

308 --- Q199702060013
Scientific Notebook #113,
Regional Tectonics Research

RECORD

37%

R

130

SATX 78238, 210522-3747

Any correspondence should include the code number printed at the bottom of this page as well as the book title stamped at the bottom of the spine.

CNWRA
CONTROLLED
COPY # 113

RMI290393

140

Contributors:

D. Brent Henderson

Chad Fleming pg. 58-61

Danny Skelton pg. 118-121

Scott Mack pg. 122-125

Project: Regional Tectonics Research 20-5704-163
PI: David Ferrill x 6082

Objective: Data processing and visualization to support investigations of tectonics processes and relationships in regional area.

Area of investigation includes 120°W to 114°W longitude and 32°N to 40°N latitude.

Utilize various digital elevation model (DEM) databases from USGS including 3-arcsecond and 30-meter resolution DEMs.

Also use Landsat Thematic Mapper (TM) imagery as well as data from various mapping and modeling packages such as ARC/INFO and DGI Earth Vision.

Current research includes investigation of maximum horizontal stress, relationships of faults and seismic events, and relationships of faults and topography.

Brent Henderson
8-5-94

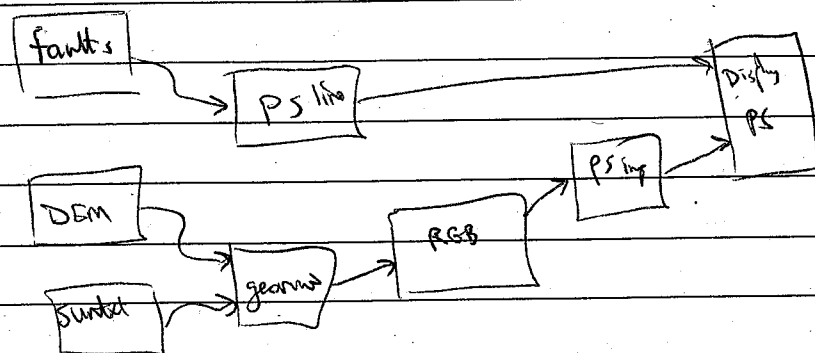
Two approaches for investigating relationship of DEMs and faults are considered.

A. Synthetic shading of DEMs at various azimuths and elevations.

Compare results with known faults (digitized).

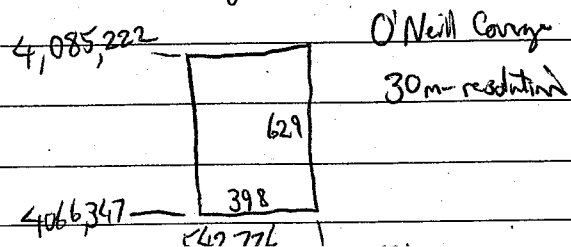
Procedure:

1. Compute raster image of synthetically shaded DEM.
2. ~~Overlay~~ ^{PS} Convert raster image to PostScript.
3. Overlay fault lines with image in PostScript.
4. Visually compare correlation of shading and fault lines.



Area of interest is Yucca Mt.

Time to synthesize image = 21 seconds on SGI Oxy/REL



Brut Hahn
8-5-94

Compare synthetic shading of DEM with 2 fault coverages: O'Neill 92 and FRB20 (from GIS database).

1. Do synthetic shading from Az 0-360 in 10 deg steps.
2. Merge fault lines with DEM shading.
3. Use SGI movie program to view files.

Factors - 1. Scale elevation 1.0-8.0
2. Sun Azimuth
3. Sun Elevation
4. Image process shaded DEM images:
threshold, find edges (Roberts) using imgworks.

Show = roberts 0,1,2... display Az80 - movie
sun060 AllFlts
sun070 AllFlts
az000, az090, az180, az270

Suggestions from David Ferrell:

1. digitize sundance fault from USGS OFR 94-49,
The Sundance Fault: A Newly Recognized
Shear Zone At Yucca Mtn, Nev.
2. look at TM
3. aerial photos

Brut Hahn

Problem: Display 3D surface (topography) data along with line data such as faults + outlines.

Approach #1:

1. DEM \rightarrow geotiff \rightarrow GEM
2. GEM \rightarrow geotiff \rightarrow RGB [can add lines here]
3. GEM + RGB \rightarrow geotiff \rightarrow GEMTEX
4. GEMTEX \rightarrow geotiff \rightarrow DIT

3D Model Range 545010 - 550980 m UTM East
407400 - 408300 m UTM North

* Compare 'warden' at Yuma Mtn big riffs are fault controlled + some faults exist in "flat" valleys.

Butcher
8-9-94

Aug 29, 30th discuss how 3D Model was built.

\rightarrow adeg. 8 8mp .8 twt - horizon names

Filter DEMs:

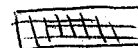
1. Process raw 7.5min DEMs (USGS) UTM projection.

Problem: 7.5min DEMs are scattered data, i.e. they don't have uniform # elevations per row.

Solution: Use Sphylous Transform to load 7.5min DEM and save as regular mesh.

Load ~~7.5min xyz~~ ^{BH 10-3-94} file (44 MB)

Rows(Y) 1401



Elev Min 740

Columns(X) 1151

1124 Max 2178

Range 34,500 m

33,690

Range 42,000 m

Min 533,000 m

533,370

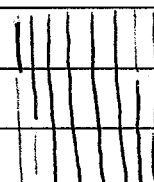
Min 4,053,000 m

Max 567,500 m

567,060

Max 4,095,000 m

Y



Range

Min 4,053,390

Max 4,0

End

Display only peaks - filter out smaller peaks, find FFT of peak
List of features looking for: peaks, harmonics, slopes, sequence, dips
11.11 11.11

Filter 7.5m DEMs

1. Run 7.5m DEMs, merge using dem-merge:

dem-merge -in m.txt -out ~~9-7m-xyz~~ ^{BH 10-3-94}

where m.txt: C 9

~~bigdune.dem~~

~~amargoserv.dem~~

~~stripkills.dem~~

~~craterflat.dem~~

~~hustelbroke.dem~~ ^{BH 10-3-94}

~~jacksflat.dem~~

~~eastbeatty.dem~~

~~topopahspgnt.dem~~

~~topopahspg.dem~~

^{BH 10-3-94}
and ~~9-7m-xyz~~ format is:

533370.00 4094280.00 1288

533370.00 4094310.00 1291

;

;

;

East

North

Elev.

2. Run stats on ~~9-7m-xyz~~ ^{BH 10-3-94}:

X (East) 533370 - 567060 m

Range 33690.0 m

1124 cnt

Y (North) 4,053,150 - 4,094,910 m

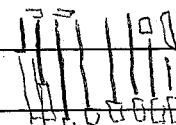
Range 41760 m

Cnt 1393 cnt

Z (elev) 740 - 2178 (1438) m Burt Hersh 8-17-94

Filter 7.5m DEM cont. BH 10-3-94

3. Fill in data for ~~9-7m-xyz~~ to make regular grid



for now use Zmin to fill in data

Read x,y if (y > y_{min}) then fill in y_{min}..y

~~9-7m-xyz~~ ^{BH 10-3-94} → ~~9-7m-dat~~ ^{BH 10-3-94}
(43 MB) (50 MB)

4. Remove X,Y data and only keep elevation Z values

~~9-7m-dat~~ ^{BH 10-3-94} → ~~9-7m-z~~ ^{BH 10-3-94}
(50 MB) (17 MB)

Matlab - [row, column]

fscanf(fid, '%g', [3,4])
m n

test.z

T

Matlab

t(1,1)=1

2

t(2,1)=2

3

t(3,1)=3

4

t(1,2)=4

5

t(2,2)=5

6

i

7

t(3,4)=12

8

9

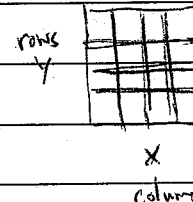
10

11

12

view(0,0) - looking up y-axis z

view(-90,0) - looking up x-axis z



Burt Hersh 8-17-94

Filter DEMs cont.

1. Loaded ~~9.7min-2~~^{10-3-94 BH} in matlab,
now extract subregions of ~~9.7min-2~~^{10-3-94 BH} and
save to disk files so these can be processed

later: save frame var -ascii

Then use load to load disk files

load frame

Read 9.7min-2:

~~fid = fopen('9.7min-2', 'r');~~^{BH}

~~data = fscanf(fid, '%g', [1393, 1124]);~~

Y X
NORTH EAST

Subregion 542000 - 552500 X 287 - 638 (352)
4065000 - 4085000 Y 395 - 1062 (668)

Faults are indicated by ridges - peaks + valleys.

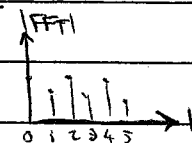
Gradient - ran out of memory.



Brent Henderson 8-17-94

Filter DEMs cont.

1. FFT is



where $f = \frac{1}{NT}$

$N = \# \text{ of samples}$

$T = \text{time interval between samples}$

for example: $f = \frac{1}{(6)(0.015625)} = 10.6 \text{ Hz}$

2. Shaded RGB img:

sun: red → ←

gm ↓ ↑

using imworks:

color → bw

negate

~~imworks~~ BH 8-19-94

Contrast

threshold 63%

edge Roberts

Brent Henderson 8-19-94

Filter DEMs

Summarize progress:

1. Goals

- 1.1 Identify probable locations of faults from DEMs
- 1.2 Classify DEM regions based on their feature order.

2. Approach

- 2.1 Color shade DEMs, coregister with map of faults.
- 2.2. 3D views of shaded DEMs
- 2.3 FFT analysis of DEMs
- 2.4 Smoothing/Filtering DEMs

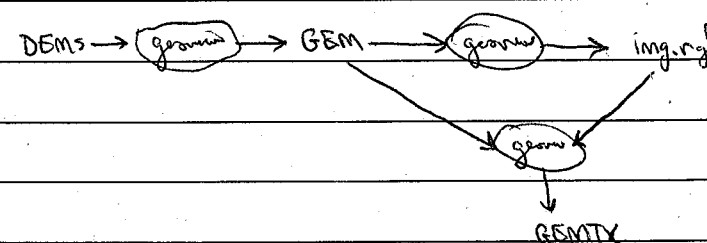
3. Results

Matlab isn't working due to
license manager problem.

rad27- oth
rad83-TM

Burt Henrich 8-23-94

3D View of Surface FT



runGemToGemtx

```

#!/bin/csh
#-----
if ($1 == "MAKE") then
    rm ../gem/9_7min_oneill.gemtx

    geoview \
        -intype 5 \
        -in ../gem/9_7min_oneill.gen \
        \
        -outtype 4 \
        -out ../gem/9_7min_oneill.gemtx \
        \
        -nodraw \
        \
        -alpha 0xff \
        -img ../rgb/oneill_faults_bw3.rgb \
        -imgalign 12390.00 -12330.00 30.00 30.00 \
        -imgtex \
        \
        -scale 1.0 1.0 1.0
endif

#-----
if ($1 == "VIEW") then
    geoview \
        -intype 7 \
        -in ../gem/9_7min_oneill.gemtx \
        \
        -nodraw \
        \
        -wnd2 \
        -bkgnd1 0xffbd794d \
        -eyelnit 33620.48 13345.31 -22948.39 141.15 -34.25 \
        -scale 1.0 3.0 1.0
endif

echo "Finished \007"
  
```

BH
8-24-94

Burt Henrich 8-24-94

DEM Processing

Example:

1. Color shade DEMs, compare with fault lines ^{BA}
 160tectonics/shadedDems/FFTs/rgb/sun060A.tif/10-3-94
 sunMovie

2. 3D Views of shaded DEMs

160tectonics/shadedDems/FFTs/bin/runGemToGembox VIEW

3. FFT Analysis of DEMs

160tectonics/fftDems

4. Smoothing/Filtering DEMs

Meet with D. Ford

1. Suggest thinner faults lines.

2. Decided on topics for AAR poster session.

Application of DEMs for Basin + Range
 Tectonic Record

Brent Jordan
 8-24-94

Semi-Annual Report Figures

Prepare Landis Slides figures:

Images 975 by 644 pixels

$$644 \times \frac{6.5 \text{ inch}}{975 \text{ pix}} = 4.3 \text{ inches ht}$$

$$\text{ht} = 11 \text{ inches} - 1.0 - 2.0 = 8.0 \text{ inch} \quad 6 \times 4 \text{ inch}$$

$$8.5 \text{ inch} - 6 = 2.5 \text{ inch} / 2 = 1.25 \text{ inch}$$

Stress Map Mods

✓ Symbols color → bw

✓ Legend — bw symbols

✓ Remove caption

✓ - Remove - thrust faulting legend

✓ - Add stress regions unknown

New UTM img - brighter, alignment

Outline arrows

✓ Inferred from: fold, fault slip, hydraulic

✓ - Clip around edges of images.

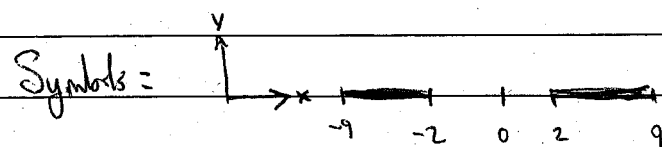
✓ Shrink tic labels

Outline symbol

Brent Jordan 9-6-94

SemiAnnual FiguresStress Map Fig 8-8 cont...

- ✓ Outline Arrows
- ✓ Outline Symbols
- ✓ New UTM img - lighten up



DEM Files:

10-3-94 BH	10-3-94 BH
trona-w-e	cedar-w
10-3-94 BH	10-3-94 BH
death valley-w-e	las Vegas-w
10-3-94 BH	10-3-94 BH
goldfield	kingman-w
9-7-94	kingman-w

Burt Hendon
9-7-94

SemiAnnual Figures

Fig 8-8 — thinner outlines on arrows, lighter DEM,
color symbols

Fig 8-1 All Eq DEM was 1400 x 2178

2000 x 1979 ^{BH} 9-8-94 2740

New DEM

- ✓ Remove caption
- New circles?
- ✓ Clipping

Fig 8-2

Moving files to publication:

- ✓ Add digital lines
- ✓ Add labels
- ✓ Black circles
- ✓ Remove caption

O:\PUBS\BENT

25 44182.37

New DEM

- New circles?
- ✓ Clipping

Burt Hendon
9-8-94

NRC Review Mtg

All Eq Poster

• DEM rdy ingrowths Brightness 1.8
Contrast Shadows 0.6

Highlights - 0.1

EQs

#EQs 30

Mags 0.0 to 8.3

Dates 18/2/93 to 1994.11/27/94

Mag Size MagLabel

<1.0 0.1 M0

<2.0 0.5 M1

<3.0 0.6 M2

<4.0 0.7 M3

<5.0 1.0 M4

<6.0 2.5 M5

<7.0 4.0 M6

<8.0 6.0 M7

<9.0 8.0 M8

Sorted by - depth

Burt Hahn

9-16-94

Epicenter data is sorted:

runSortEqDepth.bh

Page 1

Sort by: (1) Field 2 = depth
(2) Field 4 = date
(3) Field 3 = magnitude

sort +2r -3 +4 +3r -4 part053_sortedEqMag.bh.ps > part053_sortedEqDepth.bh.ps

runSortEqMag

Sort by: (1) Field 2 = depth
(2) Field 4 = date
(3) Field 3 = magnitude

sort +3 -4 +7 -8 part053_sortedEqMag.new.ps > ac

runEqStats

#!/bin/sh

nawk '{

mag = \$4 + 0.0

cnt1++

if (mag < 2.0)

cnt1++

else

if (mag < 3.0)

cnt2++

else

if (mag < 4.0)

cnt3++

else

if (mag < 5.0)

cnt4++

else

if (mag < 6.0)

cnt5++

else

if (mag < 7.0)

cnt6++

else

if (mag < 8.0)

cnt7++

else

cnt8++

}

END {

cnts = cnt1

printf "

cnts += cnt2

printf " 2.0 <= Mag < 3.0 %6d %6d \n", cnt1, cnts

cnts += cnt3

printf " 3.0 <= Mag < 4.0 %6d %6d \n", cnt2, cnts

cnts += cnt4

printf " 4.0 <= Mag < 5.0 %6d %6d \n", cnt3, cnts

cnts += cnt5

printf " 5.0 <= Mag < 6.0 %6d %6d \n", cnt4, cnts

cnts += cnt6

printf " 6.0 <= Mag < 7.0 %6d %6d \n", cnt5, cnts

cnts += cnt7

printf " 7.0 <= Mag < 8.0 %6d %6d \n", cnt6, cnts

cnts += cnt8

printf " Mag >= 8.0 %6d %6d \n", cnt7, cnts

printf "Total %6d %6d \n", cnt8, cnts

}' part053_sortedEqMag.ps

MAG 1 CNT

2.0 <= Mag < 3.0 609 9519

3.0 <= Mag < 4.0 8910 26501

4.0 <= Mag < 5.0 16982 30130

5.0 <= Mag < 6.0 3629 30635

6.0 <= Mag < 7.0 505 30738

7.0 <= Mag < 8.0 103 30754

Mag >= 8.0 16 30756

Total 30756

Burt Hahn 9-16-94

Look Up Tell Script + Assign Colors to Cards

runEpicentersLUT.bh

Page 1

```
#!/bin/sh

# nawk 'BEGIN { firstOne = 1
#               lastLineNo = 9999 }
#               { if (substr($1,1,1) != "#") {
#                   printf "%20.10f %20.10f %20.10f %6.1f %12.6f dCirc\n", $1, $2, $3,
# $5, $6
#               } }' ./part0rs.pdat > o
#               }' eqLanders.pdat > ../partsUtm/part052_epicenters.ps

nawk 'BEGIN {
    colorIdx = 0
    while (getline <"color.lut") {
        if (substr($1,1,1) != "#") {
            color[colorIdx] = $1 + 0.0
            color[colorIdx+1] = $2 + 0.0
            color[colorIdx+2] = $3 + 0.0
            colorIdx++
        }
    }
    colorIdxLast = colorIdx
    close("color.lut")

    magIdx = 0
    while (getline <"mag.bhlarge.lut") {
        if (substr($1,1,1) != "#") {
            magRng[magIdx] = $1 + 0.0
            magValue[magIdx] = $2 + 0.0
            magIdx++
        }
    }
    magIdxLast = magIdx
    close("mag.lut")
}
{
    if (substr($1,1,1) != "#") {
        printf "%20.10f %20.10f %20.10f ", $1, $2, $3

        magInput = $4 + 0.0

        for (magIdx = 0; magIdx < magIdxLast; magIdx++) {
            if (magInput < magRng[magIdx]) break;
        }

        if (magIdx == magIdxLast)
            magOutput = magValue[magIdx-1]
        else
            magOutput = magValue[magIdx]

        printf " %6.1f ", magOutput

        date = $5 + 0.0

# Mags Min 0.0 Max 8.3
# Date Min 1812.937050 Max 1994.112730

dFirst = 1812.937050
```

BA
9-16-94

runEpicentersLUT.bh

Page 2

```
# Note: Be sure the dLast date is later than all of your data.
# In other words, do not use the dLast date.

dLast = 1994.113000
dRng = dLast - dFirst
colCntdate = 64.0
dScale = colCntdate / dRng

pos = (date - dFirst) * dScale
pp = int (pos) * 3

printf " %0.3f %0.3f %0.3f %12.6f dCirc\n", color[pp], color[pp+1],
color[pp+2], date
}
}' ./partsUtm/part052_epicenters.ps > ../partsUtm/part053_sortedEqMag.bh.ps

echo "DONE \007"
```

But John
9-16-94

runAssemble.bh

```
#!/bin/sh

# Usage: runAssemble -or-
# runAssemble A B C
# where:
# A = bw or color [color]
# B = letter or ledger [ledger]
# C = portrait or landscape [landscape]

# demFile=../partsUtm/part021_demIngBw.ps
# demFile=../partsUtm/part022_demIngColor.ps
# demFile=../partsUtm/part023_demIngBwDate.ps
# demFile=../partsUtm/part024_demIngColorDate.ps

echo Begin

if (test $# = 0)
then
    demFile=../partsUtm/part023_demIngBwDate.ps
    legendFile=../partsUtm/part061_dateLegend.ps
    pselect=1
else
    if (test $# = 3)
    then
        if (test $1 = "bw")
        then
            demFile=../partsUtm/part023_demIngBwDate.ps
            legendFile=../partsUtm/part061_dateLegend.ps
        else
            demFile=../partsUtm/part024_demIngColorDate.ps
            legendFile=../partsUtm/part061_dateLegend.ps
        fi
        if ((test $2 = "letter") && (test $3 = "portrait"))
        then
            pselect=1
        else
            if ((test $2 = "letter") && (test $3 = "landscape"))
            then
                pselect=2
            else
                if ((test $2 = "ledger") && (test $3 = "portrait"))
                then
                    pselect=3
                else
                    pselect=4
                fi
            fi
        fi
    fi
else
    echo "Usage: runAssemble -or-
    echo " runAssemble A B C"
    echo " where:
    echo " A = bw or color [color]"
    echo " B = letter or ledger [ledger]"
    echo " C = portrait or landscape [landscape]"
    exit
fi

echo $demFile $pselect

nawk '{ if ($1 == "/pageSelect") {
    print "/pageSelect", psel, "def"
} else {
    print $0
}'
```

BA
9-16-94

mag.bhlarge.lut

#	Range	Value
1.0	.1	
2.0	.5	
3.0	.6	
4.0	.7	
5.0	1.0	
6.0	2.5	
7.0	4.0	
8.0	6.0	
99.0	8.0	

BA
9-16-94

runAssemble.bh

```
} psel=$pselect ./partsUtm/part010_top.ps > run.tmp
mv run.tmp ./partsUtm/part010_top.ps

echo "Assemble with $demFile..."
cat ./partsUtm/part010_top.ps \
./partsUtm/part014_topTitle.ps \
./partsUtm/part013_clip.ps \
$demFile \
./partsUtm/part031_statesHdr.ps \
./partsUtm/part032_states.ps \
./partsUtm/part081_ymBox.ps \
./partsUtm/part051_epicentersHdr.ps.new \
./partsUtm/part053_sortedEqDepth.bh.ps \
./partsUtm/part041_faultsHdr.ps \
./partsUtm/part042_faults.ps \
./partsUtm/part060_endClip.ps \
$legendFile \
./partsUtm/part011_ticsHdr.ps \
./partsUtm/part012_tics.ps \
./partsUtm/part063_title.ps \
./partsUtm/part064_scale.ps \
./partsUtm/part065_magLegend.ps.large \
./partsUtm/part066_blackBox.ps \
./partsUtm/part070_showPage.ps \
>
fig8-1.ps
```

But John
9-16-94

Creating poster of DEM, faults, stress map data:

-120°-114°W

32°-48°N

Components

no epicenters

add stress data,

need header file

data file

legend file

x imgXfirst sub imgXsch mul

y imgYfirst sub imgYsch mul

modify horiz stress legend to mpt to
bottom of image.

modify title of poster

Burt Hahn 10-3-94

Edit Slip Plots PS

1. Comment out plotext (N)

2. center x

3. center y

4. rotate

5. plot scale

6. legend rel. mag.

7. legend dot size

80 -100 51

110 -10 52

-50 -120 52

-10 105 N

Burt Hahn

10-3-94

Poster 4

DEM, Faults, Stress, Slip Plots

Poster 2

DEM, Faults, EQ Hypocenters

Creative Communications, Jay Jamison

415-957-0333

Space 1/4" tape - tar

Std 48 hrs processing

Printer - Cactus/Versatel wide 52"

Synergy 34" wide

\$12/sq ft plot

\$5/sq ft hot lamination

please include scale factor

Charrette, Laurie 617-547-9484

Printer DigiMax 42" wide 400 dpi

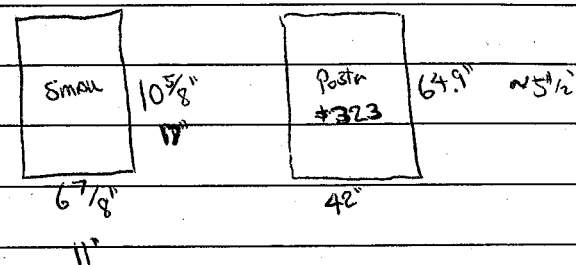
42x65 \$252

hot lamination \$110.25

3-days

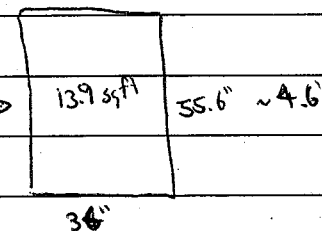
Syquest disk, no tape

D. But Hahn 10-5-94

Poster 2 All Eq

\$238 →

Price Qty 2



Origin? Center on page.

Poster 4 Slip Tending

- X1. Sigma symbol - std char set in ps
- X2. $\sigma_1 = \text{vert}$
- X3. wht circle backgnd → rectangle
- 4. circle-dot-line 3

					MAX SN
X5.	Plot Ym	$\sigma_1 = \text{vert} = 100$	$\sigma_2 = \text{NS} = 90$	$\sigma_3 = \text{EN} = 18$	0.955
X6.	Plot DV	$\sigma_2 = \text{vert} = 99$	$\sigma_1 = \text{NS} = 100$	$\sigma_3 = \text{EW} = 18$	0.964
X7.	Plot SM	$\sigma_2 = \text{vert} = 50$	$\sigma_1 = \text{NS} = 100$	$\sigma_3 = \text{EN} = 18$	0.955

X8. Display max S/N value

X9. Rotate plots

X10. Top title

X11. North Arrow/tic

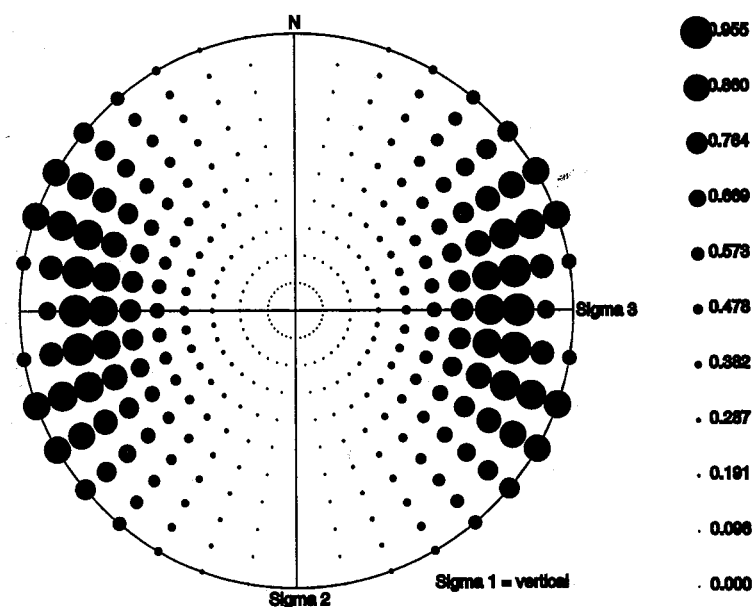
X12. Legend X 145 / 156

But Hahn 10-5-94

BH 10-5-94
Dor

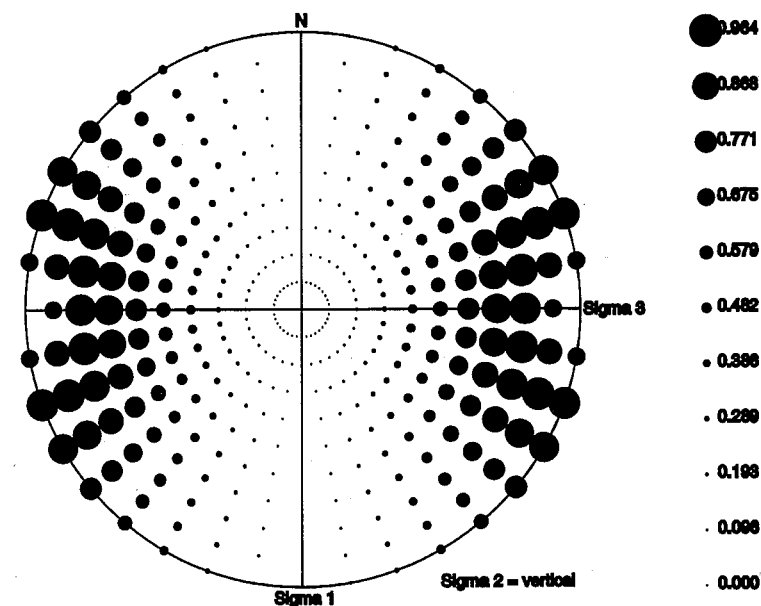
Yucca Mountain

Stress Domain



Slip Tendency
Relative Magnitudes of Principal Stresses:
Vertical Sigma 1 1
N-S Sigma 2 0.9
E-W Sigma 3 0.18

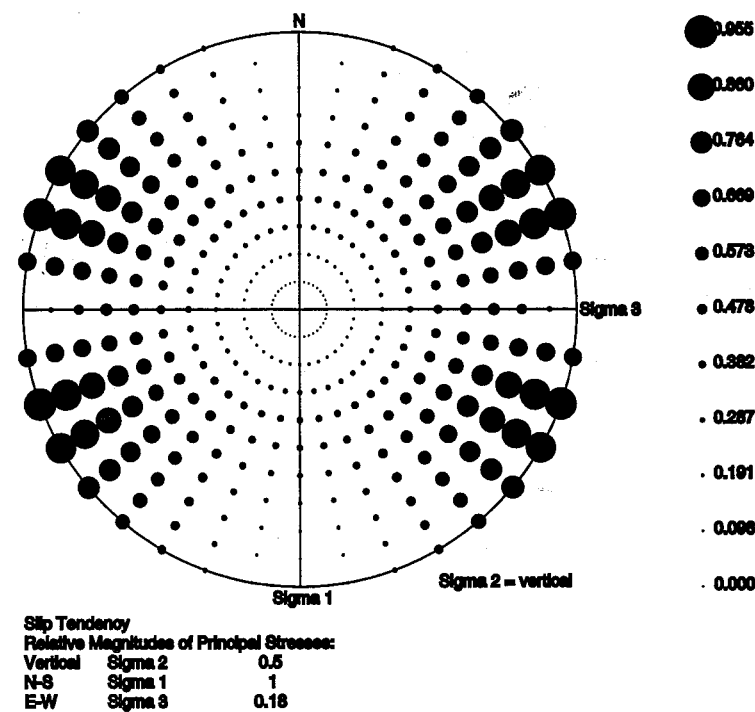
Burt Hahn
10-5-94

Death Valley
Stress
Domain

Slip Tendency
Relative Magnitudes of Principal Stresses:
Vertical Sigma 2 0.99
N-S Sigma 1 1
E-W Sigma 3 0.18

Burt Hahn
10-5-94

Southern Mojave
Stress
Domain



Burt Lahr
10-5-94

~~Poster 4 - add circle(dot) lines~~

Yucca Mountain
Southern Mojave
Death Valley

~~Poster 5 - Landers~~

$\begin{matrix} x & y \\ 4.0 & 2.4 \\ -5.7 & -2.0 \end{matrix}$ scaled points

☐ Poster 3 - Add 9 plots
☐ Poster 6 - Add refs

Burt Lahr 10-6-94

part091_slipYM.ps

```

% Pole plot
gsave % BEGIN poleplot

/inch [72 mul] def

% User defs
/xCenter 4.0 inch def
/yCenter 5.6 inch def

/plotScale 0.2700 def
/dotScale 1.1000 def

/linered 0.0 def
/linegrn 0.0 def
/lineblu 0.0 def

/sldred 0.0 def
/sldgrn 0.0 def
/sldblu 0.0 def

/rotangle -29.0 def

/font1 [
  /Helvetica findfont
  20.0 scalefont setfont
] def

/font2 [
  /Symbol findfont
  30.0 scalefont setfont
] def

0.500 setlinewidth

% Begin plot
xCenter yCenter translate

plotScale dup scale

% Background box
newpath
  1.0 1.0 1.0 setrgbcolor
  -120.0 -280.0 moveto
  -120.0 160.0 lineto
  215.0 160.0 lineto
  215.0 -280.0 lineto
  fill

/plotCircle [
  % Usage: x y rad red grn blu plotCircle
  /blu exch def
  /grn exch def
  /red exch def
  /rad exch def
  /y exch def
  /x exch def
  newpath
  x y rad dotScale mul 0 360 arc
  gsave
  linered linegrn lineblu setrgbcolor

```

part091_slipYM.ps

```

stroke
grestore
% red grn blu setrgbcolor
sldred sldgrn sldblu setrgbcolor
fill
] def

/plotText [
  % Usage: x y rt tstr plotText
  /tstr exch def
  /rt exch def
  /y exch def
  /x exch def
  x y moveto
  gsave
  rt rotate
  tstr true charpath
  gsave
  linered linegrn lineblu setrgbcolor
  stroke
  grestore
  sldred sldgrn sldblu setrgbcolor
  fill
  grestore
] def

gsave % BEGIN plot
rotangle rotate

```

ole	100	0	0.0125	0.0025
le	90	0	2.94	0.6
le	80	0	5.51	1.12

```

% 0.0 setgray
sldred sldgrn sldblu setrgbcolor

```

```

newpath
-100.0 0.0 moveto
100.0 0.0 lineto
stroke

newpath
0.0 -100.0 moveto
0.0 100.0 lineto
stroke

newpath
0.0 0.0 100.0 0 360 arc
stroke

font2 115.0 -10.0 rotangle neg (s3) plotText
font2 -15.0 -140.0 rotangle neg (s2) plotText

grestore % END plot

font2 50.0 -130.0 0.0 (s1) plotText
font1 80.0 -127.0 0.0 (= vert) plotText

font1 -95.0 140.0 0.0 (Yucca Mountain Domain) plotText
font1 -6.0 120.0 0.0 (N) plotText

```

But John
10-7-94

part091_slipYM.ps

Page 1

```

newpath
sldred sldgrn sldblu setrgbcolor
0.0 100.0 moveto
0.0 115.0 lineto
stroke

% Dot Size Legend
% 145 -100 0.0125 3.66e-11 0.9 0 plotCircle
e
156 -101 0.0 (0.000) plotText
145 -80 0.113 0.0195 0.881 0 plotCircle
156 -81 0.0 (0.096) plotText
145 -60 0.312 0.039 0.861 0 plotCircle
156 -61 0.0 (0.191) plotText
145 -40 0.613 0.0585 0.842 0 plotCircle
156 -41 0.0 (0.287) plotText
145 -20 1.01 0.078 0.822 0 plotCircle
156 -21 0.0 (0.382) plotText
145 1.88e-08 1.51 0.0975 0.803 0 plotCircle
156 -1 0.0 (0.478) plotText
145 20 2.11 0.117 0.783 0 plotCircle
156 19 0.0 (0.573) plotText
145 40 2.81 0.136 0.764 0 plotCircle
156 39 0.0 (0.669) plotText
145 60 3.61 0.156 0.744 0 plotCircle
156 59 0.0 (0.764) plotText
145 80 4.51 0.175 0.725 0 plotCircle
156 79 0.0 (0.860) plotText
145 100 5.51 0.195 0.705 0 plotCircle
156 99 0.0 (0.955) plotText

```

```

% Sigma Mags Legend
font1 -100.0 -150.0 0.0 (slip Tendency) plotText
font1 -100.0 -170.0 0.0 (Max Stress Ratio:) plotText
font1 85.0 -170.0 0.0 ( 0.955) plotText
font1 -100.0 -190.0 0.0 (Rel. Mag. of Principal Stresses:) plotText
font1 -100.0 -215.0 0.0 (Vertical) plotText
font2 -20.0 -215.0 0.0 (s1) plotText
font1 20.0 -215.0 0.0 (s1) plotText
font1 -100.0 -245.0 0.0 (N-S) plotText
font2 -20.0 -245.0 0.0 (s2) plotText
font1 20.0 -245.0 0.0 ( 0.90) plotText
font1 -100.0 -275.0 0.0 (E-W) plotText
font2 -20.0 -275.0 0.0 (s3) plotText
font1 20.0 -275.0 0.0 ( 0.18) plotText

```

```

% Pointer line-dot
gsave % BEGIN line-dot
4.0 setlinewidth
0.0 0.0 0.0 setrgbcolor
/dotx -5.7 inch def
/doty -6.3 inch def

% Bottom of dot
newpath
dotx doty 5.0 0 360 arc
gsave
6.0 setlinewidth
1.0 1.0 1.0 setrgbcolor
stroke

```

part091_slipYM.ps

```

grestore

% Line
newpath
-80.0 -70.0 moveto
dotx doty lineto
gsave
6.0 setlinewidth
1.0 1.0 1.0 setrgbcolor
stroke
grestore
stroke

% Top of dot
newpath
dotx doty 5.0 0 360 arc
fill

grestore % END line-dot
grestore % END poleplot

```

But John
10-7-94

ps2eps.txt

```

%%PS-Adobe-2.0 EPSF-2.0
%%BoundingBox: 20 144 530 630
%%Pages: 1
%%EndComments
%%BeginProlog
%%BeginResource ShowcaseResource
%%EndResource
%%EndProlog
%%BeginSetup
/ncopies 1 def
%%EndSetup
%%Page: label 1
gsave % BEGIN EPS file

```

```

grestore % END EPS file
showpage
%%EOF

```

Pages 1 through 30 of this Scientific Notebook were reviewed for compliance with QAP-001 in response to Corrective Action Request 94-02. Corrections and clarifications were made as appropriate. In some cases, the date of a change will reflect the date of this review rather than the date of the original Scientific Notebook entry.

Randy Zolch
SWRE-QA
12/02/94

Burt Hendon 12-2-94

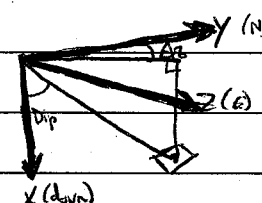
Slip Tendency Plots

Normal stress $\sigma_n = \sigma_x l^2 + \sigma_y m^2 + \sigma_z n^2$ from Folding and Fracturing of Rocks, John G. Ramsay, McGraw-Hill, 1967, p. 358

where: $l = \cos(\text{dip})$

$m = \sin(\text{dip}) \cos(\alpha_2)$

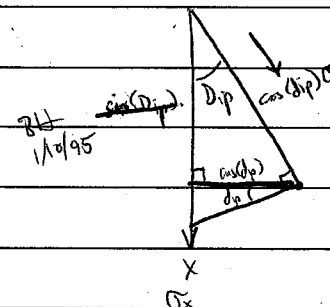
$n = \sin(\text{dip}) \sin(\alpha_2)$



Key point: stress = force per unit area

force = stress * area

For example: $\text{force}_x = \text{stress}_x * \text{area}_x = \sigma_x \cos(\text{dip}) * \text{area}_x$
 $= \sigma_x \cos(\text{dip}) * \cos(\text{dip})$
 $= \sigma_x l^2$



Shear stress $= \tau = (s^2 - \sigma_n^2)^{1/2}$ where $s^2 = \sigma_x^2 l^2 + \sigma_y^2 m^2 + \sigma_z^2 n^2$
 $= ((\sigma_x - \sigma_y)^2 l^2 m^2 + (\sigma_y - \sigma_z)^2 m^2 n^2 + (\sigma_x - \sigma_z)^2 l^2 n^2)^{1/2}$

Burt Hendon 1/10/95

Rotation of Axis:

P' is P in terms of rotated XYZ-axis system

Rot X

$$Y \Rightarrow \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos X & \sin X \\ 0 & -\sin X & \cos X \end{pmatrix}$$

$$Z \Rightarrow \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos X & \sin X \\ 0 & -\sin X & \cos X \end{pmatrix}$$

Rot Y

$$\begin{pmatrix} \cos Y & 0 & -\sin Y \\ 0 & 1 & 0 \\ \sin Y & 0 & \cos Y \end{pmatrix}$$

Rot Z

$$\begin{pmatrix} \cos Z & \sin Z & 0 \\ -\sin Z & \cos Z & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$\begin{pmatrix} \cos Z & \sin Z & 0 \\ -\sin Z & \cos Z & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$\begin{pmatrix} \cos Z & \sin Z & 0 \\ -\sin Z & \cos Z & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Order of rotations 1. Rot X 2. Rot Z 3. Rot Y

Rot Matrix

$$\begin{pmatrix} \cos Y \cos Z & \cos X \cos Y \sin Z + \sin X \sin Y & \sin X \cos Y \sin Z - \cos X \sin Y \\ -\sin Z & \cos X \cos Z & \sin X \cos Z \\ \sin Y \cos Z & \cos X \sin Y \sin Z - \sin X \cos Y & \sin X \sin Y \sin Z + \cos X \cos Y \end{pmatrix}$$

$$P' = \text{Rot Matrix} * P$$

Burt Jerh 1-10-95

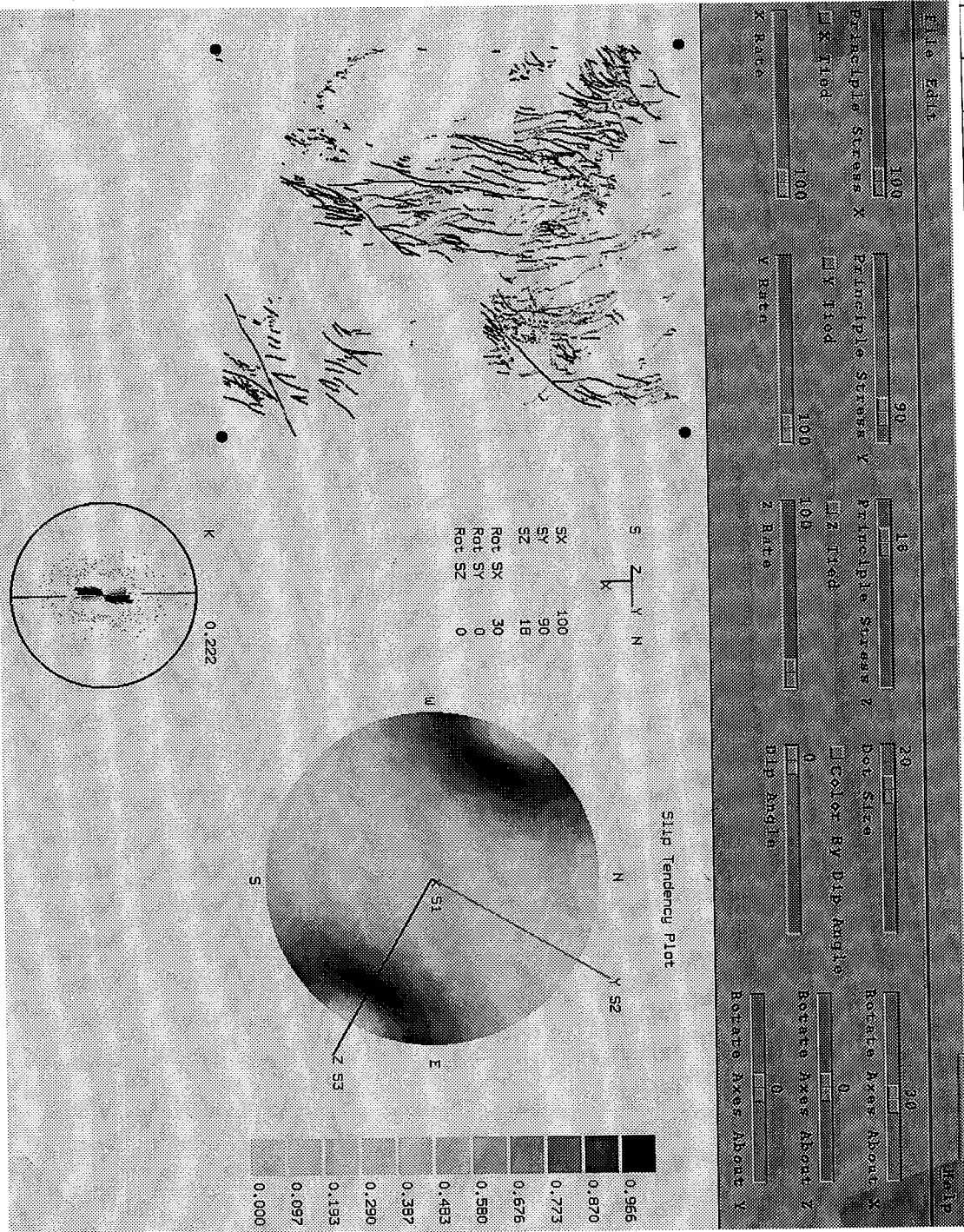
Find Slip Vector Az, Dip

1. Find stress vector $\sigma_x, \sigma_y, \sigma_z$
2. Find normal vector for fault surface
3. Rotate surface normal to rotated stress axes
4. Null stress on X (surface normal) direction
5. Rotate back to world coords
6. Compute Az, dip:

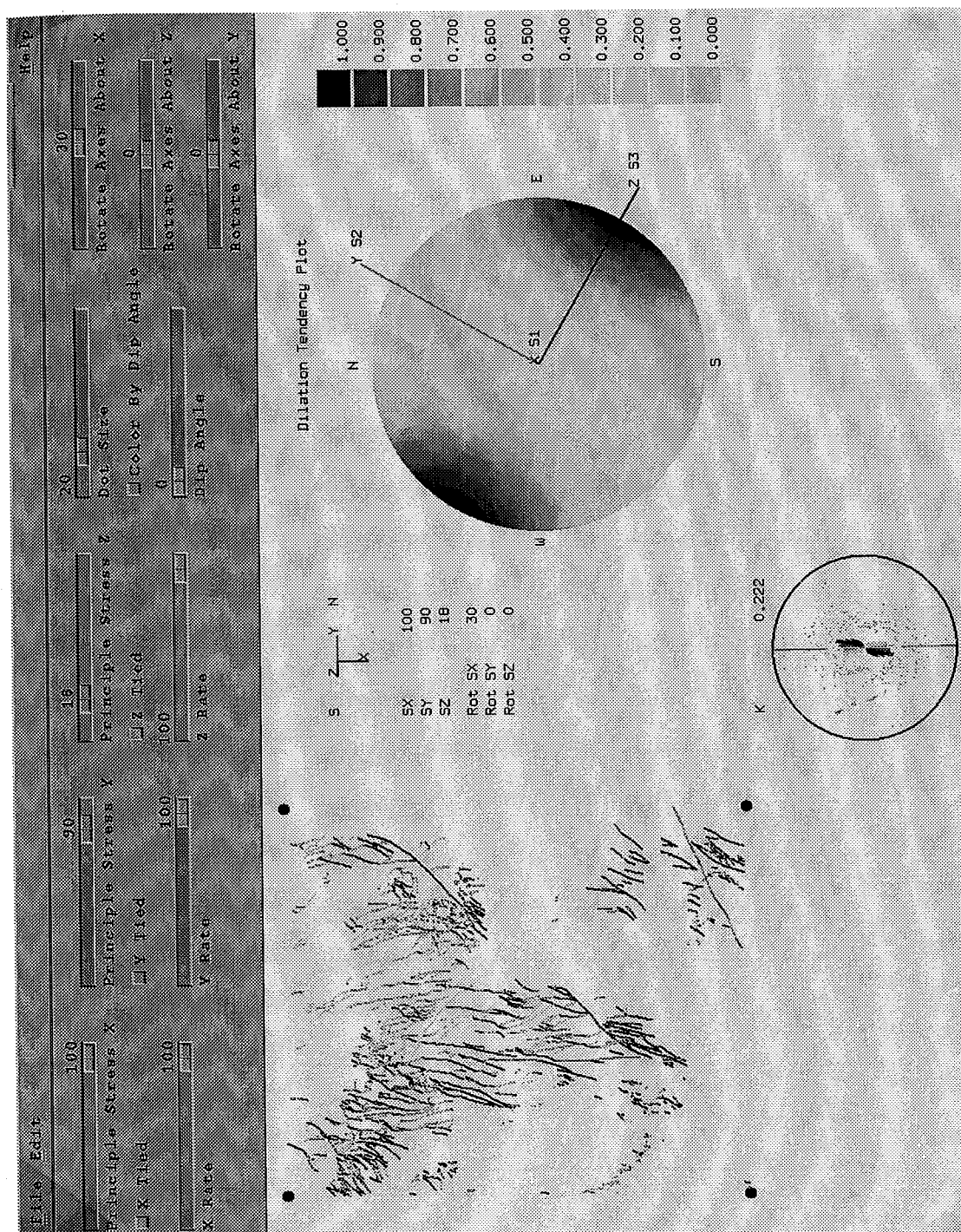
$$Az = \arctan^2(\text{stress}_Z, \text{stress}_Y)$$

$$\text{dip} = \arctan^2(\sqrt{\text{stress}_Y^2 + \text{stress}_Z^2}, \text{stress}_X)$$

Burt Jerh 1-16-95



Buss J. 1-16-95



Burt John 1-18-95

X	X S1	1
Y	Y S2	0.9
Z	Z S3	0.18
ROT X	30	
ROT Y	0	
ROT Z	0	
AZ	DIP	Dilation Tendency
0	0	0
0	10	0.0103
0	20	0.0399
0	30	0.0854
0	40	0.141
0	50	0.2
0	60	0.256
0	70	0.302
0	80	0.331
0	90	0.341
10	0	0
10	10	0.00677
10	20	0.0263
10	30	0.0562
10	40	0.0928
10	50	0.132
10	60	0.168
10	70	0.198
10	80	0.218
10	90	0.225
20	0	0
20	10	0.00448
20	20	0.0174
20	30	0.0371
20	40	0.0613
20	50	0.0871
20	60	0.111
20	70	0.131
20	80	0.144
20	90	0.148
30	0	0
30	10	0.00368
30	20	0.0143
30	30	0.0305
30	40	0.0504
30	50	0.0716
30	60	0.0915
30	70	0.108
30	80	0.118
30	90	0.122
40	0	0
40	10	0.00448
40	20	0.0174
40	30	0.0371
40	40	0.0613
40	50	0.0871
40	60	0.111
40	70	0.131
40	80	0.144
40	90	0.148
50	0	0
50	10	0.00677
50	20	0.0263
50	30	0.0562
50	40	0.0928
50	50	0.132
50	60	0.168
50	70	0.198
50	80	0.218
50	90	0.225
60	0	0
60	10	0.0103
60	20	0.0399
60	30	0.0854
60	40	0.141
60	50	0.2
60	60	0.256
60	70	0.302
60	80	0.331
60	90	0.341
70	0	0
70	10	0.0146
70	20	0.0567
70	30	0.121
70	40	0.2
70	50	0.284
70	60	0.364
70	70	0.428
70	80	0.47
70	90	0.485
80	0	0
80	10	0.0192
80	20	0.0745
80	30	0.159
80	40	0.263
80	50	0.374
80	60	0.478
80	70	0.563
80	80	0.618
80	90	0.637
90	0	0
90	10	0.0146
90	20	0.0567
90	30	0.121
90	40	0.2
90	50	0.284
90	60	0.364
90	70	0.428
90	80	0.47
90	90	0.485

But Jhr 1-16-98

210	10	0.00368
210	20	0.0143
210	30	0.0305
210	40	0.0504
210	50	0.0716
210	60	0.0915
210	70	0.108
210	80	0.118
210	90	0.122
220	0	0
220	10	0.00448
220	20	0.0174
220	30	0.0371
220	40	0.0613
220	50	0.0871
220	60	0.111
220	70	0.131
220	80	0.144
220	90	0.148
230	0	0
230	10	0.00677
230	20	0.0263
230	30	0.0562
230	40	0.0928
230	50	0.132
230	60	0.168
230	70	0.198
230	80	0.218
230	90	0.225
240	0	0
240	10	0.0103
240	20	0.0399
240	30	0.0854
240	40	0.141
240	50	0.2
240	60	0.256
240	70	0.302
240	80	0.331
240	90	0.341
250	0	0
250	10	0.0146
250	20	0.0567
250	30	0.121
250	40	0.2
250	50	0.284
250	60	0.364
250	70	0.428
250	80	0.47
250	90	0.485
260	0	0
260	10	0.0192
260	20	0.0745
260	30	0.159
260	40	0.263
260	50	0.374
260	60	0.478
260	70	0.563
260	80	0.618
260	90	0.637
270	0	0
270	10	0.0235
270	20	0.0913
270	30	0.195
270	40	0.322
270	50	0.458
270	60	0.585
270	70	0.689
270	80	0.757
270	90	0.78
280	0	0
280	10	0.0192
280	20	0.0745
280	30	0.159
280	40	0.263
280	50	0.374
280	60	0.478
280	70	0.563
280	80	0.618
280	90	0.637
290	0	0
290	10	0.0271
290	20	0.105
290	30	0.224
290	40	0.371
290	50	0.527
290	60	0.673
290	70	0.792
290	80	0.87
290	90	0.897
300	0	0
300	10	0.0235
300	20	0.0913
300	30	0.195
300	40	0.322
300	50	0.458
300	60	0.585
300	70	0.689
300	80	0.757
300	90	0.78
310	0	0
310	10	0.0192
310	20	0.0745
310	30	0.159
310	40	0.263
310	50	0.374
310	60	0.478
310	70	0.563
310	80	0.618
310	90	0.637
320	0	0
320	10	0.0146
320	20	0.0567
320	30	0.121
320	40	0.2
320	50	0.284
320	60	0.364
320	70	0.428
320	80	0.47
320	90	0.485

-2-

320	30	0.224
320	40	0.371
320	50	0.527
320	60	0.673
320	70	0.752
320	80	0.87
320	90	0.897
330	0	0
330	10	0.0235
330	20	0.0913
330	30	0.195
330	40	0.322
330	50	0.458
330	60	0.585
330	70	0.689
330	80	0.757
330	90	0.78
340	0	0
340	10	0.0192
340	20	0.0745
340	30	0.159
340	40	0.263
340	50	0.374
340	60	0.478
340	70	0.563
340	80	0.618
340	90	0.637
350	0	0
350	10	0.0146
350	20	0.0567
350	30	0.121
350	40	0.2
350	50	0.284
350	60	0.364
350	70	0.428
350	80	0.47
350	90	0.485

X	X S1	1
Y	Y S2	0.9
Z	Z S3	0.18
ROT X	30	
ROT Y	0	
ROT Z	0	
AZ	DIP	Slip Tendency
0	0	0
0	10	0.0729
0	20	0.144
0	30	0.212
0	40	0.275
0	50	0.33
0	60	0.375
0	70	0.407
0	80	0.427
0	90	0.433
10	0	0
10	10	0.0513
10	20	0.101
10	30	0.147
10	40	0.189
10	50	0.223
10	60	0.259
10	70	0.28
10	80	0.284
10	90	0
20	0	0.0299
20	10	0.0583
20	20	0.0836
20	30	0.105
20	40	0.12
20	50	0.131
20	60	0.137
20	70	0.139
20	80	0.139
20	90	0.14
30	0	0
30	10	0.0172
30	20	0.0325
30	30	0.0444
30	40	0.0514
30	50	0.0523
30	60	0.0468
30	70	0.0353
30	80	0.0189
30	90	1.99e-10
40	0	0
40	10	0.0299
40	20	0.0583
40	30	0.0836
40	40	0.105
40	50	0.12
40	60	0.131
40	70	0.137
40	80	0.139
40	90	0.14
50	0	0
50	10	0.0513
50	20	0.101
50	30	0.147
50	40	0.189
50	50	0.223
50	60	0.25
50	70	0.269
50	80	0.28
50	90	0.284

But Jhr 1-16-98

-3-

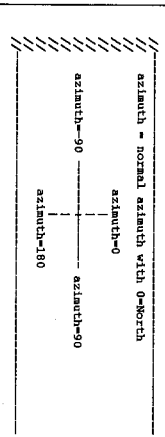
60	0	0	170	20	0.185
60	10	0.0729	170	30	0.274
60	20	0.144	170	40	0.359
60	30	0.212	170	50	0.437
60	40	0.275	170	60	0.501
60	50	0.33	170	70	0.55
60	60	0.375	170	80	0.579
60	70	0.407	170	90	0.588
60	80	0.427	180	0	0
60	90	0.433	180	10	0.0729
70	0	0	180	20	0.144
70	10	0.0928	180	30	0.212
70	20	0.185	180	40	0.275
70	30	0.274	180	50	0.33
70	40	0.359	180	60	0.375
70	50	0.437	180	70	0.407
70	60	0.501	180	80	0.427
70	70	0.55	180	90	0.433
70	80	0.579	190	0	0
70	90	0.588	190	10	0.0513
80	0	0	190	20	0.101
80	10	0.11	190	30	0.147
80	20	0.221	190	40	0.189
80	30	0.331	190	50	0.223
80	40	0.438	190	60	0.25
80	50	0.54	190	70	0.269
80	60	0.627	190	80	0.28
80	70	0.693	190	90	0.284
80	80	0.731	200	0	0
80	90	0.742	200	10	0.0299
90	0	0	200	20	0.0583
90	10	0.125	200	30	0.0836
90	20	0.25	200	40	0.105
90	30	0.379	200	50	0.12
90	40	0.508	200	60	0.131
90	50	0.633	200	70	0.137
90	60	0.744	200	80	0.139
90	70	0.823	200	90	0.14
90	80	0.859	210	0	0
90	90	0.866	210	10	0.0172
100	0	0	210	20	0.0325
100	10	0.135	210	30	0.0444
100	20	0.273	210	40	0.0514
100	30	0.415	210	50	0.0523
100	40	0.563	210	60	0.0468
100	50	0.71	210	70	0.0353
100	60	0.84	210	80	0.0189
100	70	0.917	210	90	2.88e-09
100	80	0.909	220	0	0
100	90	0.876	220	10	0.0299
110	0	0	220	20	0.0583
110	10	0.142	220	30	0.0836
110	20	0.287	220	40	0.105
110	30	0.439	220	50	0.12
110	40	0.598	220	60	0.131
110	50	0.76	220	70	0.137
110	60	0.901	220	80	0.139
110	70	0.954	220	90	0.14
110	80	0.809	230	0	0
110	90	0.61	230	10	0.0513
120	0	0	230	20	0.101
120	10	0.144	230	30	0.147
120	20	0.292	230	40	0.189
120	30	0.447	230	50	0.223
120	40	0.611	230	60	0.25
120	50	0.778	230	70	0.269
120	60	0.922	230	80	0.28
120	70	0.955	230	90	0.284
120	80	0.685	240	0	0
120	90	1.09e-08	240	10	0.0729
130	0	0	240	20	0.144
130	10	0.142	240	30	0.212
130	20	0.287	240	40	0.275
130	30	0.439	240	50	0.33
130	40	0.598	240	60	0.375
130	50	0.76	240	70	0.407
130	60	0.901	240	80	0.427
130	70	0.954	240	90	0.433
130	80	0.809	250	0	0
130	90	0.61	250	10	0.0928
140	0	0	250	20	0.185
140	10	0.135	250	30	0.274
140	20	0.273	250	40	0.359
140	30	0.415	250	50	0.437
140	40	0.563	250	60	0.501
140	50	0.71	250	70	0.55
140	60	0.84	250	80	0.579
140	70	0.917	250	90	0.588
140	80	0.909	260	0	0
140	90	0.876	260	10	0.11
150	0	0	260	20	0.221
150	10	0.125	260	30	0.331
150	20	0.25	260	40	0.438
150	30	0.379	260	50	0.54
150	40	0.508	260	60	0.627
150	50	0.633	260	70	0.693
150	60	0.744	260	80	0.731
150	70	0.823	260	90	0.742
150	80	0.859	270	0	0
150	90	0.866	270	10	0.125
160	0	0	270	20	0.25
160	10	0.11	270	30	0.379
160	20	0.221	270	40	0.508
160	30	0.331	270	50	0.633
160	40	0.438	270	60	0.744
160	50	0.54	270	70	0.823
160	60	0.627	270	80	0.859
160	70	0.693	270	90	0.866
160	80	0.731	280	0	0
160	90	0.742	280	10	0.135
170	0	0	280	20	0.273
170	10	0.0928	280	30	0.415

Burt Hahn
1-16-95

-4-

280	40	0.563
280	50	0.71
280	60	0.84
280	70	0.917
280	80	0.909
280	90	0.876
290	0	0
290	10	0.142
290	20	0.287
290	30	0.439
290	40	0.598
290	50	0.76
290	60	0.901
290	70	0.954
290	80	0.809
290	90	0.61
300	0	0
300	10	0.144
300	20	0.292
300	30	0.447
300	40	0.611
300	50	0.778
300	60	0.922
300	70	0.955
300	80	0.685
300	90	2.3e-08
310	0	0
310	10	0.142
310	20	0.287
310	30	0.439
310	40	0.598
310	50	0.76
310	60	0.901
310	70	0.954
310	80	0.809
310	90	0.61
320	0	0
320	10	0.135
320	20	0.273
320	30	0.415
320	40	0.563
320	50	0.71
320	60	0.84
320	70	0.917
320	80	0.909
320	90	0.876
330	0	0
330	10	0.125
330	20	0.25
330	30	0.379
330	40	0.508
330	50	0.633
330	60	0.744
330	70	0.823
330	80	0.859
330	90	0.866
340	0	0
340	10	0.11
340	20	0.221
340	30	0.331
340	40	0.438
340	50	0.54
340	60	0.627
340	70	0.693
340	80	0.731
340	90	0.742
350	0	0
350	10	0.0928
350	20	0.185
350	30	0.274
350	40	0.359
350	50	0.437
350	60	0.501
350	70	0.55
350	80	0.579
350	90	0.588

Burt Hahn
1-16-95

[illegible]

```

// A = Azimuth Angle
// double sinaz = sin(az * plotDPR);
// double cosaz = cos(az * plotDPR);

// Calculate slip tendency values for Surfaces
double rotMat[9];
calcRotMat[9] ( attributes.rotValue,
               attributes.rotValue,
               attributes.rotValue,
               rotMat );

// Dip angle
// double sinDip = sin(dip * plotDPR);
// double cosDip = cos(dip * plotDPR);
// Calculate surface normal vector
// double surfX = cosDip * cosaz;
// double surfY = sinDip * cosaz;
// double surfZ = sinDip * sinaz;

// Rotate surface normal vector to match rotated stress axes
// double surfRotX = surfX * rotMat[0] +
// surfY * rotMat[3] +
// surfZ * rotMat[6];
// double surfRotY = surfX * rotMat[1] +
// surfY * rotMat[4] +
// surfZ * rotMat[7];
// double surfRotZ = surfX * rotMat[2] +
// surfY * rotMat[5] +
// surfZ * rotMat[8];

// Find rotated surface normal az/dip
// double surfRotAz = atan2(surfRotZ, surfRotY);
// double surfRotDip = acos(surfRotZ);

// Calc direction cosines
// double dirCosX = cos(surfRotDip);
// double dirCosY = sin(surfRotDip);
// double dirCosZ = sin(surfRotDip);

// Calc slipNormalStress = stressX * dirCosX + dirCosY *
// stressY + dirCosZ * dirCosZ;

// Calc slipStress = sqrt(slipTerm + a2Term + a3Term);
if (slipNormalStress == 0.0) {
    slipStressRatio = slipStress/0.000001;
    cont << "*****" << endl << flush;
    else {
        slipStressRatio = slipShearStress/slipNormalStress;
    }
}

// Find slip direction
// 1. -RotX
// 2. -RotY
// 3. -RotZ

// Order of rotations: 1. X-axis
// 2. Y-axis
// 3. Z-axis

// Rotate axis system so that XYZ-pt can be expressed
// in terms of rotated axis system.
Mat[16]:
m0 m3 m6
m1 m4 m7
m2 m5 m8

RotX = RotZ * RotX (RotX first, RotZ second)
RotY = RotZ * RotY
RotZ = 0 0 0

```

Given faults ungenerated from ARC/INFO, find elevation at each point in fault file.

Method:

1. Load 30-meter DEM data for BaseMtn in Matlab
2. Find elevations by using 2D interpolation in Matlab

ARC Info Coverage from Kathy Spivey =

bringenges
~~infants~~ ^{2-1095 B2} (Worm zone II)

X Min	506685	Max	533946	m
Y Min	406697	Max	4086572	m

~~longeng~~ ^{longeng} ^{2-10-15 04} (KTM Zon II) geology polygons

X	Min	520664	Max	534100	m
Y	Min	4066997	Max	4085487	m

Bmt Hahn 2-1098

Bwt Jahn 1-17-95

[illegible]

```

14 04 -11 1
Name
COUN "BACOS 50 (LITTON)"
END
made 1, print 53
1 31.

```

But John 2-27-95

[illegible]

BWA Hlm 227-95

C-Script cont...

```

print $0
if [ $0 = 506685.0 \
  $1 = 534100.1 \
  $2 = 406697.0 \
  $3 = 4086572.5 ]
then
  death_valley-e.xyz > death_valley-e_sub.xyz
fi

./demarcaseo/runarc010demtoxyz
fi/bin/sh

demotodat -demfile death_valley-e -datfile death_valley-e.xyz -proj 2

./demarcaseo/run020MinMaxXYZ
fi/bin/sh

if [ $? -lt 1 ]
then
  echo "No data of 1"
  echo "USAGE $0 <filename>\n\n"
  exit
fi

newk 'BEGIN {
  needfirstOne = 1
  if (NF > 1) {
    x = $1 + 0.0
    y = $2 + 0.0
    z = $3 + 0.0
    if (needfirstOne) {
      minx = x
      miny = y
      minz = z
      needfirstOne = 0
    }
    if (x < minx) minx = x
    if (x > maxx) maxx = x
    if (y < miny) miny = y
    if (y > maxy) maxy = y
    if (z < minz) minz = z
    if (z > maxz) maxz = z
  }
}
END {
  printf "X Min %12.4f Max %12.4f Avg %12.4f Cnt %12.4f\n",
    minx, maxx, (maxx-minx)/20.0,
    printf "Y Min %12.4f Max %12.4f Avg %12.4f Cnt %12.4f\n",
    miny, maxy, (maxy-miny)/20.0,
    printf "Z Min %12.4f Max %12.4f Avg %12.4f\n", minz, maxz
  }
}' $1

./demarcaseo/runarc030xyztoGrid
fi/bin/sh

echo "Creating 2grd...\n"
cp death_valley-e_sub.xyz death_valley-e.dat

FEATURE=PV-QD/SVP
ev_2grd \
  -d death_valley-e.2grd \
  -F SPATIAL \
  -r 506605.000000,534100.000000,406697.000000,4086670.000000 \
  -s 460.329 \
  -d death_valley-e.dat \
  -f "z" \
  -m 4

rm death_valley-e.dat
ls -l death_valley-e.2grd
ev_export -o death_valley-e_grid.dat death_valley-e.2grd

newk '{
  if (substr($0,1,1) != "z") {
    printf "%12.4f %12.4f %12.4f\n", $1, $2, $3
  }
}' death_valley-e_grid.dat > death_valley-e_grid.xyz

rm death_valley-e_grid.dat
ls -l death_valley-e_grid.xyz
run020MinMaxXYZ death_valley-e_grid.xyz

echo "Finished, created death_valley-e_grid.xyz\n\n"

./demarcaseo/run020MinMaxXYZ
fi/bin/sh

if [ $? -lt 1 ]
then
  echo "No data of 1"
  echo "USAGE $0 <filename>\n\n"
  exit
fi

newk 'BEGIN {
  needfirstOne = 1
  if (NF > 1) {
    x = $1 + 0.0
    y = $2 + 0.0
    z = $3 + 0.0

```

```

  maxx = x
  miny = y
  minz = z
  needfirstOne = 0
  if (x < minx) minx = x
  if (x > maxx) maxx = x
  if (y < miny) miny = y
  if (y > maxy) maxy = y
  if (z < minz) minz = z
  if (z > maxz) maxz = z
}
END {
  printf "X Min %12.4f Max %12.4f Avg %12.4f Cnt %12.4f\n",
    minx, maxx, (maxx-minx)/20.0,
    printf "Y Min %12.4f Max %12.4f Avg %12.4f Cnt %12.4f\n",
    miny, maxy, (maxy-miny)/20.0,
    printf "Z Min %12.4f Max %12.4f Avg %12.4f\n", minz, maxz
  }
}' $1

./demarcaseo/runMerge
fi/bin/sh

cat dem30m.dat demarcaseo.xyz > demMerge.dat

ls -l demMerge.dat

FEATURE=PV-QD/SVP
ev_2grd \
  -d demMerge.2grd \
  -F SPATIAL \
  -r 506605.000000,534176.000000,4066900.000000,4086670.000000 \
  -s 520.660 \
  -d demMerge.dat \
  -f "z" \
  -m 4

cp demMerge.dat

ls -l demMerge.2grd
ev_export -o demMergeGrid.dat demMerge.2grd

newk '{
  if (substr($0,1,1) != "z") {
    printf "%12.4f %12.4f %12.4f\n", $1, $2, $3
  }
}' demMergeGrid.dat > demMergeGrid.xyz

rm demMergeGrid.dat
ls -l demMergeGrid.xyz
run020MinMaxXYZ demMergeGrid.xyz > demMergeGridMinMax.txt
cat demMergeGridMinMax.txt

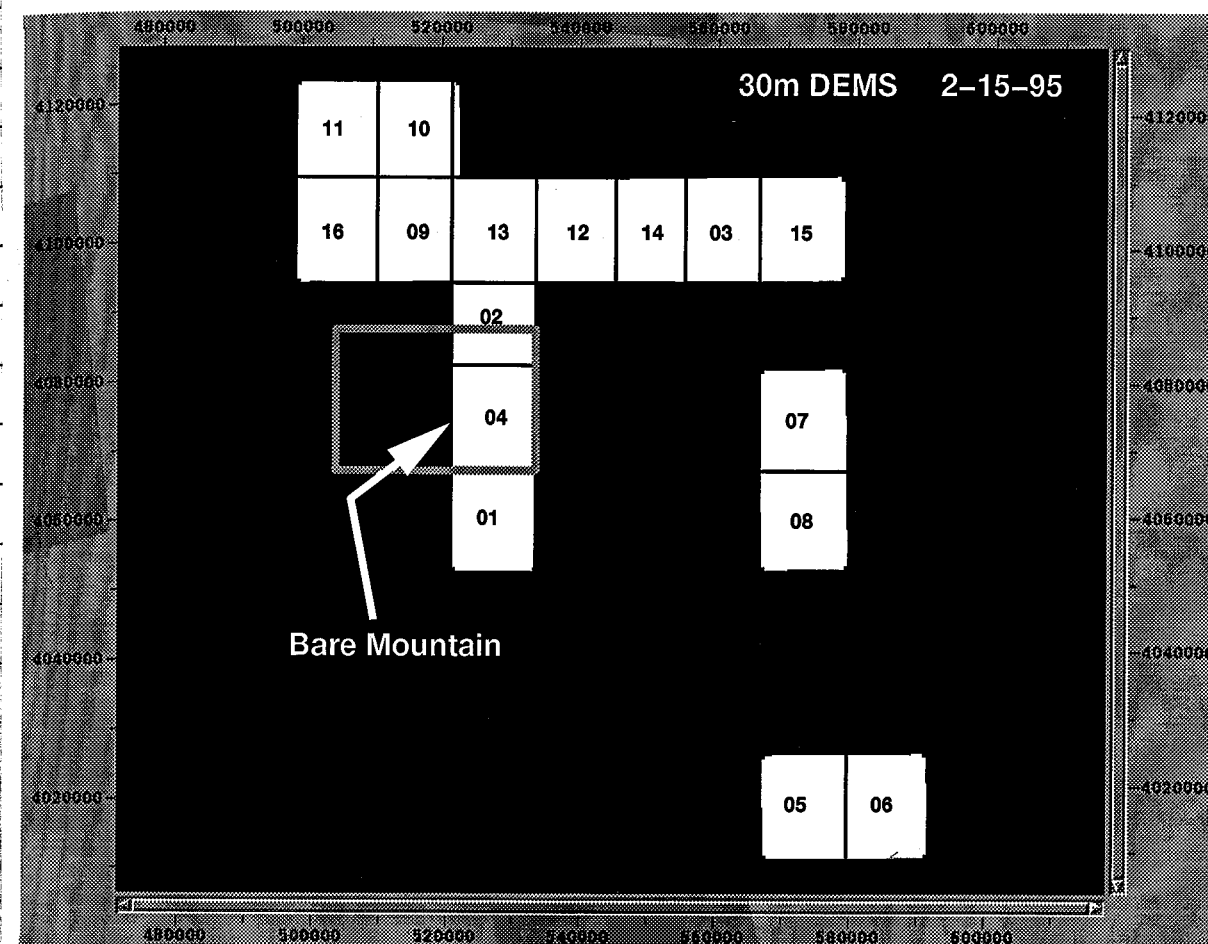
echo "Finished, created demMergeGrid.xyz\n\n"

./demarcaseo/runrgb
fi/bin/sh

for f in baremo.rgb baremo30m.rgb
do
  pe="echo $f | newk -F -" `printf %11s` pa
  echo $f Spe
  cp $f tmp.rgb
  enhance tmp.rgb $f &
  read filename
  rm tmp.rgb
  mv $f $filename
done

```

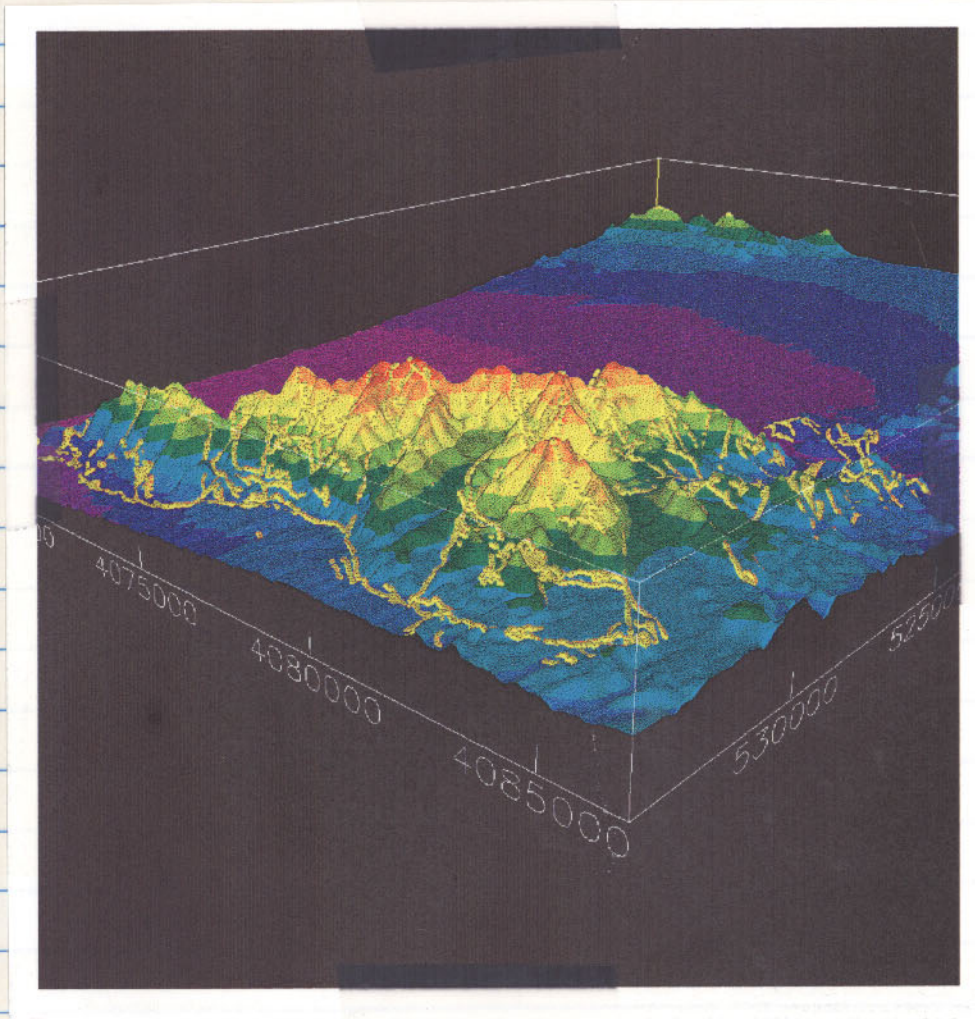
30-meter DEMs



Burt J. H. 2-27-95

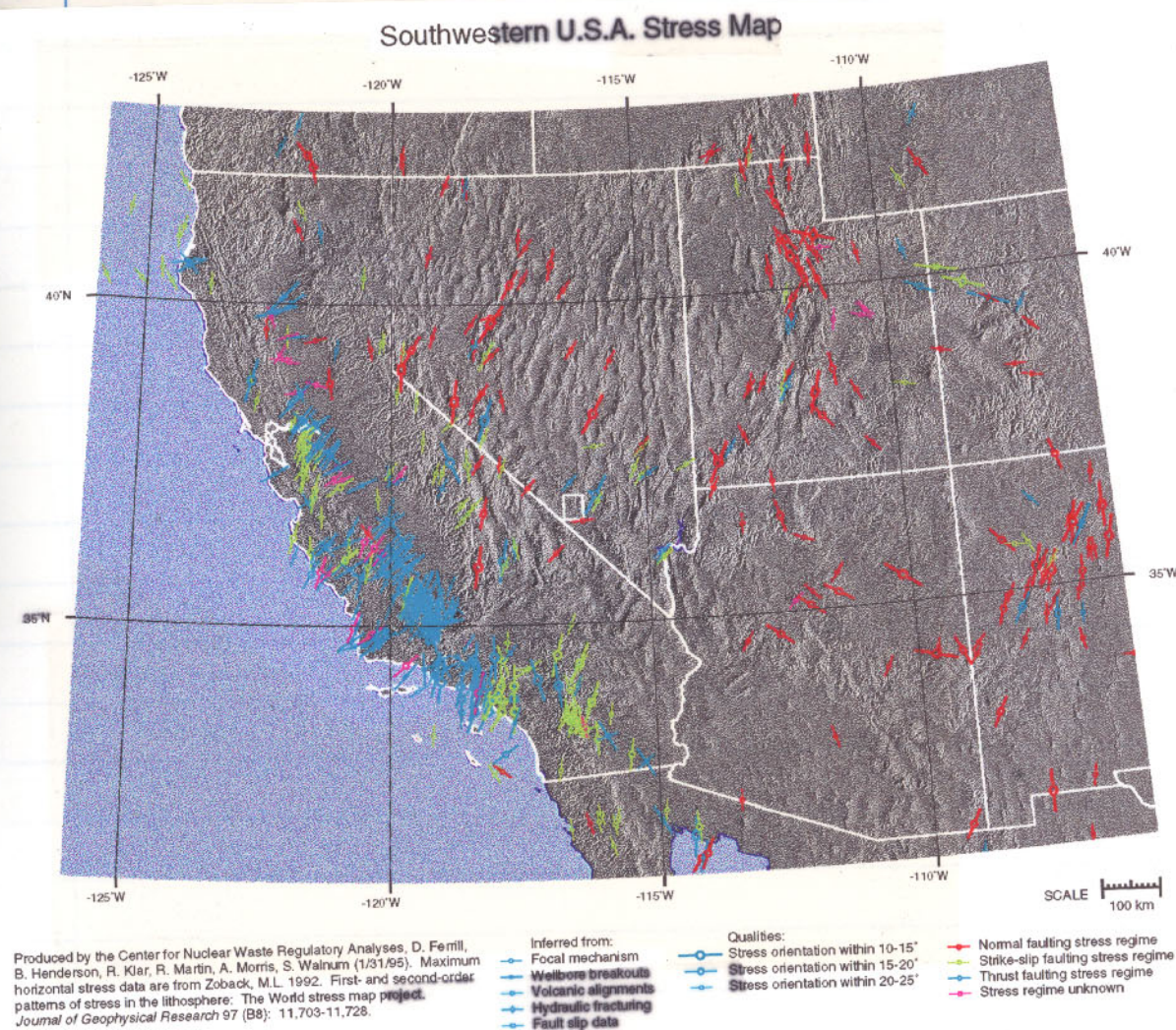
Burt J. H. 2-27-95

Bare Mtn XYZ faults Visualization from EarthVision
of topography and digital faults
at Bare Mtn.



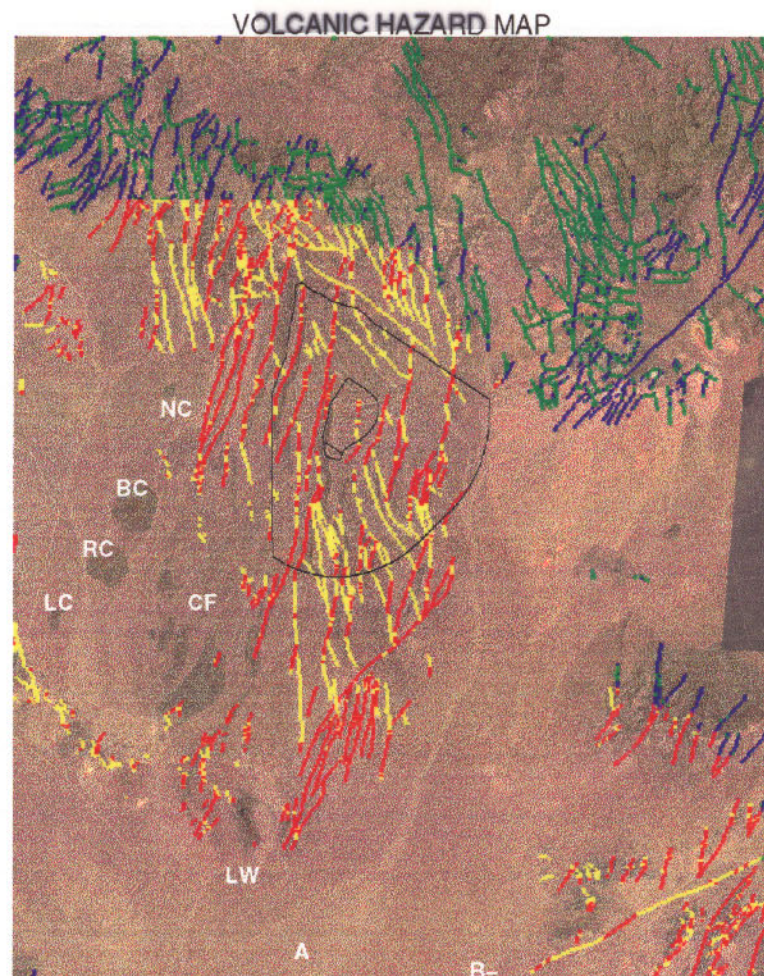
Bmt Jhr 2-27-95

Stress Map



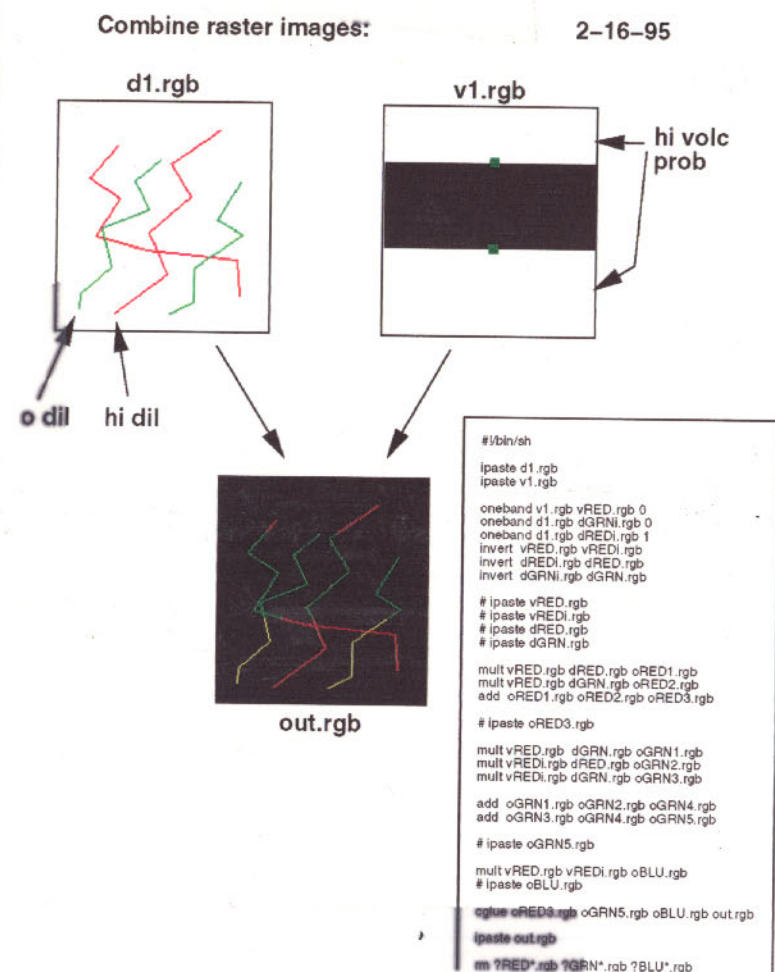
Bmt Jhr 2-27-95

Volcanic Hazard Map Background is Landsat TM image.
 Probabilities from Chuck Connor
 faults colored by 3dstress program



High dilation tendency - high probability zone
 Low dilation tendency - high probability zone
 High dilation tendency - low probability zone
 Low dilation tendency - low probability zone

Volcanic Hazard Map Method for producing the map.



Burt Hahn 2-27-95

Burt Hahn 2-27-95

Visualization of
topography from
USGS 30m
resolution,
visualization from
geoview.



```

OGRADME 3-27-95

Steps:

1. Merge 7.5-minute DEM data, dem --> xyz

#####
runDemMerge
#####

rm dem16.xyz
dem_merge -in dem_16_m1a_merge.txt -out dem16.xyz > dem16.log

2. Convert xyz file to rgb file, xyz --> rgb

#####
runXyzToRgb
#####
#!/bin/csh

rm dem16.rgb

geoview \
-in dem16.xyz \
-intype 1 \
-outtype 3 \
-out dem16.rgb \
-nodraw \
-f 1 \
-hckgnd1 0xffff00d0d0 \
-axes \
-eyenit -20718.97 23535.83 -17637.21 -121.48 -29.64 \
-elev brownToGrn.ect \
-sunfile sun.txt \
-scale 1.0 1.0 -1.0

imgworks dem16.rgb

#####
sun.txt
#####
# Sun illuminations
#
# AINT 0.0 to 1.0
# ACF 0.0 (little diffusion) to 1.0 (lots of diffusion)
#
# RED/GRN/BLU 0.0 to 1.0
#
# DCF 0.0 (little diffusion) to 1.0 (lots of diffusion)
#
# M 0-illum and shadows 1-illum 2-shadows
#
# I illumination function
# S shadow function
# where:
# 0 = cosA
# 1 = cosA^3
# 2 = cosA^7
# 3 = linear
# 4 = sqrt(cosA)
#
# SRC 0-local light src, 1-infinite light src

# AZ EL DIST AINT ACF RED GRN BLU DCF M I S SRC
105.0 5.0 500000.0 -0.4 1.0 0.9 0.9 0.9 1.0 0 0 0 1

#####
brownToGrn.ect
#####
#####
-10000.00000000 ff000000
730.00000000 ff0053a8
756.16666667 ff0059ab
782.33333333 ff005faf
808.50000000 ff0065b2
834.66666667 ff006cb6
860.83333333 ff0072b9
887.00000000 ff0079bd
913.16666667 ff0080d0
939.33333333 ff0087c4
965.50000000 ff008ec7
991.66666667 ff0096cb
1017.83333333 ff009ece
1044.00000000 ff00a5d2
1070.16666667 ff00ad55
1096.33333333 ff00b4d9
1122.50000000 ff00bcdd
1148.66666667 ff00c7e9
1174.83333333 ff00dfe3
1201.00000000 ff00d8e7
1227.16666667 ff00e1ea
1253.33333333 ff00eeee
1279.50000000 ff00f1ee
1305.66666667 ff00f5ec
1331.83333333 ff00f8e9
1358.00000000 ff00fcd5
1384.16666667 ff00ffe3
1410.33333333 ff00fbd8
1436.50000000 ff00f7cc
1462.66666667 ff00f3cl
1488.83333333 ff00e7b7
1515.00000000 ff00eccc
1541.16666667 ff00e8a2
1567.33333333 ff00e498
1593.50000000 ff00e08e
1619.66666667 ff00dc85
1645.83333333 ff00d87c
1672.00000000 ff00d473
1698.16666667 ff00d06a
1724.33333333 ff00ccc2
1750.50000000 ff00c95a
1776.66666667 ff00c552
1802.83333333 ff00c14a
1829.00000000 ff00b742
1855.16666667 ff00b33b
1881.33333333 ff00b534
1907.50000000 ff00b12e
1933.66666667 ff00ad27

```

```

1959.83333333 ff00aa21
1986.00000000 ff00a61b
2012.16666667 ff00a215
2038.33333333 ff009e10
2064.50000000 ff009a0a
2090.66666667 ff009605
2116.83333333 ff009200
2143.00000000 ff008e00
2169.16666667 ff008a00
2195.33333333 ff008700
2221.50000000 ff008300
2247.66666667 ff007f00
2273.83333333 ff007b00
2300.00000000 ff007700

3. Convert xyz file to gen file, xyz --> gen

#####
runXyzToGen
#####
#!/bin/csh

rm dem16.gen

geoview \
-in dem16.xyz \
-intype 1 \
-outtype 1 \
-out dem16.gen \
-nodraw \
-f 1 \
-hckgnd1 0xffff00d0d0 \
-axes \
-eyenit -20718.97 23535.83 -17637.21 -121.48 -29.64 \
-elev brownToGrn.ect \
-sunfile sun.txt \
-scale 1.0 1.0 -1.0

```

Other scripts:

```

#####
run010demMerge
#####
rm bareMtn.xyz
time dem_merge -in demMerge2.txt -out bareMtn.xyz

#####
run020demMergeXYZ
#####
#!/bin/sh

if [ $# -lt 1 ]
then
echo "Usage: %s of 1"
echo "USAGE $0 <filename>\n\n"
exit
fi

awk 'BEGIN {
needFirstOne = 1
}
{
if (NF > 1) {
x = $1 + 0.0
y = $2 + 0.0
z = $3 + 0.0
if (needFirstOne) {
minx = x
bigx = x
miny = y
bigy = y
minz = z
bigz = z
needFirstOne = 0
}
if (x < minx) minx = x
if (x > bigx) bigx = x
if (y < miny) miny = y
if (y > bigy) bigy = y
if (z < minz) minz = z
if (z > bigz) bigz = z
}
}
END {
printf "X Min %12.4f Max %12.4f Avg %12.4f Cnt %12.4f\n",
minx, bigx, bigx-minx, (bigx-minx)/30.0
printf "Y Min %12.4f Max %12.4f Avg %12.4f Cnt %12.4f\n",
miny, bigy, bigy-miny, (bigy-miny)/30.0
printf "Z Min %12.4f Max %12.4f\n", minz, bigz
}' $1

#####
run030checkProfiles
#####
#!/bin/sh

awk 'BEGIN {
firstTime = 1
}
{
x = $1 + 0.0
y = $2 + 0.0
z = $3 + 0.0

if (firstTime) {
firstTime = 0
lastx = x - 30.0
}

if (x != lastx) {
lastx = x
}
else {
if (y != lasty) {
lasty = y
print lastline
}
}
}' $1

```

BWT LHM 3-28-95

BWT LHM 3-28-95

Scripts

```

print $0
print ""
}
lasty = y
lastline = $0
} bareMtn.xyz

#####
run040PillEdges
#####
#!/bin/sh

nawk 'BEGIN {
}
{
x = $1 + 0.0
y = $2 + 0.0
z = $3 + 0.0

if (x != lastx) {

} else {
if (y > yMin) {
for (yf=yMin; yf<y; yf+=30.0) {
printf "%14.6f %14.6f %14.6f\n", x,yf,elev
}
}
printf "%14.6f %14.6f %14.6f\n", x,y,z
}

}' xMin=500010.0 \
xMax=599030.0 \
yMin=4011750.0 \
yMax=4122420.0 \
elev=0.0 \
bareMtn.xyz

#####
run050SubBng
#####
#!/bin/sh

nawk 'BEGIN {
xMin = x0 - 100.0
xMax = x1 + 100.0
yMin = y0 - 100.0
yMax = y1 + 100.0
}
{
x = $1 + 0.0
y = $2 + 0.0
z = $3 + 0.0
if ((x >= xMin) && (x <= xMax) &&
(y >= yMin) && (y <= yMax)) {
print $0
}
}' x0=506685.0 \
x1=534100.125 \
y0=4066997.0 \
y1=4088572.5 \
bareMtn.xyz > bareMtn_sub.xyz

#####
run060XYZtoz
#####
#!/bin/sh

if [ $# -lt 1 ]
then
echo "USAGE $0 <infile>"
exit
fi

nawk '{
print $3
}' $1

#####
run2dems
#####
#!/bin/sh

rm dem30m2.dat

for dem in file04.dat file02.dat
do
datFile=xyz$dem
echo $dem $datFile
echo "C 1" > demMerge.tmp
echo $dem >> demMerge.tmp
dem_merge -in demMerge.tmp -out $datFile
rm demMerge.tmp
cat dem30m2.dat $datFile > tmp
rm $datFile
mv tmp dem30m2.dat
done

echo "Finished, created dem30m2.dat \007"
#####
runDemMerge
#####

rm dem16.xyz
dem_merge -in dem_16_min_merge.txt -out dem16.xyz > dem16.log

runXyzToRgb

```

Scripts

```

exit

rm dem9.xyz
dem_merge -in dem_9_min_merge.txt -out dem9.xyz > dem9.log

#####
runRach
#####
#!/bin/sh

rm dem30m16.dat

for dem in file*.dat
do
datFile=xyz$dem
echo $dem $datFile
echo "C 1" > demMerge.tmp
echo $dem >> demMerge.tmp
dem_merge -in demMerge.tmp -out $datFile
rm demMerge.tmp
cat dem30m16.dat $datFile > tmp
rm $datFile
mv tmp dem30m16.dat
done

#####
runGeoView
#####
geoView -nodraw \
-in dem9.xyz -intype 1 -c 0x00ff0000 0x0000ff -out test.rgb -outtype 3

ipaste test.rgb

#####
runTest
#####
rm demTest.xyz
dem_merge -in dem_test_merge.txt -out demTest.xyz

rm test1.rgb
geoView \
-in demTest.xyz \
-intype 1 \
-outtype 1 \
-out test1.rgb \
-f 1 \
-nodraw \
-bkgnd 0xffffd0d0 \
-axes \
-eyelut -20718.97 23535.83 -17637.21 -121.48 -29.64 \
-c 0x000000 0x000000 \
-sunfile sun.txt \
-scale 1.0 1.0 -1.0

ipaste test1.rgb

#####
runViewGem
#####
#!/bin/csh

geoView \
-in dem16.gem \
-intype 5 \
-f 1 \
-bkgnd 0xffffd0d0 \
-axes \
-eyelut 22491.45 42068.46 -38000.34 179.91 -37.34 \
-scale 1.0 1.0 -1.0 \
-offset -4053120.0 0.0 0.0

#####
runXyzToGem
#####
#!/bin/csh

rm dem16.gem

geoView \
-in dem16.xyz \
-intype 1 \
-outtype 1 \
-out dem16.gem \
-nodraw \
-f 1 \
-bkgnd 0xffffd0d0 \
-axes \
-eyelut -20718.97 23535.83 -17637.21 -121.48 -29.64 \
-elev browToGrn.ect \
-sunfile sun.txt \
-scale 1.0 1.0 -1.0

#####
runXyzToRgb
#####
#!/bin/csh

rm dem16.rgb

geoView \
-in dem16.xyz \
-intype 1 \
-outtype 3 \
-out dem16.rgb \
-nodraw \
-f 1 \
-bkgnd 0xffffd0d0 \
-axes \
-eyelut -20718.97 23535.83 -17637.21 -121.48 -29.64 \
-elev browToGrn.ect \
-sunfile sun.txt \
-scale 1.0 1.0 -1.0

#####
runXyzToRct
#####
#!/bin/sh

# USAGE runXyzToRct <chr file>

min=730
max=2300

nawk 'BEGIN { elev = el
step = (e2-el)/60.0
sc = 0
first = 0
}
{ if ($1 >= 12) sc = 1
if (sc == 1) {
if (first == 0) {
first = 1
printf "%-20.8f %f02x%02x%02x\n", -10000.0, 0, 0, 0
}
printf "%-20.8f %f02x%02x%02x\n", elev, $4, $3, $2
elev += step
}
}' el=$min e2=$max $1

#####
dem_16_min_merge.txt
#####
# demerge16.txt

# List DEM 7.5 minute filenames to merge.
# C = total file count.

C 16

file01.dat
bigdune.dem
amarjoseval.dem
stripedhills.dem

file04.dat
craterflat.dem
bustedbutte.dem
jackassflats.dem

file02.dat
eastbeatty.dem
topopahppgw.dem
topopahppg.dem

file13.dat
file13.dat
file14.dat
file03.dat

```

Burt Hoke 3-28-95

Burt Hoke 3-28-95

Shaded relief map shaded by geoview software.

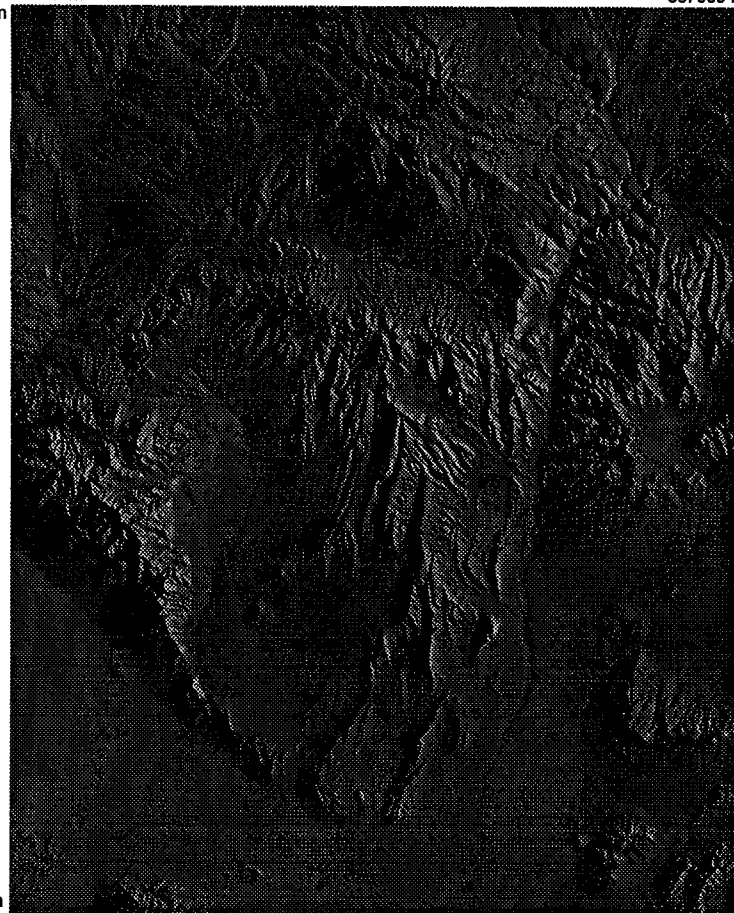
84 4/9/95
4108770m

30-meter DEM Shaded Relief Map 3-27-95

522210m
1094940m

567060 m

1053120m



Elevations: 740 to 2259 m

file01.dat
bigdune.dem
amargosaval.dem
stripedhills.dem

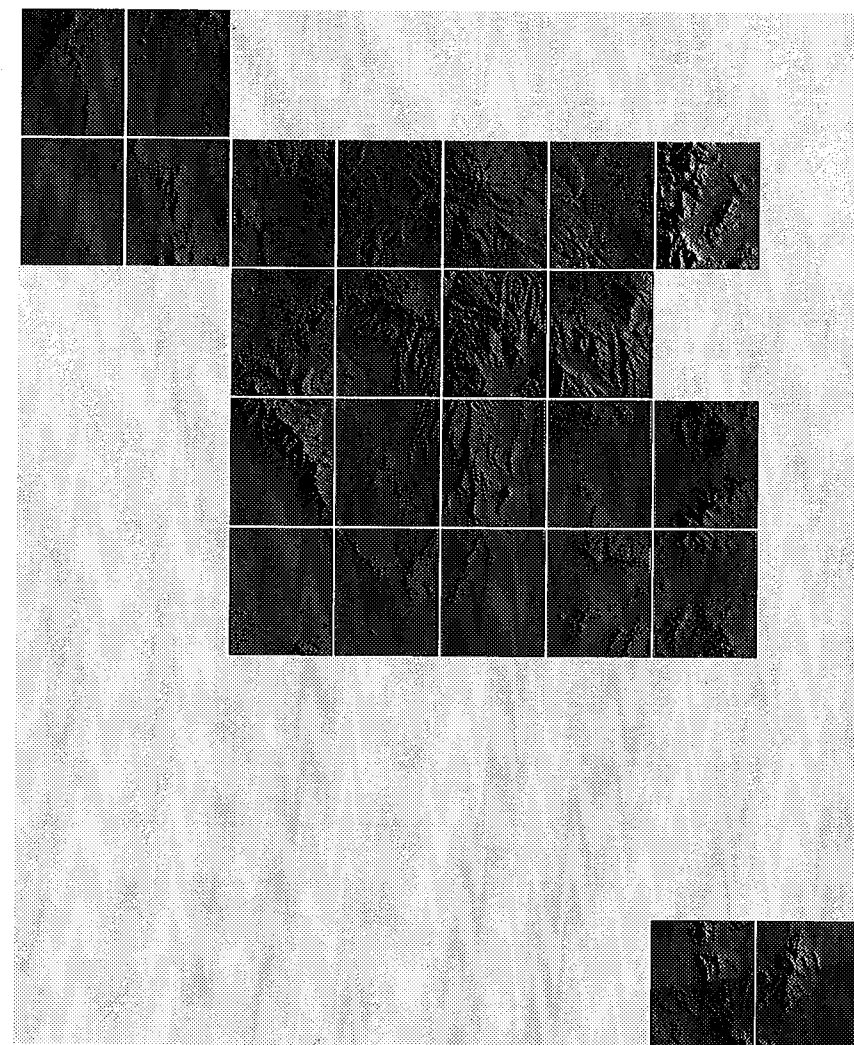
file04.dat
craterflat.dem
bustedbutte.dem
jackassflats.dem

file02.dat
eastbeatty.dem
topopahspgnw.dem
topopahspg.dem

file13.dat
file12.dat
file14.dat
file03.dat

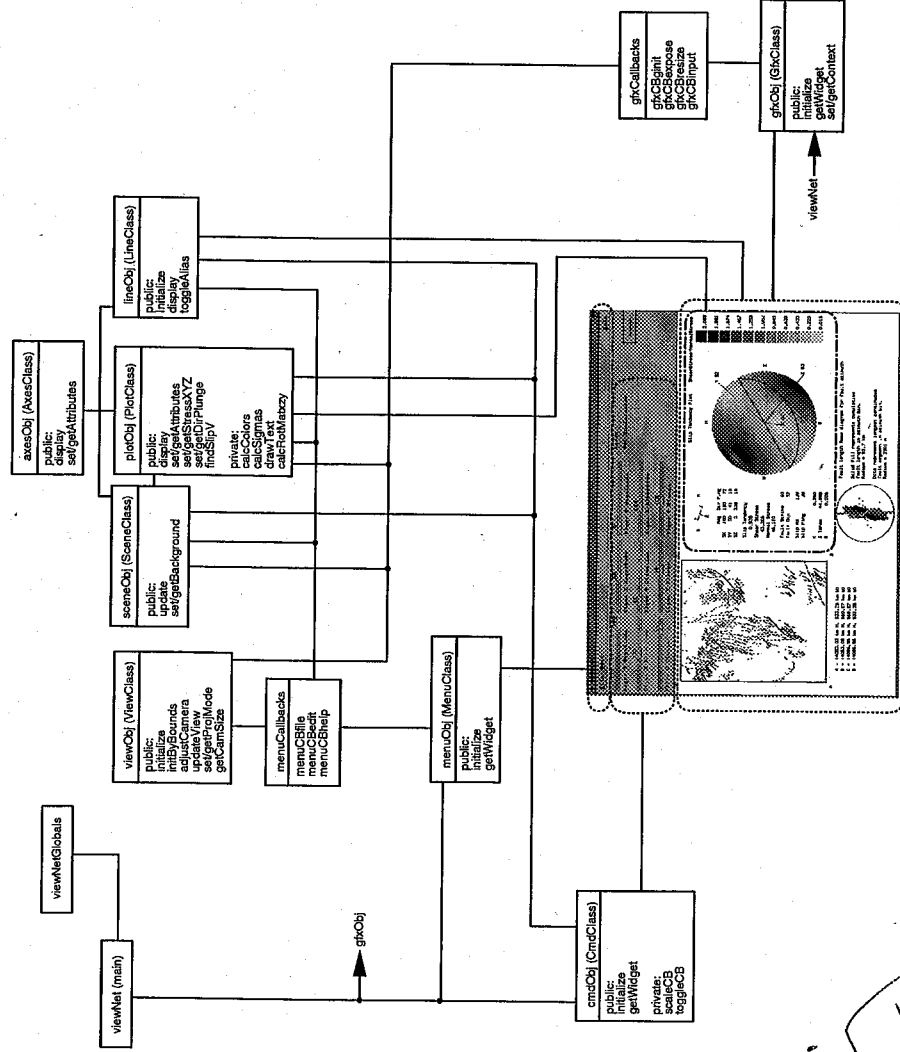
R. 1 L. 3-27-95

30-m DEMs:



TR. 21. L. 3-27-95

Block Diagram of ViewNet Objects



ViewNet Block Diagram blockDiagram.sc 5-29-95

1/3

1/3

```
1/3/95
out-1
for any in 0
do
  case some in
  1) ax=axy
  2) ay=axy
  3) az=axy
  4) tx=axy
  5) ty=axy
  6) tz=axy
  7) fstrike=axy
  8) fdip=axy
  9) atrike=axy
  10)
  11)
  12)
  13)
  14)
  15)
  16)
  17)
  18)
  19)
  20)
  21)
  22)
  23)
  24)
  25)
  26)
  27)
  28)
  29)
  30)
  31)
  32)
  33)
  34)
  35)
  36)
  37)
  38)
  39)
  40)
  41)
  42)
  43)
  44)
  45)
  46)
  47)
  48)
  49)
  50)
  51)
  52)
  53)
  54)
  55)
  56)
  57)
  58)
  59)
  60)
  61)
  62)
  63)
  64)
  65)
  66)
  67)
  68)
  69)
  70)
  71)
  72)
  73)
  74)
  75)
  76)
  77)
  78)
  79)
  80)
  81)
  82)
  83)
  84)
  85)
  86)
  87)
  88)
  89)
  90)
  91)
  92)
  93)
  94)
  95)
  96)
  97)
  98)
  99)
  100)
  101)
  102)
  103)
  104)
  105)
  106)
  107)
  108)
  109)
  110)
  111)
  112)
  113)
  114)
  115)
  116)
  117)
  118)
  119)
  120)
  121)
  122)
  123)
  124)
  125)
  126)
  127)
  128)
  129)
  130)
  131)
  132)
  133)
  134)
  135)
  136)
  137)
  138)
  139)
  140)
  141)
  142)
  143)
  144)
  145)
  146)
  147)
  148)
  149)
  150)
  151)
  152)
  153)
  154)
  155)
  156)
  157)
  158)
  159)
  160)
  161)
  162)
  163)
  164)
  165)
  166)
  167)
  168)
  169)
  170)
  171)
  172)
  173)
  174)
  175)
  176)
  177)
  178)
  179)
  180)
  181)
  182)
  183)
  184)
  185)
  186)
  187)
  188)
  189)
  190)
  191)
  192)
  193)
  194)
  195)
  196)
  197)
  198)
  199)
  200)
  201)
  202)
  203)
  204)
  205)
  206)
  207)
  208)
  209)
  210)
  211)
  212)
  213)
  214)
  215)
  216)
  217)
  218)
  219)
  220)
  221)
  222)
  223)
  224)
  225)
  226)
  227)
  228)
  229)
  230)
  231)
  232)
  233)
  234)
  235)
  236)
  237)
  238)
  239)
  240)
  241)
  242)
  243)
  244)
  245)
  246)
  247)
  248)
  249)
  250)
  251)
  252)
  253)
  254)
  255)
  256)
  257)
  258)
  259)
  260)
  261)
  262)
  263)
  264)
  265)
  266)
  267)
  268)
  269)
  270)
  271)
  272)
  273)
  274)
  275)
  276)
  277)
  278)
  279)
  280)
  281)
  282)
  283)
  284)
  285)
  286)
  287)
  288)
  289)
  290)
  291)
  292)
  293)
  294)
  295)
  296)
  297)
  298)
  299)
  300)
  301)
  302)
  303)
  304)
  305)
  306)
  307)
  308)
  309)
  310)
  311)
  312)
  313)
  314)
  315)
  316)
  317)
  318)
  319)
  320)
  321)
  322)
  323)
  324)
  325)
  326)
  327)
  328)
  329)
  330)
  331)
  332)
  333)
  334)
  335)
  336)
  337)
  338)
  339)
  340)
  341)
  342)
  343)
  344)
  345)
  346)
  347)
  348)
  349)
  350)
  351)
  352)
  353)
  354)
  355)
  356)
  357)
  358)
  359)
  360)
  361)
  362)
  363)
  364)
  365)
  366)
  367)
  368)
  369)
  370)
  371)
  372)
  373)
  374)
  375)
  376)
  377)
  378)
  379)
  380)
  381)
  382)
  383)
  384)
  385)
  386)
  387)
  388)
  389)
  390)
  391)
  392)
  393)
  394)
  395)
  396)
  397)
  398)
  399)
  400)
  401)
  402)
  403)
  404)
  405)
  406)
  407)
  408)
  409)
  410)
  411)
  412)
  413)
  414)
  415)
  416)
  417)
  418)
  419)
  420)
  421)
  422)
  423)
  424)
  425)
  426)
  427)
  428)
  429)
  430)
  431)
  432)
  433)
  434)
  435)
  436)
  437)
  438)
  439)
  440)
  441)
  442)
  443)
  444)
  445)
  446)
  447)
  448)
  449)
  450)
  451)
  452)
  453)
  454)
  455)
  456)
  457)
  458)
  459)
  460)
  461)
  462)
  463)
  464)
  465)
  466)
  467)
  468)
  469)
  470)
  471)
  472)
  473)
  474)
  475)
  476)
  477)
  478)
  479)
  480)
  481)
  482)
  483)
  484)
  485)
  486)
  487)
  488)
  489)
  490)
  491)
  492)
  493)
  494)
  495)
  496)
  497)
  498)
  499)
  500)
  501)
  502)
  503)
  504)
  505)
  506)
  507)
  508)
  509)
  510)
  511)
  512)
  513)
  514)
  515)
  516)
  517)
  518)
  519)
  520)
  521)
  522)
  523)
  524)
  525)
  526)
  527)
  528)
  529)
  530)
  531)
  532)
  533)
  534)
  535)
  536)
  537)
  538)
  539)
  540)
  541)
  542)
  543)
  544)
  545)
  546)
  547)
  548)
  549)
  550)
  551)
  552)
  553)
  554)
  555)
  556)
  557)
  558)
  559)
  560)
  561)
  562)
  563)
  564)
  565)
  566)
  567)
  568)
  569)
  570)
  571)
  572)
  573)
  574)
  575)
  576)
  577)
  578)
  579)
  580)
  581)
  582)
  583)
  584)
  585)
  586)
  587)
  588)
  589)
  590)
  591)
  592)
  593)
  594)
  595)
  596)
  597)
  598)
  599)
  600)
  601)
  602)
  603)
  604)
  605)
  606)
  607)
  608)
  609)
  610)
  611)
  612)
  613)
  614)
  615)
  616)
  617)
  618)
  619)
  620)
  621)
  622)
  623)
  624)
  625)
  626)
  627)
  628)
  629)
  630)
  631)
  632)
  633)
  634)
  635)
  636)
  637)
  638)
  639)
  640)
  641)
  642)
  643)
  644)
  645)
  646)
  647)
  648)
  649)
  650)
  651)
  652)
  653)
  654)
  655)
  656)
  657)
  658)
  659)
  660)
  661)
  662)
  663)
  664)
  665)
  666)
  667)
  668)
  669)
  670)
  671)
  672)
  673)
  674)
  675)
  676)
  677)
  678)
  679)
  680)
  681)
  682)
  683)
  684)
  685)
  686)
  687)
  688)
  689)
  690)
  691)
  692)
  693)
  694)
  695)
  696)
  697)
  698)
  699)
  700)
  701)
  702)
  703)
  704)
  705)
  706)
  707)
  708)
  709)
  710)
  711)
  712)
  713)
  714)
  715)
  716)
  717)
  718)
  719)
  720)
  721)
  722)
  723)
  724)
  725)
  726)
  727)
  728)
  729)
  730)
  731)
  732)
  733)
  734)
  735)
  736)
  737)
  738)
  739)
  740)
  741)
  742)
  743)
  744)
  745)
  746)
  747)
  748)
  749)
  750)
  751)
  752)
  753)
  754)
  755)
  756)
  757)
  758)
  759)
  760)
  761)
  762)
  763)
  764)
  765)
  766)
  767)
  768)
  769)
  770)
  771)
  772)
  773)
  774)
  775)
  776)
  777)
  778)
  779)
  780)
  781)
  782)
  783)
  784)
  785)
  786)
  787)
  788)
  789)
  790)
  791)
  792)
  793)
  794)
  795)
  796)
  797)
  798)
  799)
  800)
  801)
  802)
  803)
  804)
  805)
  806)
  807)
  808)
  809)
  810)
  811)
  812)
  813)
  814)
  815)
  816)
  817)
  818)
  819)
  820)
  821)
  822)
  823)
  824)
  825)
  826)
  827)
  828)
  829)
  830)
  831)
  832)
  833)
  834)
  835)
  836)
  837)
  838)
  839)
  840)
  841)
  842)
  843)
  844)
  845)
  846)
  847)
  848)
  849)
  850)
  851)
  852)
  853)
  854)
  855)
  856)
  857)
  858)
  859)
  860)
  861)
  862)
  863)
  864)
  865)
  866)
  867)
  868)
  869)
  870)
  871)
  872)
  873)
  874)
  875)
  876)
  877)
  878)
  879)
  880)
  881)
  882)
  883)
  884)
  885)
  886)
  887)
  888)
  889)
  890)
  891)
  892)
  893)
  894)
  895)
  896)
  897)
  898)
  899)
  900)
  901)
  902)
  903)
  904)
  905)
  906)
  907)
  908)
  909)
  910)
  911)
  912)
  913)
  914)
  915)
  916)
  917)
  918)
  919)
  920)
  921)
  922)
  923)
  924)
  925)
  926)
  927)
  928)
  929)
  930)
  931)
  932)
  933)
  934)
  935)
  936)
  937)
  938)
  939)
  940)
  941)
  942)
  943)
  944)
  945)
  946)
  947)
  948)
  949)
  950)
  951)
  952)
  953)
  954)
  955)
  956)
  957)
  958)
  959)
  960)
  961)
  962)
  963)
  964)
  965)
  966)
  967)
  968)
  969)
  970)
  971)
  972)
  973)
  974)
  975)
  976)
  977)
  978)
  979)
  980)
  981)
  982)
  983)
  984)
  985)
  986)
  987)
  988)
  989)
  990)
  991)
  992)
  993)
  994)
  995)
  996)
  997)
  998)
  999)
  1000)
  1001)
  1002)
  1003)
  1004)
  1005)
  1006)
  1007)
  1008)
  1009)
  1010)
  1011)
  1012)
  1013)
  1014)
  1015)
  1016)
  1017)
  1018)
  1019)
  1020)
  1021)
  1022)
  1023)
  1024)
  1025)
  1026)
  1027)
  1028)
  1029)
  1030)
  1031)
  1032)
  1033)
  1034)
  1035)
  1036)
  1037)
  1038)
  1039)
  1040)
  1041)
  1042)
  1043)
  1044)
  1045)
  1046)
  1047)
  1048)
  1049)
  1050)
  1051)
  1052)
  1053)
  1054)
  1055)
  1056)
  1057)
  1058)
  1059)
  1060)
  1061)
  1062)
  1063)
  1064)
  1065)
  1066)
  1067)
  1068)
  1069)
  1070)
  1071)
  1072)
  1073)
  1074)
  1075)
  1076)
  1077)
  1078)
  1079)
  1080)
  1081)
  1082)
  1083)
  1084)
  1085)
  1086)
  1087)
  1088)
  1089)
  1090)
  1091)
  1092)
  1093)
  1094)
  1095)
  1096)
  1097)
  1098)
  1099)
  1100)
  1101)
  1102)
  1103)
  1104)
  1105)
  1106)
  1107)
  1108)
  1109)
  1110)
  1111)
  1112)
  1113)
  1114)
  1115)
  1116)
  1117)
  1118)
  1119)
  1120)
  1121)
  1122)
  1123)
  1124)
  1125)
  1126)
  1127)
  1128)
  1129)
  1130)
  1131)
  1132)
  1133)
  1134)
  1135)
  1136)
  1137)
  1138)
  1139)
  1140)
  1141)
  1142)
  1143)
  1144)
  1145)
  1146)
  1147)
  1148)
  1149)
  1150)
  1151)
  1152)
  1153)
  1154)
  1155)
  1156)
  1157)
  1158)
  1159)
  1160)
  1161)
  1162)
  1163)
  1164)
  1165)
  1166)
  1167)
  1168)
  1169)
  1170)
  1171)
  1172)
  1173)
  1174)
  1175)
  1176)
  1177)
  1178)
  1179)
  1180)
  1181)
  1182)
  1183)
  1184)
  1185)
  1186)
  1187)
  1188)
  1189)
  1190)
  1191)
  1192)
  1193)
  1194)
  1195)
  1196)
  1197)
  1198)
  1199)
  1200)
  1201)
  1202)
  1203)
  1204)
  1205)
  1206)
  1207)
  1208)
  1209)
  1210)
  1211)
  1212)
  1213)
  1214)
  1215)
  1216)
  1217)
  1218)
  1219)
  1220)
  1221)
  1222)
  1223)
  1224)
  1225)
  1226)
  1227)
  1228)
  1229)
  1230)
  1231)
  1232)
  1233)
  1234)
  1235)
  1236)
  1237)
  1238)
  1239)
  1240)
  1241)
  1242)
  1243)
  1244)
  1245)
  1246)
  1247)
  1248)
  1249)
  1250)
  1251)
  1252)
  1253)
  1254)
  1255)
  1256)
  1257)
  1258)
  1259)
  1260)
  1261)
  1262)
  1263)
  1264)
  1265)
  1266)
  1267)
  1268)
  1269)
  1270)
  1271)
  1272)
  1273)
  1274)
  1275)
  1276)
  1277)
  1278)
  1279)
  1280)
  1281)
  1282)
  1283)
  1284)
  1285)
  1286)
  1287)
  1288)
  1289)
  1290)
  1291)
  1292)
  1293)
  1294)
  1295)
  1296)
  1297)
  1298)
  1299)
  1300)
  1301)
  1302)
  1303)
  1304)
  1305)
  1306)
  1307)
  1308)
  1309)
  1310)
  1311)
  1312)
  1313)
  1314)
  1315)
  1316)
  1317)
  1318)
  1319)
  1320)
  1321)
  1322)
  1323)
  1324)
  1325)
  1326)
  1327)
  1328)
  1329)
  1330)
  1331)
  1332)
  1333)
  1334)
  1335)
  1336)
  1337)
  1338)
  1339)
  1340)
  1341)
  1342)
  1343)
  1344)
  1345)
  1346)
  1347)
  1348)
  1349)
  1350)
  1351)
  1352)
  1353)
  1354)
  1355)
  1356)
  1357)
  1358)
  1359)
  1360)
  1361)
  1362)
  1363)
  1364)
  1365)
  1366)
  1367)
  1368)
  1369)
  1370)
  1371)
  1372)
  1373)
  1374)
  1375)
  1376)
  1377)
  1378)
  1379)
  1380)
  1381)
  1382)
  1383)
  1384)
  1385)
  1386)
  1387)
  1388)
  1389)
  1390)
  1391)
  1392)
  1393)
  1394)
  1395)
  1396)
  1397)
  1398)
  1399)
  1400)
  1401)
  1402)
  1403)
  1404)
  1405)
  1406)
  1407)
  1408)
  1409)
  1410)
  1411)
  1412)
  1413)
  1414)
  1415)
  1416)
  1417)
  1418)
  1419)
  1420)
  1421)
  1422)
  1423)
  1424)
  1425)
  1426)
  1427)
  1428)
  1429)
  1430)
  1431)
  1432)
  1433)
  1434)
  1435)
  1436)
  1437)
  1438)
  1439)
  1440)
  1441)
  1442)
  1443)
  1444)
  1445)
  1446)
  1447)
  1448)
  1449)
  1450)
  1451)
  1452)
  1453)
  1454)
  1455)
  1456)
  1457)
  1458)
  1459)
  1460)
  1461)
  1462)
  1463)
  1464)
  1465)
  1466)
  1467)
  1468)
  1469)
  1470)
  1471)
  1472)
  1473)
  1474)
  1475)
  1476)
  1477)
  1478)
  1479)
  1480)
  1481)
  1482)
  1483)
  1484)
  1485)
  1486)
  1487)
  1488)
  1489)
  1490)
  1491)
  1492)
  1493)
  1494)
  1495)
  1496)
  1497)
  1498)
  1499)
  1500)
  1501)
  1502)
  1503)
  1504)
  1505)
  1506)
  1507)
  1508)
  1509)
  1510)
  1511)
  1512)
  1513)
  1514)
  1515)
  1516)
  1517)
  1518)
  1519)
  1520)
  1521)
  1522)
  1523)
  1524)
  1525)
  1526)
  1527)
  1528)
  1529)
  1530)
  1531)
  1532)
  1533)
  1534)
  1535)
  1536)
  1537)
  1538)
  1539)
  1540)
  1541)
  1542)
  1543)
  1544)
  1545)
  1546)
  1547)
  1548)
  1549)
  1550)
  1551)
  1552)
  1553)
  1554)
  1555)
  1556)
  1557)
  1558)
  1559)
  1560)
  1561)
  1562)
  1563)
  1564)
  1565)
  1566)
  1567)
  1568)
  1569)
  1570)
  1571)
  1572)
  1573)
  1574)
  1575)
  1576)
  1577)
  1578)
  1579)
  1580)
  1581)
  1582)
  1583)
  1584)
  1585)
  1586)
  1587)
  1588)
  1589)
  1590)
  1591)
  1592)
  1593)
  1594)
  1595)
  1596)
  1597)
  1598)
  1599)
  1600)
  1601)
  1602)
  1603)
  1604)
  1605)
  1606)
  1607)
  1608)
  1609)
  1610)
  1611)
  1612)
  1613)
  1614)
  1615)
  1616)
  1617)
  1618)
  1619)
  1620)
  1621)
  1622)
  1623)
  1624)
  1625)
  1626)
  1627)
  1628)
  1629)
  1630)
  1631)
  1632)
  1633)
  1634)
  1635)
  1636)
  1637)

```

```

, xx=$x yy=$y zz=$z rx=$x ry=$y rz=$z fatike=$otike fdio=$dio sat=$sat bo=$bo

```

Script Tempang Mark

27 June

NAWK SCRIPT FOR DETERMINING STRIKES AND DIPS OF POLES TO A

Continuation of ARC Info AML:

04/13/95

13:52:57

basins2.

```

/* linecolor blue
/* arcs runhgeo

patch 3.50 1.5 6.55 2.0
linesymbol 5
linecolor green
line 3.55 1.95 4.05 1.95
linecolor red
line 3.55 1.65 4.05 1.65
/* linecolor cyan
/* line 3.55 1.35 4.05 1.35
/* linesymbol 9
/* linecolor yellow
/* line 3.55 1.05 4.05 1.05
/* linecolor blue
/* line 3.55 .75 4.05 .75

textsize 10 pt
move 4.25 1.90
text "Bedinger, Sargent, & Langer (1990)"
move 4.25 1.60
text "Barbey and Prudic (1991)"
/* move 4.25 1.30
/* text "Rice (1984)"
/* move 4.25 1.00
/* text "Waddell, Robison, Blankennagel (1984)"
/* move 4.25 .70
/* text "Rush (1970)"

map end
display 1040
docuscale.gra
plot docuscale.map
quit
extension 9999
display 9999 3
/* postscript docuscale.gra docuscale.pa
end

```

Buckler

12-20-95

Continuation of ARC Info AML:

04/13/95

13:52:48

s1.wat

```

Arc: |> lc <|
Workspace: /U5/RELAR

Available Coverages
-----
ARC: |> describe mexcitybox <|
Description of SINGLE precision coverage mexcitybox

ARCS                                POLYGONS
Arcs = 4                            Polygons = 0
Segments = 4                        There is NO Polygon Topology.
0 bytes of Arc Attribute Data      0 bytes of Polygon Attribute Data

NODES                                POINTS
Nodes = 0                            Label Points = 1
0 bytes of Node Attribute Data

TOLERANCES                          SECONDARY FEATURES
Fuzzy = 0.002 N                      Tics = 4
Dangle = 0.000 N                      Links = 0

COVERAGE BOUNDARY
Xmin = -109.000                      Ymin = 18.000
Xmax = -98.000                      Ymax = 20.000

STATUS
The coverage has not been Edited since the last BUILD or CLEAN.

NO COORDINATE SYSTEM DEFINED
Arc: |> cp /pscr0/brent/mexico2/bwRgs/mexSmall.ras . <|
Submitting command to Operating System ...
Arc: |> ls <|
Submitting command to Operating System ...
Arc: |> register <|
Usage: REGISTER <image> [coverage] [linecolor] [band | COMPOSITE] [red_band]
[green_band] [blue_band]
Arc: |> register mexSmall.ras mexcitybox <|

```

Buckler

12-20-95

Lineament extractions from DEMs:

Title:
**Matching Geological Faults Using
 Image Processing Techniques
 Applied to Hill-Shaded Digital Elevation Models**

Technical Area:
Geologic and Remote Sensing Applications

Authors:
**D. Brent Henderson
 David A. Ferrill**

Center for Nuclear Waste Regulatory Analyses
 Southwest Research Institute
 6220 Culebra Road, San Antonio, Texas 78238
 Tel: (210) 522-3747 Fax: (210) 522-5155
 Email: BHenderson@swri.edu

Summary submitted to IEEE Southwest Symposium on
 Image Analysis and Interpretation,
 April 8-9, 1996, San Antonio, Texas

November 13, 1995

Brent Henderson
 1/22/96

Cont...

**Matching Geological Faults Using
 Image Processing Techniques
 Applied to Hill-Shaded Digital Elevation Models**

D. Brent Henderson and David A. Ferrill

Center for Nuclear Waste Regulatory Analyses,
 Southwest Research Institute, 6220 Culebra Road, San Antonio, Texas 78238

Introduction

Information potentially subject to copyright protection
 was redacted from this location. The redacted material
 is from an article written for the referenced IEEE
 Symposium.

Brent Henderson
 1/22/96

cont...

Methodology

Results

Information potentially subject to copyright protection was redacted from this section and the methodology section. The redacted material is from the referenced material on pp. 66-67.

Brent Henderson
1/22/96

Image processing code:

01/22/96
10:54:13

imageProc.c++

1

```

//=====
// Filename: imageProc.c++
// Author: Brent Henderson
// Date: 11-29-95
//
// Developed by the Center for Nuclear Waste Regulatory
// Analyses (CNWRA), Southwest Research Institute (SWRI),
// San Antonio, Texas, USA.
//
// Copyright 07/20/95 Southwest Research Institute
// All rights reserved.
//
// This software is a trade secret owned by Southwest Research
// Institute, with access limited except as required for use by
// authorized users.
//
// This program was developed under sponsorship of the U.S.
// Nuclear Regulatory Commission, contract number NEC-02-93-005.
//
// Purpose:
// Image processing routines.
//
// Header:
//
// Revision History
// $Log: imageProc.c++,v $
//
//=====
//
// Includes
#include <stdio.h>
#include <iostream.h>
#include <math.h>

#include "imageObj.hh"
#include "viewerObj.hh"
#include "kernel.hh"

// Function prototypes
void ipGet8(unsigned short *, // data
int, // x
int, // y
int, // width
int, // height
unsigned short, // threshold
unsigned short [8]);

void ipGet9(unsigned short *, // data
int, // x
int, // y
int, // width
int, // height
float [9]);

short ipTransCnt(unsigned short [8]);
short ipNonZeros(unsigned short [8]);

// Functions
void ipThreshold(unsigned short threshold) {
    unsigned short *d = NULL;
    int width;
    int height;
    // Begin
    cout << "Begin ipThreshold." << endl << flush;
    imageObj.GetData(d, width, height);
    if (d == NULL) return;
    int n = 0;
    for (int y=0; y < height; y++) {
        for (int x=0; x < width; x++) {
            if (*(d+n) < threshold) {
                *(d+n) = 0;
            } else {
                *(d+n) = 65535;
            }
            n++;
        }
    }
}

```

```

}
viewerObj.display();
} // End

void ipSkeletonize(void) {
    unsigned short *d = NULL;
    int width;
    int height;
    unsigned short threshold = 32768;
    unsigned short p[8];
    short *f;
    short tr;
    short nz;
    short foundNodeToDelete = 1;
    short loopCnt = 0;

    // Begin
    cout << "Begin ipSkeletonize." << endl << flush;
    imageObj.GetData(d, width, height);
    if (d == NULL) return;

    // Allocate delete flag space
    f = (short *)XtMalloc(sizeof(short) * width * height);
    if (f == NULL) {
        cout << "Error cannot malloc f - "
        << width*height << endl << flush;
    }

    while ((foundNodeToDelete) && (loopCnt < 10000)) {
        foundNodeToDelete = 0;
        loopCnt++;
        //-----
        // Step 1: flag border points for deletion
        //-----
        int n = 0;
        for (int y=0; y < height; y++) {
            for (int x=0; x < width; x++) {
                *(f+n) = 0;
                if (*(d+n) < threshold) {
                    // Get 8 neighbors p2-p9
                    ipGet8(d, x, y, width, height, threshold, p);
                    tr = ipTransCnt(p);
                    nz = ipNonZeros(p);
                    if ((nz < 7) &&
                        (nz > 1) &&
                        ((tr == 1) &&
                         ((p[0] * p[2] * p[4]) == 0) &&
                         ((p[2] * p[4] * p[6]) == 0))) {
                        // Mark node to be deleted
                        *(f+n) = 1;
                        foundNodeToDelete++;
                    } // end if d < threshold
                }
                n++;
            } // end for x
        } // end for y

        // Step 2: delete flagged points from step 1
        //-----
        n = 0;
        for (y=0; y < height; y++) {
            for (int x=0; x < width; x++) {
                if (*(f+n) == 1) {
                    *(d+n) = 65535;
                }
                n++;
            } // end for x
        } // end for y

        // Step 3: flag remaining border points for deletion
        //-----
        n = 0;
        for (y=0; y < height; y++) {
            for (int x=0; x < width; x++) {
                *(f+n) = 0;
            }
        }
    }
}

```

Brent Henderson 1/22/96

01/22/96
10:54:13

imageProc.c++

```

if ((d+n) < threshold) {
    // Get 8 neighbors p2-p9
    ipGet8nd(d, x, y, width, height, threshold, p);
    tr = ipTransCn(p);
    nz = ipNonZero(p);
    if ((nz < 7) &&
        (nz > 1) &&
        (tr == 1) &&
        ((p[0] * p[2] * p[6]) == 0) &&
        ((p[1] * p[4] * p[6]) == 0)) {
        // Mark node to be deleted
        *(f+n) = 1;
        foundNodeToDelete++;
    }
} // end if d < threshold
n++;
} // end for x
} // end for y

// Step 4: Delete flagged points from step 3
n = 0;
for (int y=0; y < height; y++) {
    for (int x=0; x < width; x++) {
        if (*(f+n) == 1) {
            *(d+n) = 55535;
            n++;
        } // end for x
    } // end for y

    cout << " Found " << foundNodeToDelete << " nodes to delete."
    << endl << flush;
    viewerObj.display();
    // while foundNodeToDelete
    if (f != NULL) XFree((char *)f);
} // End

void ipInvert( void ) {
    unsigned short *d = NULL;
    int width;
    int height;
    // Begin
    cout << "Begin ipInvert." << endl << flush;
    imageObj.getData(d, width, height);
    if (d == NULL) return;
    int n = 0;
    for (int y=0; y < height; y++) {
        for (int x=0; x < width; x++) {
            *(d+n) = 65535 - *(d+n);
            n++;
        }
    }
    viewerObj.display();
} // End

void ipAdd( int z ) {
    unsigned short *d = NULL;
    int width;
    int height;
    // Begin
    cout << "Begin ipAdd." << endl << flush;
    imageObj.getData(d, width, height);
    if (d == NULL) return;
    int n = 0;
    for (int y=0; y < height; y++) {
        for (int x=0; x < width; x++) {
            int ai = *(d+n) + z;
            if (ai < 0) ai = 0;
            if (ai > 65535) ai = 65535;
            *(d+n) = ai;
            n++;
        }
    }
    cout << endl << flush;
    viewerObj.display();
} // End

void ipMult( float z ) {
    unsigned short *d = NULL;
    int width;
    int height;
    // Begin
    cout << "Begin ipMult." << endl << flush;
    imageObj.getData(d, width, height);
    if (d == NULL) return;
    int n = 0;
    for (int y=0; y < height; y++) {
        for (int x=0; x < width; x++) {
            float mf = (float)*(d+n) * z;
            int ni = (int)floor(mf);
            if (ni < 0) ni = 0;
            if (ni > 65535) ni = 65535;
            *(d+n) = (unsigned short)ni;
            n++;
        }
    }
    viewerObj.display();
} // End

void ipListEq( void ) {
    unsigned short *d = NULL;
    int width;
    int height;
    unsigned short p;
    // Begin
    cout << "Begin ipListEq." << endl << flush;
    imageObj.getData(d, width, height);
    if (d == NULL) return;
    // Step 1: compute histogram of original image
    int *hist = (int *)XtMalloc(sizeof(int)*65536);
    if (hist == NULL) {
        cout << "Error, could not malloc hist."
        << endl << flush;
        return;
    }
    int n;
    for (n = 0; n < 65536; n++) {
        *(hist+n) = 0;
    }
    n = 0;
    for (int y=0; y < height; y++) {
        for (int x=0; x < width; x++) {
            p = *(d+n);
            *(hist+p) += 1;
            n++;
        }
    }
    // Step 2: compute new gray values
    float pixscale = 1.0/(float)(width*height);
    float ng;
    float *newgray =
        (float *)XtMalloc(sizeof(float)*65536);
    if (newgray == NULL) {
        cout << "Error, could not malloc newgray."
        << endl << flush;
        return;
    }
    int sum = 0;
    for (n = 0; n < 65536; n++) {
        sum += *(hist+n);
        ng = ((float)sum*65535.0)*pixscale;
        if (ng < 0.0) ng = 0.0;
        if (ng > 65535.0) ng = 65535.0;
        *(newgray+n) = ng;
    }
}

```

R A Lth 1/22/96

01/22/96
10:54:13

imageProc.c++

cont...

```

// Step 3: apply new gray values to image
n = 0;
for (int y=0; y < height; y++) {
    for (int x=0; x < width; x++) {
        p = *(d+n);
        *(d+n) = (unsigned short)floorf(*(newgray+p));
        n++;
    }
}

if (newgray != NULL) XFree((char *)newgray);
if (hist != NULL) XFree((char *)hist);
viewerObj.display();
} // End

void ipKernel3a( void ) {
    unsigned short *d = NULL;
    int width;
    int height;
    // Begin
    cout << "Begin ipKernel3." << endl << flush;
    imageObj.getData(d, width, height);
    if (d == NULL) return;
    // Prompt user for kernel
    kernelFile();

    void ipKernel3b( float n[9] ) {
        unsigned short *d = NULL;
        int width;
        int height;
        // Begin
        imageObj.getData(d, width, height);
        if (d == NULL) return;

        // Allocate filtered data space
        float *f = NULL;
        f = (float *)XtMalloc(sizeof(float) * width * height);
        if (f == NULL) {
            cout << "Error cannot malloc f = "
            << width*height << endl << flush;
        }

        // Kernel
        float m0 m3 m6;
        float m1 m4 m7;
        float m2 m5 m8;

        // Mult filters
        float p[9];
        int n = 0;
        for (int y=0; y < height; y++) {
            for (int x=0; x < width; x++) {
                *(f+n) = 0.0;
                // Get 8 neighbors
                ipGet8nd(d, x, y, width, height, p);
                for (int i = 0; i < 9; i++) {
                    // cout << i << " "
                    // << *(f+n)/256.0 << " "
                    // << p[i]/256.0 << endl;
                    *(f+n) += m[i] * p[i];
                }
                if (*(f+n) > 65535.0) *(f+n) = 65535.0;
                if (*(f+n) < 0.0) *(f+n) = 0.0;
                cout << *(f+n)/256.0 << endl << flush;
                n++;
            }
        }
        // Save results
        n = 0;
        for (int y=0; y < height; y++) {
            for (int x=0; x < width; x++) {
                *(d+n) = (unsigned short)floorf(*(f+n));
                n++;
            }
        }
    }

    // Internal functions
    void ipGet8n(unsigned short *d,
        int x,
        int y,
        int width,
        int height,
        unsigned short threshold,
        unsigned short p[8]) {
        // Begin
        // P7 P0 P1
        // P6 P D P2
        // P5 P4 P3
        // End
        if (x < 1) {
            // Lower-left edge
            p[0] = *(d+(x+1)*width);
            p[1] = *(d+(x+1)*(y+1)*width);
            p[2] = *(d+(x+1)*y*width);
            p[3] = *(d+(x+1)*y*width);
            p[4] = *(d+(x+1)*y*width);
            p[5] = *(d+(x+1)*y*width);
            p[6] = *(d+(x+1)*y*width);
            p[7] = *(d+(x+1)*y*width);
        } else if (y > (height-2)) {
            // Upper-left edge
            p[0] = *(d+(x)*width);
            p[1] = *(d+(x+1)*y*width);
            p[2] = *(d+(x+1)*y*width);
            p[3] = *(d+(x+1)*(y+1)*width);
            p[4] = *(d+(x+1)*y*width);
            p[5] = *(d+(x+1)*y*width);
            p[6] = *(d+(x+1)*y*width);
            p[7] = *(d+(x+1)*y*width);
        } else {
            // Left edge
            p[0] = *(d+(x)*width);
            p[1] = *(d+(x+1)*(y+1)*width);
            p[2] = *(d+(x+1)*y*width);
            p[3] = *(d+(x+1)*(y+1)*width);
            p[4] = *(d+(x+1)*y*width);
            p[5] = *(d+(x+1)*y*width);
            p[6] = *(d+(x+1)*y*width);
            p[7] = *(d+(x+1)*y*width);
        }
        // End
        if (x > (width-2)) {
            // Lower-right edge
            p[0] = *(d+(x-1)*width);
            p[1] = *(d+(x-1)*(y+1)*width);
            p[2] = *(d+(x-1)*y*width);
            p[3] = *(d+(x-1)*y*width);
            p[4] = *(d+(x-1)*y*width);
            p[5] = *(d+(x-1)*y*width);
            p[6] = *(d+(x-1)*y*width);
            p[7] = *(d+(x-1)*y*width);
        }
    }
}

```

R A Lth 1/22/96

Raster-to-vector conversion code:

01/22/96
10:54:44

ras2vec.c++

```

// =====
// File: ras2vec.c++
// Author: Brent Henderson
// Date: 11-29-95
//
// Developed by the Center for Nuclear Waste Regulatory
// Analyses (CNWRA), Southwest Research Institute (SwRI),
// San Antonio, Texas, USA.
//
// Copyright 07/20/95 Southwest Research Institute
// All rights reserved.
//
// This software is a trade secret owned by Southwest Research
// Institute, with access limited except as required for use by
// authorized users.
//
// This program was developed under sponsorship of the U.S.
// Nuclear Regulatory Commission, contract number NRC-02-93-005.
//
// Reference:
// Digital Image Processing, Rafael C. Gonzales
// ISBN 0-201-11026-1, Addison-Wesley, 1987
// pp. 398-402
//
// Purpose:
// Image processing routines.
//
// Header: 5
//
// Revision History
// $Log: ras2vec.c++,v $
//
// =====
// UNIX includes
#include <stdio.h>
#include <iostream.h>
#include <fstream.h>
#include <string.h>
//
// Xt/Motif includes
#include <Xm/Xm.h>
#include <Xm/Files.h>
//
// Application includes
#include "imageObj.hh"
#include "linObj.hh"
#include "linVecObj.hh"
#include "viewerObj.hh"
//
typedef struct {
    int lineid;
    int x;
    int y;
} RasLineType;
//
// Global vars
static RasLineType *rasLines = NULL;
static int rasFlag = NULL;
static int rasId = 100;
static int rasIdCnt = 0;
static int rasN[8] = { 0, 1, -1, 2, -2, 3, -3, 4 };
extern Widget topLevel;
//
// Function prototypes
//
void rasGet8(unsigned short *, // data
            int, // x
            int, // y
            int, // width
            int, // height
            unsigned short, // threshold
            unsigned short [8]);
void rasSrchNeighbors( unsigned short *, // data
            int, // width
            int, // height
            int, // threshold
            int, // x
            int, // y
            int, // lastn
            unsigned short p[8]);
void rasWriteVert( int, // x
            int, // y
            int, // width
            );
void rasFindXY ( int, // n
            int, // x
            int, // y
            int *, // xn
            int *, // yn
            );
void rasWriteLines( void);
void rasAcceptCB(Widget, XtPointer, XtPointer);
void rasCancelCB(Widget, XtPointer, XtPointer);
//
// =====
// Functions
//
void ras2vec( void ) {
    unsigned short *d = NULL;
    unsigned short threshold = 32768;
    int width;
    int height;
    //
    // Begin
    cout << "Begin ras2vec. * << endl << flush;
    imageObj *getdata(d, width, height);
    if (d == NULL) return;
    //
    // Allocate space for line rasFlag for each node
    rasFlag = (int *)XtMalloc(sizeof(int) * width * height);
    if (rasFlag == NULL) {
        cout << "Error cannot malloc rasFlag - *
        << width*height << endl << flush;
    }
    //
    // Allocate space for lineid data
    rasLines = (RasLineType *)XtMalloc(sizeof(RasLineType) * width * height);
    if (rasLines == NULL) {
        cout << "Error cannot malloc rasLines - *
        << width*height << endl << flush;
    }
    //
    // Step 1: initialize line numbers to 0
    //
    int n = 0;
    for (int y=0; y < height; y++) {
        for (int x=0; x < width; x++) {
            *(rasFlag+n) = 0;
            *(rasLines+n) = lineid = 0;
            n++;
        }
    }
    //
    // Step 2: search pixel by pixel for connectivity
    //
    for (y=0; y < height; y++) {
        for (x=0; x < width; x++) {
            if ((*(rasFlag+n) == 0) &&
                (*(d+n) < threshold)) {
                rasWriteVert(x,y,width);
                rasSrchNeighbors(d,width,height,threshold,x,y,0);
                rasId++;
            }
            // end if d < threshold
        }
    }
}

```

Brent Henderson 1/22/96

Cont...

01/22/96
10:54:44

ras2vec.c++

```

// =====
// Internal functions
//
void rasGet8(unsigned short *d,
            int x,
            int y,
            int width,
            int height,
            unsigned short threshold,
            unsigned short p[8]) {
    //
    // Begin
    //
    p[0] = d;
    p[1] = d;
    p[2] = d;
    p[3] = d;
    p[4] = d;
    p[5] = d;
    p[6] = d;
    p[7] = d;
    //
    // Lower-left edge
    if (x < 1) {
        if (y < 1) {
            // Lower-left edge
            offset[0] = (x*(y-1)*width);
            offset[1] = ((x-1)*(y-1)*width);
            offset[2] = ((x-1)*y*width);
            offset[3] = (x*(y-1)*width);
            offset[4] = -1;
            offset[5] = -1;
            offset[6] = -1;
            offset[7] = -1;
        }
        // Not on the edge
        offset[0] = (x*(y-1)*width);
        offset[1] = ((x-1)*(y-1)*width);
        offset[2] = ((x-1)*y*width);
        offset[3] = (x*(y-1)*width);
        offset[4] = (x*(y-1)*width);
        offset[5] = (x*(y-1)*width);
        offset[6] = (x*(y-1)*width);
        offset[7] = (x*(y-1)*width);
    }
    //
    // Left edge
    offset[0] = (x*(y-1)*width);
    offset[1] = ((x-1)*(y-1)*width);
    offset[2] = ((x-1)*y*width);
    offset[3] = (x*(y-1)*width);
    offset[4] = (x*(y-1)*width);
    offset[5] = (x*(y-1)*width);
    offset[6] = (x*(y-1)*width);
    offset[7] = (x*(y-1)*width);
    //
    // Lower-right edge
    if (x > (width-2)) {
        if (y < 1) {
            // Lower-right edge
            offset[0] = (x*(y-1)*width);
            offset[1] = ((x-1)*(y-1)*width);
            offset[2] = ((x-1)*y*width);
            offset[3] = (x*(y-1)*width);
            offset[4] = (x*(y-1)*width);
            offset[5] = (x*(y-1)*width);
            offset[6] = (x*(y-1)*width);
            offset[7] = (x*(y-1)*width);
        }
        // Not on the edge
        offset[0] = (x*(y-1)*width);
        offset[1] = ((x-1)*(y-1)*width);
        offset[2] = ((x-1)*y*width);
        offset[3] = (x*(y-1)*width);
        offset[4] = (x*(y-1)*width);
        offset[5] = (x*(y-1)*width);
        offset[6] = (x*(y-1)*width);
        offset[7] = (x*(y-1)*width);
    }
    //
    // Right edge
    offset[0] = (x*(y-1)*width);
    offset[1] = ((x-1)*(y-1)*width);
    offset[2] = ((x-1)*y*width);
    offset[3] = (x*(y-1)*width);
    offset[4] = (x*(y-1)*width);
    offset[5] = (x*(y-1)*width);
    offset[6] = (x*(y-1)*width);
    offset[7] = (x*(y-1)*width);
    //
    // Upper-left edge
    if (x < 1) {
        if (y > (height-2)) {
            // Upper-left edge
            offset[0] = (x*(y-1)*width);
            offset[1] = ((x-1)*(y-1)*width);
            offset[2] = ((x-1)*y*width);
            offset[3] = (x*(y-1)*width);
            offset[4] = (x*(y-1)*width);
            offset[5] = (x*(y-1)*width);
            offset[6] = (x*(y-1)*width);
            offset[7] = (x*(y-1)*width);
        }
        // Not on the edge
        offset[0] = (x*(y-1)*width);
        offset[1] = ((x-1)*(y-1)*width);
        offset[2] = ((x-1)*y*width);
        offset[3] = (x*(y-1)*width);
        offset[4] = (x*(y-1)*width);
        offset[5] = (x*(y-1)*width);
        offset[6] = (x*(y-1)*width);
        offset[7] = (x*(y-1)*width);
    }
    //
    // Upper-right edge
    if (x > (width-2)) {
        if (y > (height-2)) {
            // Upper-right edge
            offset[0] = (x*(y-1)*width);
            offset[1] = ((x-1)*(y-1)*width);
            offset[2] = ((x-1)*y*width);
            offset[3] = (x*(y-1)*width);
            offset[4] = (x*(y-1)*width);
            offset[5] = (x*(y-1)*width);
            offset[6] = (x*(y-1)*width);
            offset[7] = (x*(y-1)*width);
        }
        // Not on the edge
        offset[0] = (x*(y-1)*width);
        offset[1] = ((x-1)*(y-1)*width);
        offset[2] = ((x-1)*y*width);
        offset[3] = (x*(y-1)*width);
        offset[4] = (x*(y-1)*width);
        offset[5] = (x*(y-1)*width);
        offset[6] = (x*(y-1)*width);
        offset[7] = (x*(y-1)*width);
    }
    //
    // Not on the edge
    offset[0] = (x*(y-1)*width);
    offset[1] = ((x-1)*(y-1)*width);
    offset[2] = ((x-1)*y*width);
    offset[3] = (x*(y-1)*width);
    offset[4] = (x*(y-1)*width);
    offset[5] = (x*(y-1)*width);
    offset[6] = (x*(y-1)*width);
    offset[7] = (x*(y-1)*width);
    //
    // Get 8 neighbors p0-p7
    rasGet8(d, x, y, width, height, threshold, p);
    // Search neighbors starting with lastn
}

```

Brent Henderson 1/22/96

Cont...

01/22/96

10:54:44

ras2vec.c++

```

for (int i = 0; i < 8; i++) {
    int n = lastn + ras[i];
    if (n < 0) {
        n += 8;
    } else if (n > 7) {
        n -= 8;
    }
    if (p[n] == 1) {
        rasFindXY(n, x, y, xn, yn);
        rasWriteVert(xn, yn, width);
        rasSchMghbrs(d, width, height, threshold, xn, yn, n);
        break;
    }
} // end rasSchMghbrs

void rasWriteVert( int x,
                  int y,
                  int width ) {
    // Begin
    (rasLines+rasidCnt)-->lineid = rasid;
    (rasLines+rasidCnt)-->x = x;
    (rasLines+rasidCnt)-->y = y;
    rasidCnt++;
    // Mark node's rasflag
    *(rasFlags+(y*width)) = 1;
}

void rasFindXY ( int n,
                int x,
                int y,
                int *xn,
                int *yn ) {
    // Begin
    switch (n) {
        case 0:
            *xn = x;
            *yn = y+1;
            break;
        case 1:
            *xn = x+1;
            *yn = y+1;
            break;
        case 2:
            *xn = x+1;
            *yn = y;
            break;
        case 3:
            *xn = x+1;
            *yn = y-1;
            break;
        case 4:
            *xn = x;
            *yn = y-1;
            break;
        case 5:
            *xn = x-1;
            *yn = y-1;
            break;
        case 6:
            *xn = x-1;
            *yn = y;
            break;
        case 7:
            *xn = x-1;
            *yn = y+1;
            break;
    } // end switch
}

void rasWriteLines(void) {
    static Widget rasFileWidget;
    // Begin
    if (!rasFileWidget) {
        Arg args[1];
        XmString msg = XmStringCreateLtoR ("*.lin",

```

```

XmFONTLIST_DEFAULT_TAG);
XtSetArg (args[0], XmPattern, msg);
rasFileWidget = XmCreateFileSelectionDialog (
    topLevel,
    "LIN",
    args,
    1);
XtAddCallback (rasFileWidget, XmOkCallback,
               rasAcceptCB, NULL);
XtAddCallback (rasFileWidget, XmCancelCallback,
               rasCancelCB, NULL);
XmStringFree(msg);
}
XtManageChild (rasFileWidget);
XtPopup (XtParent (rasFileWidget), XtGrabNone);
} // end rasWriteLines

void rasAcceptCB(Widget w,
                XtPointer callData) {
    char *file = NULL;
    char buf[1024];
    XmFileSelectionBoxCallbackStruct *cb =
        (XmFileSelectionBoxCallbackStruct *) callData;
    // Begin
    if (cb) {
        if (!XmStringGetLtoR(cb->value, XmFONTLIST_DEFAULT_TAG, &file))
            return; // internal error
        (void) strcpy (buf, file);
        XtFree (file); // free allocated data from XmStringGetLtoR()

        // Open output file
        ofstream outFile(buf, ios::out);
        if (!outFile) {
            cerr << "File not opened: " << buf << endl;
            return;
        }

        // Get scale data so that we can transform
        // from pixel to map space.
        LinFileAttributeType *lattr = linObj.getAttributes();
        float scalex = 1.0/(lattr->scale[0]);
        float scaley = 1.0/(lattr->scale[1]);
        cout << "Pixel to map scale = " << " " << scalex
              << " " << scaley << endl << flush;

        // Need out lines with only 1 vertex
        int firstOne = 1;
        int lastId;
        int lastN;
        for (int n = 0; n < rasidCnt; n++) {
            if ((rasLines+n)-->lineid == 0) {
                break;
            } else {
                if (firstOne) {
                    lastId = (rasLines+n)-->lineid;
                    lastN = n;
                    firstOne = 0;
                } else {
                    if ((lastId != (rasLines+n)-->lineid) {
                        if ((n - lastN) < 2) {
                            (rasLines+lastN)-->lineid = 0;
                            lastId = (rasLines+n)-->lineid;
                            lastN = n;
                        }
                    }
                }
            }
        } // end for loop

        if ((rasLines+n-2)-->lineid != (rasLines+n-1)-->lineid) {
            (rasLines+n-1)-->lineid = 0;
        }

        // Write data to file
        firstOne = 1;
        lastId;
        for (n = 0; n < rasidCnt; n++) {

```

Cont...

01/22/96

10:54:44

ras2vec.c++

```

if ((rasLines+n)-->lineid != 0) {
    if (firstOne) {
        outFile << " " << (rasLines+n)-->lineid << endl;
        lastId = (rasLines+n)-->lineid;
        firstOne = 0;
    } else {
        if (lastId != (rasLines+n)-->lineid) {
            outFile << "END" << endl;
            outFile << " " << (rasLines+n)-->lineid << endl;
            lastId = (rasLines+n)-->lineid;
        }
        outFile << ((rasLines+n)-->x * scalex) + lattr->mapMin[0]
                  << " "
                  << ((rasLines+n)-->y * scaley) + lattr->mapMin[1]
                  << endl;
    } // end for loop
    outFile << "END" << endl;
    outFile << "END" << endl;
    outFile << flush;
    linVecObj.read(buf, 0);
    if (rasLines != NULL) XtFree((char *)rasLines);
    XtUnmanageChild (w);
    viewerObj.display();
} // end rasAcceptCB

void rasCancelCB ( Widget w,
                  XtPointer callData ) {
    XtUnmanageChild (w);
}

```

Butcher 1/22/96

Butcher 1/22/96

Program for finding candidate stress states:

Method for finding candidate stress states 12-22-95

1. Record sample slip vectors

For each stress state:

Sample lower hemisphere every 10 deg $(10 \times 36) + 1 = 361$:

Record 9-bit fltstrike
7-bit fltdip
9-bit slipaz
7-bit slippg $9+7+9+7\text{bits} = 4 \text{ bytes}$

Bytes / stress state = $361 \times 4 = 1444 \text{ bytes}$

Stress states at 10mag/10deg steps =

sx 0 - 100 = 11
sy 0 - 100 = 11
sz 0 - 100 = 11
dx 0 - 360 = 36

$(11^3) \times 36 = 47916 \text{ states}$

File size = $1444 \times 47916 = 70\text{MB}$ 140 MB for 4 disks (8.4gb)

(Assumes sx is always vertical)

2. Application for finding candidate stress states

Given: fltstrike fltdip slipaz slippg

Find: candidate stress states

```
findStress data1.bin 250 10 328 10 | sort > t1
=> t1 <=
0 Sxyz = 1 20 30 Rx = 10 Petrike = 250 Fdip = 10 Sz = 328 Spg = 10
0 Sxyz = 1 20 30 Rx = 20 Petrike = 250 Fdip = 10 Sz = 328 Spg = 10
0 Sxyz = 1 20 30 Rx = 30 Petrike = 250 Fdip = 10 Sz = 328 Spg = 10
0 Sxyz = 1 20 30 Rx = 190 Petrike = 250 Fdip = 10 Sz = 328 Spg = 10
0 Sxyz = 1 20 30 Rx = 200 Petrike = 250 Fdip = 10 Sz = 328 Spg = 10
0 Sxyz = 1 20 30 Rx = 210 Petrike = 250 Fdip = 10 Sz = 328 Spg = 10
0 Sxyz = 1 20 50 Rx = 50 Petrike = 250 Fdip = 10 Sz = 328 Spg = 10
0 Sxyz = 1 20 50 Rx = 230 Petrike = 250 Fdip = 10 Sz = 328 Spg = 10
0 Sxyz = 1 30 20 Rx = 100 Petrike = 250 Fdip = 10 Sz = 328 Spg = 10
0 Sxyz = 1 30 20 Rx = 110 Petrike = 250 Fdip = 10 Sz = 328 Spg = 10
```

```
findStress data1.bin 50 30 128 29 | sort > t2
=> t2 <=
0 Sxyz = 1 10 70 Rx = 40 Petrike = 50 Fdip = 30 Sz = 128 Spg = 29
0 Sxyz = 1 10 70 Rx = 220 Petrike = 50 Fdip = 30 Sz = 128 Spg = 29
0 Sxyz = 1 10 80 Rx = 40 Petrike = 50 Fdip = 30 Sz = 128 Spg = 29
0 Sxyz = 1 10 80 Rx = 220 Petrike = 50 Fdip = 30 Sz = 128 Spg = 29
0 Sxyz = 1 10 90 Rx = 40 Petrike = 50 Fdip = 30 Sz = 128 Spg = 29
0 Sxyz = 1 10 90 Rx = 220 Petrike = 50 Fdip = 30 Sz = 128 Spg = 29
0 Sxyz = 1 10 100 Rx = 40 Petrike = 50 Fdip = 30 Sz = 128 Spg = 29
0 Sxyz = 1 10 100 Rx = 220 Petrike = 50 Fdip = 30 Sz = 128 Spg = 29
0 Sxyz = 1 20 30 Rx = 50 Petrike = 50 Fdip = 30 Sz = 128 Spg = 29
0 Sxyz = 1 20 30 Rx = 160 Petrike = 50 Fdip = 30 Sz = 128 Spg = 29
```

```
findStress data1.bin 210 50 345 40 | sort > t3
=> t3 <=
0 Sxyz = 1 10 40 Rx = 50 Petrike = 210 Fdip = 50 Sz = 345 Spg = 40
0 Sxyz = 1 10 40 Rx = 240 Petrike = 210 Fdip = 50 Sz = 345 Spg = 40
0 Sxyz = 1 20 70 Rx = 110 Petrike = 210 Fdip = 50 Sz = 345 Spg = 40
0 Sxyz = 1 20 70 Rx = 290 Petrike = 210 Fdip = 50 Sz = 345 Spg = 40
0 Sxyz = 1 40 10 Rx = 150 Petrike = 210 Fdip = 50 Sz = 345 Spg = 40
0 Sxyz = 1 40 10 Rx = 330 Petrike = 210 Fdip = 50 Sz = 345 Spg = 40
0 Sxyz = 1 40 90 Rx = 80 Petrike = 210 Fdip = 50 Sz = 345 Spg = 40
0 Sxyz = 1 40 90 Rx = 90 Petrike = 210 Fdip = 50 Sz = 345 Spg = 40
0 Sxyz = 1 40 90 Rx = 260 Petrike = 210 Fdip = 50 Sz = 345 Spg = 40
0 Sxyz = 1 40 90 Rx = 270 Petrike = 210 Fdip = 50 Sz = 345 Spg = 40
```

But H 1/22/96

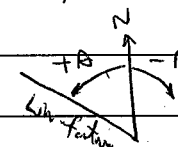
Image processing literature review:

From Lithologic and Structural Analysis of a part of Western Turkey By Using Landsat TM data, MS Thesis, Suleyman Sami Nalbant, Dec 91, U. of Nanch, Reno

p.77

$$\cos A * \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix} + \sin A * \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix} = \text{filter value}$$

where A is angle of linear feature to be enhanced



(from Johns '87)

Eliminate lineaments caused by drainage patterns.

From Lineament Analysis of Satellite Images Using a Segment Tracing Algorithm (STA), Katsuki Koike, S. Nagano, and M. Ohmi, Comp. & Geosciences, V21, N9, pp 1091-1104, 1995

faults + crustal fractures correspond with lineaments.

DEM → shaded relief map (60m resolution)

Landsat TM → (30m resolution)

Correlation coefficients between DEM shaded relief map vs. TM bands

Band 4 had highest correlation with DEM, therefore band 4 most suitable for lineament analysis. Also account for tree-covered slopes vs bare slopes.

Segment tracing - non-filtering approach, testing similarities + differences of pixel neighbors.

But H 1/22/96

Literature review cont.---

Preliminary draft of article by Evaluation of Aerial Photographic Evidence
for Faulting in the Area of the Potential Repository, YM, Nevada
by Charles W. Weisenburg.

Ghost Dance fault is major fault in local area.

Mostly Basin + Range is N-S striking faults, extensional.

Photolineaments mostly picked up cooling joints rather than
faults.

Selective Detection of Linear Features in Geological Remote Sensing Data

Jo Ann Parikh + John S. DePinto, E.G. D'Nieder + R.A. Pedersen

SPIE V.1709, Appl. of Art. Neural Netw. III 1992/963-972

Candidate regions for linear elements - image preprocessing

1. Smoothing filter 3×3 median

2. Sobel edge detector masks:

$$\begin{array}{ccc|ccc} -1 & -2 & -1 & 1 & 0 & -1 \\ 0 & 0 & 0 & 2 & 0 & -2 \\ 1 & 2 & 1 & 1 & 0 & -1 \\ \hline S_x & & & S_y & & \end{array}$$

$$MAG = [S_x^2 + S_y^2]^{1/2}$$

$$DIR = \arctan(S_y/S_x)$$

3. threshold edge map

4. thin threshold edge map

Butt Hahn 1/22/96

Write extracted lines to file, and keep only lines \geq three #verts

```
void rasWriteFile ( char *fname ) {
    // Begin
    // Open output file
    ofstream outFile(fname, ios::out);
    if (!outFile) {
        cerr << "File not opened: " << fname << endl;
        return;
    }

    // Get scale data so that we can transform
    // from pixel to map space.
    LinFileAttributeType *lattr = linObj.getAttributes();
    float scalex = 1.0/(lattr->scale[0]);
    float scaley = 1.0/(lattr->scale[1]);
    cout << "Pixel to map scale = " << " " << scalex
        << " " << scaley << endl << flush;

    //-----
    // Weed out lines with less than min verts
    //-----
    int firstOne = 1;
    int lastId;
    int lastN;
    int vcnt;
    for (int n = 0; n < rasVertCnt; n++) {
        if ((rasLines+n)->lineid != 0) {
            if (firstOne) {
                lastId = (rasLines+n)->lineid;
                lastN = n;
                vcnt = 1;
                firstOne = 0;
            } else {
                if (lastId == (rasLines+n)->lineid) {
                    // lastId equals current lineid
                    vcnt++;
                } else {
                    // lastId not equal to current lineid
                    if (vcnt < rasMinVerts) {
                        for (int v=lastN; v<n; v++) {
                            if ((rasLines+v)->lineid == lastId) {
                                (rasLines+v)->lineid = 0;
                            }
                        }
                    } else {
                        for (int v0=lastN; v0<n; v0++) {
                            cout << "v0 = " << v0 << endl << flush;
                            if ((rasLines+v0)->lineid != 0) {
                                for (int v1=v0+1; v1<n; v1++) {
                                    // cout << "v1 = " << v1 << endl << flush;
                                    if ((rasLines+v1)->lineid != 0) {
                                        for (int v2=v1+1; v2<n; v2++) {
                                            // cout << "v2 = " << v2 << endl << flush;
                                            if ((rasLines+v2)->lineid != 0) {
                                                float mx = (rasLines+v1)->x -
                                                    (rasLines+v0)->x;
                                                float my = (rasLines+v1)->y -
                                                    (rasLines+v0)->y;
                                                VectorClass mv(mx, my, 0.0);
                                                float md = mv.mag();

                                                float nx = (rasLines+v2)->x -
                                                    (rasLines+v1)->x;
                                                float ny = (rasLines+v2)->y -
                                                    (rasLines+v1)->y;
                                                VectorClass nv(nx, ny, 0.0);
                                                float nd = nv.mag();

                                                float dtop = md + nd;

                                                float px = (rasLines+v2)->x -
                                                    (rasLines+v0)->x;
                                                float py = (rasLines+v2)->y -
                                                    (rasLines+v0)->y;
                                                VectorClass pv(px, py, 0.0);
                                            }
                                        }
                                    }
                                }
                            }
                        }
                    }
                }
            }
        }
    }
}
```

Butt Hahn 2/12/96

*Skeleton section**Skeleton section*

Information potentially subject to copyright protection was redacted from pages 82 through 84. The redacted material is from the following reference:

Gonzalez, R.C. Representation and Description. "Digital Image Processing." Reading, Massachusetts: Addison-Wesley. pp. 398-401. 1987

*Bob Hahn 7/1/01**Bob Hahn 7/1/01*

Write line files &
cont...

Information potentially subject to copyright protection was redacted from this page. The redacted material is from the referenced cited on pages 82 and 83.

Brent H. 2/12/96

**MAPPING GEOLOGICAL FAULTS USING IMAGE PROCESSING TECHNIQUES
APPLIED TO HILL-SHADED DIGITAL ELEVATION MODELS**

D. Brent Henderson¹, David A. Ferrill¹, and Keith C. Clarke²

¹Center for Nuclear Waste Regulatory Analyses
Southwest Research Institute
San Antonio, Texas 78238 U.S.A.
bhenderson@swri.edu

²Department of Geology and Geography
Hunter College - City University of New York
New York, NY 10021 U.S.A.

Information potentially subject to copyright protection was redacted from pages 85-90. The redacted material is from the following reference:

Henderson, D.B., Ferrill, D.A., Clarke, K.C. "Mapping Geological Faults Using Image Processing Techniques Applied to Hill-Shaded Digital Elevation Models." Proceedings of the IEEE Southwest Symposium on Image Analysis, San Antonio, Texas. pp. 1-6. 1996.

Brent H. 2/12/96

Information potentially subject to copyright protection was redacted from these pages. The reference is located on page 85 of this scientific notebook.

Butcher 2/12/96

Butcher 2/12/96

Information potentially subject to copyright protection was redacted from these pages. The reference for the redacted material is on page 85 of this scientific notebook.

Paul H. 2/12/96

Paul H. 2/12/96

Information potentially subject to copyright protection was redacted from this page. The reference for the redacted material is on page 85 of this scientific notebook.

Burt H. 2/12/96

3DStress installation testing:

Verified operation of 3DStress on SGI Onyx:

install_test.v1.1 3-6-96

1. Launch 3dstress: % 3dstress

2. Set the sigma magnitudes and orientations using the sliders to the following settings:

Sigma X Magnitude 90
Sigma Y Magnitude 70
Sigma Z Magnitude 10

Sigma X Direction 28
Sigma X Plunge 90

Sigma Y Direction 28
Sigma Y Plunge 0

Sigma Z Direction 118
Sigma Z Plunge 0

3. Move the mouse over the slip tendency plot so that the white box corresponds to a fault with strike 198 and dip 72.

4. The following values should appear on the display near the slip tendency plot:

	Mag	Dir	Plng
SX	90	28	90
SY	70	28	0
SZ	10	118	0

Slip Tendency
1.301

Shear Stress
24.546

Normal Stress
18.870

Fault Strike 198
Fault Dip 72

Slip Az 341
Slip Plng 62

K 0.184
% TsMax 97.591
R 0.250

Software installation was
verified to be correctly functioning,
passed the tests.

Burt H. 3/6/96

SLIPTENDENCY SPREADSHEET

	A	B	C	D	E
1	sliptend.wks				=NOW()
2					
3					
4	Given:	FltStrike 50		Sx	100
5		FltDip 80		Sy	50
6				Sz	5
7					
8	Compute:	Flt dircos:		Ax	=SIN((90-C5)*PI()/180)
9				Ay	=COS((90-C5)*PI()/180)*COS((C4+90)*PI()/180)
10				Az	=COS((90-C5)*PI()/180)*SIN((C4+90)*PI()/180)
11				SN	=E8*E8*E4+E9*E9*E5+E10*E10*E6
12				Dx	=E4*E12
13				Dy	=E5*E12
14				Dz	=E6*E12
15				A	=SQRT(E8*E8*E14*E14+E9*E9*E15*E15+E10*E10*E16*E16)
16				Bx	=E8*E14/E17
17				By	=E9*E15/E17
18				Bz	=E10*E16/E17
19		Slip dircos:			
20				Ts	=(E8*E8*E14*E14+E9*E9*E15*E15+E10*E10*E16*E16)/E17
21					
22					
23					
24					
25		SlipTend			=E23/E12
26					
27	Find:	SlipAz			=(ATAN(E21/E20)*180/PI())
28					=IF(E31<0,E27-180,E27)
29					=IF(E28<0,E28+360,IF(E28>360,E28-360,E28))
30					
31		SlipPng			=(ASIN(E19)*180/PI())
32					=IF(E31<0,-E31,E31)
33					
34	Check:	FltStrike	=90+(ATAN(E36/E35)*180/PI())	Nx	=E8
35		FltDip	=90-(ASIN(E34)*180/PI())	Ny	=E9
36			Nz		=E10

Bmt H... 3/12/96

Cont...

sliptend.wks				3/12/96
Given:	FltStrike	50	Sx	100.000
	FltDip	80	Sy	50.000
			Sz	5.000
Compute:	Flt dircos:		Ax	0.174
			Ay	-0.754
			Az	0.633
			SN	33.475
			Dx	66.525
			Dy	16.525
			Dz	-28.475
			A	24.774
	Slip dircos:		Bx	0.466
			By	-0.503
			Bz	-0.728
			Ts	24.774
		SlipTend		0.740
Find:		SlipAz		55.333
				55.333
				55.333
		SlipPIng		27.793
				27.793
Check:	FltStrike	50	Nx	0.174
	FltDip	80	Ny	-0.754
			Nz	0.633

Bmt H... 3/12/96

RUNNING 3DSTRESS FROM THE COMMAND LINE (BATCH MODE)

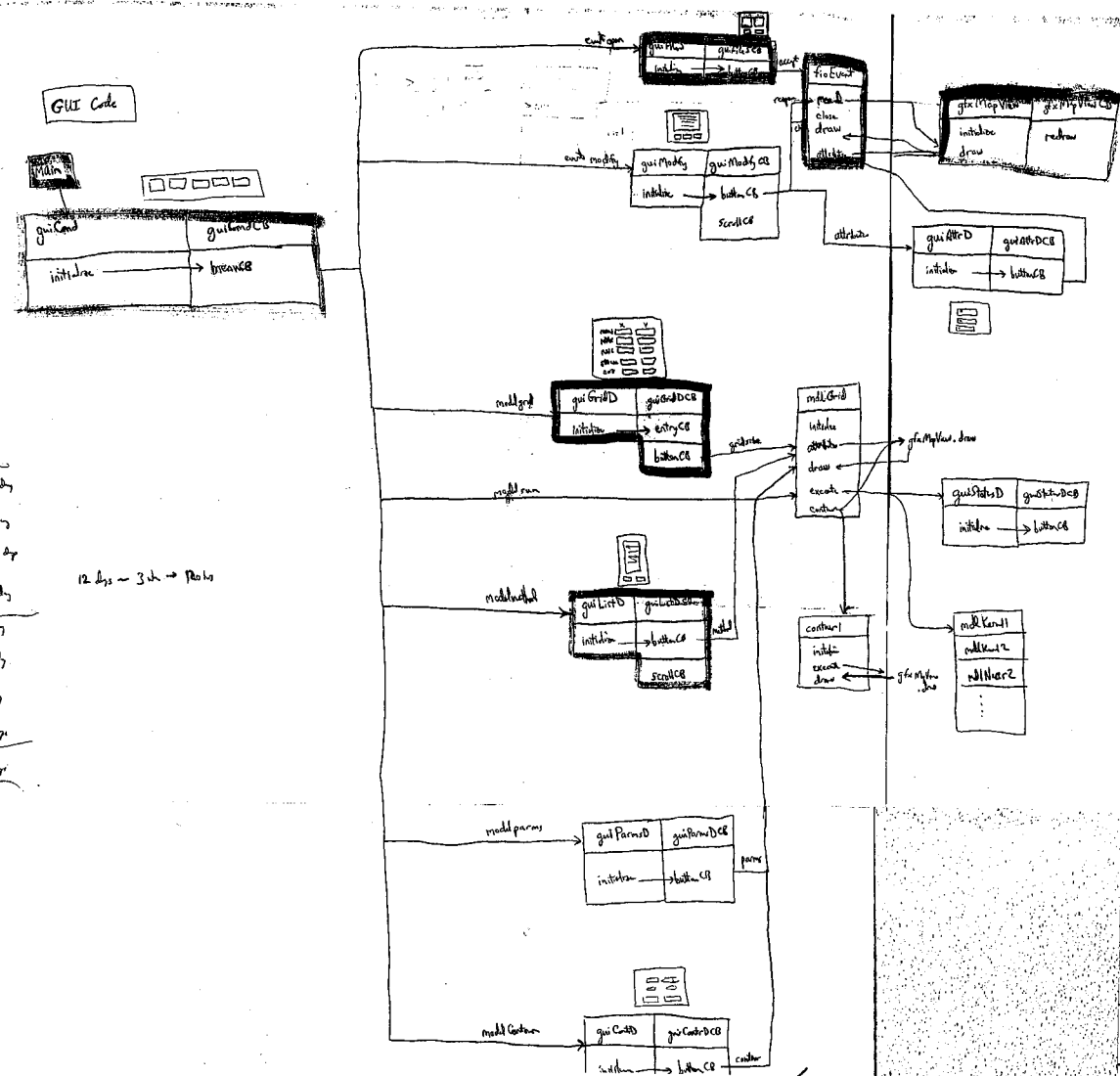
Examples:

```
% 3dstress.exp -stress 100 50 20 0 0 0 -fault 80 50
sxyz = 100 50 20 rxyz = 0 0 0 flt = 80 50
slip = 156.292 49.1828 sliptend = 0.361794

% 3dstress.exp -stress 100 50 5 0 0 0 -fault 50 80
sxyz = 100 50 5 rxyz = 0 0 0 flt = 50 80
slip = 55.3328 27.7932 sliptend = 0.74008
```

Volcanic Hazard Analysis Code

Design:



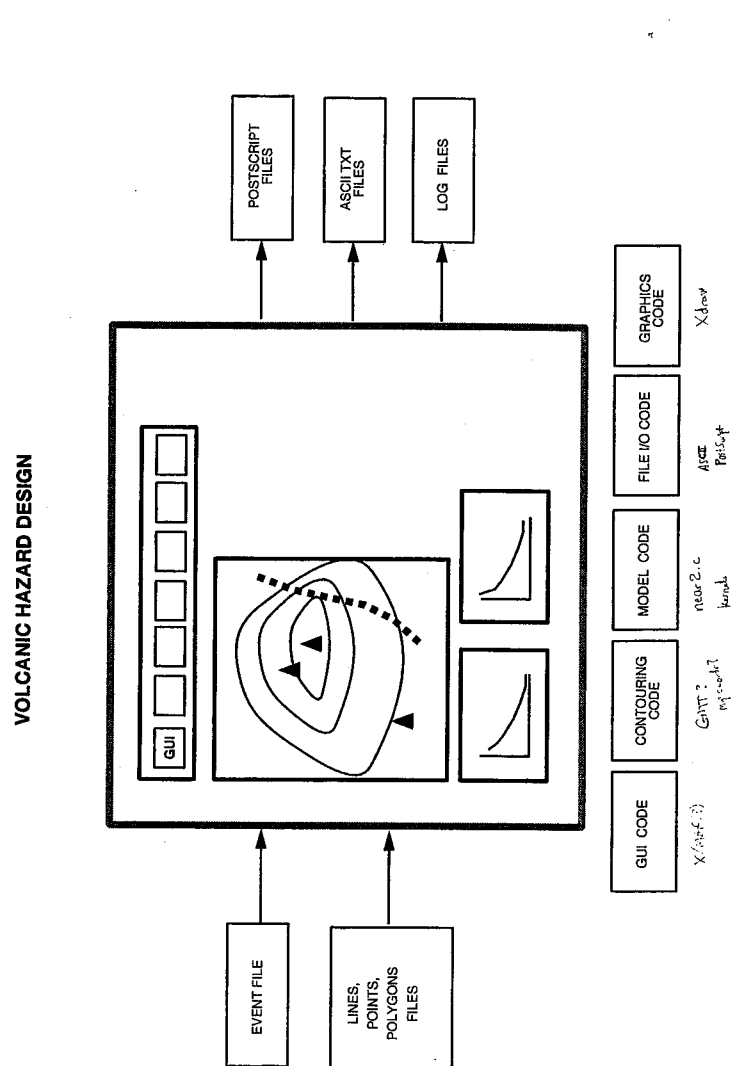
But Hendon 4-26-96

[illegible]

04/26/96
09:02:14

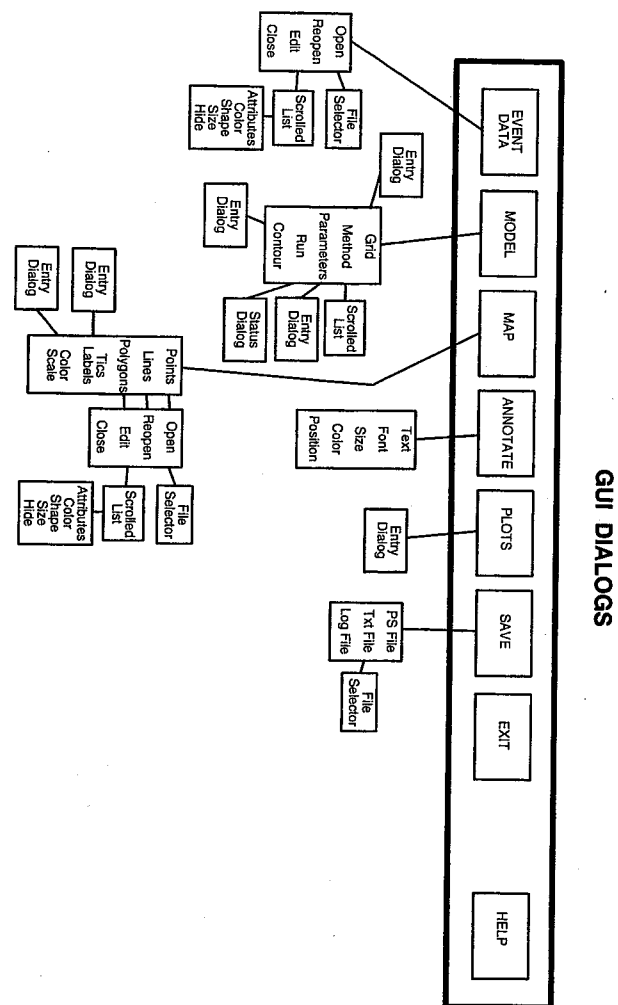
calcNear2.c++

Burt Hahn 4/26/96



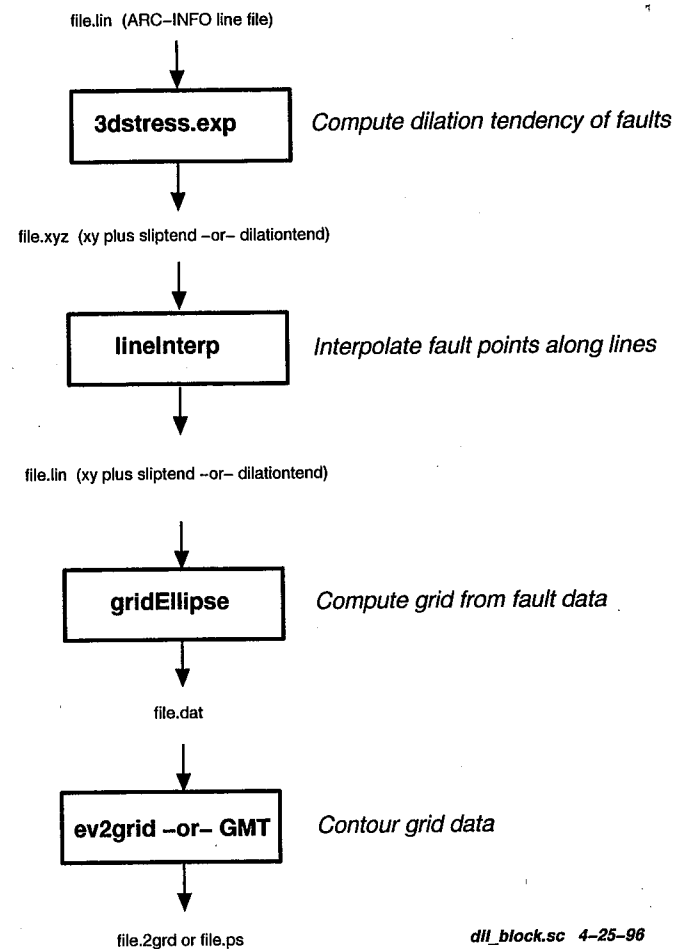
3-15-96 DBH

Rmt Henderson 4/26/96

Code Modules

3-15-96 DBH

Ravi Hunk 4/26/96

Utilities for Analyzing Dilation Tendency of Faults:

dll_block.sc 4-25-96

Ravi Hunk 4/26/96

Burt Hahn

lineInterp.c++

I

04/26/96
09:02:21

Line interpolation program

Grid based on an ellipse

[illegible][illegible]

04/26/96
09:02:29

gridellipse.c++



gridEclipse.c++

04/26/96 09:02:29

```

// Find fault pt in terms of rotated ellipse coords
float xroted = xrot*PR;
float yroted = yrot*PR;
float sum = cos(eccrad);
float sum2 = sin(eccrad);
float fix = dx * cosr - dy * sinr;
float fry = dy * cosr + dx * sinr;
if (insideEllipse == 0)
    fix = 0.0;
// Fault pt is along rotated y-axis
if (fabs(fry) <= y1) {
    insideEllipses = 1;
} else {
    // Find slope of rotated fault line
    float m1 = fry/fix;
    float m2 = -yrot/fix;
    float ey2 = eyref;
    float ex2 = exref;
    float m = m2 * ey2 * (m2 + 1.0);
    float nr = m2 * ex2 - ey2;
    // Find distance from grid pt to ellipse edge
    float dist = sqrt(nr*nr/(m2*m2+1));
    if (fabs(m1) <= dist) {
        insideEllipses = 1;
    }
}
if (insideEllipses) {
    float h = exnd;
    float h2 = 2.0*exnd*exr;
    sum = ku / (h+h2);
    // float h = exnd;
    // float h2 = exnd;
    // float ed = fdist/h;
    // float ku = 2.0*(1.0-ed)/h*PR;
    // sum += f0>2 * ku / (h+h2);
    // printf("sum = %f fdist = %f h = %f ed = %f ku = %f\n",
    //        sum, fdist, h, ed, ku);
}
if (endit >= 0.8)
    // end of fault <= end
}
// end of 1
// end of 3
// end of 5
// end of 7
// end of 9
// end of 11
// end of 13
// end of 15
// end of 17
// end of 19
// end of 21
// end of 23
// end of 25
// end of 27
// end of 29
// end of 31
// end of 33
// end of 35
// end of 37
// end of 39
// end of 41
// end of 43
// end of 45
// end of 47
// end of 49
// end of 51
// end of 53
// end of 55
// end of 57
// end of 59
// end of 61
// end of 63
// end of 65
// end of 67
// end of 69
// end of 71
// end of 73
// end of 75
// end of 77
// end of 79
// end of 81
// end of 83
// end of 85
// end of 87
// end of 89
// end of 91
// end of 93
// end of 95
// end of 97
// end of 99
// end of 101
// end of 103
// end of 105
// end of 107
// end of 109
// end of 111
// end of 113
// end of 115
// end of 117
// end of 119
// end of 121
// end of 123
// end of 125
// end of 127
// end of 129
// end of 131
// end of 133
// end of 135
// end of 137
// end of 139
// end of 141
// end of 143
// end of 145
// end of 147
// end of 149
// end of 151
// end of 153
// end of 155
// end of 157
// end of 159
// end of 161
// end of 163
// end of 165
// end of 167
// end of 169
// end of 171
// end of 173
// end of 175
// end of 177
// end of 179
// end of 181
// end of 183
// end of 185
// end of 187
// end of 189
// end of 191
// end of 193
// end of 195
// end of 197
// end of 199
// end of 201
// end of 203
// end of 205
// end of 207
// end of 209
// end of 211
// end of 213
// end of 215
// end of 217
// end of 219
// end of 221
// end of 223
// end of 225
// end of 227
// end of 229
// end of 231
// end of 233
// end of 235
// end of 237
// end of 239
// end of 241
// end of 243
// end of 245
// end of 247
// end of 249
// end of 251
// end of 253
// end of 255
// end of 257
// end of 259
// end of 261
// end of 263
// end of 265
// end of 267
// end of 269
// end of 271
// end of 273
// end of 275
// end of 277
// end of 279
// end of 281
// end of 283
// end of 285
// end of 287
// end of 289
// end of 291
// end of 293
// end of 295
// end of 297
// end of 299
// end of 301
// end of 303
// end of 305
// end of 307
// end of 309
// end of 311
// end of 313
// end of 315
// end of 317
// end of 319
// end of 321
// end of 323
// end of 325
// end of 327
// end of 329
// end of 331
// end of 333
// end of 335
// end of 337
// end of 339
// end of 341
// end of 343
// end of 345
// end of 347
// end of 349
// end of 351
// end of 353
// end of 355
// end of 357
// end of 359
// end of 361
// end of 363
// end of 365
// end of 367
// end of 369
// end of 371
// end of 373
// end of 375
// end of 377
// end of 379
// end of 381
// end of 383
// end of 385
// end of 387
// end of 389
// end of 391
// end of 393
// end of 395
// end of 397
// end of 399
// end of 401
// end of 403
// end of 405
// end of 407
// end of 409
// end of 411
// end of 413
// end of 415
// end of 417
// end of 419
// end of 421
// end of 423
// end of 425
// end of 427
// end of 429
// end of 431
// end of 433
// end of 435
// end of 437
// end of 439
// end of 441
// end of 443
// end of 445
// end of 447
// end of 449
// end of 451
// end of 453
// end of 455
// end of 457
// end of 459
// end of 461
// end of 463
// end of 465
// end of 467
// end of 469
// end of 471
// end of 473
// end of 475
// end of 477
// end of 479
// end of 481
// end of 483
// end of 485
// end of 487
// end of 489
// end of 491
// end of 493
// end of 495
// end of 497
// end of 499
// end of 501
// end of 503
// end of 505
// end of 507
// end of 509
// end of 511
// end of 513
// end of 515
// end of 517
// end of 519
// end of 521
// end of 523
// end of 525
// end of 527
// end of 529
// end of 531
// end of 533
// end of 535
// end of 537
// end of 539
// end of 541
// end of 543
// end of 545
// end of 547
// end of 549
// end of 551
// end of 553
// end of 555
// end of 557
// end of 559
// end of 561
// end of 563
// end of 565
// end of 567
// end of 569
// end of 571
// end of 573
// end of 575
// end of 577
// end of 579
// end of 581
// end of 583
// end of 585
// end of 587
// end of 589
// end of 591
// end of 593
// end of 595
// end of 597
// end of 599
// end of 601
// end of 603
// end of 605
// end of 607
// end of 609
// end of 611
// end of 613
// end of 615
// end of 617
// end of 619
// end of 621
// end of 623
// end of 625
// end of 627
// end of 629
// end of 631
// end of 633
// end of 635
// end of 637
// end of 639
// end of 641
// end of 643
// end of 645
// end of 647
// end of 649
// end of 651
// end of 653
// end of 655
// end of 657
// end of 659
// end of 661
// end of 663
// end of 665
// end of 667
// end of 669
// end of 671
// end of 673
// end of 675
// end of 677
// end of 679
// end of 681
// end of 683
// end of 685
// end of 687
// end of 689
// end of 691
// end of 693
// end of 695
// end of 697
// end of 699
// end of 701
// end of 703
// end of 705
// end of 707
// end of 709
// end of 711
// end of 713
// end of 715
// end of 717
// end of 719
// end of 721
// end of 723
// end of 725
// end of 727
// end of 729
// end of 731
// end of 733
// end of 735
// end of 737
// end of 739
// end of 741
// end of 743
// end of 745
// end of 747
// end of 749
// end of 751
// end of 753
// end of 755
// end of 757
// end of 759
// end of 761
// end of 763
// end of 765
// end of 767
// end of 769
// end of 771
// end of 773
// end of 775
// end of 777
// end of 779
// end of 781
// end of 783
// end of 785
// end of 787
// end of 789
// end of 791
// end of 793
// end of 795
// end of 797
// end of 799
// end of 801
// end of 803
// end of 805
// end of 807
// end of 809
// end of 811
// end of 813
// end of 815
// end of 817
// end of 819
// end of 821
// end of 823
// end of 825
// end of 827
// end of 829
// end of 831
// end of 833
// end of 835
// end of 837
// end of 839
// end of 841
// end of 843
// end of 845
// end of 847
// end of 849
// end of 851
// end of 853
// end of 855
// end of 857
// end of 859
// end of 861
// end of 863
// end of 865
// end of 867
// end of 869
// end of 871
// end of 873
// end of 875
// end of 877
// end of 879
// end of 881
// end of 883
// end of 885
// end of 887
// end of 889
// end of 891
// end of 893
// end of 895
// end of 897
// end of 899
// end of 901
// end of 903
// end of 905
// end of 907
// end of 909
// end of 911
// end of 913
// end of 915
// end of 917
// end of 919
// end of 921
```

[illegible]

103

Bruce Hahn 4/26/96

Bert Hahn
4/26/96

Test run 7:

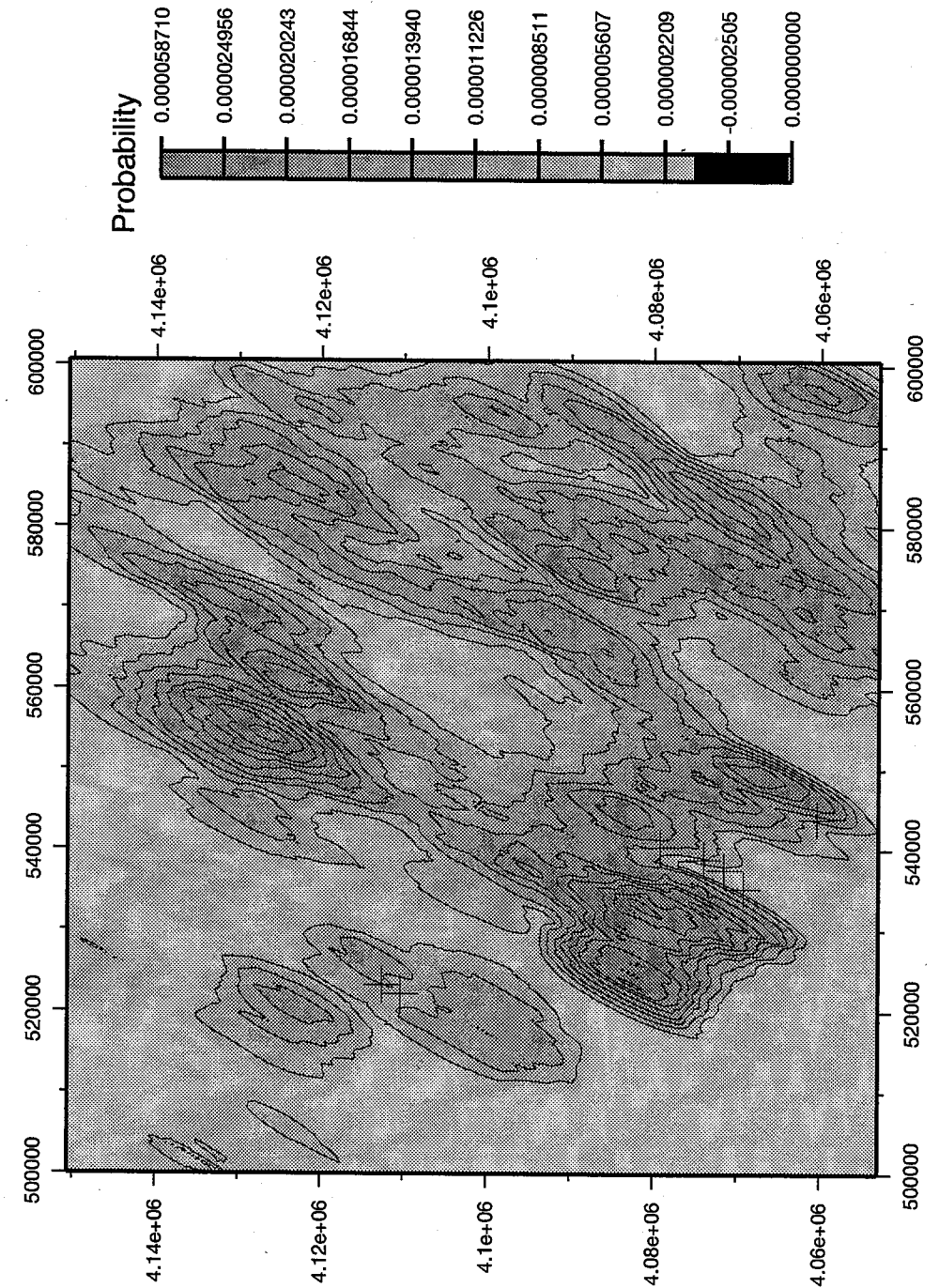
```

#!/bin/sh
date
# First use 3dstress.exp to load file.lin
# and save lines to file.xyz
echo "\nlineInterp..."
lineInterp 50 frizbmflts_dil.xyz
mv frizbmflts_dil_50.lin friz7.lin
echo "\ngridEllipse..."
# Usage:  gridEllipse
#         <e_x-axis> <e_y-axis> <e_rotation from Y-axis>
#         <xmin> <xmax> <ymin> <ymax>
#         <xinterval> <yinterval> <file.lin>
gridEllipse.sav \
  2000.0 /
  10000.0 /
  28.0 /
  500000.0 /
  600500.0 /
  4053000.0 /
  4150500.0 /
  333 /
  333 /
  friz7.lin
date
echo "Finished"

```

But John 4/29/96

Test run 7:



R. A. L. 4/29/91

Test run 8:

```

#!/bin/sh

date

# First use 3dstress.exp to load file.lin
# and save lines to file.xyz

echo "\nlineInterp..."
lineInterp 50 frizbmflts_dil.xyz

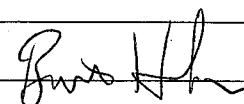
mv frizbmflts_dil_50.lin friz8.lin

echo "\ngridEllipse..."
# Usage:  gridEllipse
#         <e_x-axis> <e_y-axis> <e_rotation from Y-axis>
#         <xmin> <xmax> <ymin> <ymax>
#         <xinterval> <yinterval> <file.lin>
gridEllipse.sav \
  4460.8 /
  4460.8 /
  0.0 /
  500000.0 /
  600500.0 /
  4053000.0 /
  4150500.0 /
  333 /
  333 /
  friz8.lin

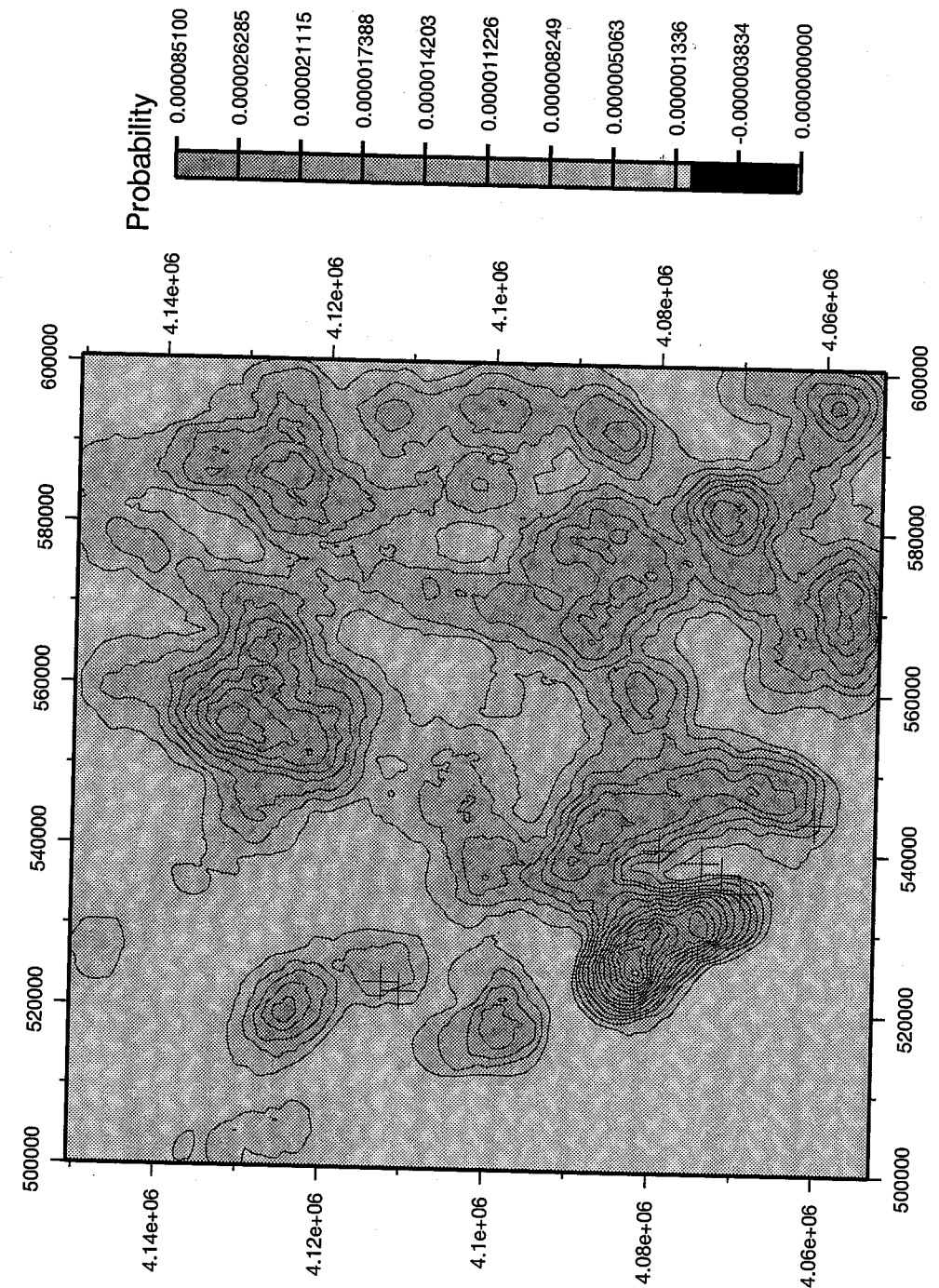
date

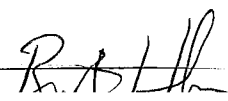
echo "Finished"

```

 4/29/96

Test run 8:



 4/29/96

Use GMT to make contour map:

```
rm frizlout.ps

gmtset ANOT_FONT_SIZE 8
gmtset LABEL_FONT_SIZE 12

# Create binary grid file...
xyz2grd /usr3/brent/462igneous/volHazCode/bin/frizbmflts_dil_100_grd.dat \
  -Cpvtmp.grd \
  -I333.00000/333.00000 \
  -R520000.00000/570283.00000/4055000.00000/4095293.00000

# Create color palette file...
grd2cpt pvtmp.grd \
  -M0.95/0.65 \
  > pvtmp.cpt

# Create color filled contour image PS...
grdimage pvtmp.grd \
  -JX4.500000 \
  -P \
  -Cpvtmp.cpt \
  -K \
  > frizlout.ps

# Create contour lines PS...
grdcontour pvtmp.grd \
  -JX4.500000 \
  -C0.00100 \
  -T \
  -P \
  -Bf2000.000000a10000.000000 \
  -O \
  -K \
  >> frizlout.ps

# Create points in PS...
psxy \
  cones.xy \
  -JX4.500000 \
  -O \
  -R520000.00000/570283.00000/4055000.00000/4095293.00000 \
  -Sx0.200000 \
  -P \
  -K \
  >> frizlout.ps

# Create lines in PS...
psxy \
  frizbmflts_noid.lin \
  -JX4.500000 \
  -O \
  -R520000.00000/570283.00000/4055000.00000/4095293.00000 \
  -MEND \
  -P \
  -K \
  >> frizlout.ps

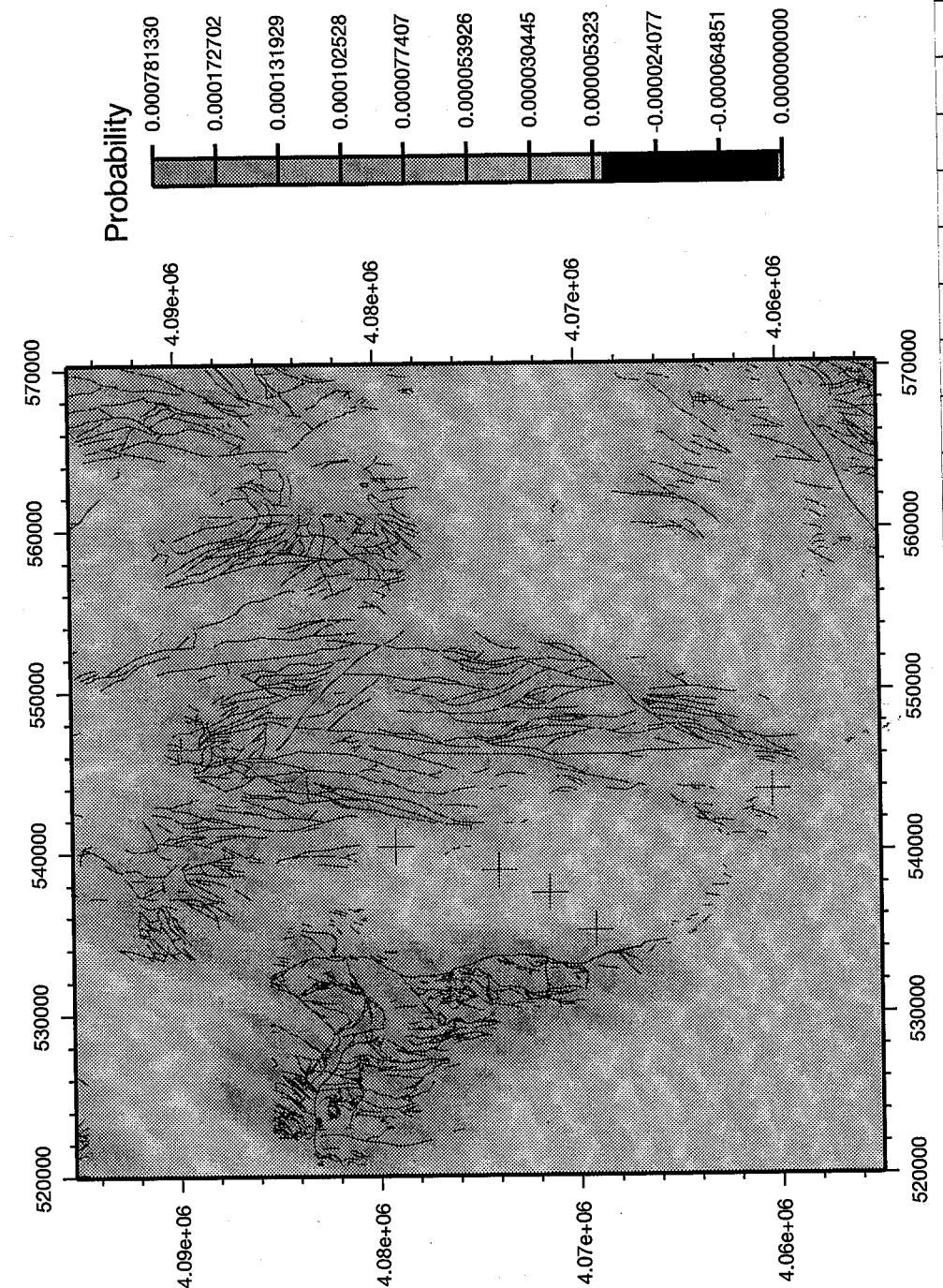
# Add scale to PS...

# Add scale to PS...
psscale \
  -D5.500000/2.250000/3.500000/0.150000 \
  -Cpvtmp.cpt \
  -L -B:"Probability": \
  -P \
  -O >> frizlout.ps

rm pvtmp.cpt pvtmp.grd
echo Created frizlout.ps
```

Brent H 5/2/96

Contour map:



Brent H 5/2/96

Process data from 3D Stress:

```
#!/bin/sh

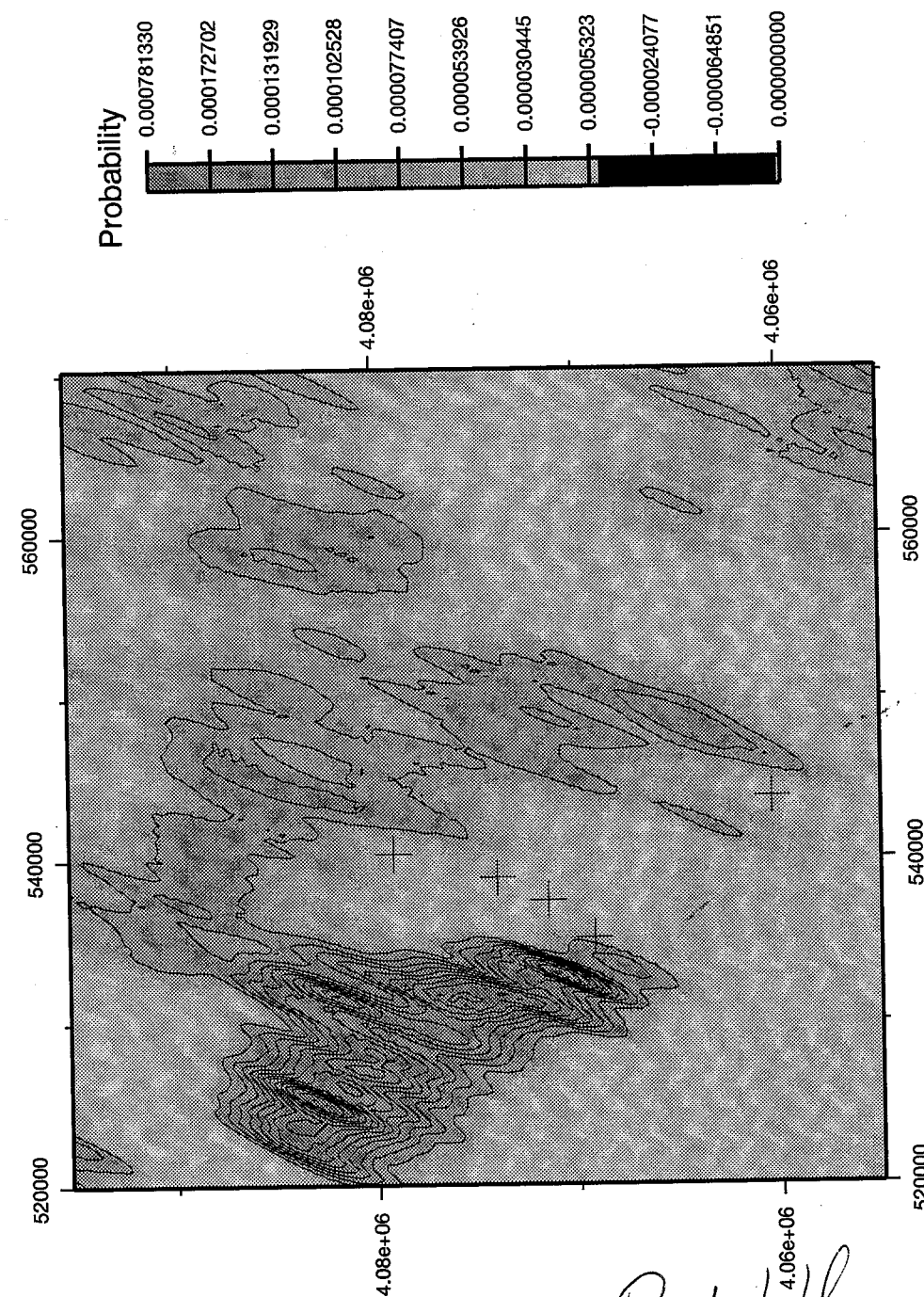
# First use 3dstress.exp to load file.lin
# and save lines to file.xyz

echo "\nlineInterp..."
lineInterp 100 frizbmflts_dil.xyz

echo "\ngridEllipse..."
# Usage:  gridEllipse
#          <e_x-axis> <e_y-axis> <e_rotation from Y-axis>
#          <xmin> <xmax> <ymin> <ymax>
#          <xinterval> <yinterval> <file.lin>
gridEllipse \
  500.0 \
  1000.0 \
  28.0 \
  545010.0 \
  550980.0 \
  4074000.0 \
  4083000.0 \
  200 \
  200 \
  frizbmflts_dil_100.lin
echo "Finished"
```

Boat / Hm 5/2/96

Contour map of elliptic kernel output:



Boat / Hm 5/2/96

16/2/99 *[Signature]*

FRIZZELL BM FLTS DILATION MINUS FAULT DENSITY 4-30-96

rwfriz8

#/bin/sh

date

First use 3dstress.exp to load file.lin
and save lines to file.xyz

echo "unlineinterp..."
lineinterp 50 frizbmfts_dil.xyz

mv frizbmfts_dil_50.lin friz8.lin

echo "ngridellipse..."

Usage: gridellipse
<e_x-axis> <e_y-axis> <e_rotation from Y-axis>
<xmin> <xmax> <ymin> <ymax>
<xinterval> <yinterval> <file.lin>
gridellipse.sav \

4460.8 \
4460.8 \
0.0 \
500000.0 \
600500.0 \
4053000.0 \
4150500.0 \
333 \
333 \
friz8.lin

date

echo "Finished"

rwfriz10

#/bin/sh

date

First use 3dstress.exp to load file.lin
and save lines to file.xyz

echo "unlineinterp..."
lineinterp 50 frizbmfts.xyz

mv frizbmfts_50.lin friz10.lin

echo "ngridellipse..."

Usage: gridellipse
<e_x-axis> <e_y-axis> <e_rotation from Y-axis>
<xmin> <xmax> <ymin> <ymax>
<xinterval> <yinterval> <file.lin>
gridellipse.sav \

4460.8 \
4460.8 \
0.0 \
500000.0 \
600500.0 \
4053000.0 \
4150500.0 \
333 \
333 \
friz10.lin

date

echo "Finished"

rwfriz10.dat

runNormalize friz8_grd.dat > friz8_norm.dat
runNormalize friz10_grd.dat > friz10_norm.dat

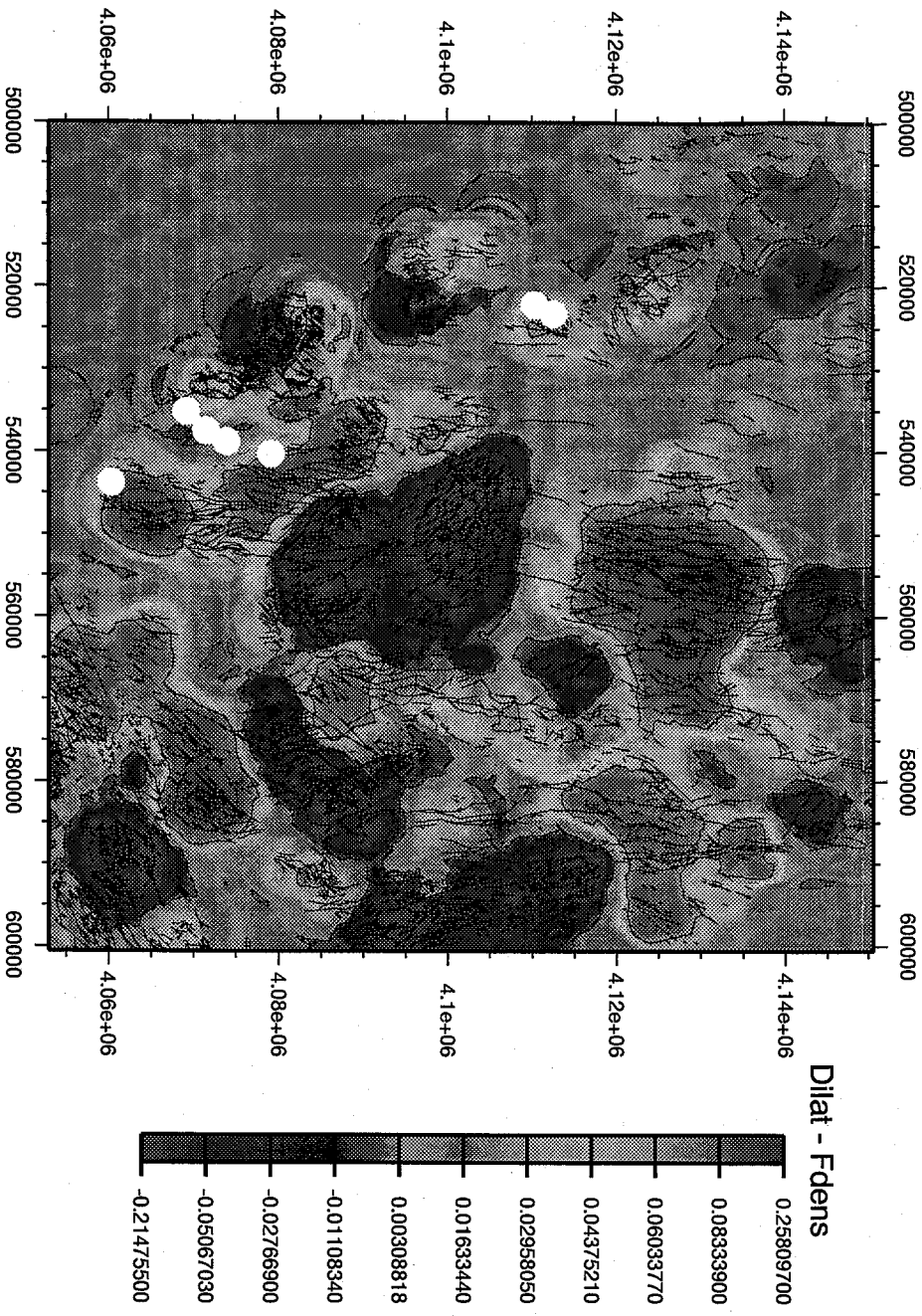
runStats friz8_norm.dat
runStats friz10_norm.dat

runGridDiff friz8_norm.dat friz10_norm.dat >
friz8minus10.dat

runStats friz8minus10.dat

friz8minus10run.sh

echo "unFinished"



FRIZZELL BM FLTS DILATION MINUS FAULT DENSITY 4-30-96

rm friz8minus10out.ps

gmtset ANOT_FONT_SIZE 8
gmtset LABEL_FONT_SIZE 12

Create binary grid file...
xyz2grd /usr3/brent/462igneous/volHazCode/bin/friz8minus10.dat \
-Gpvtmp.grd \
-I333.00000/333.00000 \
-R500000.00000/600566.00000/4053000.00000/4150569.00000

Create color palette file...
grd2cpt pvtmp.grd \
-C300/0 \
> pvtmp.cpt

Create color filled contour image PS...
grdimage pvtmp.grd \
-JX4.500000 \
-P \
-Cpvtmp.cpt \
-K \
> friz8minus10out.ps

Create contour lines PS...
grdcontour pvtmp.grd \
-JX4.500000 \
-C0.064360 \
-T \
-P \
-Bf5000.000000a20000.000000 \
-O \
-K \
>> friz8minus10out.ps

Create points in PS...
psxy \
-MEND \
/usr3/brent/462igneous/volHazCode/bin/frizbmflts_noid.lin \
-JX4.500000 \
-O \
-R500000.00000/600566.00000/4053000.00000/4150569.00000 \
-P \
-K \
>> friz8minus10out.ps

psxy \
/usr3/brent/462igneous/volHazCode/bin/cones.xy \
-JX4.500000 \
-O \
-G255 \
-R500000.00000/600566.00000/4053000.00000/4150569.00000 \
-Sc0.15 \
-P \
-K \
>> friz8minus10out.ps

Add scale to PS...
psscale \
-D5.500000/2.250000/3.500000/0.150000 \
-Cpvtmp.cpt \
-L -B:"Dilat - Fdens": \
-P \
-O >> friz8minus10out.ps

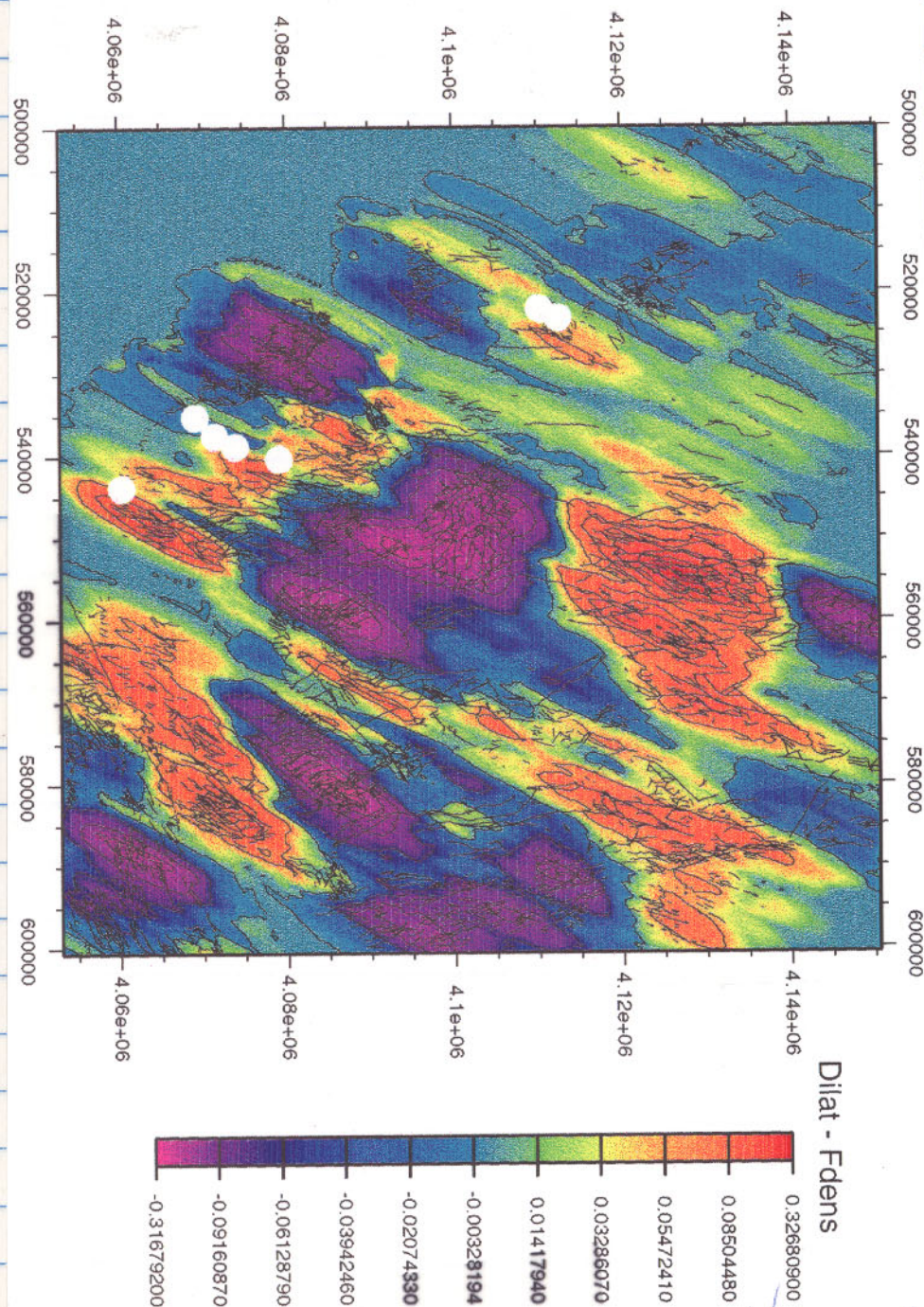
rm pvtmp.cpt pvtmp.grd

echo Created friz8minus10out.ps

PROCESSING 34 DATA

115

DILATION TENDENCY PLOT:



Brent Hiller 5/2/96

Graphic User Interface (GUI) Builders

Jun 24 1996 16:51

gui_builder.eval

Page 1

GRAPHIC USER INTERFACE BUILDERS TOOLKIT EVALUATIONS

1. This paper compares basic information for two market available software packages used to build graphic user interfaces (GUI). Comparisons are based on price, and operating system (OS) platform availability.

a. Intergrated Computer Solutions Inc. OSF/Motif:

- (1) Price: \$350.00 plus \$99.00 per year for support and upgrades.
- (2) OS Platforms Available: Solaris SPARC, Sun Operating System. Silicon Graphics (SGI) not available.
- (3) OSF/Motif Toolkit, User Interface Language (UIL), Widget Meta Language (WML), and Motif Window Manager. Develops Motif applications. Lists only C and C++ as source languages.

b. Qualix Co. SCO Premier Motif:

- (1) Price: \$695.00; one year support and upgrades included.
- (2) OS Platforms Available: SunOS, Solaris, SGI, MS Windows.
- (3) Platforms can be switched at any time with no additional cost. Languages include UIL, C, C++.

2. The above information was submitted by vendors via fax or telephone conversation.

Danny Skelton
Student Analyst

6/24/96

Danny Skelton

User Interface Management Tools (UIM/X) and Graphic User Interface Builders

Jun 24 1996 16:41

develop.eval

Page 1

GRAPHIC USER INTERFACE BUILDERS EVALUATION DEVELOPER SOFTWARE AND USER INTERFACE MANAGEMENT TOOLS

1. This paper compares basic information for several market available software packages used to build graphic user interfaces (GUI). Comparisons are based on price, and operating system (OS) platform availability.

- a. Intergrated Computer Solutions Inc. Builder Xcessory:
 - (1) Price: \$3,200.00 (no support and upgrade price given).
 - (2) OS Platforms Available: SunOS SPARCworks, SGI.
 - (3) GUI developer. Supports C, C++, UIL, or ADA.
- b. Thomson Software Products, TeleUSE;
 - (1) Price: \$7,500.00 plus 20% yearly for support and upgrades.
 - (2) OS Platforms Available: SunOS, SGI.
 - (3) User interface management system. Supports UIL, C, C++, and ADA source languages.
- c. Imperial Software Technology Ltd. X-Designer:
 - (1) Price: 3,500.00; one year support and upgrades included. Support is 15% of purchase price for subsequent years.
 - (2) OS Platforms Available: All platforms available, to include Apple Macintosh.
 - (3) GUI builder. Supports UIL, C, C++, and Java.
- d. Black and White Software. UIM/X 2.9:
 - (1) Price: \$5,000.00 no support quoted.
\$7,700.00 with XMove software
\$2,500.00 for cross platfor toolset.
 - (2) OS Platforms Available: Sun, Solaris, SGI, MS Windows.
 - (3) User Interface Management system. Supports C, C++ and UIL languages. Includes Ux Library, which can convert to UIL code.
 - (4) Evaluation Package: All graphics are maintained on a palette, novice or expert. Windows and menus are easy to build. System is user friendly, and applications are simple to modify once UIL is learned.
- e. Bluestone Inc. UIM/X:
 - (1) Price: \$5,000.00.
\$1,500.00 annual maintenance contract.
\$ 500.00 for Deluxe Package
 - (2) OS Platforms Available: SunOS, HP, IBM, SGI.
 - (3) User Interface Management system. UIL, C and C++ languages. Deluxe Package includes extended text editor integration, Online help, and full documentation, XRT integration. XRT is the set of widgets for displaying tables, charts, and three dimensional images.

2. The above information was submitted by vendors via fax or telephone conversation. Only the UIM/X package from Black and White Software was submitted for hands-on evaluation.

Visual ToolKit (vtk) Evaluation pg 1 of 3

Jun 24 1996 12:24

vtk.eval

Page 1

From: Danny Skelton

Date: 6/24/96

Subj: Evaluation of Visual ToolKit (vtk)

To: Brent Henderson

1. Visual ToolKit was obtained from the Internet at internet address www.cs.rpi.edu/~martink/ (this was only the home page for the program). Vtk was downloaded with several support programs as listed below. Preparations for installation are also listed.

- a. The following programs were downloaded as support programs for vtk:
 - (1) Mesa-1.2.7 (Mesa-1.2.7.tar.gz) was downloaded as a Graphis Library (GL) program.
 - (2) Tcl7.5 (tcl7.5.tar.gz) was downloaded as an interpretive language, to be used as an alternative to C++.
 - (3) Tk4.1 (tk4.1.tar.gz) was downloaded as a graphical user interface toolkit based on Tcl.
 - (4) VTK11 (VTK11.ZIP) was downloaded as the Windows executable for vtk.
 - (5) vtk1.0 data files to provide the data files for the program.
 - (a) vtk1.0.data.part1.tar.gz
 - (b) vtk1.0.data.part2.tar.gz
 - (c) vtk1.0.data.part3.tar.gz
 - (d) vtk1.0.data.part4.tar.gz
 - (6) The vtk source file vtk1.1.src.tar.gz.
- b. Files were unzipped and expanded using following commands:
 - (1) Unzipped;
 - (a) .gz files -- gunzip <filename>
 - (b) .ZIP file -- unzip <filename>
 - (2) Untar -- tar xvfo <filename>

2. Configuration and Compilation: Files were configured and compiled as follows:

- a. Mesa-1.2.7 -- used the README file, Unix instructions.
 - (1) typed "make irix5" to configure program for bemore and compile.
- b. Tcl7.5 -- used the README file in the unix directory for instructions.
 - (1) Typed "./configure" to run configuration script, configuring program to SGI system and bemore.
 - (2) Typed "make" to create library archives libtcl.a and libtcl.so.
 - (3) Typed "make install" to install Tcl binaries and script files in the standard places.
- c. Tk4.1 -- used the README file in the unix directory for instructions.
 - (1) Typed "./configure" to run configuration script, configuring program to SGI system and bemore.
 - (2) Typed "make" to create library archives libtk.a and libtk.so.

6/24/96

Visual Tool/Kit (VTK) Evaluation

Pg 2 of 3

Jun 24 1996 12:24

vtk.eval

Page 2

- (3) Typed "make install" to install Tcl binaries and script files in the standard places.
- d. vtk -- used the README file in the main vtk1.1 directory for instructions.
- (1) Configured program by typing "./configure --with-tcl --with-mesa". This command builds the vtk library, and binds vtk with tcl and mesa.
 - (2) Modify Config.mk file by entering the following:
 - (a) CC=CC sets compiler line to a CC prefix.
 - (b) CXX=CC
 - (c) TCL_INCLUDE=/userdata/vtk/include/tcl7.5a2/include this was the location of the tcl7.5a2 include file
 - (d) TCL_LIB=/userdata/vtk/lib/tcl7.5a this was the location of the tcl7.5a2 library file
 - (e) TK_INCLUDE=/userdata/vtk/include/tk4.1a/include this was the location of the tk4.1a include file
 - (f) TK_LIB=/userdata/vtk/lib/tk4.1a this was the location of the tk4.1a library file
 - (3) Type "make" to compile vtk1.1 and create the vtk executable file.

3. Problems: All programs with the exception of vtk1.1 compiled. After several attempts to compile vtk1.1, it was determined that it would not compile on the SGI machines.

- a. Bemoire did not have a c++ compiler, so compilation had to be accomplished on performer system.
- b. vtk1.1 contained several virtual functions, which caused problems in the compiler for performer, as documented in the release notes:

- (1) Error code received was:

03): "/pscr4/danny/vtk1.1/include/vtkStructuredGrid.hh", line 67: error(34

object of abstract class type is not allowed
 vtkDataSet *MakeObject() {return new vtkStructuredGrid(*this);};

1 error detected in the compilation of "vtkStructuredGrid.cc".
 *** Error code 1 (bu21)
 *** Error code 1 (bu21) "

- (2) Reason for error code was that the class type "vtkDataSet *MakeObject()" is a pure virtual function in the file "vtkDataSet.hh".

- (3) Release notes for the compiler state:

"5.2 Runtime_Linker_(rld(1))

- o It is an error for a program to declare an external symbol (only possible in different *.o's) both as a function (TEXT) and as data (DATA). A common cause of this is to take the address of a function but fail to write the function prototype defaulting the symbol to be data. Normally, the linker would catch such an error when a program is built. However, it is possible to link a program with one set of *.so's and then run it with another set creating the conflict at execution

VTK Evaluation

Pg 3 of 3

Jun 24 1996 12:24

vtk.eval

Page 3

time. Under Irix 5.2, rld wouldn't always catch the error and if the program did not happen to execute the code path which referenced the symbol, then the program would run "correctly". Under Irix 5.3, attempting to execute the erroneous program produces the following error:

```
rld: Fatal Error: search_for_undefineds: symbol
<name> should not have any relocation entry
```

The appropriate action is to correct the error in the source code and rebuild it."

" 6. Known_Problems_and_Workarounds

This section deals with known problems with C++ Version 4.0 that have not yet been fixed.

- o Incorrect layout of object or compiler abort with certain virtual base classes

CC sometimes lays out an object with virtual base classes incorrectly, or crashes when processing the class."

4. Evaluation: Incomplete due to non-compilation of vtk1.1. This program has potential, however bugs in our compilers will have to be worked out before we can even try it. I will maintain a compressed copy of all programs for awhile until I am positive that we cannot find a work around for the compiler problem.

Danny Skelton
 Student Analyst

[Signature]
 6/24/96

Calculate anomaly pg. 1 of 3

MATTIN

anomaly.c++	
anomaly.Globals.h	

CALCULATIONS	Calc Anomaly - vtt
--------------	--------------------

to PolygonObj	write penUp draw goTo(x,y) get set setValues addPoly select moveVert sizeObj deleteVert resetVer
to PolygonProc	penUp selectObj sizeObj addPoly intensity topLevel bottomLevel xLevel xLevelObj setValues setVer

CALCULATIONS

Calc Anomaly. vtt

```

JUN 26 1996 09:11      calcAnomaly.C++      Page 3
*/
// rise: inclination of remnant magnetization in degree, down is positive
// rise: inclination of remnant magnetization in ena/cc
// top: elevation of the top of the polygon in meters
// bottom: elevation of the bottom of the polygon in meters (positive upward)
// this is followed by an enumeration of the vertices of the polygon
// one coordinate per line
// entered as Northing, Easting
// these are entered in clockwise order
// if coordinates are not entered in clockwise order, the
// line integral will be done incorrectly
// the information for the next polygon follows the last coordinate
// the coordinates for the next polygon are entered

floatPolygonType *inPolyData;
int
inPolyObj::get(inPolyData, inPolyCnt);

int vcnt=0;
for (int poly=0; poly<inPolyCnt; poly++) {
    // printf("Polygon %d\n", poly);
    // printf("  numSides      %d\n", inPolyData[poly].numSides);
    // printf("  susceptibility %d\n", inPolyData[poly].susceptibility);
    // printf("  inclination      %d\n", inPolyData[poly].inclination);
    // printf("  declination      %d\n", inPolyData[poly].declination);
    // printf("  intensity         %d\n", inPolyData[poly].intensity);
    // printf("  topElev          %d\n", inPolyData[poly].topElev);
    // printf("  bottomElev       %d\n", inPolyData[poly].bottomElev);
    inPoly = inPolyData[poly].numSides;
    x[poly] = inPolyData[poly].topElev + inFact;
    z[poly] = inPolyData[poly].bottomElev + inFact;
    for (int w=0; w<inPolyData[poly].numSides; w++) {
        // printf("  x = %f\n", x[poly]);
        x[wvcnt] = inPolyData[poly].x[w];
        y[wvcnt] = inPolyData[poly].y[w];
        // find the lengths of the sides of the polygons and store
        // in the matrix detvec[]; This is used in the calculation
        // of <math>\tau</math> in eq. 1
        detvec[vcnt] = sqrt ((inPolyData[poly].x[w+1]-inPolyData[poly].x[w]) *
                             (inPolyData[poly].x[w+1]-inPolyData[poly].x[w]) +
                             (inPolyData[poly].y[w+1]-inPolyData[poly].y[w]) *
                             (inPolyData[poly].y[w+1]-inPolyData[poly].y[w]));
        // else
        detvec[vcnt] = sqrt ((inPolyData[poly].x[0]-inPolyData[poly].x[w]) *
                             (inPolyData[poly].x[0]-inPolyData[poly].x[w]) +
                             (inPolyData[poly].y[0]-inPolyData[poly].y[w]) *
                             (inPolyData[poly].y[0]-inPolyData[poly].y[w]));
    }
    // find the shortest polygon side
    if (detvec[vcnt] < shortsides) shortsides = detvec[vcnt];
    vcnt++;
} // endfor

// Pass the information to dir_cos to calculate
// dir_dir, dir_r: direction cosine for the polygon, dir_induced field
// and [x1mm], [y1mm], [z1mm]: the vector of magnetization for the
// remnant magnetization
// dir_cos(xdc, xsc, sus, s, rdec, rsc, ssc, dir_r, dir_rmm, dir_rmm);
// dir_dir = dir_dir;
// inPolyData[poly].susceptibility,
// inPolyData[poly].inclination,
// inPolyData[poly].declination,
// inPolyData[poly].intensity,
// dir_r, dir_rmm, dir_rmm, [x1mm],
// dir_r, dir_rmm, dir_rmm, [x1mm], [y1mm], [z1mm]);

// find the thinnest polygon

```

```

Jun 28 1986 09:11 calcAnomaly.C++ Page 2
void dir_osc(float xdc, float xinc, float sus, float b, float rdc,
            float rinc, float nrm, float *dir1, float *dir2, float *dirn, int m)
{
    void calc1(int num_pts, int mm, float shortSide, float rdx1,
              double *yp, float *yp,
              float *xb, float *yb, int *ll, float *dstnc, float *zom, float *zm);
    void target(float p, float xdl, float r1, float xdr,
              float r2, float xdl, float r3, float xdr, float r4, float *t12);
}

////////////////////////////////////
// calcAnomaly
////////////////////////////////////
void calcAnomaly(AnomalyType *in) {
    // cout << "calcAnomaly begin... \n" << flush;

    float shortSide = 1e20;
    float rdx1 = 1e20;
    float dir1, dirn, dir2;
    double xp[1000], yp[1000];
    float zp[1000];
    float xb[1000], yb[1000];
    int ll[1000];
    float xzl[200];
    float xzm[200];
    float dstnc[200];

    // double t_field[1000];
    // double dirnc[200], xzm[200], zsm[200], xdst1[1000];
    // double xb[200], yb[200], zp[1000];
    // xp[1000], yp[1000], obs_mag[1000];

    // int nand = 0;
    // int nm = 0;

    // float sus,nrm;
    // num,rdc,rinc,top,bottom;

    //-----
    // Get input anomaly data
    //-----
    // this file contains the list of field point locations
    // and the form northings, elevation (increases upward)
    // all have the same units
    AnomalyType *inMag;
    double *inDist;
    int inMagCnt;

    floatAnomalyObj.in.get(&inMag, &inDist, &inMagCnt);

    for (int i=0; i<inMagCnt; i++) {
        xp[i] = (inMag->).x;
        yp[i] = (inMag->).y;
        zp[i] = 1005.0;
    }

    //-----
    // Get input polygon data
    //-----
    // this file contains the list of polygons
    // for each polygon in the file, first there is a
    // line with general information about the polygon
    // this information line follows the format:
    // num: number of sides of the polygon
    // sus: susceptibility of the polygon in emu
    // rdc: declination of remanent magnetization, in degrees,

```

```

//      east of North is positive
Jun 26 1996 09:11      calcAnomaly.c++      Page 4

float zd = inPolyData[ply].topElev-inPolyData[ply].bottomElev;
if (isd < zdel) zdel = zd;

} //endiforloop

//-----
//      Write output file
//-----
//      calculate the field anomaly
//      this subroutine returns the horizontal (delta_x,delta_y,
//      and vertical (delta_z) components of the magnetic anomaly for
//      each field point, and the volume
//      integrals for each field point
//-----
//      calcNum_pts, res, shortSide, xdel,yp,yp,xby,y1,dstnc,see,zm;
calc(inMagCnt, inPolyCnt, shortSide, zdel,
    yp, yp, xp, x0, y0, y1, distnc, see, zm);

//-----
//      the total field anomaly is calculated at each field point
//      from the components of
//      the anomaly and the direction cosine data
//      this is multiplied by 100000 to convert to nT
floatAnomalyType *outData = (floatAnomalyType *)2*Malloc(
    inMagCnt * inPolyCnt * distnc * distnc/(floatAnomalyType));
if (outData == NULL) {
    fprintf(stderr,"Error, could not malloc outData %d\n", inMagCnt);
    return;
}

for (int s = 0; s < inMagCnt; s++) {
    (outData[s])>xc = (inMagCnt)>xc;
    (outData[s])>y = (inMagCnt)>y;
    (outData[s])>z = (inMagCnt)>z;
    (outData[s])>dir = (dir*(delta_x[s] +
        dir*(delta_y[s] +
        dir*(delta_z[s]))*100000.0;

} // endifor

floatAnomalyObjObj.set(outData, inMagCnt);
floatAnomalyObjObj.write(in->outFile);

} // end calcAnomaly

//-----
//      Calculate the Direction Cosines
//-----
//-----
void dir_cos(float xdel, float xinc, float xsm, float N,
    float xdim, float xinc, float xrm, float *dir1,
    int nm) {
//      calculates the direction cosines for the magnetic field for
//      a given polygon
//-----
//      by C. Connor, 10/95
//-----
//      Input:
//      xdel - declination of the regional field in degrees (if N is positive)
//      xinc - inclination of the regional field in degrees (down is positive)
//      xsm - susceptibility of the polygon (uniform in the
//      polygon, emu (e.g., 0.005)
//      N - north's magnetic field strength in Gs (e.g., 0.5)
//      rdel - declination of the remanent vector of magnetization
//      in degrees (if 0 or 0 is positive)
//      rinc - inclination of the remanent vector of magnetization
//      in degrees (down is positive)
//      xrm - intensity of remanent magnetization in emu/cc
//-----
//      Output:
//      returns the direction cosines for the regional field
//      dir1, dir2, dir3
//-----
//      returns the components of the vector of magnetization for the remanent

```

calcAnomaly.c++

Let W 6128/96

6/28/96

Program to rotate grids =

Aug 26 1996 09:23

main.programs

```

:
:tfmgrid.usage
:
:Error argc=1, needs to be 7 or 8.
Usage: tfmgrid Infile Outfile -r x y z [ROIfile]
-or- tfmgrid Infile Outfile -t x y z [ROIfile]
-r = rotation; x y z = rotation angles in degrees
-t = translation; x y z = translation coordinates
ROIfile = file with Region of Interest coordinates
ROIfile: 6 faces consisting of 4 coords each,
ROIfile: 24 lines of x y z data

:
:fltgrid.usage
:
:Usage: fltgrid ingrid.dat outgrid.dat fault.dat property distance
:Where ingrid.dat is xyzp file
:outgrid.dat is xyzp file
:fault.dat is file with 4 xyz verts
:property is value of property near fault
:distance determines which nodes are changed

```

Aug 26 1996 09:17

functions.list

```

=====
void TranslateGrid(double, double, double, int);
=====
quad.h
-Determine if a point in the plane of a quad is inside or outside
the quad
=====
int QuadContainsPoint(Point3Df *, Point3Df[4]);
=====
surface.h
-Determine distance of a point to a surface, if point is
within threshold distance to surface then modify point's
property value
=====
void surface(int, double [4][3], double, double);
=====
plane.h
-Plane creation and distance functions, also determine if a line
and plane intersect
=====
void PlaneCreate(Point3Df *, Point3Df *, Point3Df *, Plane3Df *);
int PlaneIntersect(Point3Df *, Point3Df *, Point3Df *, Plane3Df *);
double PlanePointDist(Point3Df *, Plane3Df *);
=====
vector.h
-Vector and point functions
=====
void PointCreate(double, double, double, Point3Df *);
void PointRotate(Point3Df *, double [3][3]);
void PointTranslate(Point3Df *, Vector3Df *);
void VectorCreate(double, double, double, Vector3Df *);
void VectorNormalize(Vector3Df *);
void VectorCrossProduct(Vector3Df *, Vector3Df *, Vector3Df *);
double VectorLength(Vector3Df *);
double VectorDotProduct(Vector3Df *, Vector3Df *);
double VectorDistance(Vector3Df *, Vector3Df *);

```

Burt Hahn 9/16/96

Aug 26 1996 09:17

functions.list

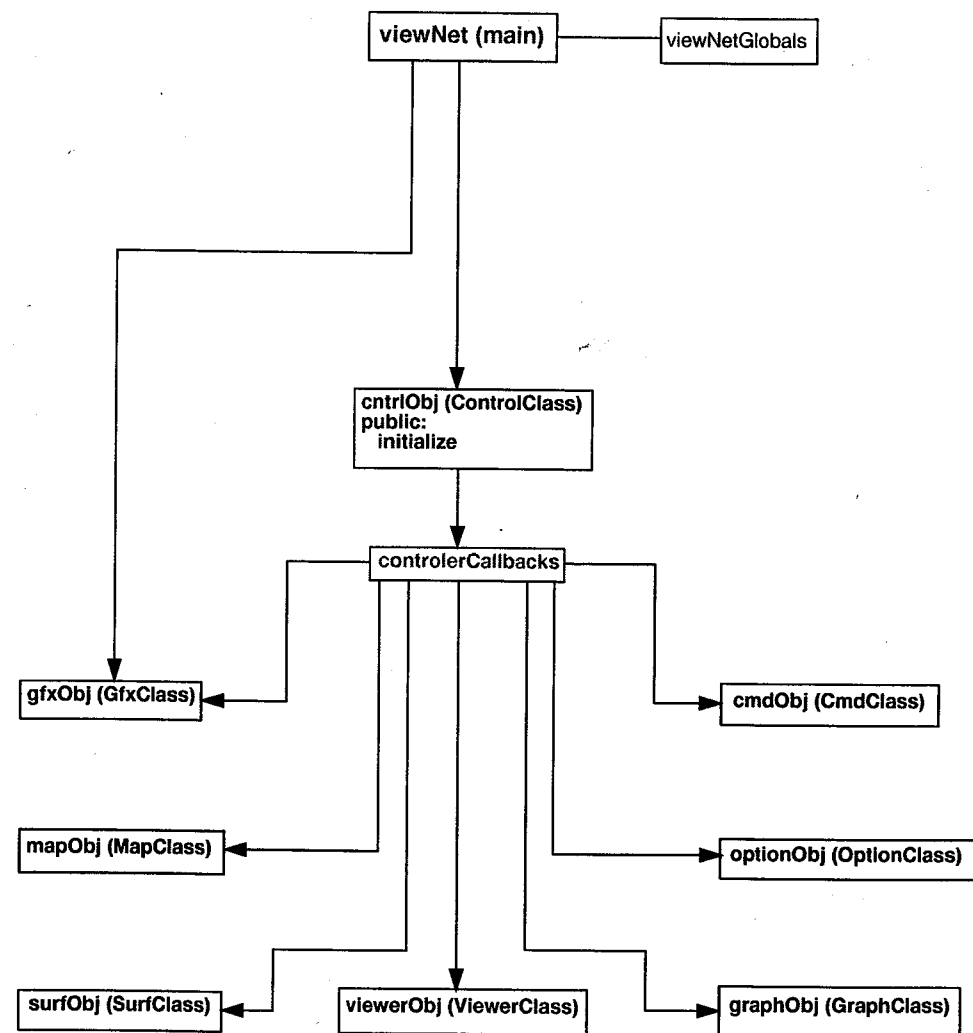
```

=====
Main programs: tfmgrid and fltgrid
=====
Data Structures
=====
typedef struct{ /* output array structure */
int x; /* x coordinate */
int y; /* y coordinate */
int z; /* z coordinate */
double p; /* property of coordinate */
} Gridpoint;
=====
Global Variables
=====
Gridpoint outgrid[4000000];
double inProp[200][200][10];
=====
cmdInput.h
-Read command line inputs
=====
int cmdCountArgs(char *[], int);
int cmdOpenFiles(char *[], int, FILE **, FILE **, FILE **);
int cmdFetchCoord(char *[], double *, double *, double *);
int cmdFetchAngles(char *[], double *, double *, double *);
=====
compute_matrix.h
-Compute rotation matrix from input rotation angles
about x,y,z axes
=====
void ComputeMatrix(double [3][3], double, double, double);
=====
grid_array.h
-Load and write 3D grid files consisting of XYZP data
=====
void GridZero();
void GridLoad(FILE *, int*, int *, int *, int *, int *, int *);
void GridArrayPrint(int);
void GridArrayWrite(FILE *, int);
=====
interpolate.h
-Perform 3D interpolation at a given point
=====
void interpSet(int, int, int, int, int, int);
double interpolate(Point3Df *);
=====
roi.h
-Read region of interest file and determine if a given
point is inside the ROI
=====
void roiLoad(FILE *);
void roiSetPoint(Point3Df *);
int roiPointInside(Point3Df *);
=====
rotate.h
-Rotate points in a 3D grid using an input rotation matrix
=====
void RotateGrid(double [3][3], int);
=====
translate.h
-Translate points in a 3D grid using input translations in x,y,z
directions

```

Burt Hahn 9/16/96

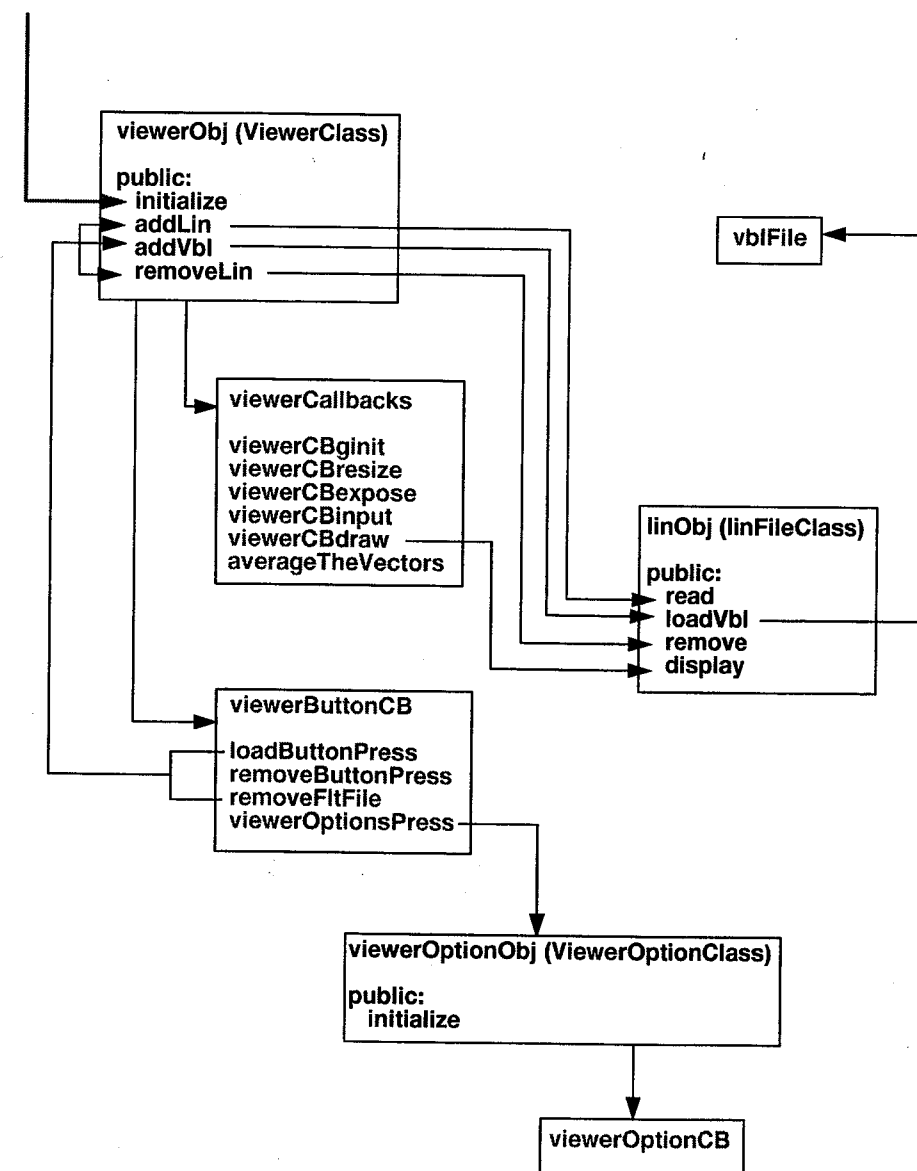
3D Stress V1.2 block diagrams



Burt H. H.

10/31/96

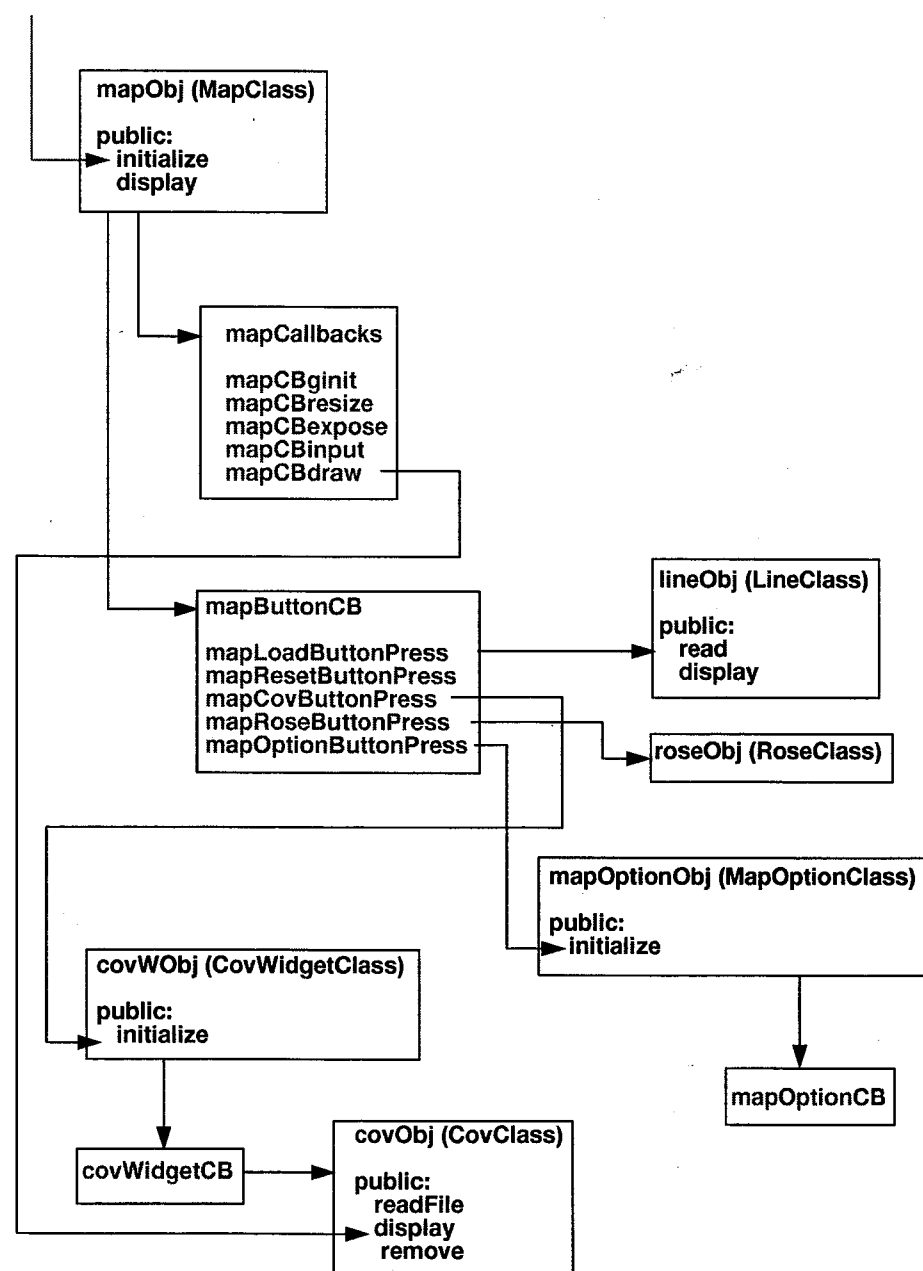
Cont'd



Burt H. H.

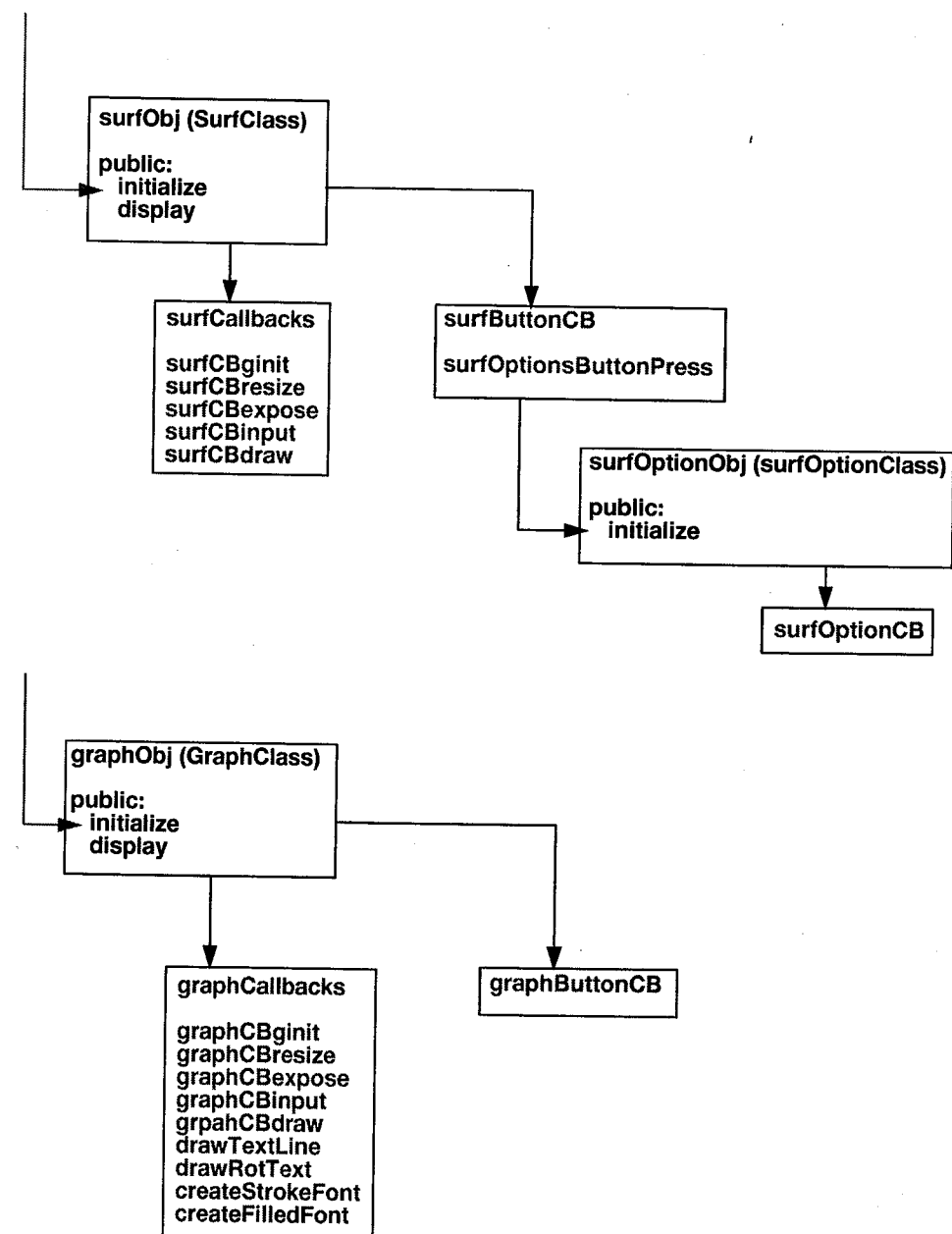
10/31/96

Cont'd



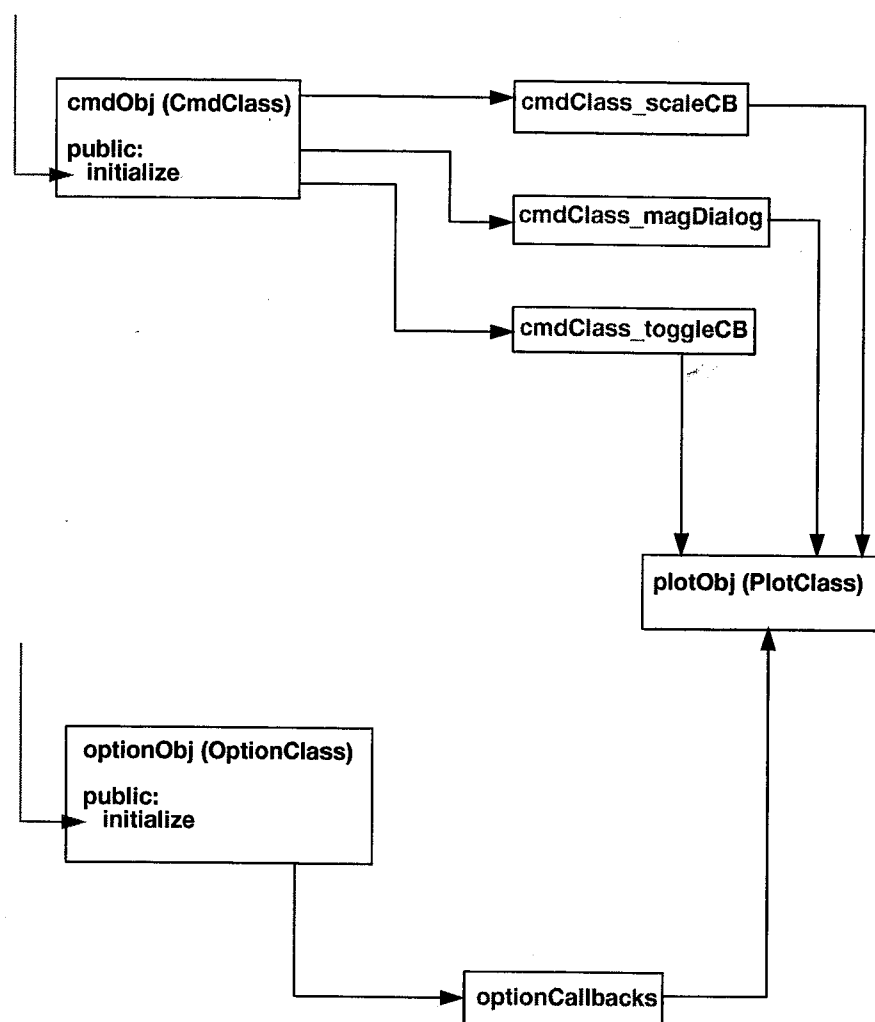
Butcher 10/31/96

Cont'd



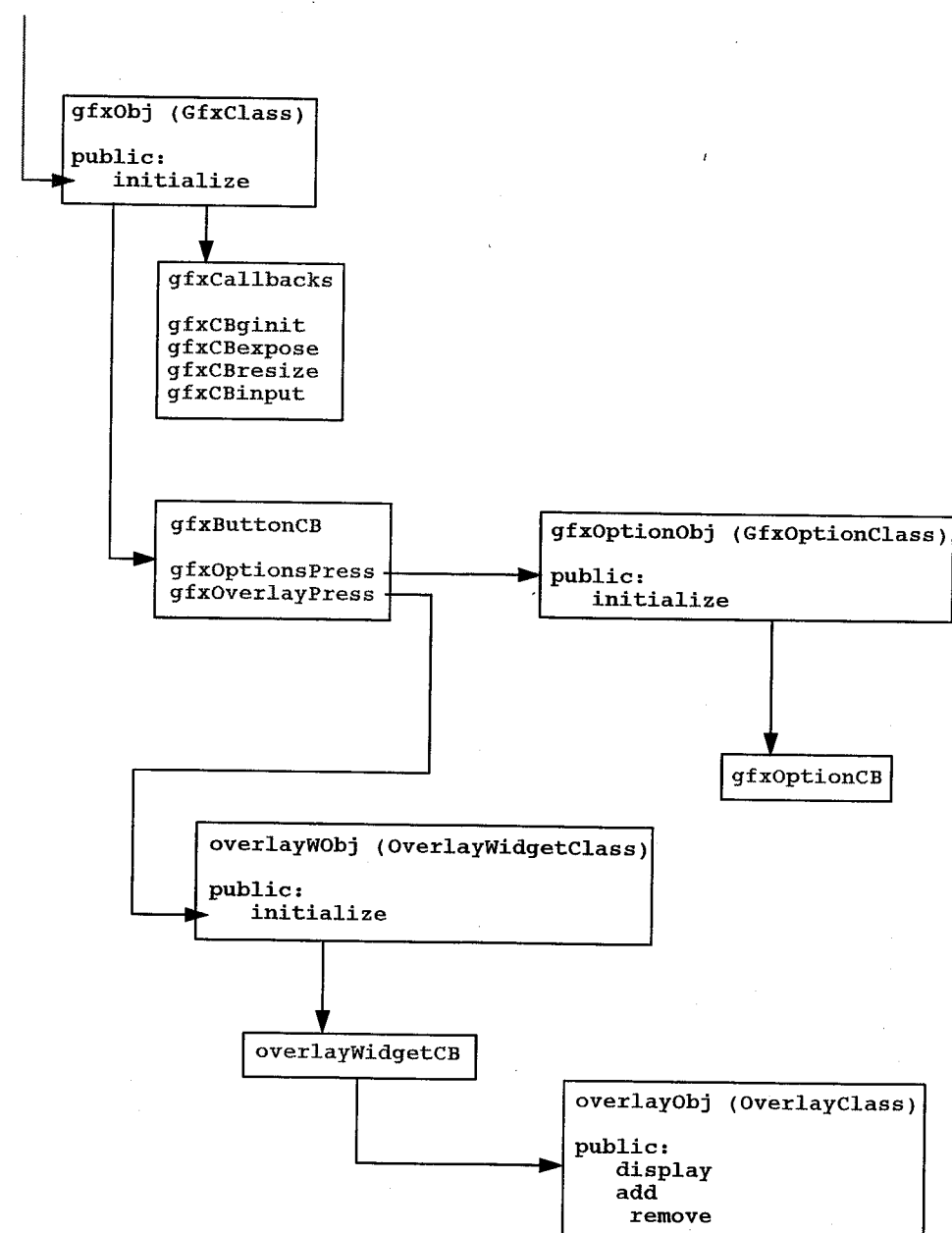
Butcher 10/31/96

Cont'd



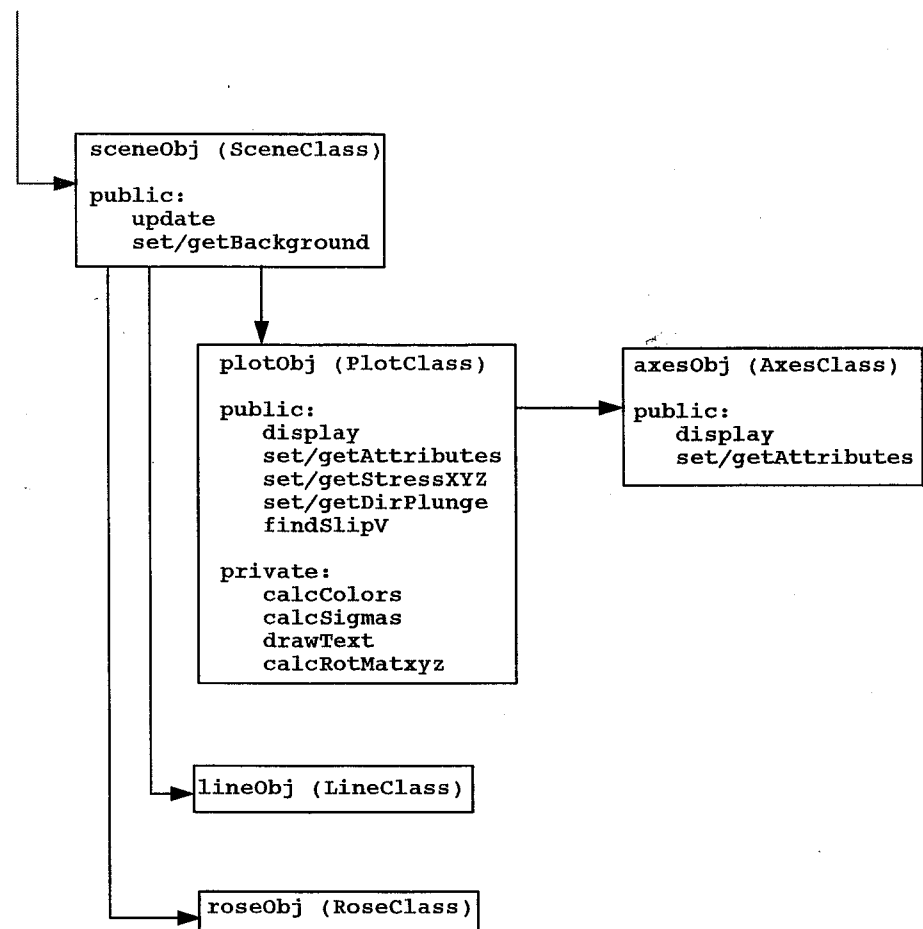
Burt Lahr 10/31/96

Cont'd



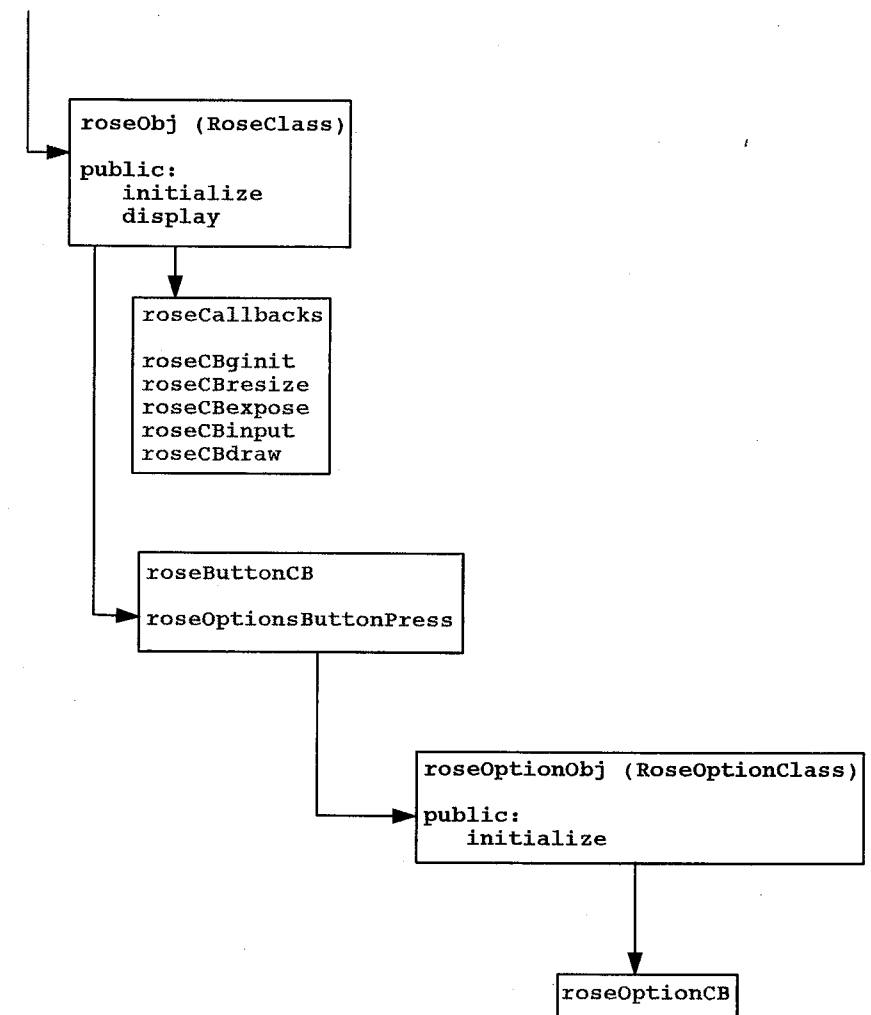
Burt Lahr 10/31/96

Cont'd



Butcher 10/31/96

Cont'd



Butcher 10/31/96

3DStress Documents

3DStress Version 1.2

Software Requirement Description

Date: 8/6/96
Location: 3DStress QA Folder, CNWRA, San Antonio

Source Code and Executable

Date: 10/31/96
Location: 4mm Tape, 3DStress QA Folder, CNWRA, San Antonio

Design Verification Report

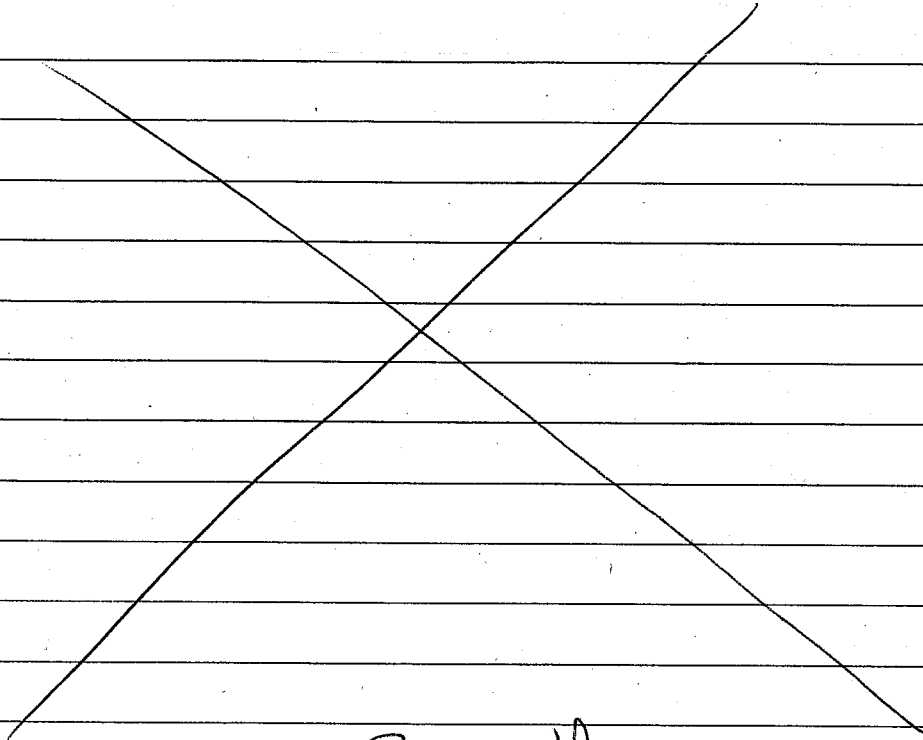
Date: 10/31/96
Location: 3DStress QA Folder, CNWRA, San Antonio

Installation Test Report

Date: 10/31/96
Location: 3DStress QA Folder, CNWRA, San Antonio

User's Manual

Date: 10/31/96
Location: 3DStress QA Folder, CNWRA, San Antonio



Rut Hm 10/31/96

3DStress Source Code Summary

Oct 31 1996 08:44

rlog

Page 1

/usr/people/rboenau/3d/src

Thu Oct 31 08:43:47 CST 1996

Filename	Rev	Line Count
1. axesClass.c++	1.3	236
2. axesClass.hh	1.2	78
3. axesObj.hh	1.2	43
4. boundBox.c++	1.6	205
5. boundBox.hh	1.2	57
6. cmdClass.c++	1.45	632
7. cmdClass.hh	1.4	62
8. cmdClass_magDialog.c++	1.12	323
9. cmdClass_plotDialog.c++	1.3	52
10. cmdClass_scaleCB.c++	1.6	297
11. cmdClass_surfDialog.c++	1.3	52
12. cmdClass_toggleCB.c++	1.24	601
13. cmdClass_uncertDialog.c++	1.5	293
14. cmdObj.hh	1.3	48
Cannot open colorLUT.hh: No such file or directory		
15. colorLUT.hh	1.3	0
16. controlClass.c++	1.10	236
17. controlClass.hh	1.1	40
18. controlerCallbacks.c++	1.8	146
19. controlerCallbacks.hh	1.1	28
20. covClass.c++	1.6	265
21. covClass.hh	1.3	59
22. covModify.c++	1.5	381
23. covObj.hh	1.1	26
24. covWObj.hh	1.1	21
25. covWidgetCB.c++	1.14	343
26. covWidgetCB.hh	1.1	33
27. covWidgetClass.c++	1.7	568
28. covWidgetClass.hh	1.2	66
29. destroyFS.c++	1.1	24
30. fileSelect.c++	1.3	126
31. fileSelect.hh	1.2	31
32. fileShower.c++	1.10	291
33. fileShower.hh	1.1	30
34. findAngle.c++	1.1	100
35. fontdef.hh	1.1	8776
36. gfxButtonCB.c++	1.11	94
37. gfxButtonCB.hh	1.5	43
38. gfxCallbacks.c++	1.47	708
39. gfxCallbacks.hh	1.3	53
40. gfxClass.c++	1.23	471
41. gfxClass.hh	1.3	66
42. gfxObj.hh	1.2	44
43. gfxOptionCB.c++	1.4	248
44. gfxOptionCB.hh	1.3	37
45. gfxOptionClass.c++	1.8	661
46. gfxOptionClass.hh	1.3	70
47. gfxOptionObj.hh	1.1	22
48. graphButtonCB.c++	1.4	62
49. graphButtonCB.hh	1.2	30
50. graphCallbacks.c++	1.14	1422
51. graphCallbacks.hh	1.4	62
52. graphClass.c++	1.22	412
53. graphClass.hh	1.3	88
54. graphObj.hh	1.2	45
55. helpWidget.c++	1.11	344
56. helpWidget.hh	1.1	26
57. infoWidget.c++	1.1	60
58. infoWidget.hh	1.1	24
59. linFileClass.c++	1.52	1408
60. linFileClass.hh	1.24	175
61. linObj.hh	1.4	53
62. lineClass.c++	1.37	780
63. lineClass.hh	1.10	101

Rut Hm 10/31/96

Cont'd

Printed by robenau from yosemite

Oct 31 1996 08:44 rlog Page 2

```

64. lineObj.hh                1.2      46
65. mapButtonCB.c++          1.22     269
66. mapButtonCB.hh           1.3       39
67. mapCallbacks.c++         1.21     813
68. mapCallbacks.hh           1.1       67
69. mapClass.c++             1.15     566
70. mapClass.hh               1.6       95
71. mapOptionCB.c++          1.4      117
72. mapOptionCB.hh           1.2       35
73. mapOptionClass.c++       1.5     289
74. mapOptionClass.hh         1.2       50
75. mapOptionObj.hh           1.1       21
Cannot open menuCallbacks.c++: No such file or directory
76. menuCallbacks.c++         1.39       0
Cannot open menuCallbacks.hh: No such file or directory
77. menuCallbacks.hh          1.3       0
Cannot open menuClass.c++: No such file or directory
78. menuClass.c++            1.21       0
Cannot open menuClass.hh: No such file or directory
79. menuClass.hh              1.3       0
Cannot open menuObj.hh: No such file or directory
80. menuObj.hh                1.3       0
81. notice.c++                1.2     160
82. notice.hh                  1.1       46
83. optionCallbacks.c++       1.9     314
84. optionCallbacks.hh         1.4       39
85. optionClass.c++           1.12     637
86. optionClass.hh             1.4       74
87. optionObj.hh              1.1       23
88. overlayClass.c++          1.12     366
89. overlayClass.hh           1.5       80
Cannot open overlayDialog.c++: No such file or directory
90. overlayDialog.c++         1.8       0
Cannot open overlayDialog.hh: No such file or directory
91. overlayDialog.hh          1.3       0
92. overlayObj.hh             1.2       46
Cannot open overlayRemove.c++: No such file or directory
93. overlayRemove.c++         1.8       0
Cannot open overlayRemove.hh: No such file or directory
94. overlayRemove.hh          1.3       0
95. overlayWidgetCB.c++       1.9     278
96. overlayWidgetCB.hh        1.1       30
97. overlayWidgetClass.c++    1.7     494
98. overlayWidgetClass.hh     1.2       63
99. ovlWObj.hh                1.2       24
Cannot open pixButton.c++: No such file or directory
100. pixButton.c++            1.1       0
101. plotClass.c++            1.54    1632
102. plotClass.hh              1.39     425
103. plotClass_display.c++     1.41    1344
104. plotClass_findSlipV.c++   1.10     295
Cannot open plotClass_output.c++: No such file or directory
105. plotClass_output.c++      1.4       0
106. plotClass_setDirPlunge.c++ 1.4     527
107. plotObj.hh                1.2       43
108. readBin.c++               1.1     148
109. remove.c++                1.1    322
110. remove.hh                 1.1       45
111. roseButtonCB.c++          1.7     119
112. roseButtonCB.hh           1.3       36
113. roseCallbacks.c++         1.8     789
114. roseCallbacks.hh          1.1       51
115. roseClass.c++             1.22     644
116. roseClass.hh              1.6       99
117. roseObj.hh                1.1       41
118. roseOptionCB.c++          1.2       92
119. roseOptionCB.hh           1.2       30
120. roseOptionClass.c++       1.5     234
121. roseOptionClass.hh        1.3       52
122. roseOptionObj.hh          1.1       21
123. rotClass.c++             1.2     213

```

Bret J. H. 10/31/96

Cont'd

Oct 31 1996 08:44 rlog Page 3

```

124. rotClass.hh              1.1       61
125. saveWindow.c++          1.15     207
126. saveWindow.hh            1.1       21
127. sceneClass.c++           1.6     139
128. sceneClass.hh             1.2       69
129. sceneObj.hh               1.2       43
130. surfButtonCB.c++         1.7     109
131. surfButtonCB.hh           1.2       30
132. surfCallbacks.c++         1.18     836
133. surfCallbacks.hh          1.5       67
134. surfClass.c++            1.23     539
135. surfClass.hh              1.6     108
136. surfObj.hh                1.2       44
137. surfOptionCB.c++         1.1     113
138. surfOptionCB.hh           1.1       26
139. surfOptionClass.c++       1.3     345
140. surfOptionClass.hh        1.2       53
141. surfOptionObj.hh          1.1       21
142. vblFile.c++              1.7     605
143. vblFile.hh                1.2       49
Cannot open vblToFlt.c++: No such file or directory
144. vblToFlt.c++             1.4       0
Cannot open vblToFlt.hh: No such file or directory
145. vblToFlt.hh              1.2       0
146. vectorClass.c++          1.5     165
147. vectorClass.hh            1.4       84
148. viewClass.c++            1.5     228
149. viewClass.hh              1.2     127
150. viewNet.c++              1.37     595
151. viewNetGlobals.hh         1.25     196
152. viewObj.hh                1.2       43
153. viewerButtonCB.c++        1.32     523
154. viewerButtonCB.hh         1.5       45
155. viewerCallbacks.c++       1.58    1315
156. viewerCallbacks.hh        1.5       62
157. viewerClass.c++           1.52    1212
158. viewerClass.hh            1.16     138
159. viewerObj.hh              1.2       44
160. viewerOptionCB.c++        1.14     587
161. viewerOptionCB.hh         1.8       63
162. viewerOptionClass.c++     1.12    1114
163. viewerOptionClass.hh      1.8     105
164. viewerOptionObj.hh        1.1       21

```

Total line count = 44613

Check Sum = 51917 3619 ../bin/3dstress.exp

```

../bin/3dstress.exp:
3dstress 10-30-96 Rev 1.2

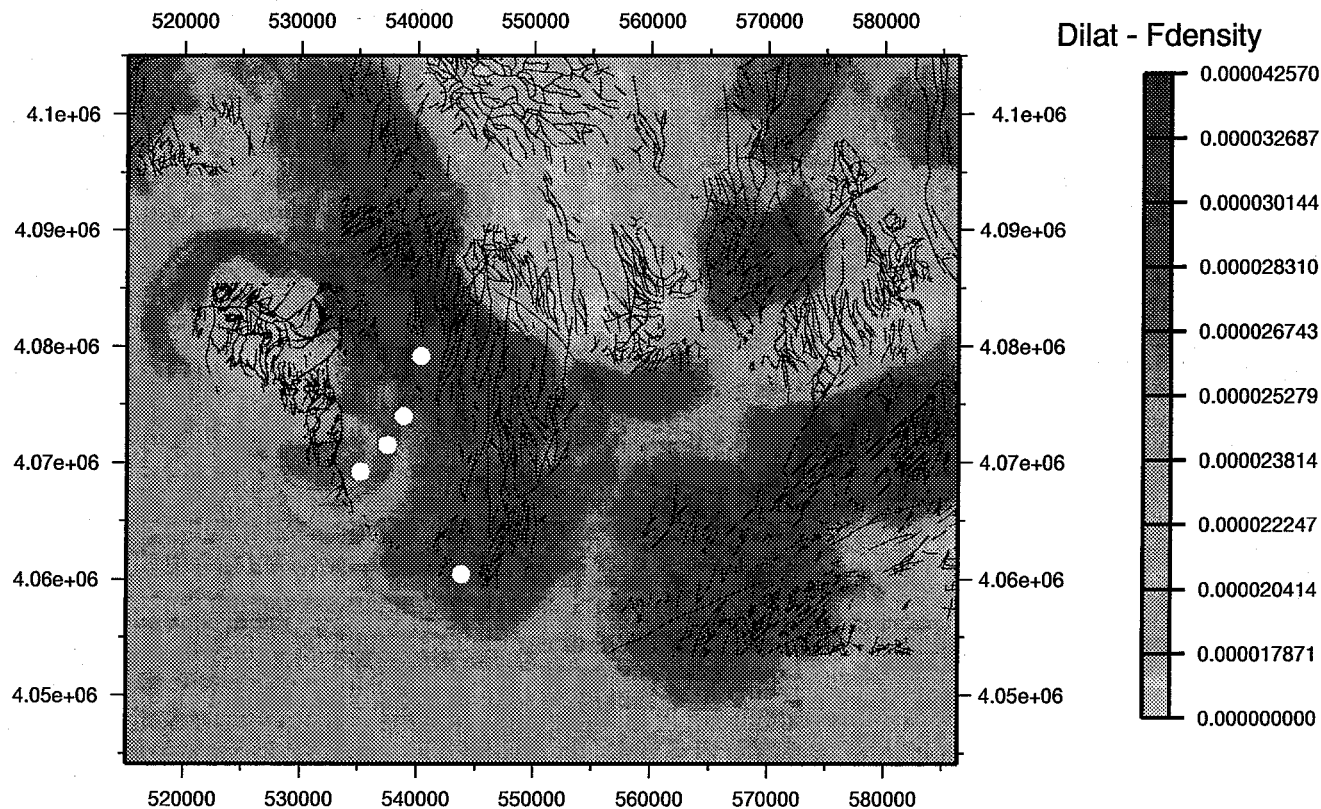
```

Bret J. H. 10/31/96

Frizzell and Bare Mtn Faults Dilation Tendency Analysis

Z = DILATION TENDENCY - FAULT DENSITY

October 24, 1996



/pscr0/brent1/volHazCode_961024/dir961024/friz961024_norm.xyz

Thu Oct 24 15:31:27 CDT 1996

10/24/96
Rt 1/10

Frizzell and Bare Mtn Faults Dilation Tendency Analysis

Z = DILATION TENDENCY - FAULT DENSITY

October 24, 1996

run961024_mkdatt.sh

```
#!/bin/sh

xmin=515085.0
xmax=586347
ymin=4044091.0
ymax=4105030.0
xsp=333.0
ysp=333.0

#-----
# Frizzell and Bare Mtn faults sets,
# z = dilation tendency - fault density
#-----

date

# First use 3dstress.exp to load file.lin
# and save lines to file.xyz

#-----
# echo "\nlineInterp..."
#-----
# ./bin/lineInterp 50 frizbmflts_dil.xyz
# mv frizbmflts_dil_50.lin friz8.lin

#-----
echo "\ngridEllipse..."
#-----
# Usage: gridEllipse
# <e_x-axis> <e_y-axis> <e_rotation from Y-axis>
# <xmin> <xmax> <ymin> <ymax>
# <xinterval> <yinterval> <file.lin>
#-----
./bin/gridEllipse.sav \
5000.0 \
5000.0 \
0.0 \
$xmin \
$xmax \
$ymin \
$ymax \
$xsp \
$ysp \
friz8.lin

date

#-----
# echo "\nlineInterp..."
#-----
# ./bin/lineInterp 50 frizbmflts.xyz
# mv frizbmflts_50.lin friz10.lin

#-----
echo "\ngridEllipse..."
#-----
# Usage: gridEllipse
# <e_x-axis> <e_y-axis> <e_rotation from Y-axis>
# <xmin> <xmax> <ymin> <ymax>
# <xinterval> <yinterval> <file.lin>
#-----
./bin/gridEllipse.sav \
5000.0 \
5000.0 \
0.0 \
$xmin \
$xmax \
$ymin \
$ymax \
$xsp \
$ysp \
friz10.lin

date

#-----
echo "runNormIntegral..."
#-----
./scripts/runNormIntegral friz8_grd.dat > friz8_norm.dat
./scripts/runNormIntegral friz10_grd.dat > friz10_norm.dat

#-----
echo "runStats..."
#-----
./scripts/runStats friz8_norm.dat
./scripts/runStats friz10_norm.dat

#-----
echo "runGridDiff..."
#-----
./scripts/runGridDiff friz8_norm.dat friz10_norm.dat > friz961024.dat

#-----
echo "runStats..."
#-----
./scripts/runStats friz961024.dat

#-----
echo "runMakePositive..."
#-----
./scripts/runMakePositive friz961024.dat > tmp.dat

#-----
echo "runNormIntegral..."
#-----
./scripts/runNormIntegral tmp.dat > friz961024_norm.dat
rm tmp.dat

#-----
echo "runStats..."
#-----
./scripts/runStats friz961024_norm.dat

echo "\nFinished, created friz961024_norm.dat\n"
```

run961024_mkeps.sh

```
gmtset ANOT_FONT_SIZE 8
gmtset LABEL_FONT_SIZE 12
gmtset D_FORMAT %.8f

psbasemap \
-JX4.500000/3.848131 \
-R515085.00000/586347.00000/4044091.00000/4105030.00000 \
-Bf50000.000000000a10000.0000000000 \
-K \
-P \
> pvout.eps

# Create binary grid file...
xyz2grd /pscr0/brent1/volHazCode_961024/dir961024/friz961024_norm.xyz \
-Gpvtmp.grd \
-I333.00000/333.00000 \
-R515085.00000/586347.00000/4044091.00000/4105030.00000

# Create color palette file...
grd2cpt pvtmp.grd \
-M0.888889/0.333333 \
> pvtmp.cpt

# Create color filled contour image PS...
grdimage pvtmp.grd \
-JX4.500000/3.848131 \
-R515085.00000/586347.00000/4044091.00000/4105030.00000 \
-P \
-O \
-Cpvtmp.cpt \
-K \
>> pvout.eps

# Add scale to PS...
psscale \
-D5.500000/2.000000/3.500000/0.150000 \
-Cpvtmp.cpt \
-L -B:"Dilat - Fdensity": \
-P \
-K \
-O >> pvout.eps

rm pvtmp.cpt pvtmp.grd

nawk 'BEGIN {
    lastNotEnd = 1
    firstv = 1
}
{
    if (NF == 1) {
        if (index($0,"END") != 0) {
            if (lastNotEnd) print $0
        }
        lastNotEnd = 0
    } else {
        print $0
        if (firstv) {
            print $0
            firstv = 0
        }
        lastNotEnd = 1
    }
}' /pscr0/brent1/volHazCode_961024/data/frizbmflts.lin > pvtmpl1.lin

# Plot line file /pscr0/brent1/volHazCode_961024/data/frizbmflts.lin...
psxy \
pvtmpl1.lin \
-JX4.500000/3.848131 \
-R515085.00000/586347.00000/4044091.00000/4105030.00000 \
-P \
-W1/0/0/0 \
-MEND \
-O \
-K \
>> pvout.eps

rm pvtmpl1.lin

# Plot points file /pscr0/brent1/volHazCode_961024/data/cones.xy...
psxy \
/pscr0/brent1/volHazCode_961024/data/cones.xy \
-JX4.500000/3.848131 \
-R515085.00000/586347.00000/4044091.00000/4105030.00000 \
-P \
-G255/255/255 \
-Sc0.10000000 \
-O \
-K \
>> pvout.eps

psbasemap \
-JX4.500000/3.848131 \
-R515085.00000/586347.00000/4044091.00000/4105030.00000 \
-Bf500000.000000000a1000000.0000000000 \
-O \
-P \
>> pvout.eps

cp pvout.eps pvtmp.eps

dtmp='date'
```


Convert USGS 3-arcsecond DEM to an EarthVision 2gnd file,
here are the script files used to build the 2gnd file:

Nov 21 1996 10:07 **README.961121** Page 1

```

.:
.:
run010_demToArcSec
.:
.:
#!/bin/sh

gunzip goldfield-e.gz

geoview \
-in goldfield-e \
-intype 4 \
-out goldfield-e.arcsec \
-outtype 6 \
-nodraw

gzip goldfield-e

exit

.:
.:
run020_arcSecToDeg
.:
.:
#!/bin/sh

nawk '{ x = $1 / 3600.0
        y = $2 / 3600.0
        z = $3 + 0.0
        printf "%20.8f %20.8f %20.8f\n", x,y,z
      }' $1

.:
.:
run030_degToUtm
.:
.:
#!/bin/sh

PROJ_LIB=/usr3/brent/geoview/lib_proj/PROJ.4/nad
export PROJ_LIB
PROJ_BIN=/usr3/brent/geoview/lib_proj/PROJ.4/bin
export PROJ_LIB

$PROJ_BIN/proj +proj=utm +zone=11 goldfield-e.deg > goldfield-e.utm
exit

$PROJ_BIN/proj +proj=utm +zone=11 death_valley-e.deg > death_valley-e.utm
exit

.:
.:
runGoldfield
.:
.:

run010_demToArcSec

run020_arcSecToDeg goldfield-e.arcsec > goldfield-e.deg

run030_degToUtm

runExtract goldfield-e.utm > ymtn_dem2.dat

.:
.:
runExtract
.:
.:
#!/bin/sh

nawk '{
    x = $1 + 0.0
    y = $2 + 0.0

```

Continued...

Nov 21 1996 10:07 **README.961121** Page 2

```

z = $3 + 0.0
if ((x > 514000) && (x < 588000) &&
    (y > 4043000) && (y < 4104000)) {
    print $0
}
}' $1

::::::::::::::::::
runMinMaxXYZ
::::::::::::::::::
#!/bin/sh

if [ $# -lt 1 ]
then
    echo "\n\n$# of 1"
    echo "USAGE runMinMaxXY <filename>\n\n"
    exit
fi

nawk 'BEGIN {
    needFirstOne = 1
}
{
    if (NF > 1) {
        x = $1 + 0.0
        y = $2 + 0.0
        z = $3 + 0.0
        if (needFirstOne) {
            minx = x
            bigx = x
            miny = y
            bigy = y
            minz = z
            bigz = z
            needFirstOne = 0
        }
        if (x < minx) minx = x
        if (x > bigx) bigx = x
        if (y < miny) miny = y
        if (y > bigy) bigy = y
        if (z < minz) minz = z
        if (z > bigz) bigz = z
    }
}
END {
    printf "X Min %12.4f   Max %12.4f\n", minx, bigx
    printf "Y Min %12.4f   Max %12.4f\n", miny, bigy
    printf "Z Min %12.4f   Max %12.4f\n", minz, bigz
}' $1

::::::::::::::::::
runEVgrd.sh
::::::::::::::::::
#!/bin/sh -e

FEATURE=ev-gmx/svp
RANGE='515085.000000,586347.000000,4044091.000000,4105030.000000'
SIZE=215,184
GRID=ymt_n_dem4.2grd
DATA=ymtn_dem4.dat
FIELD='z'
VFILT=
NVFLT=
ev_2grid -o $GRID -F $FEATURE -r $RANGE -s $SIZE -d $DATA -f "$FIELD" -G ymt_n_dem4.2grpt -m 4 -x

::::::::::::::::::
runEVexport.sh
::::::::::::::::::
```

Burt Hahn 11/21/96

Burt Hahn 11/2/91

Continued...

```

Nov 21 1996 10:07      README.961121      Page 3

#!/bin/sh

ev_export -o ymtn_dem6_repos.dat -x ymtn_dem6_repos.2grd
runMinMaxXYZ ymtn_dem6_repos.dat > ymtn_dem6_repos.ranges

exit

ev_export -o ymtn_dem5.dat -x ymtn_dem4.2grd
runMinMaxXYZ ymtn_dem5.dat > ymtn_dem5.ranges

:
runEVplyfill.sh
:
#!/bin/sh -e

ev_fill -p m60_outline.ply -o ymtn_dem6_repos.2grd -f 1098 ymtn_dem4.2grd

:
m60_outline.ply
:
# Type: polygon data
# Version: 3
# Format: free
# Field: 1 x
# Field: 2 y
# Projection: Local Rectangular
# Units: unknown
# End:
POLYGON
548237.1250 4079741.0000
548196.1875 4079703.0000
548170.1875 4079676.5000
548133.5000 4079616.0000
548111.5625 4079574.7500
548078.3750 4079529.2500
548045.3125 4079476.5000
548008.3125 4079431.2500
547964.1875 4079363.2500
547938.5625 4079314.2500
547920.2500 4079284.2500
547894.3125 4079254.0000
547872.0625 4079231.2500
547853.7500 4079197.2500
547824.5000 4079144.5000
547794.9375 4079106.7500
547758.0625 4079054.0000
547688.1250 4078948.2500
547640.4375 4078869.2500
547570.3750 4078771.0000
547541.1250 4078718.2500
547515.3125 4078680.5000
547500.6875 4078654.2500
547478.5625 4078624.0000
547460.1875 4078593.7500
547442.2500 4078541.2500
547424.2500 4078492.5000
547421.3750 4078444.0000
547411.8125 4078350.5000
547402.1250 4078264.5000
547396.0000 4078186.2500
547375.0000 4078096.2500
547361.5625 4078006.5000
547347.7500 4077939.2500
547333.5000 4077890.5000
547308.5000 4077811.7500

```

READM

Burt Hahn 11/21/96

Using Virtual Reality Modeling Language (VRML) to visualize earthquake hypocenters.

VRML is a language specification that describes 3-D scenes that can be rendered on systems with a VRML viewer or browser.

For more information on VRML see the Web site:

<http://www.sdsc.edu/vrml/>

This work is summarized for a planned poster presentation at the 97 Geospatial Information Age ASPRS-ACSM-RT-Auto Carto 13 meeting at Seattle, Washington, April, 1997.

A draft of the poster paper is presented on the following pages

Burt Hahn 12/23/96

VRML cont...

SPATIO-TEMPORAL VISUALIZATION OF EARTHQUAKE HYPOCENTERS OVER THE INTERNET USING VRML

D. Brent Henderson, Sr. Research Engineer
and H. Lawrence McKague, Manager
Center for Nuclear Waste Regulatory Analyses
Southwest Research Institute
San Antonio, Texas USA

ABSTRACT

Usually a major earthquake is preceded by many smaller earthquakes and triggers many aftershocks. Visualization of the distribution of foreshocks, aftershocks, and microseismicity, can lead to a better understanding of earthquake hazards. For example, the locations of the minor earthquakes may define the position and orientation of the causative fault and other nearby major subsurface faults that were unknown and even unsuspected before the major earthquake and its associated minor earthquakes. Future earthquakes might occur along these faults making knowledge of their locations especially critical in the analysis of future seismic hazards.

For many years scientists have measured the times and spatial locations, also called hypocenters, of major and minor earthquakes. In more recent years, many of these earthquake records have been made available electronically over the Internet. However, visualization of these data has required platform specific software making their widespread use difficult.

Recently three-dimensional (3-D) visualization tools that render Virtual Reality Modeling Language (VRML) files have been introduced that execute on any of the major computer platforms. In this paper, earthquake hypocenter visualizations are presented that were produced using the VRML specification. In addition, a hypocenter animation was also produced using VRML to display a two-day earthquake sequence recorded in 1992 in southern Nevada.

EARTHQUAKE HYPOCENTERS

Earthquakes are produced when the earth rapidly releases energy built up from strain in rocks along faults in the earth. For years scientists have measured the locations and magnitudes of earthquakes using seismographs. The sensitivity and precision of seismographs have improved over time. Today large networks of seismographs are used to monitor the earth and to detect earthquake activity. The data recorded by seismographs are called seismograms. Some

VRML cont...

seismograms can be downloaded via the Internet. To find seismograms on the Internet, check the Web sites listed below.

Council of the National Seismic System <http://quake.geo.berkeley.edu:80/cnss>
National Geophysical Data Center <http://www.ngdc.noaa.gov/seg/hazard/earthqk.html>
USGS National Earthquake Information Center <http://www.neic.cr.usgs.gov>
Seismosurfing <http://www.geophys.washington.edu/seismosurfing.html>

Visualization of earthquake hypocenters is useful for determining the locations of major faults and fractures. Scientists also plot hypocenters to predict a forthcoming major earthquake since foreshocks often precede a major earthquake as strain builds up along a fault. Besides the location of earthquakes, researchers want to represent additional information such as magnitude, date of earthquake event, and seismic wave parameters.

Figure 1 is a view of earthquake hypocenters in North America with magnitudes 6.0 and above since 1868. As shown in the figure, most earthquakes occur within narrow zones around the globe. These zones correspond to the boundaries of large slabs of the earth's crust called plates.

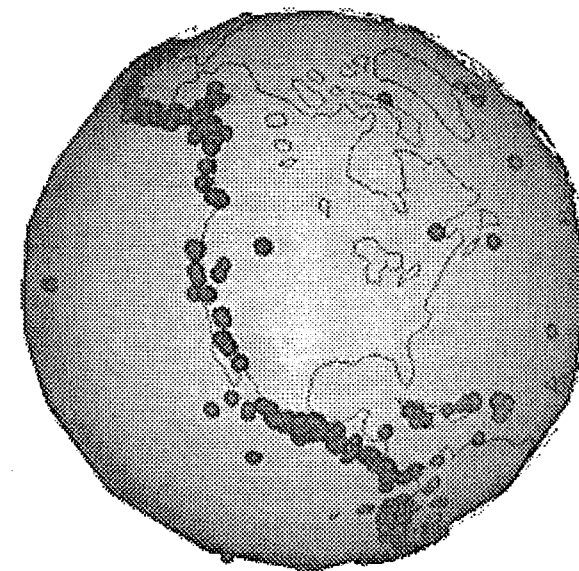


Figure 1 Earthquake hypocenters with magnitudes 6.0 and above in North America since 1869.

Brent Henderson 12/23/96

Brent Henderson 12/23/96

VRML cont.

VISUALIZATION USING VRML

Production of earthquake hypocenter visualizations often requires special software systems and programming expertise. However, a new specification for creating 3-D visualizations on the Internet has been developed. The specification is called Virtual Reality Modeling Language (VRML). VRML enables rapid development of complex hypocenter visualizations that can be used by all Internet users (Ames, 1996).

The VRML specification is a language for specifying 3-D objects and scenes that can be delivered across the Internet. VRML scenes can be created using a text editor or a VRML builder application. VRML files made available over the Internet can be downloaded and displayed by other systems on the Internet. Rendering VRML files requires a VRML viewer. Sometimes a VRML viewer can be added as a plug-in to a Web browser. Many free and commercial VRML viewers available today can be used to render VRML files. For information on VRML viewers, specifications, example files, books, and tutorials see the VRML Repository Web site at <http://www.sdsc.edu/vrml/>.

LITTLE SKULL MOUNTAIN CASE STUDY

To test the capabilities of using VRML to display hypocenters, a data set was selected that was recorded in June 1992 near Little Skull Mountain (LSM) in southern Nevada. Because this event occurred within 20 km of a potential national nuclear waste repository at Yucca Mountain, Nevada, it gained considerable attention. The seismograms from this event were used to figure out the location and orientation of a major fault in the region (Harmsen, 1994).

Figures 2 and 3 are visualizations of the LSM event data using VRML. Figure 2 presents the hypocenters with magnitudes ranging from 0.6 to 5.6 over a 47-hour time span. Each hypocenter is represented by a sphere colored by depth and sized by magnitude. An overall clustering of the hypocenters in Figure 2 can be detected when viewing the data using a VRML browser.

Figure 3 presents a subset of the LSM hypocenters shown in Figure 2. Using the hypocenters shown in Figure 2, a planar trend grid was computed. The orientation of the trend grid was N30°E, S56°W. Only those hypocenters within 750m of the trend grid were used in Figure 3. More than 70 percent of the hypocenters shown in Figure 2 are within 750m of the computed trend grid. The trend grid may represent the location and orientation of the fault where the earthquake occurred.

Brent Hendon 12/23/96

VRML cont.

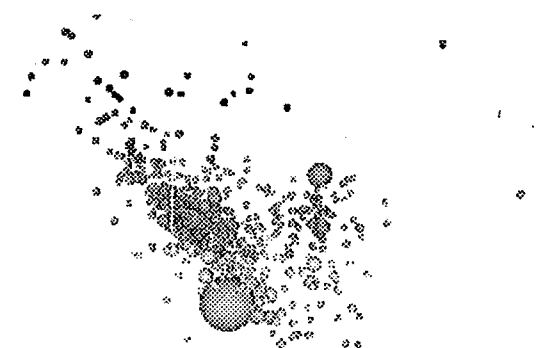


Figure 2 Visualization of the LSM hypocenters recorded on June 29 and 30, 1992.

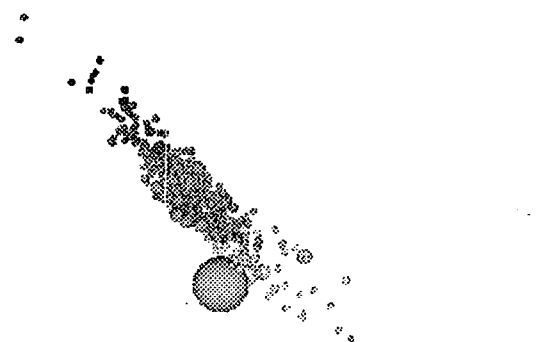


Figure 3 LSM hypocenters within 750m of a trend grid computed from the hypocenters shown in Figure 2.

Brent Hendon 12/23/96

VRML cont...

TIME SEQUENCE ANIMATIONS

The VRML version 2.0 language enables animations as well as static 3-D displays. Using the time of each earthquake hypocenter occurrence, a time animation of the LSM event was produced using the VRML language.

CONCLUSIONS

Earthquake hypocenters have been recorded for many years and numerous data sets are available over the Internet. The visualization of these hypocenters is useful in determining faults and fractures and in earthquake hazard prediction. The VRML language is an effective means of rendering hypocenter data. VRML files made available over the Internet can be visualized by anyone with a VRML viewer. Time animations of hypocenter events can also be easily created using the VRML language.

ACKNOWLEDGMENTS

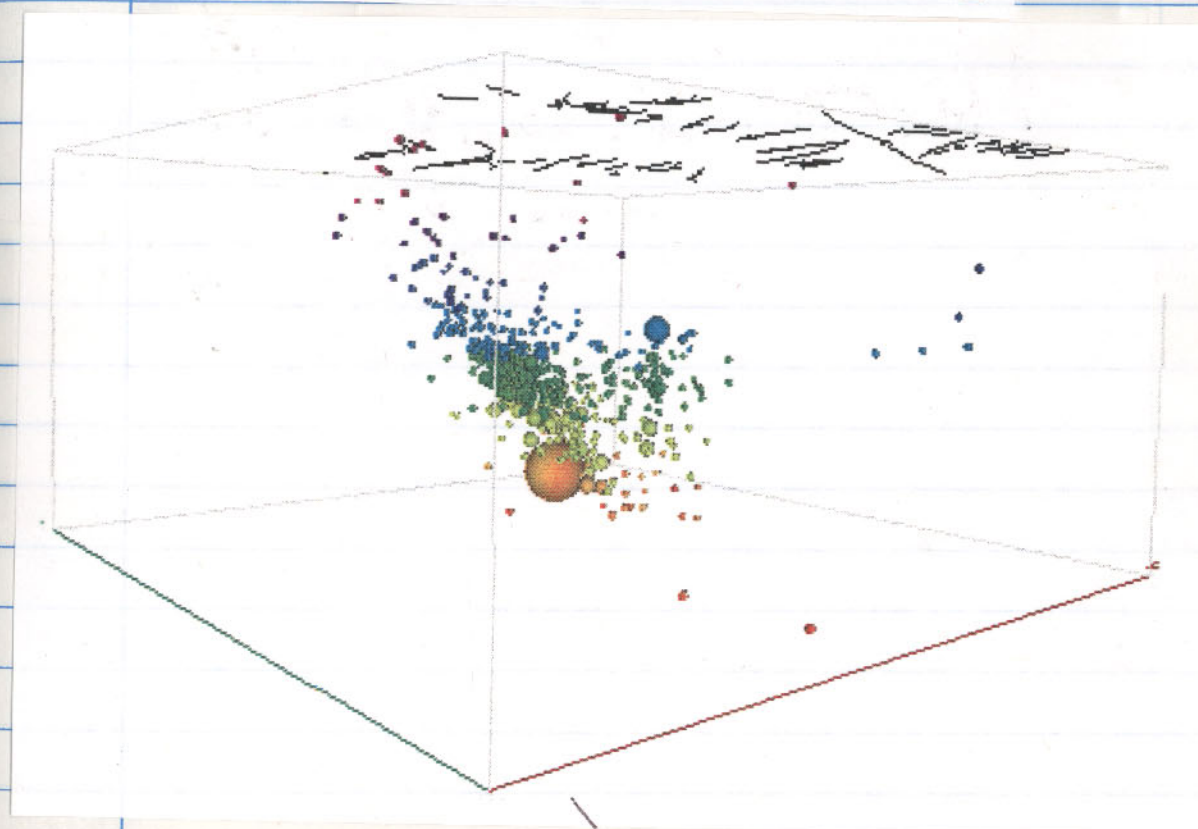
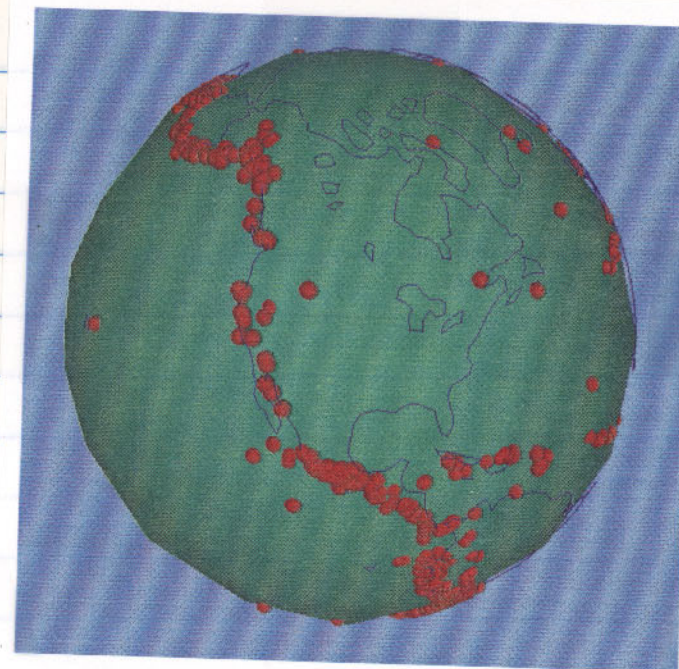
The hypocenter visualizations were developed by the CNWRA for the U.S. NRC Office of Nuclear Material Safeguards, Division of Waste Management, Contract NRC-02-93-005. This poster is an independent product of the CNWRA and does not necessarily reflect the views or regulatory position of the NRC. D. Ferrill and J. Stamatakis provided technical review. K. Smith (University of Nevada-Reno) provided the LSM hypocenter data set. The global hypocenter data set was provided by the National Earthquake Information Center.

REFERENCES

- Ames, A.L., D.R. Nadeau, and J.L. Moreland (1996). *The VRML Sourcebook*, John Wiley & Sons, Inc., New York.
- Harmsen, S.C. (1994). The Little Skull Mountain, Nevada, Earthquake of 29 June 1992: Aftershock Focal Mechanisms and Tectonic Stress Field Implications, *Bulletin of the Seismological Society of America*, 84(5):1484-1505.

Brent Hender 12/23/96

Color prints of VRML visualizations of hypocenters...



Brent Hender 12/23/96

Script files used to create VRML visualization
of hypocenters.

Dec 20 1996 15:00

README.961220

Page 1

Using VRML to visualize earthquake hypocenters.

Little Skull Mountain (LSM) June 1992 data set:

Here are the Little Skull Mountain relocations. There may be some earthquakes out side of the Little Skull Mountain aftershock zone included in this list and I wouldn't trust any of those locations. These were relocated for a set of station corrections developed for the LSM area. Magnitudes are from Harsen through September, they are Mls determined from the low gain horizontal component at the GS LSM station. M from September 1 through December are UNR coda duration magnitudes. Close is the closest station (epicentral distance) and Far is the further station that was picked. Only stations within 75 km were used in the location, and an average residual station correction was determined for all sites. Portable stations from both UNR and the USGS are included in the relocations. Let me know if you have any questions.

Ken Smith
ken@seismo.unr.edu
702-784-4218

```
=====
run000batch1
=====
#!/bin/sh

# Points within 750m of plane lsm060.pln
#-----
run011extract.sh    lsm011.rawdata    lsm021_geo.xyzm
run021geo2utm.sh    lsm021_geo.xyzm    lsm031_utm.xyzm
pntsPlane           lsm060.pln lsm031_utm.xyzm 750.0 lsm034_utm.xyzm
run031dat2wrl.sh    lsm034_utm.xyzm    lsm044.wrl
run041addlin2wrl.sh frizzell.lin       lsm044.wrl
```

```
=====
run011extract.sh
=====
#!/bin/sh

echo "$0 extracting x-y-z-mag-time(min) data..."
```

```
nawk 'BEGIN {
    firstone = 1
    }{
        if (substr($0,1,1) != "#") {
            lng = -1.0 * ($6 + ($7/60.0))
            lat = $4 + ($5/60.0)
            z = $8 * -1000.0
            mag = $9 + 0.0
            if ((z > -15000.0) &&
                (mag > 0.5)) {
                if (firstone) {
                    firstone = 0
                    min = substr($2,3,2) + 0.0
                    hr = substr($2,1,2) + 0.0
                    t0 = ($1 * 24 * 60) + (hr * 60) + min
                }
                min = substr($2,3,2) + 0.0
                hr = substr($2,1,2) + 0.0
                time = (($1 * 24 * 60) + (hr * 60) + min) - t0
                if (time < 3000) {
                    printf "%13.6f %13.6f %13.6f %6.3f %16d\n",
                        lng, lat, z, mag, time
                }
            }
        }
    }
```

Dec 20 1996 15:00

README.961220

Page 2

```

    }
} } $1 > $2

echo "Finished"

=====
run021geo2utm.sh
=====
#!/bin/sh

echo "$0 converting to UTM..."

geoFile=$1
utmFile=$2

# echo "Enter name of XY input file with lat/lon (geo) coords:"
# read geoFile

# echo "Enter name of XY output file in UTM coords:"
# read utmFile

# PERFORMER
# PROJ_LIB=/usr3/brent/geoview/lib_proj/PROJ.4/nad
# PROJ_BIN=/usr3/brent/geoview/lib_proj/PROJ.4/bin

# REDWOOD
# PROJ_LIB=/usr/people/brent/PROJ.4/nad
# PROJ_BIN=/usr/people/brent/PROJ.4/bin

export PROJ_LIB
export PROJ_LIB

#-----
# From Lng/Lat To UTM Zone 11
# -E = echo coords being converted
#-----
$PROJ_BIN/proj +proj=utm +zone=11 $geoFile > $utmFile

echo "Finished"
```

```
=====
run025pntInArea.sh
=====
#!/bin/sh
```

```
if [ $# -lt 1 ]
then
    echo "$0 file.xyz"
    exit
fi

nawk 'BEGIN {
    q0x = 559000
    q0y = 4061000

    q1x = 562000
    q1y = 4061000

    q2x = 565000
    q2y = 4068000

    q3x = 562000
    q3y = 4068000
} {
    x = $1 + 0.0
    y = $2 + 0.0
```

README.961220

1

Burt Hehn 12/22/96

Dec 20 1996 15:00

README.961220

Page 3

```
cmd = sprintf("testquad %f %f %d %d %d %d %d %d %d",
x,y,q0x,q0y,q1x,q1y,q2x,q2y,q3x,q3y)
ok = system(cmd)
if (ok) print $0
}' $1
```

```
=====
run029pntInArea.sh
=====
```

```
#!/bin/sh
```

```
if [ $# -lt 1 ]
then
echo "$0 file.xyz"
exit
fi
```

```
nawk 'BEGIN {
q0x = 558955
q0y = 4055780

q1x = 567525
q1y = 4070950
```

```
}{
x = $1 + 0.0
y = $2 + 0.0
if ((x >= q0x) && (x <= q1x) &&
(y >= q0y) && (y <= q1y)) {
print $0
}
}' $1
```

```
=====
run031dat2wrl.sh
=====
```

```
#!/bin/sh
```

```
indata=$1
outwrl=$2
today=`date | nawk '{print $6_"$2_"$3_"$4"}'
```

```
#-----
# Reads input file:
#-----
# x y z m t
# x y z m t
# x y z m t
# x y z m t
# x y z m t
#
# m = earthquake magnitude
# t = time in minutes
#-----
# Outputs WRL file
#-----
```

```
echo "$0 finding min/max..."
```

```
nawk 'BEGIN {
needFirstOne = 1
{
if (NF > 1) {
x = $1 + 0.0
y = $2 + 0.0
z = $3 + 0.0
m = $4 + 0.0
t = $5 + 0.0
```

Dec 20 1996 15:00

README.961220

Page 4

```
if (needFirstOne) {
minx = x
bigx = x
miny = y
bigy = y
minz = z
bigz = z
minm = m
bigm = m
mint = t
bigt = t
needFirstOne = 0
}
```

```
if (x < minx) minx = x
if (x > bigx) bigx = x
if (y < miny) miny = y
if (y > bigy) bigy = y
if (z < minz) minz = z
if (z > bigz) bigz = z
if (m < minm) minm = m
if (m > bigm) bigm = m
if (t < mint) mint = t
if (t > bigt) bigt = t
}
```

```
END {
printf "X Min %12.4f Max %12.4f\n", minx, bigx
printf "Y Min %12.4f Max %12.4f\n", miny, bigy
printf "Z Min %12.4f Max %12.4f\n", minz, bigz
printf "M Min %12.4f Max %12.4f\n", minm, bigm
printf "T Min %12.4f Max %12.4f\n", mint, bigt
```

```
xrng = bigx - minx
yrng = bigy - miny
if (xrng > yrng) {
scale = 20.0/xrng
} else {
scale = 20.0/yrng
}
```

```
xmin = (minx + bigx)/2.0
ymin = (miny + bigy)/2.0
printf "Xmid %f\nYmid %f\nScale %f\n",
xmin, ymin, scale
}' $indata > tmp.minmax
```

```
echo "$0 writing wrl file..."
```

```
nawk 'function prtnode(x,y,z,m,t) {
printf "DEF POS%05d PositionInterpolator {\n", NR
printf "key [ 0.0, %7.4f, %7.4f, 1.0 ]\n", t, t+0.0001
print "keyValue [ 0.01 0.01 0.01, 0.01 0.01 0.01,"
print "1.0 1.0 1.0, 1.0 1.0 1.0 ]"
print "}"
printf "ROUTE Timer1.fraction_changed TO POS%05d.set_fraction\n", NR
printf "DEF SHP%05d Transform {\n", NR
print "translation", x,y,z
print "children ["
print "Shape ["
# Select color
if (z < -7.5) {
print "appearance USE C00"
} else if (z < -3.75) {
print "appearance USE C01"
} else if (z < -2.5) {
print "appearance USE C02"
} else if (z < -1.25) {
print "appearance USE C03"
} else if (z < 0.0) {
print "appearance USE C04"
} else if (z < 1.25) {
print "appearance USE C05"
```

README.961220

Burt Hehn 12/22/96

Dec 20 1996 15:00

README.961220

Page 5

```

    } else if (z < 2.5) {
        print "      appearance USE C06"
    } else if (z < 3.75) {
        print "      appearance USE C07"
    } else {
        print "      appearance USE C08"
    }

    # Select size
    if (m < 3.0) {
        print "      geometry USE Size0"
    } else if (m < 4.0) {
        print "      geometry USE Size1"
    } else if (m < 5.0) {
        print "      geometry USE Size2"
    } else {
        print "      geometry USE Size3"
    }
    print "    } #End Shape"
    print "  }"
    print " } #End Transform"
    printf "ROUTE POS%05d.value_changed TO SHP%05d.set_scale\n", NR, NR
    print ""
}
BEGIN {
    getline <"tmp.minmax"
    x0 = $3 + 0.0
    x1 = $5 + 0.0
    getline <"tmp.minmax"
    y0 = $3 + 0.0
    y1 = $5 + 0.0
    getline <"tmp.minmax"
    z0 = $3 + 0.0
    z1 = $5 + 0.0
    getline <"tmp.minmax"
    m0 = $3 + 0.0
    m1 = $5 + 0.0
    getline <"tmp.minmax"
    t0 = $3 + 0.0
    t1 = $5 + 0.0

    xrng = x1 - x0
    yrng = y1 - y0
    zrng = z1 - z0
    trng = t1 - t0
    if (xrng > yrng) {
        scale = 20.0/xrng
    } else {
        scale = 20.0/yrng
    }
    xmin = (x0 + x1)/2.0
    ymin = (y0 + y1)/2.0
    zmin = (z0 + z1)/2.0
    zscale = 10.0/zrng

    print "#VRML V2.0 utf8"
    print ""
    print "# Author:      B. Henderson"
    print "# Date:          ", dt
    print "# Data file:    ", FILENAME
    print ""
    print "# Input data ranges:"
    print "#   X ",x0," ",x1," Rng ", xrng
    print "#   Y ",y0," ",y1," Rng ", yrng
    print "#   Z ",z0," ",z1," Rng ", zrng
    print "#   M ",m0," ",m1," Rng ", m1-m0
    print "#   T ",t0," ",t1," Rng ", trng
    print "#   XYscale ", scale
    print "#   Zscale ", zscale
    print "#   Xmin ", xmin
    print "#   Ymin ", ymin

```

Dec 20 1996 15:00

README.961220

Page 6

```

    print "#      Zmin ", zmin
    print ""
    print "#-----"
    print "# WorldInfo"
    print "#-----"
    print "WorldInfo {"
    print "  title \"Earthquake Hypocenters\""
    print "} #End WorldInfo"
    print ""
    print "#-----"
    print "# Viewing"
    print "#-----"
    print "NavigationInfo {"
    print "  type \"EXAMINE\""
    print "}"
    print "Viewpoint {"
    print "  position 0 0 30"
    print "  description \"DefaultView\""
    print "}"
    print "Viewpoint {"
    print "  position 0 -40 0"
    print "  orientation 1 0 0 1.50"
    print "  description \"ViewN\""
    print "}"
    print "Viewpoint {"
    print "  position -40 0 0"
    print "  orientation 0 1 0 -1.50"
    print "  description \"ViewE\""
    print "}"
    print ""
    print "#-----"
    print "# Background"
    print "#-----"
    print "Background {"
    print "  skyColor [ 0.0 0.0 0.0 ]"
    print "} #End Background"
    print ""
    print "#-----"
    print "# Define Colors and Sizes"
    print "#-----"
    print "Transform {"
    print "  translation -11 -11 -8"
    print "  children ["
    print "    Shape {"
    print "      appearance DEF CBLK Appearance { material Material {"
    print "        diffuseColor 0.0 0.0 0.0"
    print "      }"
    print "      geometry Sphere { radius 0.1 } },"
    print "    Shape {"
    print "      appearance DEF CWHT Appearance { material Material {"
    print "        diffuseColor 1.0 1.0 1.0"
    print "      }"
    print "      geometry Sphere { radius 0.1 } },"
    print "    Shape {"
    print "      appearance DEF CRED Appearance { material Material {"
    print "        diffuseColor 1.0 0.0 0.0"
    print "      }"
    print "      geometry Sphere { radius 0.1 } },"
    print "    Shape {"
    print "      appearance DEF CGRN Appearance { material Material {"
    print "        diffuseColor 0.0 1.0 0.0"
    print "      }"
    print "      geometry Sphere { radius 0.1 } },"
    print "    Shape {"
    print "      appearance DEF CBLU Appearance { material Material {"
    print "        diffuseColor 0.0 0.0 1.0"
    print "      }"
    print "      geometry Sphere { radius 0.1 } },"
    print "    Shape {"
    print "      appearance DEF C08 Appearance { material Material {"
    print "        diffuseColor 0.6 0.0 0.4"

```

README.961220

Burt Hehn 12/22/96

Dec 20 1996 15:00

README.961220

Page 7

```

print "    }]"
print "    geometry Sphere { radius 0.1 } ],"
print "  Shape {"
print "    appearance DEF C07 Appearance { material Material {"
print "      diffuseColor 0.1 0.0 0.7"
print "    }]"
print "    geometry Sphere { radius 0.1 } ],"
print "  Shape {"
print "    appearance DEF C06 Appearance { material Material {"
print "      diffuseColor 0.0 0.4 0.8"
print "    }]"
print "    geometry Sphere { radius 0.1 } ],"
print "  Shape {"
print "    appearance DEF C05 Appearance { material Material {"
print "      diffuseColor 0.0 0.8 0.8"
print "    }]"
print "    geometry Sphere { radius 0.1 } ],"
print "  Shape {"
print "    appearance DEF C04 Appearance { material Material {"
print "      diffuseColor 0.2 0.9 0.0"
print "    }]"
print "    geometry Sphere { radius 0.1 } ],"
print "  Shape {"
print "    appearance DEF C03 Appearance { material Material {"
print "      diffuseColor 0.8 1.0 0.0"
print "    }]"
print "    geometry Sphere { radius 0.1 } ],"
print "  Shape {"
print "    appearance DEF C02 Appearance { material Material {"
print "      diffuseColor 1.0 0.6 0.0"
print "    }]"
print "    geometry Sphere { radius 0.1 } ],"
print "  Shape {"
print "    appearance DEF C01 Appearance { material Material {"
print "      diffuseColor 1.0 0.3 0.0"
print "    }]"
print "    geometry Sphere { radius 0.1 } ],"
print "  Shape {"
print "    appearance DEF C00 Appearance { material Material {"
print "      diffuseColor 1.0 0.1 0.0"
print "    }]"
print "    geometry Sphere { radius 0.1 } ],"
print "  Shape {"
print "    appearance USE CBLK"
print "    geometry DEF Size0 Sphere { radius 0.10 } ],"
print "  Shape {"
print "    appearance USE CBLK"
print "    geometry DEF Size1 Sphere { radius 0.20 } ],"
print "  Shape {"
print "    appearance USE CBLK"
print "    geometry DEF Size2 Sphere { radius 0.40 } ],"
print "  Shape {"
print "    appearance USE CBLK"
print "    geometry DEF Size3 Sphere { radius 0.80 } ],"
print "  }]"
print "} #End Transform"
print ""
print ""
print "-----"
print "# Bounding box"
print "-----"
print "Shape {"
print "  geometry IndexedLineSet {"
print "    coord Coordinate {"
print "      point ["
print "        -10.0 -10.0 -5.0, # 0"
print "        10.0 -10.0 -5.0, # 1"
print "        -10.0 10.0 -5.0, # 2"
print "        10.0 10.0 -5.0, # 3"
print "        -10.0 -10.0 5.0, # 4"

```

Dec 20 1996 15:00

README.961220

Page 8

```

print "        10.0 -10.0 5.0, # 5"
print "        -10.0 10.0 5.0, # 6"
print "        10.0 10.0 5.0, # 7"
print "      ]"
print "    }"
print "  }"
print "  # End Coordinate"
print "  coordIndex ["
print "    # X-axis"
print "    0, 1, -1,"
print "    8, 9, 10, 11, -1,"
print "    12, 13, -1,"
print "    # Y-axis"
print "    0, 2, -1,"
print "    14, 15, 16, 17, -1,"
print "    # Z-axis"
print "    0, 4, -1,"
print "    # Box"
print "    2, 3, 7, 6, 2, -1,"
print "    6, 4, 5, 7, -1,"
print "    3, 1, 5"
print "  ]"
print "  color Color {"
print "    color ["
print "      0.2 0.2 0.2, # 0"
print "      1.0 0.0 0.0, # 1"
print "      0.0 1.0 0.0, # 2"
print "      0.0 0.0 1.0, # 3"
print "    ]"
print "  }"
print "  colorIndex ["
print "    # X-axis"
print "    1, 1, 1,"
print "    1, 1, 1, 1, 1,"
print "    1, 1, 1,"
print "    # Y-axis"
print "    2, 2, 2,"
print "    2, 2, 2, 2, 2,"
print "    # Z-axis"
print "    3, 3, 3,"
print "    # Box"
print "    0, 0, 0, 0, 0, 0,"
print "    0, 0, 0, 0, 0,"
print "    0, 0, 0"
print "  ]"
print "  colorPerVertex TRUE"
print "  } #End IndexedLineSet"
print "} #End Shape"
print ""
print "-----"
print "# Animation Nodes"
print "# Touch --> Timer --> PosInterpolator --> Shape"
print "-----"
print ""
print "-----"
print "# Touch Sensor"
print "-----"
print "Transform {"
print "  translation -10.5 -10.5 -5.5"
print "  children ["
print "    Shape {"

```

README.961220

Burt Hehn 12/23/96

Dec 20 1996 15:00

README.961220

Page 9

```

print "      appearance Appearance {"
print "        material Material {"
print "          diffuseColor 0.1 0.0 0.1"
print "        }"
print "      }"
print "      geometry Box {"
print "        size 1.0 1.0 1.0 "
print "      }"
print "    },"
print "    DEF Touch1 TouchSensor { } "
print "  }"
print "} #End Transform"
print "Transform {"
print "  translation 10.5 10.5 -5.5"
print "  children {"
print "    Shape {"
print "      appearance Appearance {"
print "        material Material {"
print "          diffuseColor 0.1 0.0 0.1"
print "        }"
print "      }"
print "      geometry Box {"
print "        size 1.0 1.0 1.0 "
print "      }"
print "    },"
print "    DEF Touch2 TouchSensor { } "
print "  }"
print "} #End Transform"
print "Transform {"
print "  translation -10.5 10.5 -5.5"
print "  children {"
print "    Shape {"
print "      appearance Appearance {"
print "        material Material {"
print "          diffuseColor 0.1 0.0 0.1"
print "        }"
print "      }"
print "      geometry Box {"
print "        size 1.0 1.0 1.0 "
print "      }"
print "    },"
print "    DEF Touch3 TouchSensor { } "
print "  }"
print "} #End Transform"
print "Transform {"
print "  translation 10.5 -10.5 -5.5"
print "  children {"
print "    Shape {"
print "      appearance Appearance {"
print "        material Material {"
print "          diffuseColor 0.1 0.0 0.1"
print "        }"
print "      }"
print "      geometry Box {"
print "        size 1.0 1.0 1.0 "
print "      }"
print "    },"
print "    DEF Touch4 TouchSensor { } "
print "  }"
print "} #End Transform"
print "-----"
print "# Timer"
print "-----"
print "DEF Timer1 TimeSensor { cycleInterval 5.0 loop FALSE }"
print "ROUTE Touch1.touchTime TO Timer1.set_startTime"
print "ROUTE Touch2.touchTime TO Timer1.set_startTime"
print "ROUTE Touch3.touchTime TO Timer1.set_startTime"

```

Dec 20 1996 15:00

README.961220

Page 10

```

print "ROUTE Touch4.touchTime TO Timer1.set_startTime"
print "-----"
print "# Sound "
print "-----"
print "Sound {"
print "  source DEF Sound1 AudioClip {"
print "    url \"sound1.wav\""
print "    description \"Sound Effect 1\""
print "    loop FALSE"
print "    startTime 1"
print "    stopTime 0"
print "  },"
print "  maxFront 200"
print "  maxBack 200"
print "  spatialize FALSE"
print "}"
print "ROUTE Touch1.touchTime TO Sound1.set_startTime"
print "ROUTE Touch2.touchTime TO Sound1.set_startTime"
print "ROUTE Touch3.touchTime TO Sound1.set_startTime"
print "ROUTE Touch4.touchTime TO Sound1.set_startTime"
print "-----"
print "# Hypocenter Nodes"
print "-----"

{
  x = ($1 - xmin)*scale
  y = ($2 - ymin)*scale
  z = ($3 - zmin)*zscale
  m = $4 + 0.0
  t = $5 + 0.0
  prtnode(x,y,z,m,t,trng)
}
END {
  print " "
  print " "
}' dt=$today $indata > $outwrl

echo "Finished"

echo "runSizes..."
runSizes $outwrl

cat tmp.minmax

=====
run041addlin2wrl.sh
=====
#!/bin/sh

lin=$1
wrl=$2

echo "$0 adding fault lines..."

if [ $# -lt 2 ]
then
  echo "Usage: $0 file.lin file.wrl"
  exit
fi

#-----
# Sample tmp.minmax file:
#-----
# X Min 558877.1400 Max 567811.5700
# Y Min 4047634.4700 Max 4071649.3200
# Z Min 2070.0000 Max 12560.0000

```

README.961220

Burt Hehn 12/23/96

Dec 20 1996 15:00

README.961220

Page 11

```
# M Min      2.6000 Max      5.6000
# T Min      0.0000 Max      2678.0000
# Xmid 563344.355000
# Ymid 4059641.895000
# Scale 0.000833

nawk 'BEGIN {
    getline <"tmp.minmax"
    # Skip X
    getline <"tmp.minmax"
    # Skip Y
    getline <"tmp.minmax"
    # Skip Z
    getline <"tmp.minmax"
    # Skip M
    getline <"tmp.minmax"
    # Skip T

    getline <"tmp.minmax"
    xm = $2+0.0
    getline <"tmp.minmax"
    ym = $2+0.0
    getline <"tmp.minmax"
    sc = $2+0.0

    print "#-----"
    print "#      Fault lines"
    print "#-----"
    print "Shape ["
    print "    appearance USE CWHT"
    print "    geometry IndexedLineSet ["
    print "        coord Coordinate ["
    print "            point ["
    i = 0
}

{
    # Scale XY data to fit display range
    # Write in WRL format
    if ((substr($0,1,1) != "#") &&
        (NF == 2)) {
        x = ($1 - xm) * sc
        y = ($2 - ym) * sc
        if ((x >= -10.0) && (x <= 10.0) &&
            (y >= -10.0) && (y <= 10.0)) {
            flag[i] = 0
            i++
            printf "          %20.8f %20.8f 5.0,\n", x, y
        }
    } else {
        if ($0 == "END") {
            flag[i-1] = 1
        }
    }
}

END {
    print "          ]"
    print "        ] #End coord"
    print "      coordIndex ["
    lasti = i
    for (i=0; i<lasti; i++) {
        if (flag[i]) {
            printf "          %d, -1,\n", i
        } else {
            printf "          %d,\n", i
        }
    }
    print "      ]"
    print "    ] #End IndexedLineSet"
    print "  ] #End Shape"

}' $lin >> $wrl
```

Dec 20 1996 15:00

README.961220

Page 12

```
echo "Finished"

=====
run060pntsPlane.sh
=====
echo "pntsPlane..."

pntsPlane lsm060.pln lsm031_utm.xyzm 750.0 lsm035_utm.xyzm

echo "Finished"

=====
runSizes
=====
#!/bin/sh

f=$1

echo "Sizes 0 - 3..."
grep Size0 $f | wc
grep Size1 $f | wc
grep Size2 $f | wc
grep Size3 $f | wc

echo "\nColors 0 - 8..."
grep C00 $f | wc
grep C01 $f | wc
grep C02 $f | wc
grep C03 $f | wc
grep C04 $f | wc
grep C05 $f | wc
grep C06 $f | wc
grep C07 $f | wc
grep C08 $f | wc

==> lsm011.rawdata <==

==> lsm021_geo.xyzm <==
-116.297833 36.712000 -11010.000000 2.300 0
-116.303833 36.713333 -10360.000000 1.500 5
-116.303000 36.731667 -6390.000000 1.900 217
-116.296833 36.709500 -11040.000000 2.700 233
-116.300167 36.712167 -10860.000000 2.200 234
-116.295833 36.715167 -10550.000000 1.600 234
-116.284000 36.710000 -13020.000000 1.900 235
-116.306333 36.712667 -10750.000000 1.700 239
-116.316667 36.724500 -6120.000000 1.600 283
-116.286167 36.716000 -12430.000000 3.000 572

==> lsm031_utm.xyzm <==
562712.87 4062951.67 -11010.000000 2.300 0
562175.91 4063095.63 -10360.000000 1.500 5
562235.51 4065129.92 -6390.000000 1.900 217
562804.22 4062675.01 -11040.000000 2.700 233
562504.27 4062968.67 -10860.000000 2.200 234
562888.91 4063304.29 -10550.000000 1.600 234
563950.01 4062738.96 -13020.000000 1.900 235
561953.16 4063020.13 -10750.000000 1.700 239
561020.83 4064326.11 -6120.000000 1.600 283
563751.51 4063403.08 -12430.000000 3.000 572

==> lsm041.wrl <==
#VRML V2.0 utf8
```

README.961220

Burt Hehn 12/22/96

Dec 20 1996 15:00

README.961220

Page 13

```
# Author:      B. Henderson
# Date:        1996_Dec_10_16:16:40
# Data file:   lsm031_utm.xyzm

# Input data ranges:
#   X  558769   567890   Rng  9121.18
#   Y  4.05125e+06  4.07731e+06   Rng  26051.8
#   Z  -14850    0       Rng  14850

==> lsm060.pln <==
560513.201000 4060844.185500 -8555.551758
565013.201000 4066844.185500 -9872.487305
561513.201000 4065344.185500 -6402.470215

/*-----
**  pntsPlane.c
**
**  12-11-96
**
**  cc pntsPlane.c -o pntsPlane vector.o plane.o
**-----*/

#include <stdio.h>
#include <stdlib.h>
#include "vector.h"
#include "plane.h"

void print_usage(char *);

int main (int argc, char *argv[]) {
    int      dbg = 0;
    int      i, n;
    FILE      *plnFid;
    FILE      *pntFid;
    FILE      *outFid;
    double    inx, iny, inz;
    Point3Df  pnt[3];
    Plane3Df  plane;
    double    dist;
    double    thres;
    char      s[81];

    /* Begin */

    /* Read command line args */
    if (argc != 5) {
        print_usage(argv[0]);
        return 0;
    }

    /* Open files */
    plnFid = fopen (argv[1], "r");
    if (plnFid == NULL) {
        fprintf(stderr, "Could not open plane file %s.\n", argv[1]);
        return 0;
    }
    pntFid = fopen (argv[2], "r");
    if (pntFid == NULL) {
        fprintf(stderr, "Could not open points file %s.\n", argv[2]);
        return 0;
    }
    thres = atof(argv[3]);
    if (thres <= 0.0) {
        fprintf(stderr, "Error, thres_dist must be > 0.0, %lf\n", thres);
        return 0;
    }
    if (dbg) printf("Threshold distance = %f\n", thres);
    outFid = fopen (argv[4], "w");
    if (outFid == NULL) {
```

Dec 20 1996 15:00

README.961220

Page 14

```
        fprintf(stderr, "Could not open output file %s.\n", argv[4]);
        return 0;
    }

    /* Read plane file */
    for (i=0; i<3; i++) {
        fgets(s, 80, plnFid);
        if (dbg) printf("READ %s", s);
        n = sscanf(s, "%lf%lf%lf", &inx, &iny, &inz);
        if (dbg) printf("Read plane file: %lf %lf %lf  %d\n",
            inx, iny, inz, n);
        if (n != 3) {
            fprintf(stderr, "Error reading plane file.\n");
            return 0;
        }
        pointCreate(inx, iny, inz, &pnt[i]);
    }
    fclose(plnFid);
    planeCreate(&pnt[0], &pnt[1], &pnt[2], &plane);
    if (1) {
        printf("Plane = %lf %lf %lf  %lf\n",
            plane.x, plane.y, plane.z, plane.d);
    }

    /* Read input points */
    while (!feof(pntFid)) {
        fgets(s, 80, pntFid);
        if (feof(pntFid)) break;
        n = sscanf(s, "%lf%lf%lf", &inx, &iny, &inz);
        if (dbg) printf("Read input point: %lf %lf %lf  %d\n",
            inx, iny, inz, n);
        pointCreate(inx, iny, inz, &pnt[0]);
        dist = planePointDist(&pnt[0], &plane);
        if (dbg) printf("Dist = %lf\n", dist);
        if (dist <= thres) {
            fprintf(outFid, "%s", s);
        }
    }

    fclose(pntFid);
    fclose(outFid);
} /* End */

void print_usage(char *s) {
    /* Begin */
    fprintf(stderr, "Usage: %s \n", s);
    fprintf(stderr, "      <plane.xyz> \n");
    fprintf(stderr, "      <pnts.xyz> \n");
    fprintf(stderr, "      <threshold_distance> \n");
    fprintf(stderr, "      <out.xyz> \n");
    fprintf(stderr, "\n");
    fprintf(stderr, "Where plane.xyz file format is:\n");
    fprintf(stderr, "      x y z\n");
    fprintf(stderr, "      x y z\n");
    fprintf(stderr, "      x y z\n");
    fprintf(stderr, "\n");
    fprintf(stderr, "Where pnts.xyz file format is:\n");
    fprintf(stderr, "      x y z m t\n");
    fprintf(stderr, "      . . . .\n");
    fprintf(stderr, "      . . . .\n");
    fprintf(stderr, "      . . . .\n");
    fprintf(stderr, "      x y z m t\n");
    fprintf(stderr, "\n");
    fprintf(stderr, "Where threshold_distance is the \n");
    fprintf(stderr, "      maximum distance that a point \n");
    fprintf(stderr, "      can be from the plane and be  \n");
    fprintf(stderr, "      saved to the output file.  \n");
    fprintf(stderr, "\n");
    fprintf(stderr, "Where out.xyz is the output file\n");
}
```

README.961220

Burt Hehn 12/22/96

Dec 20 1996 15:00

README.961220

Page 15

```

} /* End */

=====
LMS041.WRL
=====

#VRML V2.0 utf8

# Author:      B. Henderson
# Date:        1996_Dec_10_16:16:40
# Data file:   lsm031_utm.xyzm

# Input data ranges:
# X 558769 567890 Rng 9121.18
# Y 4.05125e+06 4.07731e+06 Rng 26051.8
# Z -14850 0 Rng 14850
# M 0.6 5.6 Rng 5
# T 0 2819 Rng 2819
# XScale 0.0007677
# ZScale 0.000673401
# Xmin 563330
# Ymin 4.06428e+06
# Zmin -7425

#-----
# WorldInfo
#-----
WorldInfo {
  title "Earthquake Hypocenters"
} #End WorldInfo

#-----
# Viewing
#-----
NavigationInfo {
  type "EXAMINE"
}
Viewpoint {
  position 0 0 30
  description "DefaultView"
}
Viewpoint {
  position 0 -40 0
  orientation 1 0 0 1.50
  description "ViewN"
}
Viewpoint {
  position -40 0 0
  orientation 0 1 0 -1.50
  description "ViewE"
}

#-----
# Background
#-----
Background {
  skyColor [ 0.0 0.0 0.0 ]
} #End Background

#-----
# Define Colors and Sizes
#-----
Transform {
  translation -11 -11 -8
  children {
    Shape {
      appearance DEF CBLK Appearance { material Material {
        diffuseColor 0.0 0.0 0.0
      } }
      geometry Sphere { radius 0.1 } },

```

Dec 20 1996 15:00

README.961220

Page 16

```

Shape {
  appearance DEF CWHT Appearance { material Material {
    diffuseColor 1.0 1.0 1.0
  } }
  geometry Sphere { radius 0.1 } },
Shape {
  appearance DEF CRED Appearance { material Material {
    diffuseColor 1.0 0.0 0.0
  } }
  geometry Sphere { radius 0.1 } },
Shape {
  appearance DEF CGRN Appearance { material Material {
    diffuseColor 0.0 1.0 0.0
  } }
  geometry Sphere { radius 0.1 } },
Shape {
  appearance DEF CBLU Appearance { material Material {
    diffuseColor 0.0 0.0 1.0
  } }
  geometry Sphere { radius 0.1 } },
Shape {
  appearance DEF C08 Appearance { material Material {
    diffuseColor 0.6 0.0 0.4
  } }
  geometry Sphere { radius 0.1 } },
Shape {
  appearance DEF C07 Appearance { material Material {
    diffuseColor 0.1 0.0 0.7
  } }
  geometry Sphere { radius 0.1 } },
Shape {
  appearance DEF C06 Appearance { material Material {
    diffuseColor 0.0 0.4 0.8
  } }
  geometry Sphere { radius 0.1 } },
Shape {
  appearance DEF C05 Appearance { material Material {
    diffuseColor 0.0 0.8 0.8
  } }
  geometry Sphere { radius 0.1 } },
Shape {
  appearance DEF C04 Appearance { material Material {
    diffuseColor 0.2 0.9 0.0
  } }
  geometry Sphere { radius 0.1 } },
Shape {
  appearance DEF C03 Appearance { material Material {
    diffuseColor 0.8 1.0 0.0
  } }
  geometry Sphere { radius 0.1 } },
Shape {
  appearance DEF C02 Appearance { material Material {
    diffuseColor 1.0 0.6 0.0
  } }
  geometry Sphere { radius 0.1 } },
Shape {
  appearance DEF C01 Appearance { material Material {
    diffuseColor 1.0 0.3 0.0
  } }
  geometry Sphere { radius 0.1 } },
Shape {
  appearance DEF C00 Appearance { material Material {
    diffuseColor 1.0 0.1 0.0
  } }
  geometry Sphere { radius 0.1 } },
Shape {
  appearance USE CBLK
  geometry DEF Size0 Sphere { radius 0.10 } },
Shape {
  appearance USE CBLK
  geometry DEF Size1 Sphere { radius 0.20 } },

```

README.961220

Burt Hehn 12/22/96

Dec 20 1996 15:00

README.961220

Page 17

```

Shape {
  appearance USE CBLK
  geometry DEF Size2 Sphere { radius 0.40 } },
Shape {
  appearance USE CBLK
  geometry DEF Size3 Sphere { radius 0.80 } },
} #End Transform

```

```

#-----
# Bounding box
#-----

```

```

Shape {
  geometry IndexedLineSet {
    coord Coordinate {
      point [
        -10.0 -10.0 -5.0, # 0
        10.0 -10.0 -5.0, # 1
        -10.0 10.0 -5.0, # 2
        10.0 10.0 -5.0, # 3
        -10.0 -10.0 5.0, # 4
        10.0 -10.0 5.0, # 5
        -10.0 10.0 5.0, # 6
        10.0 10.0 5.0, # 7
        #----- E -----
        11.7 -10.5 -5.0, # 8
        11.0 -10.5 -5.0, # 9
        11.0 -9.5 -5.0, #10
        11.7 -9.5 -5.0, #11
        11.0 -10.0 -5.0, #12
        11.7 -10.0 -5.0, #13
        #----- N -----
        -10.7 11.0 -5.0, #14
        -10.7 12.0 -5.0, #15
        -10.0 11.0 -5.0, #16
        -10.0 12.0 -5.0, #17
      ]
    }
  } #End Coordinate
  coordIndex [
    # X-axis
    0, 1, -1,
    8, 9, 10, 11, -1,
    12, 13, -1,
    # Y-axis
    0, 2, -1,
    14, 15, 16, 17, -1,
    # Z-axis
    0, 4, -1,
    # Box
    2, 3, 7, 6, 2, -1,
    6, 4, 5, 7, -1,
    3, 1, 5
  ]
  color Color {
    color [
      0.2 0.2 0.2, # 0
      1.0 0.0 0.0, # 1
      0.0 1.0 0.0, # 2
      0.0 0.0 1.0, # 3
    ]
  }
  colorIndex [
    # X-axis
    1, 1, 1,
    1, 1, 1, 1, 1,
    1, 1, 1,
    # Y-axis
    2, 2, 2,
    2, 2, 2, 2, 2,
  ]
}

```

Dec 20 1996 15:00

README.961220

Page 18

```

# Z-axis
3, 3, 3,
# Box
0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0,
0, 0, 0
]
colorPerVertex TRUE
} #End IndexedLineSet
} #End Shape

```

```

#-----
# Animation Nodes
# Touch --> Timer --> PosInterpolator --> Shape
#-----

```

```

#-----
# Touch Sensor
#-----
Transform {
  translation -10.5 -10.5 -5.5
  children [
    Shape {
      appearance Appearance {
        material Material {
          diffuseColor 0.1 0.0 0.1
        }
      }
      geometry Box {
        size 1.0 1.0 1.0
      }
    },
    DEF Touch1 TouchSensor { }
  ]
} #End Transform

```

```

Transform {
  translation 10.5 10.5 -5.5
  children [
    Shape {
      appearance Appearance {
        material Material {
          diffuseColor 0.1 0.0 0.1
        }
      }
      geometry Box {
        size 1.0 1.0 1.0
      }
    },
    DEF Touch2 TouchSensor { }
  ]
} #End Transform

```

```

Transform {
  translation -10.5 10.5 -5.5
  children [
    Shape {
      appearance Appearance {
        material Material {
          diffuseColor 0.1 0.0 0.1
        }
      }
      geometry Box {
        size 1.0 1.0 1.0
      }
    },
    DEF Touch3 TouchSensor { }
  ]
} #End Transform

```

```

Transform {

```

README.961220

Burt Hehr 12/22/96

Dec 20 1996 15:00

README.961220

Page 19

```

translation 10.5 -10.5 -5.5
children [
  Shape {
    appearance Appearance {
      material Material {
        diffuseColor 0.1 0.0 0.1
      }
    }
    geometry Box {
      size 1.0 1.0 1.0
    }
  },
  DEF Touch4 TouchSensor { }
] #End Transform

#-----
# Timer
#-----
DEF Timer1 TimeSensor { cycleInterval 5.0 loop FALSE }
ROUTE Touch1.touchTime TO Timer1.set_startTime
ROUTE Touch2.touchTime TO Timer1.set_startTime
ROUTE Touch3.touchTime TO Timer1.set_startTime
ROUTE Touch4.touchTime TO Timer1.set_startTime

#-----
# Sound
#-----
Sound {
  source DEF Sound1 AudioClip {
    url "sound1.wav"
    description "Sound Effect 1"
    loop FALSE
    startTime 1
    stopTime 0
  },
  maxFront 200
  maxBack 200
  spatialize FALSE
}
ROUTE Touch1.touchTime TO Sound1.set_startTime
ROUTE Touch2.touchTime TO Sound1.set_startTime
ROUTE Touch3.touchTime TO Sound1.set_startTime
ROUTE Touch4.touchTime TO Sound1.set_startTime

#-----
# Hypocenter Nodes
#-----
DEF POS00001 PositionInterpolator {
  key [ 0.0, 0.0000, 0.0001, 1.0 ]
  keyValue [ 0.01 0.01 0.01, 0.01 0.01 0.01,
    1.0 1.0 1.0, 1.0 1.0 1.0 ]
}
ROUTE Timer1.fraction_changed TO POS00001.set_fraction
DEF SHP00001 Transform {
  translation -0.473694 -1.01924 -2.41414
  children [
    Shape {
      appearance USE C03
      geometry USE Size0
    } #End Shape
  ]
} #End Transform
ROUTE POS00001.value_changed TO SHP00001.set_scale

DEF POS00002 PositionInterpolator {
  key [ 0.0, 0.0018, 0.0019, 1.0 ]
  keyValue [ 0.01 0.01 0.01, 0.01 0.01 0.01,
    1.0 1.0 1.0, 1.0 1.0 1.0 ]
}

```

Dec 20 1996 15:00

README.961220

Page 20

```

ROUTE Timer1.fraction_changed TO POS00002.set_fraction
DEF SHP00002 Transform {
  translation -0.885918 -0.908719 -1.97643
  children [
    Shape {
      appearance USE C03
      geometry USE Size0
    } #End Shape
  ]
} #End Transform
ROUTE POS00002.value_changed TO SHP00002.set_scale

#-----
# Fault lines
#-----
Shape {
  appearance USE CWHT
  geometry IndexedLineSet {
    coord Coordinate {
      point [
        -7.41817920      8.18682624 5.0,
        -7.51835520      8.10964224 5.0,
        -7.62769920      7.99962624 5.0,

        -3.09016320      -0.53112576 5.0,
        -2.40438720      0.09921024 5.0,
        -2.34587520      0.00801024 5.0,

        ]
      ] #End coord
    coordIndex [
      0,
      1,
      2,
      15,
      16, -1,
      17,
      18, -1,
      19,
      20,
      21, -1,
      22,

      1623,
      1624, -1,
      1625,
      1626, -1,

      ]
    ] #End IndexedLineSet
  } #End Shape
}

```

Burt Hahn 12/22/96

THIS SCIENTIFIC NOTEBOOK
(# 113) HAS BEEN REVIEWED AND FOUND SATISFACTORY
AND IS TO BE ARCHIVED AS
OF 1/2/97.

A. Laurence McKay

