DEPARTMENT OF THE INTERIOR

U.S. GEOLOGICAL SURVEY

WELLMAP, a FORTRAN computer program for the generation of maps of specified geographic areas with a variety of well symbols available to depict the current status of given oil and gas drilling.

By E.G. Boyce, Jr., and Robert M. Turner

Open-File Report 83-155

1983

CONTENTS

APPENDICES

Introduction

This paper will describe the WELLMAP program and give the information needed to use the program effectively.

WELLMAP was written to enable the user to plot oil and gas well data on a geographic base map as an aid in data evaluation and presentation. Prior to its existence, well information was transferred to a map by tedious and time-consuming manual effort. With industry rapidly expanding and new data coming in constantly, manual methods proved insufficient. The ability to select a geographic "window" and view the amount of drilling which has already taken place is readily achieved through WELLMAP. The WELLMAP program is based in part on the Mappit program for data display (Turner, in preparation). The WELLMAP program employs the subroutine "PICT" (Cukrowicz, 1980) to produce annotations providing some descriptive information as to the name, completion date, and total depth of the plotted hole. The plot may be previewed on a Tektronix screen to verify map results although when satisfactory output is obtained, a plot may be generated. See figures 1, 2 and 3 for sample input and output.

WELLMAP has provided a means for viewing raw data which truly simplifies the tasks of examining and viewing the most current information available. This report instructs the potential user in the use of WELLMAP, from the time data are received until the map is plotted.

Acknowledgements

The well-symbol-plotting routine PICT was written by Ron Cukrowicz of the Cities Services Company. He may be contacted at:

Cities Services Company
Room 215 - Koger Building
Box 300
Tulsa, Oklahoma 74102

tel: (918) 561-3217

```
15.583    36.792    51 Hole-1        78 10035 3
15.167    36.850    06 Hole-2        82  9628 2
14.250    36.700    08 Hole-3        79  8220 1
15.000    37.978    21 Hole-4        76 10202 3
12.039    44.958    21 Hole-5        75  7356 5
11.208    43.789    00 Hole-6        72 10220 4
14.000    42.000    07 Hole-7        68  9871 3
15.708    40.942    53 Hole-8        55  5004 4
16.083    37.933    02 Hole-9        59  7310 3
16.267    40.933    55 Hole-10       73  9929 3
18.258    40.922    01 Hole-11       80 12050 3
17.133    38.917    17 Hole-12       77  8432 3
```

Figure 1.--Sample WELLMAP input.

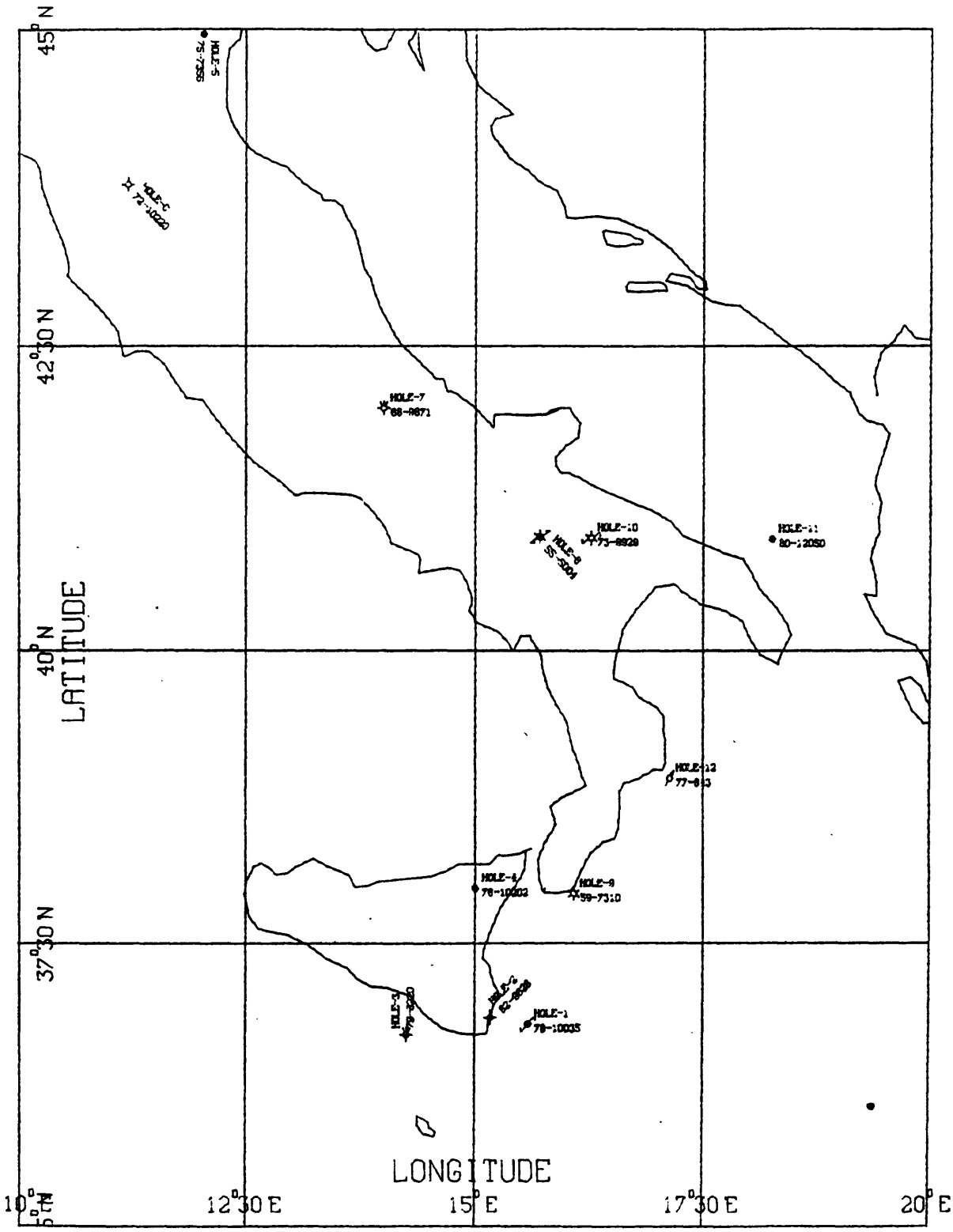

Figure 2.--Sample WELLMAP output.

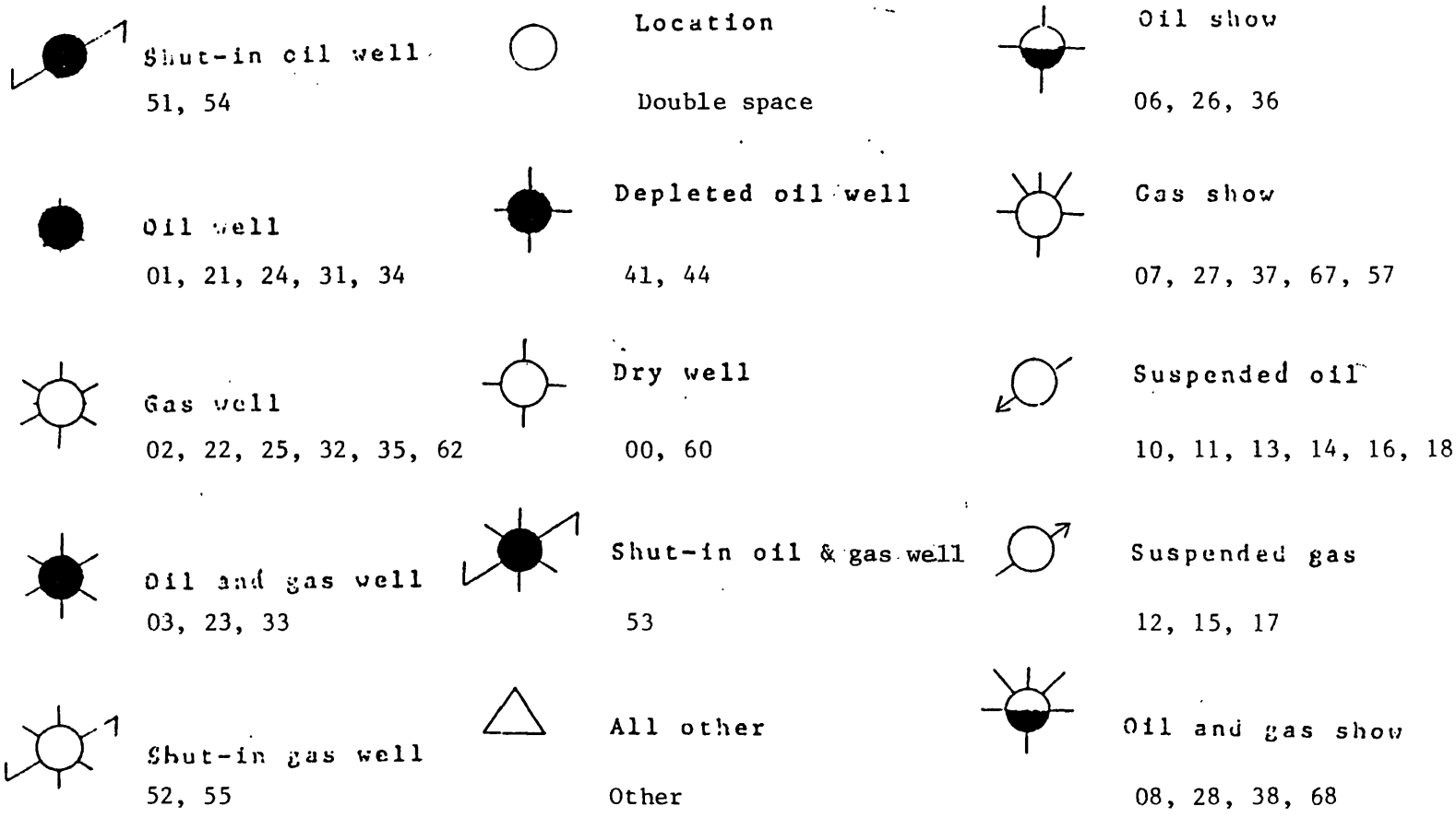

Figure 3.--Well symbols and corresponding status codes.

We would also like to commend the late Nancy A. Wright for her efforts in the initial development of WELLMAP. Her diligence and research was instrumental in the conception of the WELLMAP program.

Computer restrictions

The WELLMAP program was written for the Honeywell Multics system and is designed for interactive use. The program incorporates several system subroutines which render it machine dependent. Refer to Appendix G for a list of these subroutines. It was written to be used in conjunction with either a CALCOMP plotter or a Tektronix screen display and, therefore, one of these must be used in plot production. As a minimum, the user must know the "login" procedure for the Honeywell Multics System and how to link to WELLMAP and the various files required with WELLMAP for execution. Refer to Appendix A for instructions regarding the necessary links and the proper commands to obtain them.

Program limitations

WELLMAP was written for use on the Honeywell Multics computer located in Reston, Virginia, and utilizes several pieces of software which may not be available elsewhere. These are:

DISSPLA - a proprietary package employed for the purpose of plotting all coastline and political boundaries as well as the map border, title, grid, and axis labels.

TEKTRONIX - The "prevuu" routines make DISSPLA compatible with the Tektronix hardware thereby allowing the user to preview the plot before generating a CALCOMP plot.

CALCOMP - software giving DISSPLA the ability to interface with the CALCOMP plotter.

Input data

Input to WELLMAP is by a Multics segment created through GRASP, the Geologic Retrieval And Synopsis Program (Bowen, 1976). GRASP is a retrieval system which has its own command language and rules of operation providing interactive access to a pre-existing data base. Therefore, the user must either have access to a GRASP data base, have knowledge of how to create one, or already have a data input segment in his/her working directory. Assuming enough information is available, the user may enter data into a segment via a text editor prior to the invocation of the program. The name of this file may be no more than eight characters in length. The input format is as follows:

DATA NAME	POSITION	DESCRIPTION	UNITS
X	1 - 10	X-axis coordinate	longitude degrees
Y	11 - 20	Y-axis coordinate	latitude degrees
IST	29 - 30	well status	(see Figure 3)
WELLNAM	32 - 43	well identification	well_id
COMPDAT	50 - 51	completion year	two-digit (19 assumed)
TOTDEPTH	52 - 57	total depth	feet or meters
IROTANOT	59	annotation rotation factor	1-5

The type of well symbol plotted is determined by the two-digit code contained in the well status field. The well identification, year of completion, and the well depth are all plotted adjacent to the well symbol and their position may be rotated anywhere from 90 degrees above the point to 90 degrees below it. This is accomplished by inserting a number from 1 through 5 in the annotation rotation factor field to indicate a 90, 45, 0, -45, or -90 degree rotation respectively.

Coordinates of west longitude and south latitude are defined as being negative as a means of distinguishing between hemispheres. These longitudes and latitudes are the "X" and "Y" values listed above.

Design and usage

After linking to WELLMAP as described in Appendix A, enter "WELLMAP" to initiate execution. The WELLMAP program will then request the transmission rate of the terminal currently being used. You will then be asked whether the output is to be directed to a Tektronix screen, a CALCOMP plotter or a Multics segment entitled "pfile". If a CALCOMP plot is desired, the tape should have already been taken to the computer room; otherwise a message to this effect is issued and the execution is terminated.

Once these preliminaries are established, you are prompted for a projection type. The various projections available are listed in Appendix F along with their appropriate explanations. For some of the more distorted projections, a calculation to cause the projection center to be at the map center may be required. This feature is accounted for and is accomplished by answering "yes" when so prompted. The default is the Greenwich-equator intersection. Next you will be requested to supply a title for the map. The response may contain as many as 48 characters and will appear in upper case letters on the map, regardless of the form in which it was entered. You will then be asked whether a state map with county boundaries is desired. If so, enter the postal abbreviation (first letter upper case, second letter lower case) of the state to be mapped. An additional prompt is issued to enable you to select a geographic window of your own choice. In this case, you must enter the range of longitude, from the minimum to the maximum, as well as the interval number of degrees between grid lines. This same information is then required for the range of latitude and grid

intervals. You are then prompted to specify which, if any, coastline and/or political files are to be employed in the generation of your map. Refer to Appendix C for a list of the available files. The program will then inquire as to whether you have a file of points which you would like to plot and, if so, will subsequently prompt for the name of the file. This is the file discussed earlier on page five. A simple base map is produced when no point file is specified. At this point all necessary data is present and the program gives the option of restarting should there have been errors in entering information.

Additional well information

GRASPWELL is a FORTRAN program to generate a report listing details about the wells in a specified geographic window or of a given well-id. Normally, this report would accompany a map produced by the WELLMAP program and there would be a one-for-one correlation between symbols on the map and information in the report.

GRASPWELL prompts the user for either a geographical window or for specific well-id's which then become the selection criteria for a subsequent GRASP retrieval. The well data extracted from the data base is the most frequently requested information, although there are in fact more details available within the system.

The program creates an absentee job generating two output segments entitled "graspfl1" and "graspfl2". These are to be examined together, as there is a line-by-line correspondence between the listings. Output is generated using the GRASP list command and will appear in tabular format.

A copy of the program and the absentee job it produces may be found in Appendix E.

Appendix A

- Links -

Enter "link" followed by:

>udd>ORA>library>WELLMAP

>udd>ORA>library>assoc

>udd>ORA>library>closer

>udd>ORA>library>statwndo

>udd>ORA>library>grasp

>udd>ORA>library>pict

>udd>ORA>library>Ak

through

>udd>ORA>library>Wy

- Search Rules -

Enter "asr" (assign search rules) followed by:

>mr82_fortran -be >system_library_unbundled

Appendix B

- Sample Terminal Session -

```
[WELLMAP]
Enter terminal speed ('9600' '1200' '300' etc.):[300]
choose either a calcomp plot 'cal'
or a tektronix screen display 'tek':[cal]
choose a plot tape with a 6 digit name:['_____']
or an output file named:['pfile'ADG623]
Is the tape upstairs...and have you asked the operator to locate it?:[yes]
Enter desired projection:[merca]
Do you want Projection Center at the Map Center?:[no]
Enter appropriate title:[borneo wells]
Do you want a STATE MAP with COUNTY BOUNDARIES ?:[no]
Enter longitude min, max,and step size:[112,116,2]
Enter latitude min, max, and step size:[2,6,2]
Do you wish a coastline file?:[yes]
Enter coastline file desired:[coastlines]
Do you wish a Political File?:[yes]
Enter political file desired:[political]
Do you have a Point File to plot? ('yes' or 'no'):[yes]
Enter name of file with grasp-list coordinates:[bornpts]
Do you wish to START OVER AGAIN? ('yes' or 'no'):[no]
```

Appendix C

- Coastline and Political Files -

Coastline Files: 'MAPDTA'

'HERSHEY'

'AFRICA'

'ANTARCTIC'

'ASIA'

'AUSTRALASIA'

'EUROPE'

'NORTH AMERICA'

'SOUTH AMERICA'

'COASTLINES' (of the world)

Political Files: 'PAFRICA'

'PASIA'

'PAUSTRALASIA'

'PNORTH AMERICA'

'PSOUTH AMERICA'

'POLITICAL' (of the world)

'USTATESx' (x = 0 thru 9,

0 = best resolution)

All files are described in the DISSPLA manual.

Appendix D

Program Listing

```
c
c     A basic mapping program using DISSPLA to
c     generate geographic map plots of data files. Output is to TEKTRONIX SCREEN or as
c     a CALCOMP plot tape or plot file on Multics.
c
c     OPTIONS:
c               Plot
c                    -CALCOMP
c                              Plottape
c                              Multics plot file
c                    -TEKTRONIX
c                               screen display
c
c               Map
c                    -Projection (12 different, + user defined )
c                          -Projection center at map center, if desired.
c
c                    -State map with county boundaries
c                          -Geographic window, select or default.
c
c                    -Geographic window and Lat/Long Grid.
c
c                    -Geographic Boundaries.
c
c                    -Political Boundaries.
c
c              Data
c
c
c                    -Point File
c
c
c
c
c
c                   Longitude_Latitude_Status_Well-ID_Completion-Date_Total-Depth_Rotation-Code
c     INPUT FORMAT: 123456.7891123456.789_______A1_A1B2C1D1E1F1_____1212123456_1
c                      (2f10.0,8x,a2,1x,a12,6x,i2,i6,1x,i1)
      external ioa_$nnl(descriptors),setup_tektronix_tcs(descriptors)
      external asr(descriptors),dsr(descriptors)
      external close_file(descriptors),project(descriptors)
      external setup_calcomp(descriptors)
      external xposn(descriptors),yposn(descriptors)
      character*20 turner,cfile,pfile
      character*8 fildegs
      integer titl(12)
      integer icode
      integer*2 picode(46)
      integer irotanot
      integer compdat, totdepth
```

```
      real degrot
      character ist*2, istat*2(46)
      integer xint, yint
      character wellnam*12
      dimension ititl(48)
      dimension x2(1500),y2(1500)
      dimension state(50),xxmin(50),xxmax(50),yymin(50),yymax(50),xo(300),yo(300)
      data iblnk,idol/" ","$"/
      data istat/"51","54","21","24","31","34","22","25","32","35","23","33",
     &     "52","55","41","44","00","60","53","26","36","27","37","11",
     &     "14","13","16","18","12","15","17","28","38","68","66","67","  ",
     &     "07","57","06","02","08","01","62","10","03"/
          data picode/ 1, 1, 3, 3, 3, 3, 5, 5, 5, 5, 8,
     &      8,14,14,23,23,25,25,37,58,58,59,59,
     &     67,67,67,67,67,68,68,68,91,91,91,58,59,19,
     &     59,59,58, 5,91, 3, 5,67, 8/
      character*3 restart
      character*3 resp
      character*3 skip
      character*3 skip1
      character*3 skip2
      character*3 skip3
      character*3 dsplq
      character*6 calcq
      character*3 tapemt
c     request for state boundary file ************
      character*3 maptyp
      character*2 mapnam
      character*8 fname
      character*8 statwndo
      character*3 window
      character*2 state
      character*3 filechk
      character*3 file2chk
      character*3 pause
101   call ioa_$nnl(" ^/Enter terminal speed ('9600' '1200' '300' etc.):")
      read(5,115)ispeed
      ispeed=ispeed/10
      call ioa_$nnl(" ^/choose either a calcomp plot 'cal' ")
      call ioa_$nnl(" ^/or a tektronix screen display 'tek':")
200   read(5,115)dsplq
      if(dsplq.eq."cal")go to 210
      if(dsplq.eq."tek")go to 220
      call ioa_$nnl(" ^/enter only 'tek'  or  'cal':")
      go to 200
210   call ioa_$nnl(" ^/choose a plot tape with a 6 digit name'_____'")
      call ioa_$nnl(" ^/or an output file named: 'pfile':")
      read(5,115)calcq
      if(calcq.eq."pfile")go to 240
      call ioa_$nnl(" ^/Is the tape upstairs...and have you asked the operator to locate it?:")
      read(5,115)tapemt
      if(tapemt.eq."yes")go to 230
      call ioa_$nnl(" ^/Go tend to it,  and restart the program")
      go to 300
```

```
230 call setup_calcomp("-un","16","-tp","-nm",calcq)
    go to 245
240 call setup_calcomp("-un","16","-fl","-nm","pfile")
245 call asr(">iml>disspla")
    call calcmp(16)
    call inch30
    call page(30.,30.)
    call nobrdr
    go to 5
220 call setup_tektronix_tcs
  4 call asr(">iml>disspla")
    call nhance(ispeed)
  5 call ioa_$nnl(" Λ/Enter desired projection:")
    read(5,100)turner
    call nochek
    call grace(0.0)
    call ioa_$nnl(" Λ/Do you want Projection Center at the Map Center?:")
    read(5,115)skip3
100 format(a20)
    skip="yes"
    skip1="yes"
    call ioa_$nnl(" Λ/Enter appropriate title:")
    read(5,110)ititl
110 format(48a1)
    l=48
    do 112 i=1,48
    if(ititl(1).ne.iblnk) go to 113
112 l=l-1
113 l=l+1
    ititl(1)=idol
    encode(titl,110)ititl
c
c    -------*** SELECT STATE MAP WITH COUNTY BOUNDARIES ***--
c

111 call ioa_$nnl(" Λ/Do you want a STATE MAP with COUNTY BOUNDARIES ?:")
    read(5,115)maptyp
    if(maptyp.ne."yes")go to 114
    print,"A single state at a time can be mapped; Select the state using the "
    print,"TWO LETTER postal abbreviation, with ONLY the First Letter CAPITALIZED"
  8 call ioa_$nnl(" Λ/Enter Abbreviation of the State desired ,(2 char.) ,(i.e. Az = Arizona):")
    read(5,115)mapnam
c
c    -------*** SELECT GEOGRAPHIC WINDOW FROM STATE FILE ***--
c
    open(15,form="formatted", file="statwndo")
    do 9 i=1,48
  6 read(15,7,end=99)state(i),xxmin(i),xxmax(i),yymin(i),yymax(i)
  7 format(a2,2f4.0,2f3.0)
    if(mapnam.eq.state(i))go to 10
  9 continue
    close(15)
    go to 99
```

```
 10 continue
    close(15)
    print,"State= ",state(i)," window= ",xxmin(i)," ",xxmax(i)," ",yymin(i)," ",yymax(i)
    call ioa_$nnl(" A/Do you wish to select your own Geographic Window?:")
    read(5,115)window
    if(window.eq."yes")go to 114
    skip="no"
    skip1="no"
    xstp=2
    ystp=2
c
c     --------*** Assign Default Window ***--
c
    xmin=xxmin(i)
    xmax=xxmax(i)
    ymin=yymin(i)
    ymax=yymax(i)
    go to 65
 99 print ,"ABBREVIATION ERROR"
    print ,"Use POSTAL ABBREVIATION"
    go to 8
114 call ioa_$nnl(" A/Enter longitude min, max, and step size:")
    read(5,115)xmin,xmax,xstp
115 format(v)
    call ioa_$nnl(" A/Enter latitude min, max, and step size:")
    read(5,115)ymin,ymax,ystp
 50 call ioa_$nnl(" A/Do you wish a coastline file?:")
    read(5,115)skip
    if(skip.eq."no")go to 55
    call ioa_$nnl(" A/Enter coastline file desired:")
    read(5,100)cfile
 55 call ioa_$nnl(" A/Do you wish a Political File?:")
    read(5,115)skip1
    if(skip1.eq."no")go to 65
 60 call ioa_$nnl(" A/Enter political file desired:")
    read(5,100)pfile
 65 call ioa_$nnl(" A/Do you have a Point File to plot? ('yes' or 'no'):")
    read(5,115)skip2
    if(skip2.eq."no")go to 600
600 if(skip2.eq."no")go to 280
    call ioa_$nnl(" A/Enter name of file with grasp-list coordinates:")
    read(5,120)fildegs
207 call ioa_$nnl(" A/Do You Wish To START OVER AGAIN? ('yes' or 'no'):")
    read(5,115)restart
    if(restart.eq."yes")go to 101
120 format(a8)
280 if(dsplq.eq."cal")go to 285
    call projct(turner)
    call physor(0.,0.)
    call title(titl,-100,"longitude",9,"latitude",8,8.,6.0)
    goto 284
285 call projct(turner)
```

```
      call physor(0.,0.)
      call title(titl,-100,"longitude",-9,"latitude",-8,30.,20.)
c
c     -------*** STATE BOUNDARY MAP***--
c
284   continue
287   call nochek
      call grace(0.0)
      if(skip3.ne."yes")go to 286
      xpole=(xmin+xmax)/2.
      ypole=(ymin+ymax)/2.
      call mapole(xpole,ypole)
286   call mapgr(xmin,xstp,xmax,ymin,ystp,ymax)
      if(skip.eq."no")go to 70
      call mapfil(cfile)
 70   if(skipl.eq."no")go to 73
      call mapfil(pfile)
 73   call grid(1,1)
      if(maptyp.ne."yes")go to 18
      fname=mapnam
      open(11,form="formatted",file=fname)
 74   n=300
      read(11,end=888)n,(xo(ii),ii=1,n),(yo(ii),ii=1,n)
      call curve(xo,yo,n,0)
      go to 74
888   close(11)
18    open(15,form="formatted",file=filedegs)
19    read(15,105,end=999)x,y,ist,wellnam,compdat,totdepth,irotanot
105   format(2f10.0,8x,a2,1x,a12,6x,i2,i6,1x,i1)
      if(xmin.lt.-180..and.x.gt.xmax)x=x-360.
      if(xmax.gt.180..and.x.lt.xmin)x=x+360.
      xcoord=xposn(x,y)
      ycoord=yposn(x,y)
      angel=0.0
      ht=.09
      do 550 j=1,46
      if(ist.ne.istat(j))go to 550
      icode=picode(j)
      go to 560
550   continue
      icode=41
560   call pict(xcoord,ycoord,ht,icode,angel)
      call height(.045)
      go to (561,562,563,564,565,566),irotanot
561   degrot=90.
      go to 566
562   degrot=45.
      go to 566
563   degrot=0.
      go to 566
564   degrot=-45.
      go to 566
```

```
565 degrot=-90.
    go to 566
566 continue
    call angle(degrot)
    if (degrot.eq.90.) call mshift(-.05, .05)
    if (degrot.eq.45.) call mshift(.025, .09)
    if (degrot.eq.0.) call mshift(.05, .05)
    if (degrot.eq.-45.) call mshift(.09, -.025)
    if (degrot.eq.-90.) call mshift(.05, -.05)
    call rlmess(wellnam,12,x,y)
    call reset("mshift")
    if (degrot.eq.90.) call mshift(.05, .05)
    if (degrot.eq.45.) call mshift(.09, .03)
    if (degrot.eq.0.) call mshift(.05, -.05)
    if (degrot.eq.-45.) call mshift(.03, -.09)
    if (degrot.eq.-90.) call mshift(-.05, -.05)
    if (compdat.ne.0) go to 567
    call rlmess("  ",2,x,y)
    go to 568
567 continue
    call rlint(compdat,x,y)
568 continue
    if (totdepth.eq.0.) go to 569
    if (degrot.eq.90.) call mshift(.05, .13)
    if (degrot.eq.45.) call mshift(.15, .09)
    if (degrot.eq.0.) call mshift(.13, -.05)
    if (degrot.eq.-45.) call mshift(.09, -.15)
    if (degrot.eq.-90.) call mshift(-.05, -.13)
    call rlint(totdepth,x,y)
569 continue
    call reset("mshift")
    call reset("angle")
    go to 19
999 continue
    close(15)
290 call endpl(0)
    if(dsplq.eq."cal")go to 260
    call ioa_$nnl(" ^/Do you want another plot?")
    read(5,115)resp
    if(resp.eq."yes")go to 5
    if(dsplq.eq."tek")go to 250
260 call setup_calcomp("-reset")
    go to 251
250 call setup_tektronix_tcs("-reset")
251 call dsr(">iml>dissplа")
300 call close_file("-all")
    call ioa_$nnl(" ^/NORMAL TERMINATION")
    go to 431
430 continue
    close(11)
    close(12)
    go to 290
431 stop
    end
```

Appendix E

GRASPWELL

```
%global ansi77;
      external ioa_$nnl(descriptors)
      character wellid*12(100)
      character alpha*26
      character*3 answer
      character*3 window
      character*1 hemins, hemiew
      character string*126, aorb*4, concat*1
      real xmin, xmax, ymin, ymax
      data alpha/"abcdefghijklmnopqrstuvwxyz"/
      data string/126*" "/
      data aorb/".or."/
  100 format("long be ",f10.3,",",f10.3)
  150 format("lat be ",f10.3,",",f10.3)
  200 format(v)
  300 format("wellid eq ",a12)
  400 format(a12)
      call ioa_("You are given the option of specifying a geographic window")
      call ioa_("or individual well id's which will become the basis for a GRASP")
      call ioa_("retrieval to provide details on those wells specified.")
      call ioa_$nnl(" ^/Would you like information about all wells")
      call ioa_$nnl(" ^/within a specific geographical window ?:")
    1 read(5,200)window
      if (window.ne."yes") go to 10
      call ioa_$nnl(" ^/Enter the range of longitude and hemisphere(E or W) (min, max, hemis.):")
      read(5,200)xmin,xmax,hemiew
      call ioa_$nnl(" ^/Enter the range of latitude and hemisphere(N or S) (min, max, hemis.):")
      read(5,200)ymin,ymax,hemins
      go to 30
   10 if (window.ne."no") go to 20
      i = 1
      j = 0
   11 call ioa_$nnl(" ^/Enter desired 'well-id':")
      read(5,400)wellid(i)
      concat = alpha(i:i)
      i = i + 1
      if (j.eq.0) go to 14
      string(j+1:j+4) = aorb
      string(j+5:j+5) =  concat
      j = j + 5
   12 call ioa_$nnl(" ^/Would you like to enter another well-id ?:")
   13 read(5,200)answer
      if (answer.eq."yes") go to 11
      if (answer.eq."no") go to 30
      call ioa_$nnl(" ^/Enter only 'yes' or 'no':")
    go to 12
14  j = 1
    string(j:j)=concat
```

```
   go to 12
20 call ioa_$nnl(" /Enter only 'yes' or 'no':")
   go to 1
30 write(15,200)"&attach"
   write(15,200)"cwd wellmap"
   write(15,200)"grasp"
   write(15,200)"defi"
   write(15,200)"read lldefi"
   write(15,200)" "
   write(15,200)"cond"
   if (window.eq."no") go to 40
   write(15,100)xmin,xmax
   write(15,200)"longew cs ",hemiew
   write(15,150)ymin,ymax
   write(15,200)"latns cs ",hemins
   write(15,200)" "
   write(15,200)"logi"
   write(15,200)"a.and.b.and.c.and.d"
   write(15,200)"sear"
   write(15,200)" "
   write(15,200)"file15"
   go to 60
40 i = i - 1
   do 50 n=1,i
   write(15,300)wellid(n)
50 continue
   write(15,200)" "
   write(15,200)"logi"
   write(15,200)string
   write(15,200)"sear"
   write(15,200)" "
   write(15,200)"file15"
60 write(15,200)"list"
   write(15,200)"file15"
   write(15,200)" "
   write(15,200)"u"
   write(15,200)"yes"
   write(15,200)"graspfll"
   write(15,200)"no"
   write(15,200)"uniqid 1 15"
   write(15,200)"country 17 4"
   write(15,200)"situati 23 1"
   write(15,200)"wellid 26 12"
   write(15,200)"wellnam 40 16"
   write(15,200)"wellno 57 3 (i3)"
   write(15,200)"longew 62 1"
   write(15,200)"longdeg 65 3 (i3)"
   write(15,200)"longmin 69 2 (i2)"
   write(15,200)"longsec  72 3 (i3)"
   write(15,200)"latns 77 1"
   write(15,200)"latdeg 80 2 (i2)"
   write(15,200)"latmin 83 2 (i2)"
   write(15,200)"latsec 86 3 (i3)"
```

```
write(15,200)"geoprov 91 10"
write(15,200)"dpthunt 101 1"
write(15,200)"td 105 5 (i5)"
write(15,200)"formatn 111 12"
write(15,200)"agelu 123 4"
write(15,200)"spuddat 128 6 (i6)"
write(15,200)" "
write(15,200)"list"
write(15,200)"file15"
write(15,200)" "
write(15,200)"u"
write(15,200)"yes"
write(15,200)"graspfl2"
write(15,200)"no"
write(15,200)"yes"
write(15,200)"compdat 1 6 (i6)"
write(15,200)"agelu 9 4"
write(15,200)"agell 13 4"
write(15,200)"fmtnl 18 12"
write(15,200)"bopdl 30 5 (i5)"
write(15,200)"mcfdl 36 6 (i6)"
write(15,200)"age2u 43 4"
write(15,200)"age2l 47 4"
write(15,200)"fmtn2 51 12"
write(15,200)"bopd2 63 5 (i5)"
write(15,200)"mcfd2 69 6 (i6)"
write(15,200)"age3u 75 4"
write(15,200)"age3l 79 4"
write(15,200)"fmtn3 84 12"
write(15,200)"bopd3 96 5 (i5)"
write(15,200)"mcfd3 102 6 (i6)"
write(15,200)" "
write(15,200)"quit"
write(15,200)"yes"
write(15,200)"ted"
write(15,200)"bl r graspfll"
```

```
                    write(15,200)"1i"
                    write(15,200)"                          S                                   E          "
                    write(15,200)"              COUNTRY I                 WELL          WELL /   LONGITUDED
tcE    GEOL    DPTH. TOT.            AGE  SPUD"
                    write(15,200)"   UNIQUE ID    CODE  T.    WELL ID      NAME           NO. W DEG.MIN.SECN
tc.SEC PROV    UNIT  DPTH.FORMATION   CODE DATE"
                    write(15,200)" "
                    write(15,200)"tf"
                    write(15,200)"w graspf11"
                    write(15,200)"b2 r graspf12"
                    write(15,200)"1i"
                    write(15,200)"        AGE  TEST  ONE       BARRELS THOUS AGE  TEST  TWO      BARRELS THOUT
tc THREE     BARRELS THOUS"
                    write(15,200)"COMPL.                          PER   CU.FT                        PER   CU.FT
                    write(15,200)"DATE    UPPER LOWER   FRMTN    DAY    DAY  UPPER LOWER   FRMTN  DAY    DAYE
tcR   FRMTN
                    write(15,200)" "
                    write(15,200)"tf"
                    write(15,200)"w graspf12"
                    write(15,200)"q"
                    write(15,200)"logout"
                    99 stop
                    end
```

Output from GRASPWELL

```
&attach
cwd wellmap
grasp
defi
read lldefi

cond
long be    118.000,    120.000
longew cs E
lat be     10.000,     12.000
latns cs N

logi
a.and.b.and.c.and.d
sear

filel5
list
filel5

u
yes
graspfll
no
uniqid 1 15
country 17 4
situati 23 1

wellid 26 12
wellnam 40 16
wellno 57 3 (i3)
longew 62 1
longdeg 65 3 (i3)
longmin 69 2 (i2)
longsec  72 3 (i3)
latns 77 1
latdeg 80 2 (i2)
latmin 83 2 (i2)
latsec 86 3 (i3)
geoprov 91 10
dpthunt 101 1
td 105 5 (i5)
formatn 111 12
agelu 123 4
spuddat 128 6 (i6)

list
filel5
u
```

```
yes
graspf12
no
yes
compdat 1 6 (i6)
agelu 9 4
agell 13 4
fmtnl 18 12
bopdl 30 5 (i5)
mcfdl 36 6 (i6)
age2u 43 4
age2l 47 4
fmtn2 51 12
bopd2 63 5 (i5)
mcfd2 69 6 (i6)
age3u 75 4
age3l 79 4
fmtn3 84 12
bopd3 96 5 (i5)
mcfd3 102 6 (i6)

quit
yes
ted
bl r graspf11
li
                    S                                    E              N
             COUNTRY I                  WELL          WELL /   LONGITUDE  /  LATITUDE L
DPTH. TOT.               AGE  SPUD
   UNIQUE ID    CODE  T.    WELL ID      NAME           NO. W DEG.MIN.SEC  S  DEG.MIN.SV
UNIT  DPTH.FORMATION   CODE DATE
logout
†fgraspf11
b2 r graspf12
li
        AGE  TEST  ONE       BARRELS THOUS AGE  TEST  TWO     BARRELS THOUS AGE  TEST  TE
BARRELS THOUS
COMPL.
DATE    UPPER LOWER   FRMTN    DAY    DAY  UPPER LOWER   FRMTN  DAY    DAY  UPPER LOWER D
                                                                                  †cAY    DAY
†f
w graspf12
q
logout
```

Appendix F

- Available Projections -

Cylindrical Projections ***

a) Cylindrical Equidistant (CYLIN) - does not project data but rather displays coordinates as they are. It checks limits for validity and is useful if limits are calculated and there is potential error in specification.

b) Mercator (MERCA) - At all points the scale is the same in all directions but does not express the variance of the separation of lines of latitude and longitude thereby giving the illusion of enlarged areas toward the poles.

c) Exact Cylindrical Equidistant (EXACT) - Similar to (A) except that coordinates are fully corrected for ellipticity of the earth and are scaled at a constant factor based on latitude and longitude at the map center.

d) Corrected Mercator (CORRE) - Similar to (B) except that whereas longitude lines are still separated by a constant factor, the local latitude scale is corrected at all points for ellipticity of the globe.

Elliptical Projections ***

a) Mollweide (MOLLW) an equal area map with parallel lines of constant latitude while the meridians (longitude) appear as ellipses equally spaced at the equator. Distortion is great near the poles.

b) Aitoff (Hammers) Projection (AITOF) - an equal area projection but is not as badly distorted near the poles as the Mollweide. Latitude lines are neither parallel nor straight, equally spaced only at the map center while meridians are equally spaced at the equator.

c) Sanson (Flamsteed) Sinusoidal (SANSO) - an equal area projection with straight, parallel and equally spaced lines of constant latitude while longitude lines are equally spaced at the equator. Distortion is severe near the edges and poles.

d) Simple Elliptical (ELLIP) - resembles Mollweide very closely but is not equal area and has the feature that latitude and longitude obey a simple equation it is thus possible to relate coordinates from the projected data with accuracy without a fine mesh grid.

Conical Projections ***

a) Bi-parallel Conformal Conic (CONFO) - a conformal projection using two reference parallels for which the meridians are straight lines, intersecting at either the north or south pole when extrapolated and parallels appear as segments of concentric circles whose centers are the poles. The separation of parallels and meridians preserves the local scale.

b) Bi-parallel Equal Area Conic (Alberts Equal Area Conic) (ALBER) - Similar to (A) except that the spacing of meridians, radii and circle segments is such as to preserve local area on the globe. A desireable feature is that although local scale is not preserved, the angle is, making it useful for maps of limited area.

c) True Polyconic Projection on an Infinitesimal Graticule (POLYC) - a conformal projection which is an extension of the simple conic projection, however it is more accurate due to the use of a series of tangent cones as opposed to a single cone.

Azimuthal Projections ***

a) Gnomic Projection (GNOMO) - a projection which is neither conformal nor equal-area with heavy distortion near map corners making useful area limited to area near the map center. All straight lines on the map represent the shortest path between two points on the globe surface.

b) Orthographic Projection (ORTHO) - a true perspective view of the globe which is neither conformal nor equal area. The view is that which would be seen by an astronaut as the projection pole lies at infinity.

c) Stereographic (STERE) - a conformal projection with projection pole on the surface diametrically opposite the map pole. Due to its conformal property it is used for maps of large areas (entire hemispheres).

d) Azimuthal Equi-Distant (AZIMU) - a non-conformal non-equal area projection for which lengths on the map plane correspond to the lengths on the sphere surface. It is often used for polar projections in which the map pole coincides with the north or south pole.

e) Azimuthal (Lambert) Equal-Area (LAMBE) - Similar to (D) except the projection is equal-area but scales are not the same in all directions. It is often used for large maps as the bearings taken from the map poles are true.

Appendix G

- Subroutines Employed-

A. setup_calcomp and setup_tektronix_tcs

Used to prepare the program to interface with standard CALCOMP and Tektronix-supplied software.

B. ioa_$nnl

Is an entry point to the ioa_subroutine and is used to format the data string entered at the terminal and write the resulting string to the user output switch.

C. add_search_rules(asr) and delete_search_rules(dsr)

Used to modify the current "search rules" established by the users start_up.ec or those acquired by default as well as to reset them upon termination.

D. close_file

Closes specified Fortran files opened by the program.

Bibliography

Bowen, R.W., Botbol, J.M, 1976, GRASP: The Geologic Retrieval and Synopsis Program, a portable data retrieval system: U.S. Geological Survey Professional Paper 966.

Cukrowicz, Ron, 1980, PICT: a subprogram of the GEO/PLOT system for plotting oil and gas well symbols: Cities Services Company internal report, May.

Turner, Robert M., Mappit: a user-oriented program for display of point and line data on geographic base maps: U.S.Geological Survey Open-File Report. (in preparation).