Appendix A: Shell Script for TERRAIN

```
# QSUB -r TERRAIN                       # request name
# QSUB -q prem                          # job queue class
# QSUB -eo                              # stdout and stderr together
# QSUB -lM 8Mw                          # maximum memory
# QSUB -lT 1000                         # time limit
# QSUB                                  # no more qsub commands
#
ja
set echo
#
#
#               ********************************************
#               ******   terrain batch C shell        ******
#               ********************************************
#
#
cd $TMPDIR
#
# define the batch output name
#
    batchname terrain.out
#
#
# this should be the user's case or experiment
# set the TERRAIN MSS outputs pathname and the retention period
# (for example TEST/EXP1/...).  Note that there is no / in front of the
# pathname.
#
    set ExpName = TEST/EXP1
#
    set RetPd   = 367
#
# give the host user and machine names to remote copy the terrain plots
# from shavano to users' local machine.
# where username is the users login name on their local machine,
# domain.host is the local machine's address.
# note : if the local machine is a VAX, please refered to appendix D of the
# TERRAIN documentation for more information on using the ftp
# instead of the rcp command.
#
    set Host    = username@host.domain:/usr/tmp/username
#
# if T, color filled terrain and landuse plots
#
    set FillCo  = F
#   set FillCo  = T
#
# if Compile = YES, compile terrain program
# note : terrain program dynamically allocates memory. Therefore,
#        if you don't have your own mods to the terrain program, there
#        is no need to compile the terrain program
#
```

```
    set Compile = NO
#   set Compile = YES
#
#  if TerPlt = F, create only the map background
#
    set TerPlt = T
#   set TerPlt = F
#
#  TERRAIN MSS output file pathname
#
    set OutTerr = $ExpName
#
#
#    ------ TERRAIN NAMELISTS  ---------------------------
#
if ( -e mif ) rm mif
cat > mif << EOF
 &MAPBG
 PHIC  =   37.0,     ; CENTRAL LATITUDE (minus for southern hemesphere)
 XLONC =  -95.0,     ; CENTRAL LONGITUDE (minus for western hemesphere)
 IEXP  =    .T.,     ; .T. EXTENDED COARSE DOMAIN, .F. NOT EXTENDED.
 AEXP  =   360.,     ; APPROX EXPANSION(KM)
 IPROJ = 'LAMCON',   ; MAP PROJECTION
 ;IPROJ = 'POLSTR',   ; MAP PROJECTION
 ;IPROJ = 'MERCAT',   ; MAP PROJECTION
 & ;-------------------------------------------------------------------
 &DOMAINS
 ;
 MAXNES = 2,           ; max no of domains
 ;
 NESTIX=   25, 34,  1,  1,  1,     ; DOMAIN  IX
 NESTJX=   28, 37,  1,  1,  1,     ; DOMAIN  JX
 DIS   =  90.,30., 1., 1., 1.,     ; GRID SIZE(KM)
 MTHRD =    1,  1,  1,  1,  1,     ; MOTHER DOMAIN ID
 NESTI =    1,  8,  1,  1,  1,     ; LOWER LEFT I OF NEST
 NESTJ =    1,  9,  1,  1,  1,     ; LOWER LEFT J OF NEST
 RID   =  1.5,1.5,1.5, 1.5,2.3,    ; RAD OF INFLUENCE IN GRID UNIT
 NTYPE =   2 ,  3,  4,  4,  4,     ; TERRAIN AND LANDUSE RESOLUTION
 ;  1:  1 deg (~111 km) global terrain and landuse
 ;  2: 30 min ( ~56 km) global terrain and landuse
 ;  3: 10 min ( ~19 km) global terrain and landuse
 ;  4;  5 min (  ~9 km) global ter. and 10 min (~19 km) global landuse
 ;  5; 30 sec ( ~.9 km) ter. over 48 states and 10 min (~19 km) global landuse
 ;
 NSTTYP=   1,   2,  2,  2,   2,    ;  1 -- ONE WAY NEST, 2 -- TWO WAY NEST
 & ;-------------------------------------------------------------------
 &OPTN
 IFTER   = .${TerPlt}.,  ; .T.-- TERRAIN,  .F.-- PLOT DOMAIN MAPS ONLY
 DATASW  = .T.,       ; .T. user specify terrain and landuse resolution (ntype)
 ;                     .F. terrain program choose the data resolution
 IFANAL  = .F.,       ; .T.-- OBJECTIVE ANALYSIS, .F.-- INTERPOLATION
 ISMTHTR =  1 ,       ;  1: 1-2-1 smoother, 2: two pass smoother/desmoother
 IFEZFUG = .T.,       ; .T. USE EZMAP WATER BODY INFO TO FUDGE THE LAND USE
 IFTFUG  = .F.,       ; .T. DON'T DO EZFUDGE WITHIN THE USER-SPECIFIED
```

```
;                       LAT/LON BOXES, need to define namelist fudget
IFFUDG  = .F.,        ; .T. POINT-BY-POINT FUDGING OF LANDUSE,
                      ;  need to define namelist fudge
IPRNTD  = .F.,        ; PRINT OUT LAT. AND LON. ON THE MESH
IPRTHT  = .F.,        ; PRINT OUT TERRAIN HEIGHT ON THE MESH
IPRTLU  = .F.,        ; PRINT OUT LANDUSE ON THE MESH
FIN     = 100.,       ; CONTOUR INTERVAL (meter) FOR TERRAIN HEIGHT PLOT
;--------------------------------------------------------------------------
;If your true latitude is not standard for the map projection you choose
;uncomment and define the TRUELAT1 and TRUELAT2 lines (delete the ;).
;standard (minus for southern hemesphere) :
;          Lambert Conformal  : TRUELAT1=30. ,TRUELAT2=60.,
;          Polar Stereographic: TRUELAT1=60.,
;          Mercater           : TRUELAT1=0.                 )
;TRUELAT1=91.,       ; TRUE LATITUDE 1
;TRUELAT2=91.,       ; TRUE LATITUDE 2, use this if IPROJ='LAMCON'
IFILL   = .$FillCo., ; COLOR-FILL THE TERRAIN HEIGHT CONTOURS AND MAP BKG
;                       (do not change)
& ;---------------------------------------------------------------------
&FUDGE
; USE ONLY IF IFFUDG = .T., POINT-BY-POINT FUDGING OF LANDUSE,
; IFFUG FOR EACH OF THE NESTS: .F. NO FUDGING, .T. FUDGING
IFFUG   = .F.,.F.,    ; FUDGE FLAGS
; NDFUG : THE NUMBER OF FUDGING POINTS FOR EACH OF NESTS
NDFUG   = 0,0,
; LOCATION (I,J) AND LANDUSE VALUES FOR EACH OF THE NESTS
; NOTE: REGARDLESS OF IFFUG AND NDFUG, 200 VALUES MUST BE GIVEN FOR
;    EACH NEST, OR ELSE THE INDEXING WILL GET MESSED UP
IFUG    =   0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
            0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
            0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
            0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
            0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
            0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
            0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
            0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
            0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
            0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
JFUG    =   0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
            0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
            0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
            0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
            0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
            0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
            0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
            0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
            0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
            0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
            0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
LNDFUG  =   0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
            0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
            0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
            0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
            0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
```

```
            0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
            0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
            0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
            0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
            0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
 & ;-------------------------------------------------------------------
 &FUDGET
 ; USE ONLY IF IFTFUG=.T., WHICH MEANS TERRAIN WON'T DO EZFUDGE WITHIN
 ;    THE USER-SPECIFIED LAT/LON BOXES.  THIS OPTION IS USED WHEN THERE
 ;    ARE INLAND BODIES OF WATER THAT ARE DEFINED IN THE LAND USE
 ;    DATA SET BUT NOT IN THE EZMAP DATA SET.  THIS OPTION PREVENTS
 ;    THOSE BODIES OF WATER FROM BEING WIPED OUT BY EZFUDGE
 NFUGBOX   =   2     ;  NUMBER OF SUBDOMAINS IN WHICH TO
                     ;     TURN OFF EZMAP LAND USE FUDGING
 STARTLAT=45.0,44.0,   ;  LATITUDES OF LOWER-LEFT CORNERS OF SUBDOMAINS
 ENDLAT  =46.5,45.0,   ;  LATITUDES OF UPPER-RIGHT CORNERS OF SUBDOMAINS
 STARTLON=-95.0,-79.8,   ;  LONGITUDES OF LOWER-LEFT CORNERS OF SUBDOMAINS
 ENDLON  =-92.6,-78.5,   ;  LONGITUDES OF UPPER-RIGHT CORNERS OF SUBDOMAINS
 & ;-------------------------------------------------------------------
 &EZFUDGE
 ; USE ONLY IF IFEZFUG=.T., WHICH TURNS ON EZMAP WATER BODY FUDGING OF LANDUSE.
 ; USERS: FEEL FREE TO ADD ANY MORE LAKE SURFACE HEIGHTS THAT YOU'LL NEED.
 ; HTPS IS THE HEIGHT IN METERS AND THE INDEX OF HTPS CORRESPONDS TO THE ID
 ; OF THE 'PS' AREA IN THE FILE ezmap_area_ids.
 ;
 HTPS(441) =  -.001   ; Oceans  --  Do NOT change this one
 HTPS(550) =  183.    ; Lake Superior
 HTPS(587) =  177.    ; Lakes Michigan and Huron
 HTPS(618) =  176.    ; Lake St. Clair
 HTPS(613) =  174.    ; Lake Erie
 HTPS(645) =   75.    ; Lake Ontario
 HTPS(480) = 1897.    ; Lake Tahoe
 HTPS(500) = 1281.    ; Great Salt Lake
 & ;-------------------------------------------------------------------
EOF
#
########################################################################
####################                             ######################
####################    END USER MODIFICATION    ######################
####################                             ######################
########################################################################
#
#       this is INTERACTIVE or BATCH
#
if ( $?ENVIRONMENT ) then
   echo "environment variable defined as $ENVIRONMENT"
else
   setenv ENVIRONMENT INTERACTIVE
   echo "environment variable defined as $ENVIRONMENT"
endif
#
   set DomId = (1 2 3 4 5 6 7 8 9 10)
#
#       have the temporary disk as default
```

```
#
   set MesoUser = ~mesouser/MM5V1/Terrain
#
#       set up correct tables for map and contours
#
cp ${MesoUser}/t_map.tbl       map.tbl
cp ${MesoUser}/t_maparea.tbl   maparea.tbl
cp ${MesoUser}/t_con.tbl       con.tbl
cp ${MesoUser}/t_luco.tbl      luco.tbl
if ( $FillCo == T ) then
cp ${MesoUser}/t_mapfi.tbl   mapfi.tbl
cp ${MesoUser}/t_confi.tbl   confi.tbl
endif
#
#
#       get sample raob station file
#
if ( ! -e raobsta ) cp ${MesoUser}/raobsta .
#
#       get ezmap area definitions file
#
if ( ! -e ezids ) then
   cp ${MesoUser}/ezmap_area_ids ezids
endif
#
if ( $Compile == NO ) then
if ( ! -e terrain.exe ) then
msread terrain.exe /MESOUSER/MM5V1/TERRAIN/TERRAIN.EXE.NEW
chmod +x terrain.exe
endif
else
#
if ( ! -e terrain.exe ) then
#
# --------- make a TERRAIN source code  --------------------------
#
#  get the update terrain source code
#
if ( ! -e terrain.s ) cp ${MesoUser}/terrain.s terrain.s
#
if ( -e terrain.mods ) rm terrain.mods
#
if ( -e t_stand.mods ) then
   echo "using local copy of t_stand.mods"
   cp t_stand.mods terrain.mods
else if ( -e ~/t_stand.mods ) then
   echo "using MY copy of t_stand.mods"
   cp ~/t_stand.mods terrain.mods
else if ( -e ${MesoUser}/t_stand.mods ) then
   echo "using standard copy of t_stand.mods"
   cp ${MesoUser}/t_stand.mods terrain.mods
else
   echo "not using any copy of t_stand.mods"
endif
```

```
if ( -e t_my.mods ) then
   echo "using local copy of t_my.mods"
   cat t_my.mods >> terrain.mods
else if ( -e ~/t_my.mods ) then
   echo "using MY copy of t_my.mods"
   cat ~/t_my.mods >> terrain.mods
else
   echo "not using any copy of t_my.mods"
endif
if ( -e t_new.mods ) then
   echo "using local copy of t_new.mods"
   cat t_new.mods >> terrain.mods
else if ( -e ~/t_new.mods ) then
   echo "using MY copy of t_new.mods"
   cat ~/t_new.mods >> terrain.mods
else if ( -e ${MesoUser}/t_new.mods ) then
   echo "using standard copy of t_new.mods"
   cat ${MesoUser}/t_new.mods >> terrain.mods
else
   echo "not using any copy of t_new.mods"
endif
#
# make a new terrain.f
#
nupdate -i terrain.s -n terrain.pl
nupdate -p terrain.pl -f -i terrain.mods -m2 -o sq -c terrain
set toast = $status
if ( $ENVIRONMENT != BATCH ) then
   echo -n "update complete, continue?     (yes)   "
   set ans = "$<"
   if (( $ans == "n" ) || ( $ans == "no" )) then
      exit (1)
   endif
else if (( $ENVIRONMENT == BATCH ) && ( $toast != 0 )) then
   echo "error in the update, stopping"
   exit(1)
endif
#
#get the mapdrv and condrv object file
#
if ( ! -e plots.o ) then
   msread plots.o /MESOUSER/plots/object/plots.o
endif
#
#               compile, load TERRAIN
#
if ( -e terrain.o ) rm terrain.o
cf77 -Wf"-a static" plots.o -L /lib,/usr/lib,/usr/local/lib \
-l ncarg,ncarg_gks,ncarg_loc,ncaro,ncarg_c,X11 \
-o terrain.exe terrain.f
set toast = $status
if ( $ENVIRONMENT != BATCH ) then
   echo -n "compile / load complete, continue?    (yes)   "
   set ans = "$<"
```

```
   if (( $ans == "n" ) || ( $ans == "no" )) then
      exit (1)
   endif
else if (( $ENVIRONMENT == BATCH ) && ( $toast != 0 )) then
   echo "error in the compile, stopping"
   exit(1)
endif
endif
#
endif
#
#  get terrain and landuse data set from MS
#
if (( ! -e terln.tar ) && ( ${TerPlt} == T ) ) then
msread terln.tar /MESOUSER/MM5V1/TERRAIN/ter.lndu.tar
tar -xf terln.tar
endif
ls -l
#
# --------- set up fortran input files for TERRAIN ----------------
#
if ( -e assign.terrain ) rm assign.terrain
setenv FILENV assign.terrain
    assign -a mif                    fort.15
    assign -a raobsta                fort.16
    assign -a ezids                  fort.18
    assign -a landu.60               fort.20
    assign -a ter.60                 fort.21
    assign -a landu.30               fort.22
    assign -a ter.30                 fort.23
    assign -a landu.10               fort.24
    assign -a ter.10                 fort.25
    assign -a ter.05                 fort.27
    assign -a ter.30s                fort.29
#
#       run TERRAIN
#
date
if ( -e acct ) rm acct
terrain.exe >&! terrain.print.out
ja -chls >! acct
cat acct >> terrain.print.out
if ( $ENVIRONMENT == BATCH ) cat terrain.print.out
#
set toast = $status
if (( $ENVIRONMENT == BATCH ) && ( $toast != 0 )) then
   echo "error in the TERRAIN, stopping"
   rm core
   exit(1)
endif
#
#       if interactive, probably do not want to zap and dispose
#
if ( $ENVIRONMENT != BATCH ) then
```

```
   echo -n “test terrain run complete, continue?     (no)    “
   set ans = “$<“
   if (( $ans != “y” ) && ( $ans != “yes” )) then
      exit (0)
   endif
endif
#
# send terrain plots back to the front end machine
#
rcp TER.PLT ${Host}/ter.plt
#
if ( $TerPlt == T ) then
#
#       terrain output files to mss
#
set Numb = 0
set IfAny = ( `ls fort.3*` )
set MAX = ${#IfAny}
if ( $MAX > 0 ) then
   foreach fil ( `ls fort.3*` )
   @ Numb ++
      echo “mswrite -t $RetPd $fil $OutTerr/TERRAIN_DOMAIN${DomId[$Numb]}”
   mswrite -t $RetPd $fil $OutTerr/TERRAIN_DOMAIN${DomId[$Numb]}
   end
endif
#
#  send terrain plots to various plotting device
#
mswrite -t $RetPd TER.PLT $OutTerr/TERRAIN_PLOT
echo “ mswrite -t $RetPd TER.PLT $OutTerr/TERRAIN_PLOT “
#
endif
#
# tar the namelist, mods, source code, executable, and output together
# save the terrain tar file on MSS
#
tar -cvf terrain.tar terrain.f terrain.exe mif terrain.mods terrain.print.out
mswrite -t $RetPd terrain.tar $OutTerr/terrain.tar
echo “ mswrite -t $RetPd terrain.tar $ExpName/terrain.tar “
rm G*
ls -ls
```