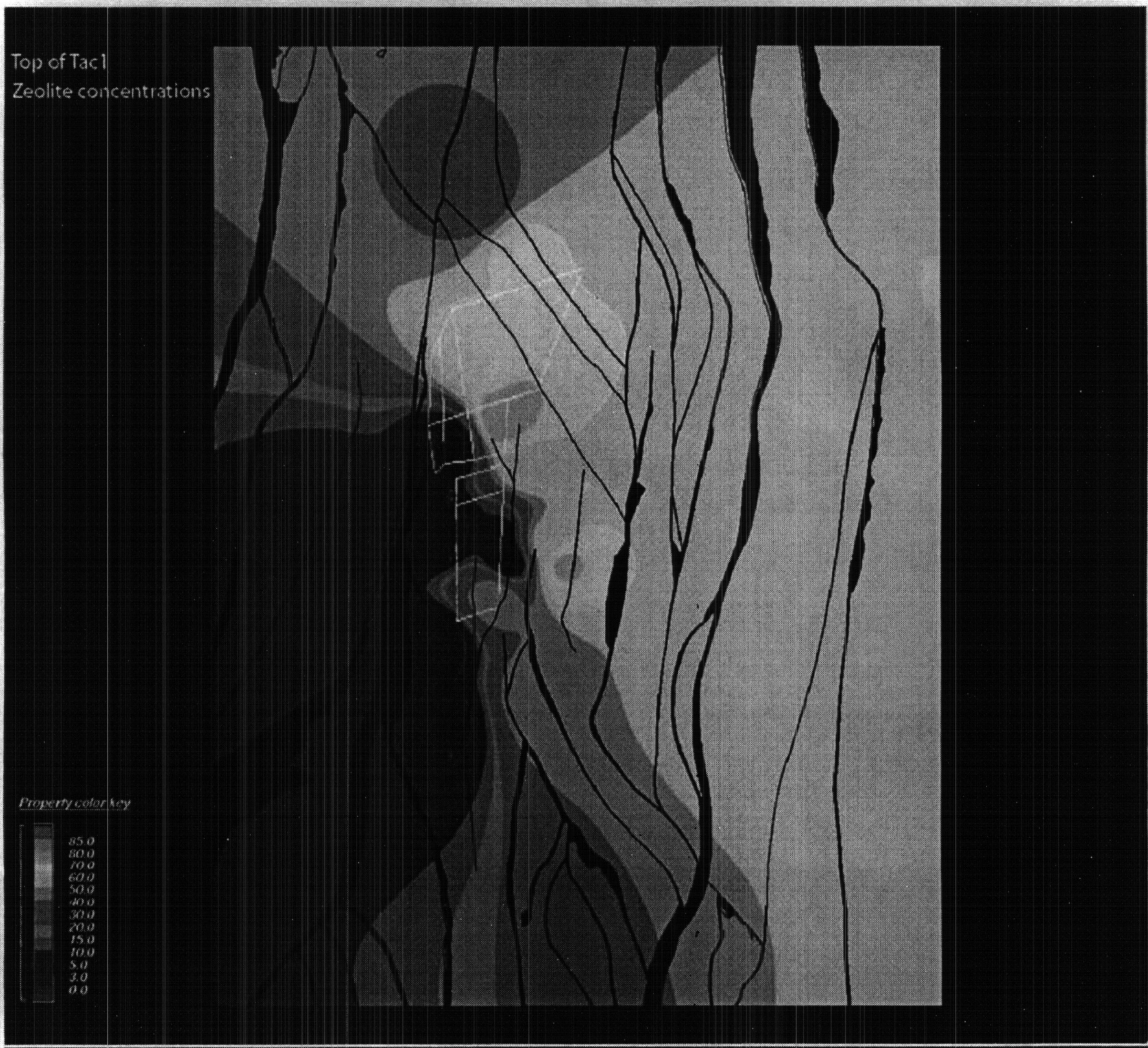


Figure XII-10. Zeolite percentage extracted from MM3.1 for Tac2 in CHn

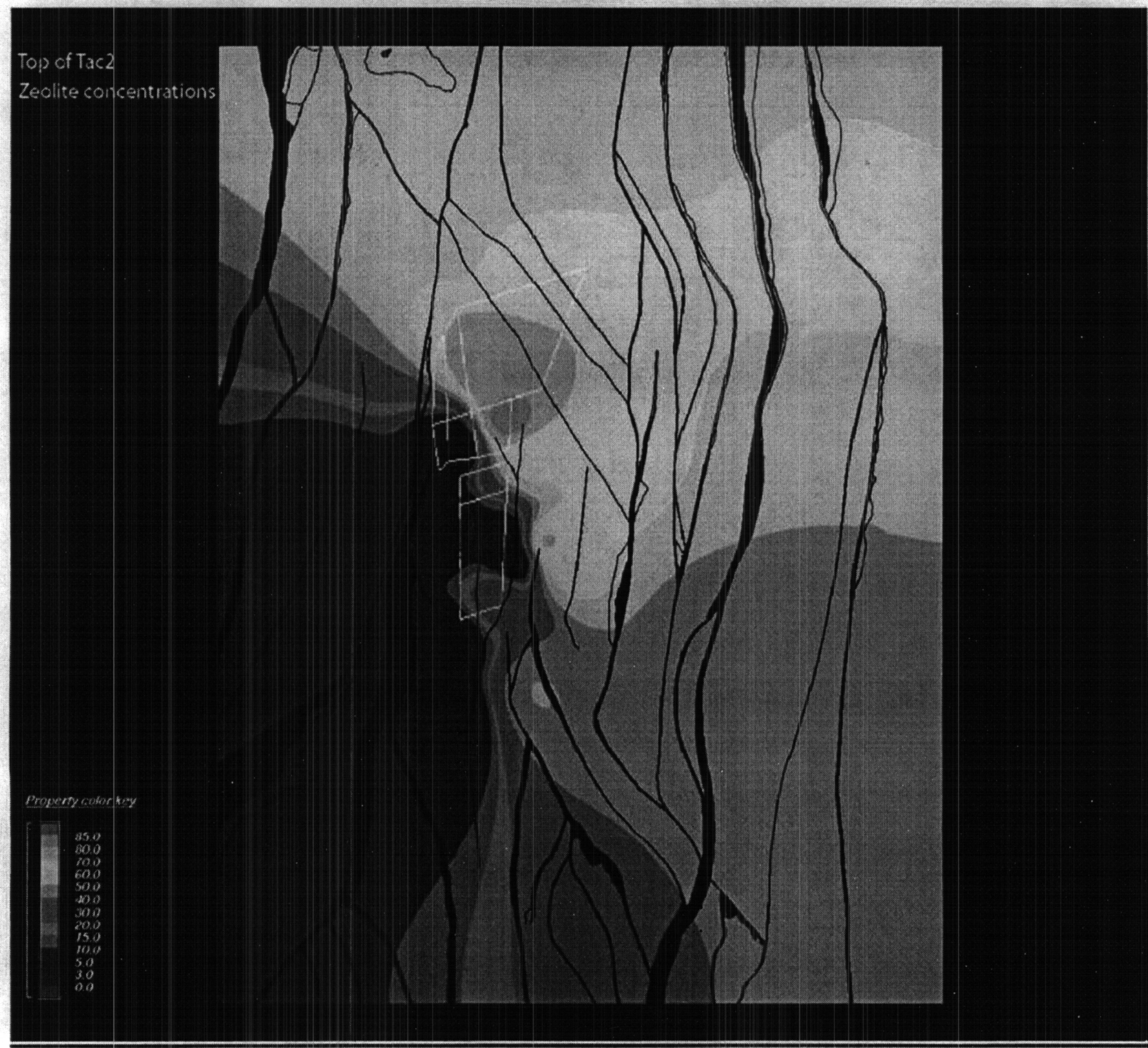


Figure XII-11. Zeolite percentage extracted from MM3.1 for Tac3 in CHn

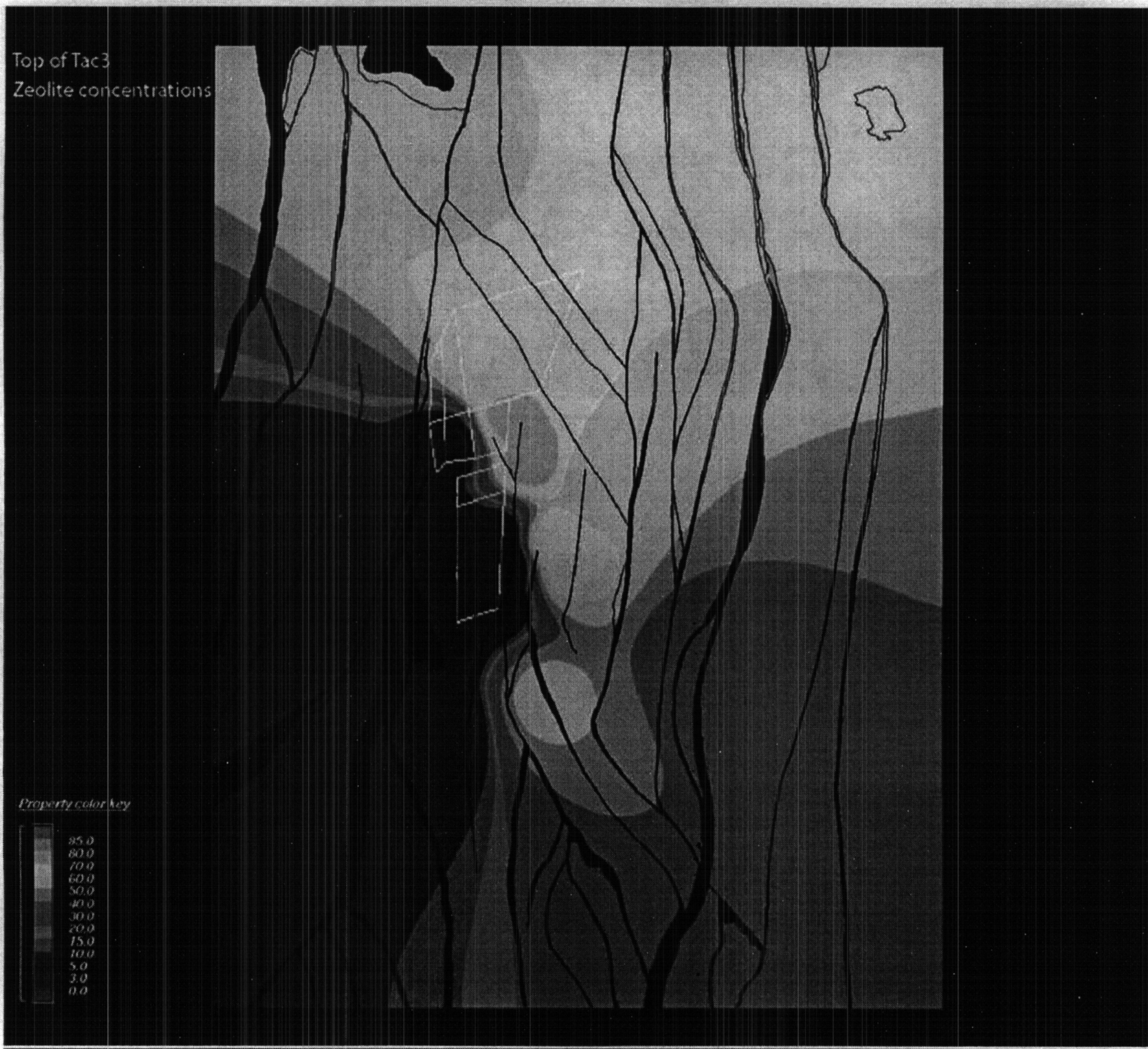


Figure XII-12. Zeolite percentage extracted from MM3.1 for Tac4 in CHn

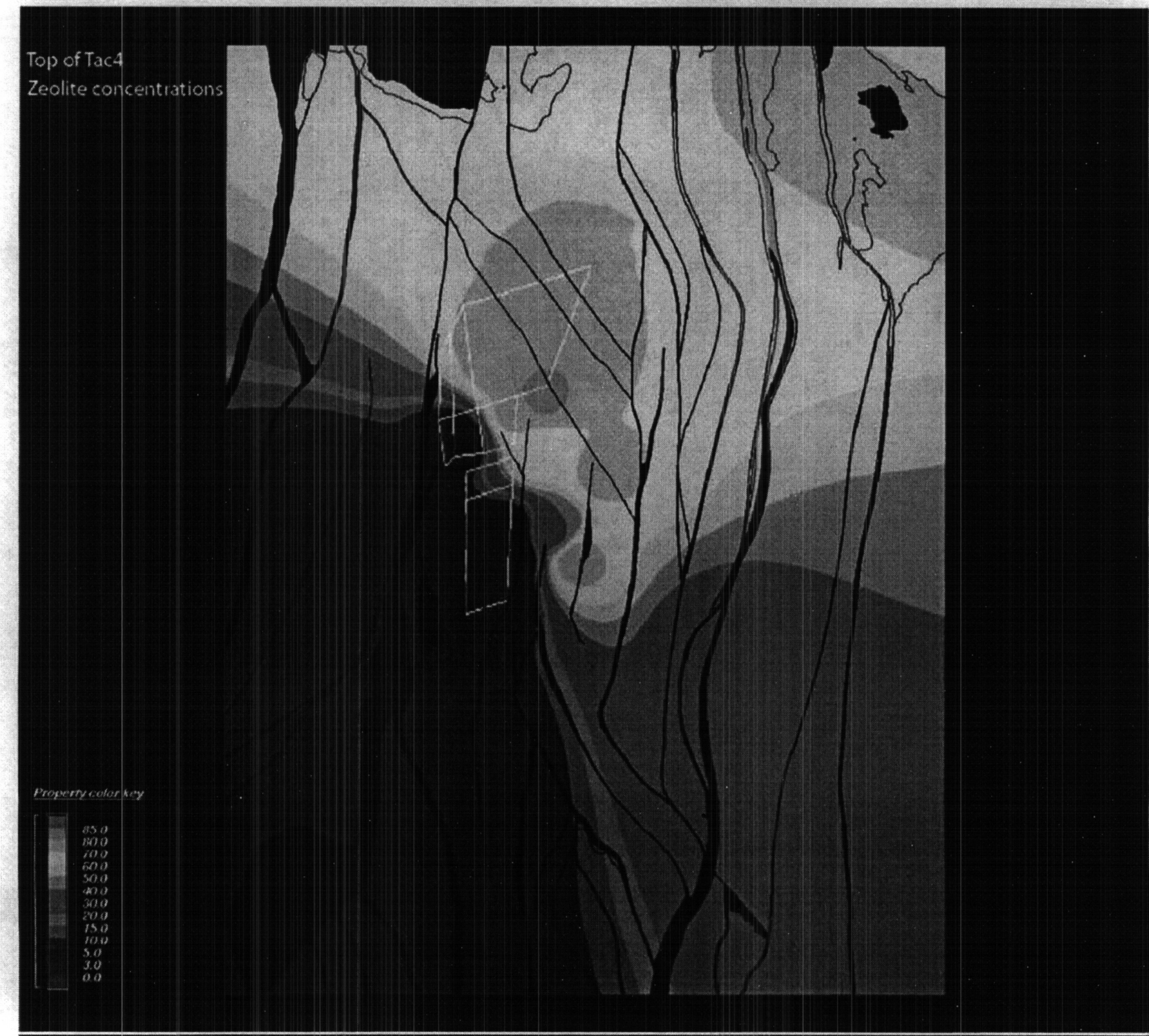
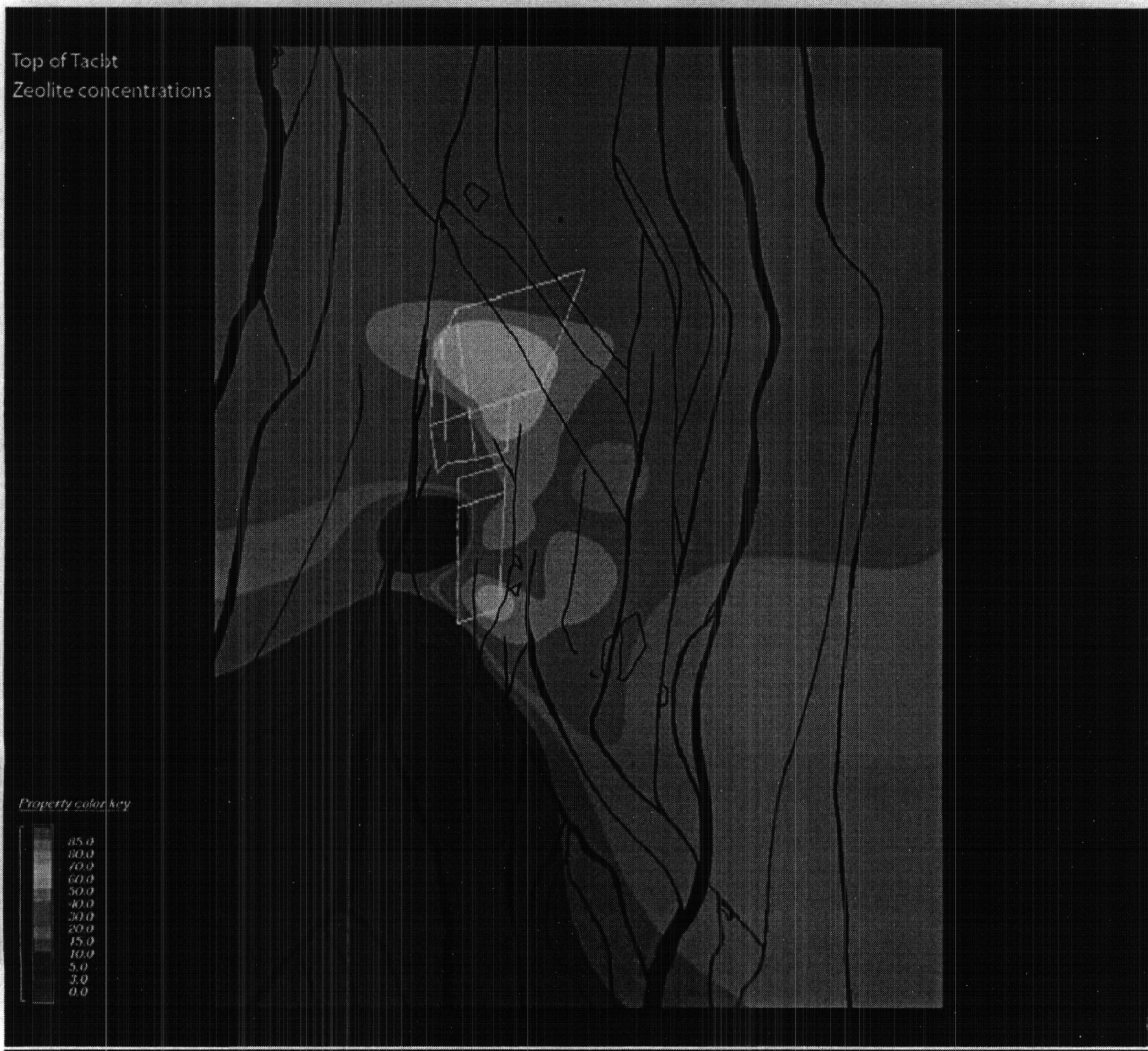


Figure XII-13. Zeolite percentage extracted from MM3.1 for Tacbt in CHn



UZ Layer Thicknesses for Subareas

Objectives – Thickness of All Layers in TPA 5.0.0, Constant Values (Paul Landis)

The following discussion outlines the various steps undertaken to revise the unsaturated zone (UZ) stratigraphy for the 2005 TPA update. Data was obtained from forty synthetic wells in Earthvision 7.5. All analyses for layer thicknesses, unless otherwise noted, were conducted by Paul Landis.

Codes
Earthvision v7.5 was used to obtain stratigraphic thicknesses from the GFM 3.1. ArcGIS 9.0 was used in creating the center nodes and calculating the drift elevation. All computer codes, utilities and programs noted in the text are commercially available and maintained in accordance with CNWRA Technical Operating Procedure TOP-018

Elevation of the Drift
The first step in revising the UZ stratigraphy was to spatially locate the points that define the repository subareas. The current repository layout (OCRWM, 2003) was divided into eight subareas based on different geologic and hydrologic characteristics. Jim Winterle defined the repository subareas and the coordinates of the 32 bounding points that define the subareas as quadrilaterals. In order to obtain representative subarea stratigraphic thicknesses for TPA units, a center node for each subarea was defined in ArcGIS 9.0.

For the purposes of performance assessment, we are only interested in the unsaturated zone from the drift elevation in the repository layout to the water table. The drift elevation and elevation of the water table are unknown, but can be calculated in ArcGIS 9.0 and Earthvision 7.5, respectively. OCRWM (2003) contains the elevations of all the proposed repository drifts and an attached ArcGIS registered shapefile of the repository layout. The layout shapefile and the subarea bounding and center points were imported into ArcGIS 9.0. Because the majority of the subarea bounding points do not lie directly on a drift, the elevation of the drift at each point was calculated by assuming a linear change in elevation between two adjacent drifts.

Elevation of the Water Table
The elevation of the water table at each point was calculated in Earthvision 7.5. A .2grd file of the water table (winterle_wt_wells_sp_ft.2grd) created by Darrel Sims was used to extract the elevation of the water table using the back interpolate function in the Earthvision formula processor. The file **UZwells.dat** contains the spatial coordinates, drift elevation and elevation of the water table at each subarea point.

Unsaturated Zone Stratigraphy
The UZ stratigraphy was extracted from the GFM 3.1 in Earthvision 7.5 using the well-structure query utility. Specifically, I used the high-resolution, unsliced faces file of the GFM 3.1 to determine the elevation of the tops of stratigraphic beds. The input file used was **UZwells.dat**. The output file contains elevations of the tops of the different lithostratigraphic units. Stratigraphic thicknesses were obtained by importing the output file into Microsoft Excel and subtracting the tops of units from one another. The unit thicknesses were then combined based on the TPA lithostratigraphic groupings in Table 1 (TPA UZ transport units and ....). A representative stratigraphic thickness for each subarea was calculated in Excel by taking the mean of each group in an individual subarea. The average thicknesses were checked in Earthvision to ensure a unit was not substantially thicker or thinner across an area. If a stratigraphic thickness did change across a subarea, the overall thickness was changed

accordingly. The file **TPAstrat.xls** is a Microsoft Excel spreadsheet containing thickness values for TPA UZ transport units at each point as well as mean thicknesses for each subarea.

Table 1. TPA UZ transport units and the corresponding lithostratigraphic units and DOE UZ model units

TPA UZ Transport Unit	Lithostratigraphic Unit	DOE UZ Model Unit	Description
TSw	Tptpmn, Tptpll, Tptpln, Tptpv3	tsw34 – tsw38	Welded
CHnv	Tptpv2,1, Tpbt1, Tac, Tacbt	tsw39, ch1-6	nonwelded vitric
CHnz	Tptpv2,1,Tpbt1, Tac, Tacbt, Tcpuv	tsw39, ch1-6	nonwelded zeolitic
Prow Pass	Tcpuc, Tcpmd, Tcplc	pp3, pp2	Welded
Upper Crater Flat	Tcplv, Tcpbt, Tcbuv, Tcblv, Tcbbt, Tctuv	pp4, pp1, bf2	moderately welded, devitrified or zeolitic
Bullfrog	Tcbuc, Tcbmd, Tcblc	bf3	Welded
Unsaturated Fault Zone	-	-	fault or intensely fractured zone

Results – Layer Thicknesses for Subareas

Table 2 (Subarea Thickness for LA Design Repository Footprint) below contains the thicknesses recommended for TPA Version 5.0.0. Note that subarea 8 is the contingency area. Entries in bold have used expert judgment to obtain a representative value that reflects the entire area of the subarea (e.g., faults may bias the five-point averages). The thickness at each of the five points for a subarea and the interpreted averages for each subarea are contained in TPAstrat.xls, which was obtained from Paul and stored as -

Bubo: D:\E_Drive\TEF_kti\TPAstuff\VitricThickness\slc_notebook\TPAstrat.xls

Table 2. Subarea Thickness for LA Design Repository Footprint

Subarea	TSw	CH	Prow	Upper Crater	Bullfrog
1	53.12	108.01	43.15	44.80	**0.00**
2	72.03	125.70	37.24	**15.00**	-
3	111.35	157.38	**10.00**	-	-
4	67.96	89.17	47.74	52.19	**15.00**
5	112.56	101.22	45.73	37.15	**3.00**
6	170.08	119.28	31.74	**3.00**	-
7	140.40	104.03	46.24	**33.00**	**5.00**
8	116.48	95.96	57.50	58.26	**15.00**

Reference

Office of Civilian Resource Waste Management (OCRWM). Underground Layout Configuration. 800-P0C-MGR0-00100-000-00D. 2003.

Vitric/Zeolitic Thicknesses for Subareas

Objectives – Vitric and Zeolite Thicknesses (Shannon Colton)

Zeolitic distribution data between drift locations and the water table was obtained from the Mineralogic Model (MM) 3.1 to assist in the development of TPA input parameters. Data was obtained at 5 points (4 polygon nodes plus a center point) per TPA subarea, and from those points minimum, maximum and average values were obtained. All analyses, unless otherwise noted, were performed by Shannon Colton.

Codes

EarthVision® 7.5 (on a Sun Microsystems SunFire V880Z server, Solaris 9) was used to obtain data from the MM3.1. EarthVision is commercially available software that is maintained in accordance with CNWRA Technical Operating Procedure TOP-018. All other codes used in this report are standard versions of commonly used commercial codes including Microsoft® Excel (2000-SP3). All shell scripts mentioned here use the Bourne shell.

Subarea Coordinates

Updated coordinates defining new TPA subareas were provided to by Jim Winterle, from which Paul Landis created an EarthVision polygon file (repository.ply), and Shannon created an EarthVision annotation file (repository.ann) and separate EarthVision polygon files for each subarea (named p1.ply-p8.ply). Landis also created center points for each subarea. Subarea polygons and center points are shown in Figure XII-14 (TPA Subareas) and coordinates are listed Table 3 (TPA Subareas) below.

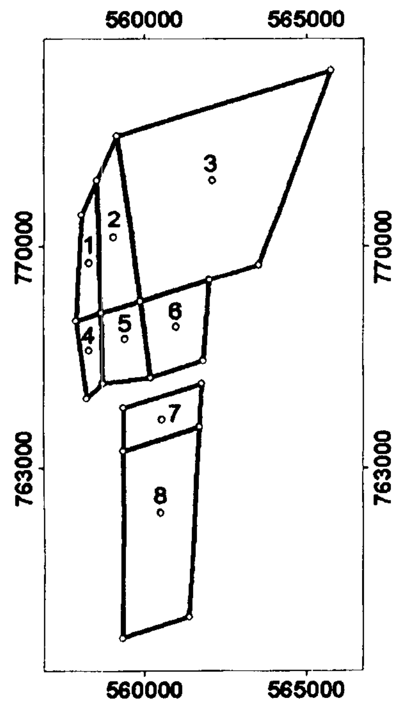


Figure 14. TPA Subareas. State Plane, NAD27

Table 3. TPA Subareas. State Plane, NAD27

Subarea	Easting	Northing	Subarea	Easting	Northing
1	558020	771032	5	558639	767893
1	558500	772130	5	559865	768273
1	558639	767893	5	560188	765859
1	557864	767646	5	558710	765661
1*	558272	769497	5*	559400	767075
2	558500	772130	6	559865	768273
2	559141	773542	6	562004	768944
2	559865	768273	6	561821	766402
2	558639	767893	6	560188	765859
2*	559036	770296	6*	560971	767472
3	559141	773542	7	559354	764894
3	565761	775597	7	561782	765677
3	563521	769415	7	561715	764308
3	559865	768273	7	559346	763532
3*	562087	772132	7*	560557	764523
4	557864	767646	8	559346	763532
4	558639	767893	8	561715	764308
4	558710	765661	8	561389	758307
4	558216	765200	8	559316	757625
4*	558283	766711	8*	560508	761571

*** indicates center points**

Zone groupings

Based on the grouping provided by Randy Fedors and the information in "file-info.txt" from the MM3.1, the 28 zones of MM3.1 were divided into groups as shown in Table 4 (Zone Groups for Zeolite Distribution Calculations) below.

Table 4. Zone groups for zeolite distribution calculations

Zone numbers in MM3.1	**Zone names in MM3.1**	**Group name for this entry**
17-20	Tptpv23, Tptpln, Tptpll, and Tptpmn	TSw
16	Tptpv2	Tptpv2
15	Tptpv1bt1	CHn_Tptpv1bt1
14	Tac4	Tac4
13	Tac3	Tac3
12	Tac2	Tac2
11	Tac1	Tac1
10	Tacbt	Tacbt
9	Tcpuv	Tcpuv
8	Tcpuclc	Prow
7	Tcplvuv	UCF_top
6	Tcbuclc	Bullfrog
5	Tcblvuv	UCF_bottom
1-4	Paleozoic, Tund, Tctlvbt, and Tctuclc	Unsat_fault_zone

Drift Elevation

Landis provided the elevation of the drift at each polygon node and center. The scattered data file, drift_elev.dat, was used to create a 2D trend grid (drift_elev.2grd) with the shell script drift_elev.2grd.sh.

Water Table

An EarthVision grid of the water table (winterle_wt_wells_sp_ft.2grd) created by Darrell Sims was used. The header was edited to specify that the z coordinates are in feet. Landis backinterpolated the water table elevation at each polygon node and center point from the winterle_wt_wells_sp_ft.2grd using the EarthVision formula processor. The file "points.dat" contains Easting, Northing, drift elevation, and water table elevations for each of the points.

Virtual Well File

Landis provided a well file called "tpawells.dat" that defines 40 virtual wells (4 nodes and 1 center point per subarea polygon). The wells are vertical and range from the interpolated water table elevation to the calculated drift elevation.

Calculating Zeolite Distributions

Fedors requested the distribution of zeolite within the Calico Hills for each subarea expressed as the percentage of rock by volume that is greater than or equal to 15% zeolite by weight. The classification into low and significant zeolite content has been used before and reflects the bimodal distribution of zeolite from XRD analyses. While no DOE reports were found that addressed the topic of how permeability varies with zeolite content (see BSC, 2003; OCRWM, 2000a,b; Loeven, 1993), staff judgment is that 15 zeolite is a reasonable threshold for changing hydrologic properties from those for vitric to those for zeolitic. The 15% delineation leads to an areal delineation for vitric CHn that is consistent with that seen in the DOE grid (BSC, 2003) for UZ transport. Several steps were involved in calculating the zeolite distributions. For each of the 40 points from points.dat (4 nodes and 1 center point per subarea polygon) zeolite content and thicknesses were determined from the MM3.1. This data was then used in an Excel spreadsheet to calculate overall zeolite distribution of the Calico Hills.

<u>Zeolite Concentrations of Each Zone</u>

As noted in the README.txt file accompanying the MM3.1, the "real model" is in the property data (pdat) files and is incorporated into EarthVision for display purposes only. For this reason, the property data files were used to extract zeolite concentrations of each zone. This was accomplished by writing and using a series of shell scripts: 01_getfromMM31.sh, 02_addheader.sh, 03_2dgrids.sh and 04_bakint.sh. The first shell script copied x, y, and weight percent zeolite data for each of the 28 zones in the MM3.1. The second script added an EarthVision header to each of those 28 files. The third script gridded each scattered data file. The fourth script estimated based on these grids the zeolite percentage at each of the 40 points from points.dat. When gridding the data, I used the same number of x and y grid nodes and the same x and y ranges as were used for 3D grid calculations from the MM3.1 (200 x 200 feet spacing; 186 x 246 grid nodes; xrange = 547000 to 584000; yrange = 738000 to 787000) based on gridALL.sh of the MM3.1). The output "points_zeolite.dat" was imported into the Excel workbook "zeolite_tpa.xls."

<u>Thicknesses of Each Zone</u>

The thicknesses of each zone at the 40 points were obtained by using EarthVision's Well Stucture Query with inputs of tpawells.dat and ISM31zeolite.unsliced.faces using the shellscript zeolite.path.sh. The output "zeolite.path" was also imported into the Excel workbook "zeolite_tpa.xls."

<u>Vitric/Zeolite Distribution in Calico Hills Zones for Each Subarea</u>

These calculations were performed with Excel using the workbook "zeolite_tpa.xls." Any zeolitic concentrations greater than or equal to 15% were flagged as 1 for zeolitic, while those with less than 15% were changed to 0 for vitric. For each zone, the thickness times zeolite concentration (1 or 0) was calculated. The zeolitic thickness was calculated by summing the thickness time zeolite values for each zone. Finally, the zeolitic thickness of the Calico Hills (between the drift elevation and water table) was divided by the total thickness of the Calico Hills (between the drift elevation and water table) to get the percent zeolite. The results are shown in Table 5 (Length Percent Zeolites at Each Point) and Table 6 (Minimum, average and maximum zeolite and vitric rock per subarea) below.

Table 5. Length Percent Zeolites at Each Point				
Subarea	Point ID	Zeolitic Thickness	Total Thickness	Percent Zeolitic
1	1	376.044	376.044	100
	2	436.107	436.107	100
	3	123.462	324.303	38.06995
	4	114.671	297.469	38.54889
	5	328.56	345.538	95.0865
2	6	436.107	436.107	100
	7	535.699	535.699	100
	8	359.982	371.769	96.82948
	9	123.462	324.303	38.06995
	10	387.275	387.275	100
3	11	535.699	535.699	100
	12	758.461	758.461	100
	13	329.375	349.652	94.20081
	14	359.982	371.769	96.82948
	15	534.88	558.659	95.74356
4	16	114.671	297.469	38.54889
	17	123.462	324.303	38.06995
	18	122.909	292.745	41.985
	19	81.81	269.059	30.40597
	20	115.828	292.703	39.57185
5	21	123.462	324.303	38.06995
	22	359.982	371.769	96.82948
	23	139.4	345.885	40.30241
	24	122.909	292.745	41.985
	25	128.025	333.846	38.34852
6	26	359.982	371.769	96.82948
	27	399.208	411.877	96.92408
	28	409.007	417.518	97.96153
	29	139.4	345.885	40.30241
	30	397.197	405.496	97.95337
7	31	73.352	290.297	25.26792
	32	388.036	397.128	97.71056
	33	374.331	390.033	95.97419
	34	0	271.6	0
	35	329.851	344.168	95.84011
8	36	0	271.6	0
	37	374.331	390.033	95.97419
	38	158.996	342.297	46.44972
	39	0	248.052	0
	40	62.18	326.182	19.06298

Table 6. Minimum, average and maximum zeolite and vitric rock per subarea (% by length).			
Subarea		CH Zeolite %	CH Vitric %
	Minimum	38	0
1	Average	74	26
	Maximum	100	62
	Minimum	38	0
2	Average	87	13
	Maximum	100	62
	Minimum	94	0
3	Average	97	3
	Maximum	100	6
	Minimum	30	58
4	Average	38	62
	Maximum	42	70
	Minimum	38	3
5	Average	51	49
	Maximum	97	62
	Minimum	40	2
6	Average	86	14
	Maximum	98	60
	Minimum	0	2
7	Average	63	37
	Maximum	98	100
	Minimum	0	4
8	Average	32	68
	Maximum	96	100

References for Zeolite/Vitric Thicknesses

BSC. Development of Numerical Grids for UZ Flow and Transport Modeling. ANL-NBS-HS-000015 Rev01. Bechtel SAIC Company, LLC. 2003.

OCRWM. Mineralogical Model (MM3.0). MDL-NBS-GS-000003 Rev00 ICN01. Office of Civilian Radioactive Waste Management. 2000a.

OCRWM. Rock Properties Model (RPM3.1). MDL-NBS-GS-000004 Rev00 ICN01. Office of Civilian Radioactive Waste Management. 2000b.

Loeven, C. A Summary and Discussion of Hydrologic Data from the Calico Hills Nonwelded Hydrogeologic Unit at Yucca Mountain, Nevada. LA-12376-MS, Los Alamos National Laboratory. 1993.

RF 3/30/05

REFLUX3 PARAMETERS

1. NEW NAMES AND SUPPORTING BASES: LIQUID VOLUME PARAMETERS (POROSITY, INITIAL AND RESIDUAL SATURATION)

2. GRADIENT OF TEMPERATURE AS A FUNCTION OF TEMPERATURE

New Names for Liquid Volume Parameters & Supporting Basis for Values

Brandi Winfrey removed reflux1 and reflux2 modules from the TPA code at the direction of Sitakanta. Reflux3, however, used reflux2 parameters for estimating the volume of water available for feeding preferential flow breaching the dryout zone. Descriptions, values, and supporting basis were needed by TSPAI staff for the updated user guide.

I used the model described in Sci Ntbk 432 Volume VII (development of 3-D model for evaluating edge cooling effect and for estimating temperature gradients to impose on CFD models of gas phase movement in drifts). This is the model referred to in the table below; for the ambient saturations, I looked at the initial saturation conditions in the 2D model (e.g., see directory ~rfedors/Metra/3D-Expanded/2Dgrid/Heat_31/. See also my summary of LA design hydrologic properties in worksheet "StratData" of

bubo: D:\E_drive\TEF_kti\ColdTrap\3D_MetraModelingSept2004\grid_sept2004.xls

Name	Description	PDF Type Value(s)	Comments
Reflux3Porosity	Porosity of rock in dry-out zone (TSw)	constant 0.14	Value is consistent with calibrated hydrological proptery used in the Multiscale Thermohydrological Model AMR (2004) for the matrix and fractures of the Tptpll unit, which comprises 75% of the LA design repository drifts. Fracture porosities are about 0.01 and matrix porosity values for the other repository units are 0.155 (Tptpul), 0.111 (Tptpmn), and 0.103 (Tptpln).
Reflux3SatInit	Initial saturation of dry-out zone rock (TSw)	constant 0.9	Value is consistent with representative saturation of matrix in staff MUTLIFLO models using LA design calibrated hydrological properties (Calibrated Properties AMR, 2003). Ambient liquid saturations for each repository unit are (i) lower Tptpul (0.81 to 0.95), (ii) Tptpmn (0.9 to 0.95), (iii) Tptpll (0.85), (iv) Tptpln (0.88). Note that Tptpll comprises 75% of repository drifts.
Reflux3SatResid	Residual saturation of dry-out zone rock (TSw)	constant 0.0	Whereas ambient residual matrix saturations are between 0.12 and 0.2 for the LA design repository, staff thermohydrological modeling indicates that the matrix essentially dries out when temperatures are above ~98 to 102 C. These thermohydrological models account for capillarity and for transient effects. Note that boiling of free water at the repository horizon occurs at about 96.3 C. The value represents the entire dryout zone and falls on the conservative side when temperatures are decreasing and approaching the boiling point.

lf 3/30/05

Temperature Gradient as a function of Temperature

Collaborator: Osvaldo Pensado (uncertainty) and Brandi Winfrey (TPA coding)

Objectives

Reflux3 was modified to include a more realistic representation of preferential flow breaching a thick dryout zone. Previously in TPA code, the temperature gradient in O.M. Phillips solution was not a function of temperature, however, we know that the temperature gradient is small near the reflux zone and increases in magnitude as the drift wall is approached (when large dryout thicknesses are present). For small dryout zones, the representative gradient in temperature should be small and the drift wall temperature should also be low. For thick dryout zones, the drift wall temperature and representative gradient in temperature would both be larger than for the small dryout zone thicknesses. Inherent in this supposition is that the temperature gradient used by O.M. Phillips is a representative gradient of the ambient thermal system (ambient, meaning no finger of water impinging on the dryout zone).

TPA code estimates a temperature at the drift wall, thus providing a means to link values of temperature gradient to the thermal environment at the current time step.

Phillips, O.M. Infiltration of a Liquid Finger Down a Fracture into Superheated Rock. Water Resources Research, 32(6): 1665–1670. 1996.

Method

To get drift wall temperature as a function of representative temperature gradient in the dryout zone, I used the results of a 2-D thermohydrological numerical model. The model is described in Sci Ntbk 432 Volume VII (development of 3-D model for evaluating edge cooling effect and for estimating temperature gradients to impose on CFD models of gas phase movement in drifts). Hence, the work here only entails extraction of appropriate temperature data from the previously completed Metra simulations. A repository center location was chosen for the 2-D model, and it was referred to as the Heat_31 location.

Metra results at different times (different files) were imported into an Excel spreadsheets and temperatures at appropriate nodes were organized into a summary worksheet. There are two spreadsheets, one with results using vapor phase lowering and another spreadsheet for results without vapor phase lowering.

see spock: ~rfedors/Metra/3D-Expanded/2Dgrid/*
- ./gradT_fit.JNB
- ./Heat_31_GradT.xls
- ./Heat_31vpl_GradT.xls
- ./Heat_31 (Metra results with no vapor pressure lowering)
- ./Heat_31/MorePrintOuts/* (more times printed with no vapor pressure lowering)
- ./Heat_31vpl (Metra results with vapor pressure lowering)

Nodal distances from the drift wall were estimated using the Mathematica 5.0 notebook used to generate the node locations for Amesh in the original grid development (Sci Ntbk Volume VII). The distances are:

node	cumulative distance, m
153	0.241
237	0.897
154	1.553
238	2.481
155	3.409
239	4.993
156	6.577
240	9.066
157	11.555
158	14.723
159	19.249

The Mathematica notebook for the node locations is
Spock: ~rfedors/Metra/3D-Expanded/Nodes1/PlaceGrid_24Sept2004.nb

Boiling point was assumed to be 96.3 C, which is appropriate for the repository elevation of about 1100 m above sea level. Choices were made on calculating gradients when the node at the drift wall was near the boiling point. Generally, a temperature near, but possibly slightly below the boiling point, was still used to estimate the temperature gradient

Results were developed for simulations with and without the vapor phase lowering option in Metra to get a sense of the variability caused by conceptual models (albeit, a small difference in conceptual models). A better illustration of variability would include variations in thermohydrological properties and heat load, as well as, spatial variations (edge-cooling effect).

To get an idea of how the temperature gradient varies within a dryout zone, Figure XII-15 below shows curves for different times. Gradually, the curves converge and the gradients shrink as the dryout zone rewets.

TL 4/11/2008

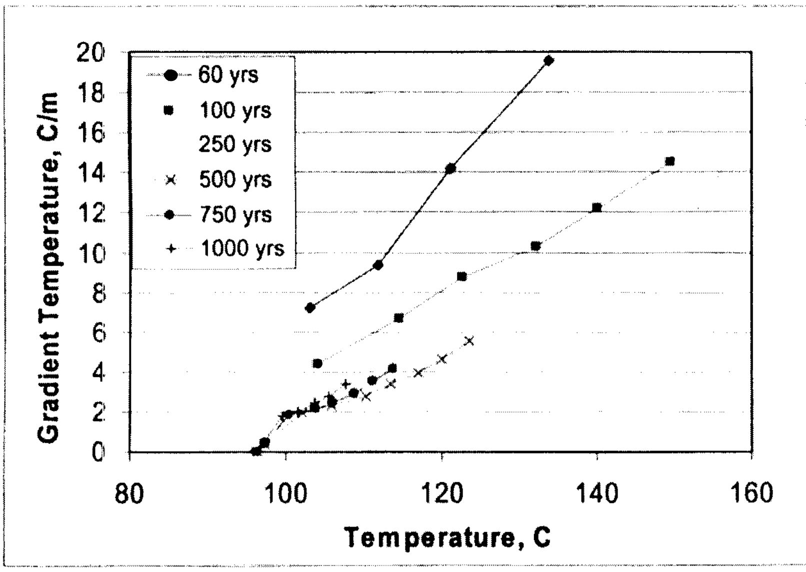


Figure XII- 15 Temperature Gradients at Different Locations in the Dryout Zone; Curves for Different Times. No Vapor Pressure Lowering Option in Metra

To derive a function between drift wall temperature and gradient and temperature, a representative gradient was calculated for each time. SigmaPlot was then used to fit the representative gradient to the drift wall temperature using a 3-parameter exponential equation

$$y = y0 + a\exp(bx)$$

where y is the fitted gradient in temperature and x is the drift wall temperature. The coefficients are:

Coefficient	No VPL	With VPL
y0	1.7157	1.29005
a	2.54E-05	.00023634
b	0.0841621	0.0699924

In fitting the Metra results (drift wall temperature plotted versus representative gradient in temperature) in SigmaPlot2000, I did not include the pre-peak temperature values; only the pairs of data on the decay side of the temperature profile.

Figure XII-16 and XII-17 show the Metra results and fitted exponential equation for the no VPL and with VPL (vapor pressure lowering) options.

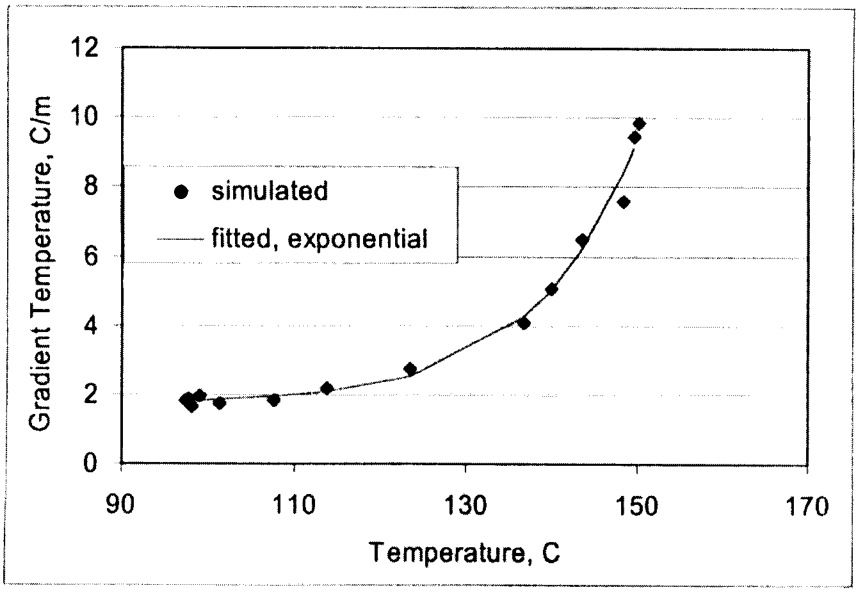


Figure XII-16. Simulated and Fitted Temperature Gradient as a Function of Temperature at the Drift Wall; No Vapor Pressure Lowering Option in Metra.

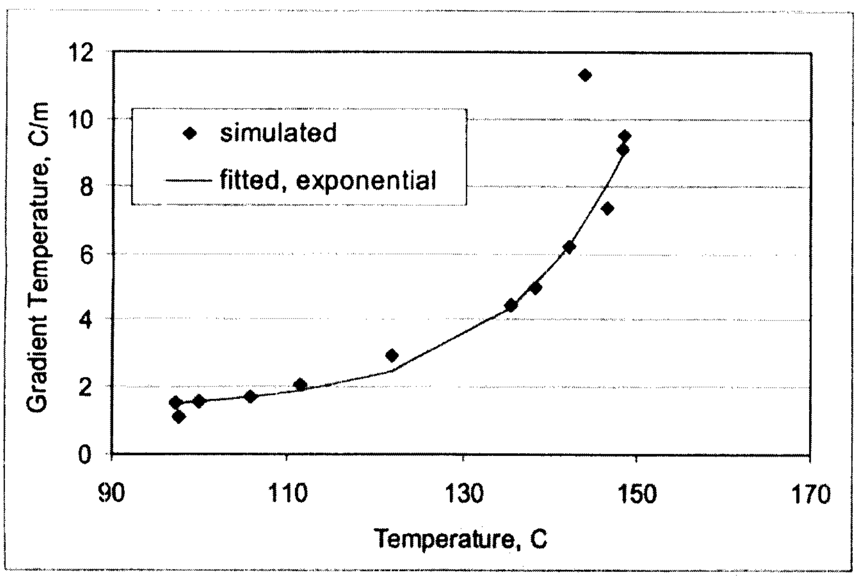


Figure XII-17. Simulated and Fitted Temperature Gradient as a Function of Temperature at the Drift Wall; Vapor Pressure Lowering Option in Metra.

The curves for no vapor phase lowering and with vapor phase lowering are plotted on the same Figure XII-18 below for visual comparison.

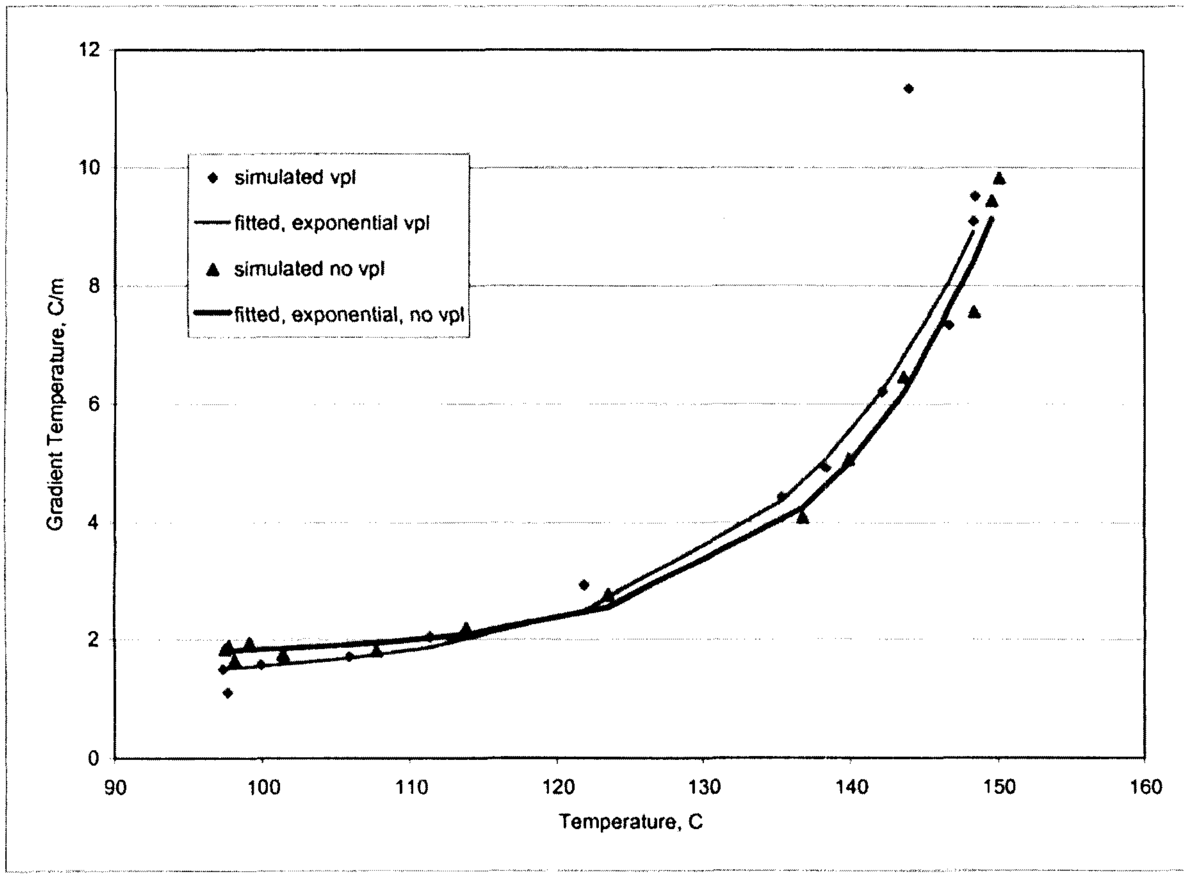


Figure XII-18. Both Curves on Same Plot for Comparison With and Without Vapor Pressure Lowering.

More analyses will be needed to support an estimate of the uncertainty parameter that will be tacked on to the gradient in temperature as a function of temperature relationship.

==

RF 04/15/05

Entries made into Scientific Notebook #432E Volume XII for the period October 2004 to April 15, 2005 have been made by Randall Fedors (April 15, 2005).

No original text or figures entered into this Scientific Notebook has been removed

RF 04/15/05

==

ADDITIONAL INFORMATION FOR SCIENTIFIC NOTEBOOK NO. 432E Vol. XII

Document Date:	4/15/05
Availability:	Southwest Research Institute® Center for Nuclear Waste Regulatory Analyses 6220 Culebra Road San Antonio, Texas 78228
Contact:	Southwest Research Institute® Center for Nuclear Waste Regulatory Analyses 6220 Culebra Road San Antonio, TX 78228-5166 Attn.: Director of Administration 210.522.5054
Data Sensitivity:	■"Non-Sensitive" ☐ Sensitive ☐"Non-Sensitive - Copyright" ☐ Sensitive - Copyright
Date Generated:	4/18/2005
Operating System: (including version number)	Windows NT
Application Used: (including version number)	Wordperfect 10.0; Excel 2002; Word 2002; Adobe; PowerPoint 2002; Lehay fortran 95; Sun fortran 77
Media Type: (CDs, 3 1/2, 5 1/4 disks, etc.)	1 CD
File Types: (.exe, .bat, .zip, etc.)	.ASCII text with various extensions; .jpg; .pdf; .wpd; .doc; .xls; .ppt; .jnb; .ai; encapsulated postscrit (.eps)
Remarks: (computer runs, etc.)	