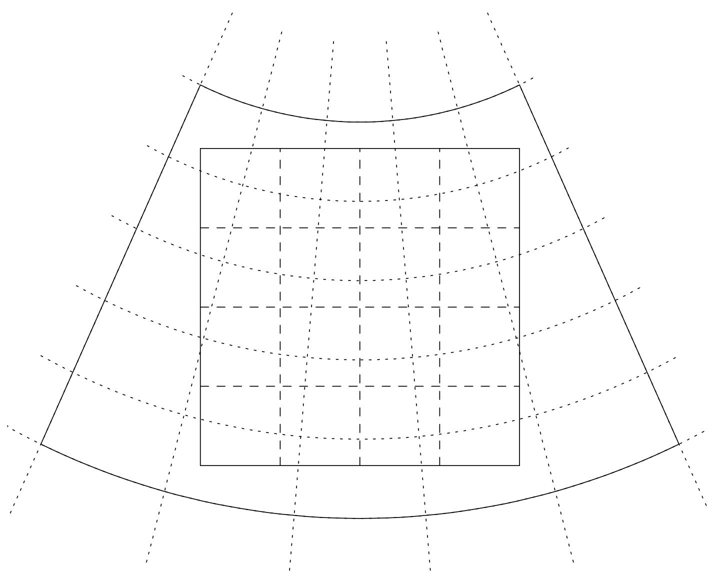
4 TERRAIN

4.1 Purpose

The program that begins any complete forecast simulation is TERRAIN. This program horizontally interpolates (or analyzes) the *latitude-longitude interval* terrain elevation and land use categories onto the chosen mesoscale domains.



The model domain settings (except for moving nests) are constructed in TERRAIN program. Users may use this program to check the correctness of the domain settings first without generating terrain height and land-use files. Once the domains are correctly set, users can then go on to run the program TERRAIN again to produce the terrain height and land-use files, which will be used by REGRID (DATAGRID) and INTERP later. Fine mesh terrain output file can also be used in MM5 (release-2-11 or later).

1. The TERRAIN program is composed of four parts: (1) source data input; (2) interpolation from lat/long source data to mesoscale grid; (3) nest interface adjustment and feed back; and (4) output terrain elevation and land use.

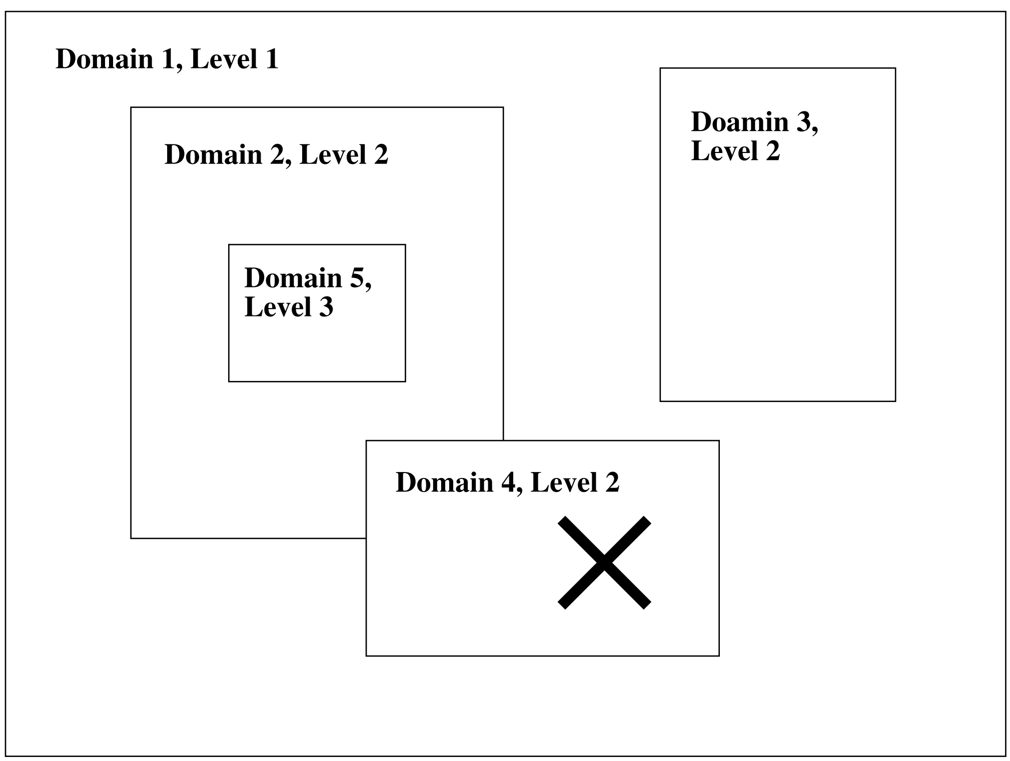
2. For MM5 version 2 release-12, the TERRAIN program can be obtained from *ftp://ftp.ucar.edu/mesouser/MM5V2/Terrain* with the file name *terrain.tar.Z.* This program now is working on multiple architectures: CRAY, SGI, DEC, SUN, HP, and IBM.

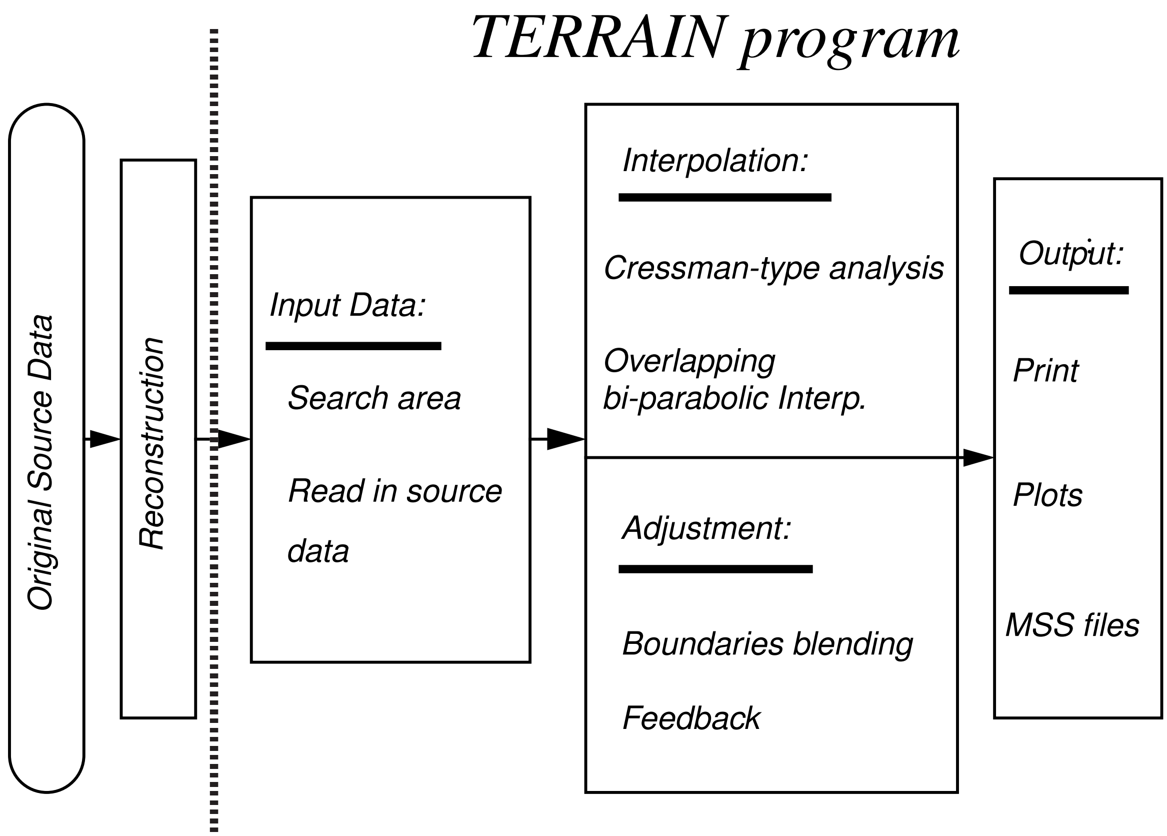
3. There are five resolutions of source elevation data: 1-degree, 30-, 10-, 5-minute and 30-second, available. For the first 4 resolutions, the data coverage is global while for 30-second data, there are two datasets available: (i) global (include 48 states in USA) and (ii) 48-states in USA.

4. There three types of source land-use data available now:

   (i) 13-category, global coverage with the resolution of 1-degree, 30- and 1-minute;
   (ii) 17-category, north-American coverage with the resolution of 1-degree, 30-, 10-, 5-minute and 30second;
   (iii) 25-category, global coverage with the resolution of 1-degree, 30-, 10-, 5-minute and 30-second.

* The terrain height and land-use files can be generated for multiple-nest domains. At present, however, they can not be produced consistently for overlapping nests even though the MM5 model allows the overlapping nests (in most cases with moving nests). For a model simulation with overlapping nests, we suggest that the terrain height and land-use for fine mesh are interpolated from their mother domain's data, which are done in MM5.





4.2 Data Format

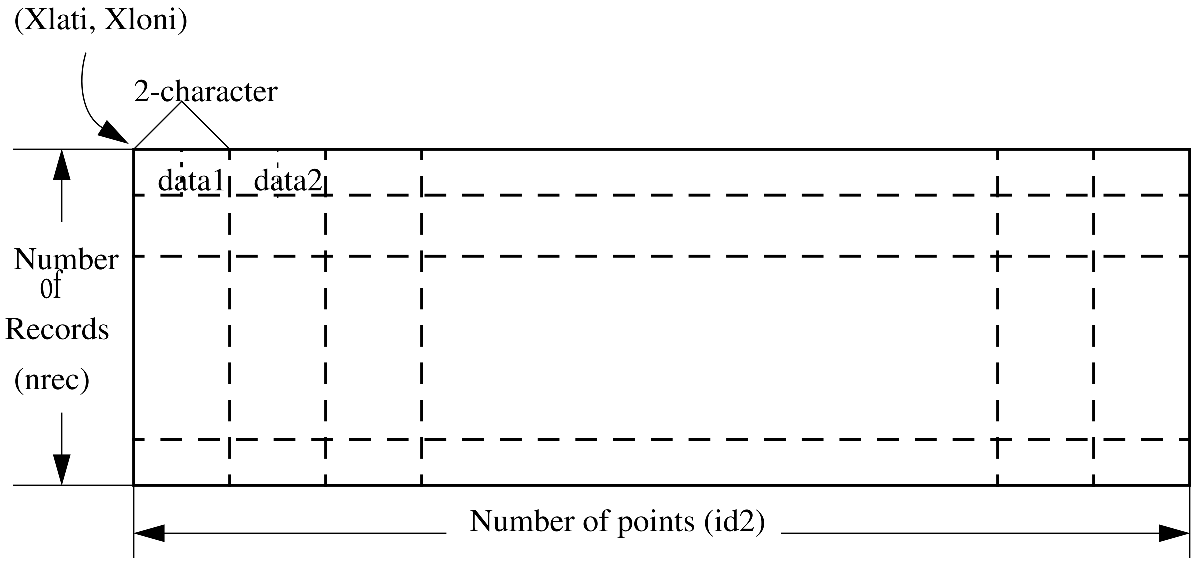
Since the original data come from different sources, they have different format and layout. These data sets are translated to a standard format which are the inputs to TERRAIN program. The data arrangement and format in the reformatted data file are as follows,

- Latitude by latitude from north to south in one latitude, the data points are arranged from west to east.

- We used 2-character array to store the elevation data (the maximum value of terrain $< 2^{15}$ (=32768) m), and 1-character array to store the land-use percentage ($<$ 100%) data with the resolution of 1-degree, 30-, 10-, and 5-minute or the category index ($<$ 25) for 30-second resolution data. This makes more convenient to read in the data in both workstation and CRAY machines.

** Because there are some errors in the 1-degree and 10 min. terrain height data, we used the 30 min. and 5 min. data to re-generate the 1-degree and 10 min. data, respectively (see WWW MM5 home page).

- **Elevation data**:



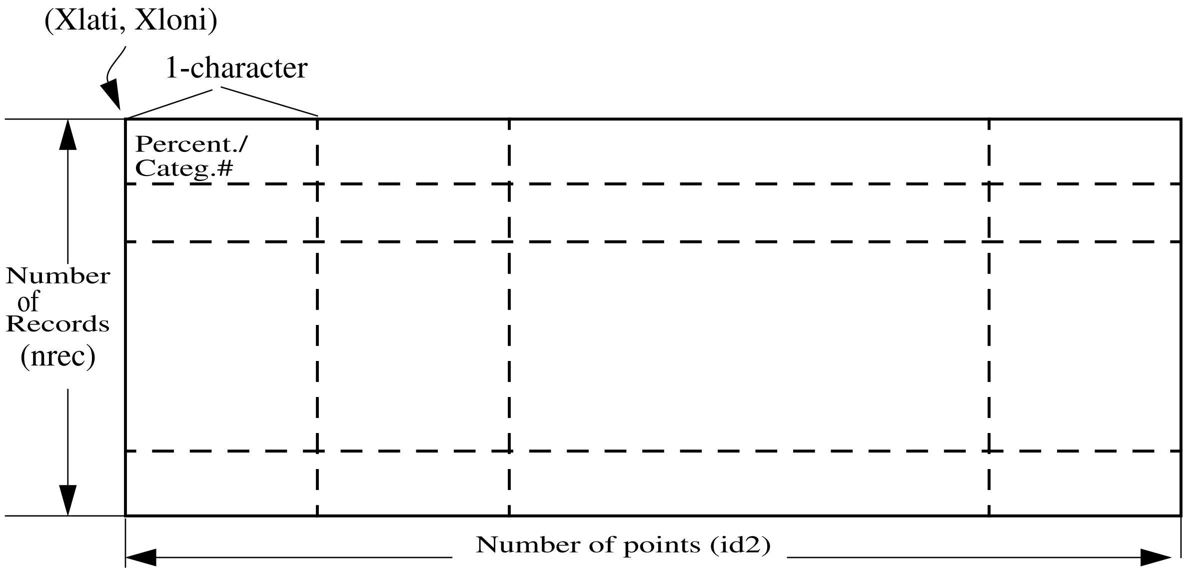
The segment of code for reading in the elevation data:

```
CHARACTER*2 INTER1_chr(id2)
INTEGER*2       INTER1(id2)

. . . . .

do jj= 1,nrec

. . . . . .

read (iunit,err-200) INTER1_chr
do ii = 1,id2
INTER1(ii) = IA(INTER1_chr(ii), 2, -1)
end do

. . . . . .

end do
```

Here the function IA(*chr, len, spv*) converts the *len*-character string *chr* to an integer, and if the integer is less than zero, a value *spv* is assigned to IA.

Because there is no 2-byte integer type in CRAY (always with 8-byte integer), we have to use type of CHARACTE*2 for elevation or CHARACTER*1 for land-use in reading in the data.

- **Land-use data**



Segment of code for reading in the land-use:

for 1-d, 30-, 10-, 5-minute data (id1=N categories) represented by N number of percentages at a point or

for 30-second data (id1 = 1) represented by a category index number at a point

```
CHARACTER*1 LANDIN_chr(id1,id2)
INTEGER*1       LANDIN (id1,id2)

. . . . . .

do jj = 1,nrec

. . . . . .

read (nunit,err=100) LANDIN_chr
do iid1 = 1,id1
do iid2 = 1,id2
LANDIN(iid1,iid2) = ichar(LANDIN_chr(iid1,iid2))
end do
end do

. . . . . .

end do
```

Here ichar is a Fortran intrinsic function to convert a character to an integer.

4.2.1 Dataset Size

•Elevation:

Table 4.1a Terrain Height Data Source*

Resolution	Data source	Coverage	Size
1 deg. (111.0 km)	PSU/NCAR	Global	180x360
30 min. (55.0 km)	PSU/NCAR	Global	360x720
10 min. (18.5 km)	Geophysical Data Center	Global	1080x2160
5 min. (9.25 km)	Geophysical Data Center	Global	2161x4320
30 sec. (0.925 km)**	GTOPO30 by U.S. Geological Survey's EROS Data Center in late 1996	Global (33 tiles: 40^o lon. x 50^o lat. or 60^o lon. x 30^o lat.)	4800x6000 or 7200x3600 for each of tiles
30 sec. (0.925 km)	Defence Mapping Agency	51^o-23^oN,130^o-60^oW	3361x8401

* Except for global 30 sec. data (GTOPO30), the data reconstruction from original source data was completed separately prior to TERRAIN.

** For details of GTOPO30 data, see *http://www.scd.ucar.edu/dss/datasets/ds758.0.html*. The data reconstruction part is included in TERRAIN job deck. The reconstructing procedure contains three steps:

(i) to determine which tiles of data file needed based on the information in namelist;
(ii) to fetch the data from MSS or ftp site tile by tile;
(iii) to reconstruct data in TERRAIN standard input format from the tiled data and provide the necessary information to TERRAIN.

•Land-use

(1) Global 13-category data from PSU/NCAR tape

Table 4.1b PSU/NCAR Land-use Data Source

Resolution	Data source	Coverage	Size
1 deg. (111.0 km)	PSU/NCAR	Global	180x360x13
30 min. (55.0 km)	PSU/NCAR	Global	360x720x13
10 min. (18.5 km)	PSU/NCAR	Global	1080x2160x13

(2) North-American 17-category data used by Simple Biosphere (SiB) model (from USGS)

Table 4.1c SiB Land-use Data Source

Resolution	Data source	Coverage	Size
1 deg. (111.0 km)	Simple Biosphere model	0^o-90^oN, 60^o-180^oW	90x120x17
30 min. (55.0 km)	Simple Biosphere model	0^o-90^oN, 60^o-180^oW	180x240x17
10 min. (18.5 km)	Simple Biosphere model	0^o-90^oN, 60^o-180^oW	540x720x17
5 min. (9.25 km)	Simple Biosphere model	0^o-90^oN, 60^o-180^oW	1080x1440x17
30 sec. (0.925 km)**	Simple Biosphere model	0^o-90^oN, 60^o-180^oW	10800x14400

(3) Global 25-category data from U.S. Geological Survey (USGS)

Table 4.1d USGS Land-use Data Source

Resolution	Data source	Coverage*	Size
1 deg. (111.0 km)	USGS	Global	180x361x25
30 min. (55.0 km)	USGS	Global	360x721x25
10 min. (18.5 km)	USGS	Global	1080x2161x25
5 min. (9.25 km)	USGS	Global	2160x4321x25
30 sec. (0.925 km)**	USGS	Global	21601x43201

* Although we called the coverage “global”, we found that there are some cavities in the data, such as south of 60^oS, east Greenland, and far east Russia.

** For 30-second data, the land-use is represented by only a category index number ranging from 1 to N (=17 for SiB or 25 for USGS) at a point, so the size equals the total number of data points.

4.2.2 Data Information:

If users have their own source data, the data can be translated to the above standard format. In addition, the following information should be provided to the TERRAIN program through data statement in setup.F or in vs_data.incl.

- Initial latitude and longitude
- Data resolution in degree
- The number of data points (longitudes) in a latitude
- Total number of records (latitudes)
- File name to be linked to the adequate Fortran unit number

* If your own data containing the missing data, you must provide the missing-value and modify the interpolation subroutine INTERP or ANAL2 for processing missing-values.

For land-use, there are 3 types of data available now with different number of categories (N = 13, 17 or 25, see Table 4.1b, 4.1c, and 4.1d). At each data point, there are N numbers of percentages for the N categories for 1-degree, 30-, 10-, and 5-minute data while there is only a category index number for 30-second data. For the data with resolution lower than or equal 5-minute, the overlapping parabolic interpolation method (see p. 4-16) is applied to obtain the percentages for each land-use categories at the mesoscale grid. For the 30-second data, we can not apply the interpolation method to the category index number to obtain the percentages for each of categories. The method for 30-second land-use data consists of two steps:

(i) we counted the total number, M, of data points within each of mesoscale grid-boxes, [I, J], and the number, m_i, of data point with Category index i for that grid-box. Then, if $M > 0$, the percentages for each of categories i (i = 1,...,N) at the grid point [I, J] are

$$PL_i = \frac{m_i}{M} \times 100$$

(ii) If $M = 0$ (no data points within a grid-box [I_0, J_0]), which will be happened when the grid distance is smaller than the data resolution (30-second or 0.925-km), we search the nearest data point from that grid-box [I_0, J_0], and assign the 100% to that data point's category and 0% to other categories for the grid [I_0, J_0].

Based on the percentages of N categories at grid points, we then assign a dominant category index number to each of grid points. If the water coverage (Category 7, 15, or 16 for 13-, 17-, 25-category data, respectively) is more than 50% at the point, the category with the maximum percentage (water) will be assigned to that point. If the water coverage is less than 50%, the category with the maximum percentage excluding the water will be assigned to that point.

The input land-use data to TERRAIN are the percentages for the N categories at the latitude/longitude grid with different resolution, and the output from TERRAIN is the category index at the mesoscale grid. In the MM5 model, the category index will be translated to the physical parameters of the surface characteristics, such as albedo, moisture availability, emissivity, roughness length, and thermal inertia, etc., as shown in Table 4.2a for 13-category, 4.2b for 17-category, and 4.2c for 25-category. For the detailed description, user may refer to the file Run/landuse.tbl from mm5v2.tar. In the Northern Hemisphere, "Summer" is defined from April 15 to October 15.

Table 4.2a Description of 13-category (PSU/NCAR) land-use categories and physical parameters for N.H. summer (15 April - 15 October) and winter (15 October - 15 April).

Landuse Integer Identification	Landuse Description	Albedo(%)		Moisture Avail. (%)		Emissivity (% at 9 μ m)		Roughness Length (cm)		Thermal Inertia (cal cm^{-2} k^{-1} s$^{-1/2}$)	
		Sum	Win	Sum	Win	Sum	Win	Sum	Win	Sum	Win
1	Urban land	18	18	5	10	88	88	50	50	0.03	0.03
2	Agriculture	17	23	30	60	92	92	15	5	0.04	0.04
3	Range-grassland	19	23	15	30	92	92	12	10	0.03	0.04
4	Deciduous forest	16	17	30	60	93	93	50	50	0.04	0.05
5	Coniferous forest	12	12	30	60	95	95	50	50	0.04	0.05
6	Mixed forest and wet land	14	14	35	70	95	95	40	40	0.05	0.06
7	Water	8	8	100	100	98	98	.01	.01	0.06	0.06
8	Marsh or wet land	14	14	50	75	95	95	20	20	0.06	0.06
9	Desert	25	25	2	5	85	85	10	10	0.02	0.02
10	Tundra	15	70	50	90	92	92	10	10	0.05	0.05
11	Permanent ice	55	70	95	95	95	95	5	5	0.05	0.05
12	Tropical or sub tropical forest	12	12	50	50	95	95	50	50	0.05	0.05
13	Savannah	20	20	15	15	92	92	15	15	0.03	0.03

Table 4.2b Description of 17-category (SiB) and-use categories and physical parameters for N.H. summer (15 April - 15 October) and winter (15 October - 15 April).

Landuse Integer Identification	Landuse Description	Albedo(%)		Moisture Avail. (%)		Emissivity (% at 9 μ m)		Roughness Length (cm)		Thermal Inertia (cal cm^{-2} k^{-1} s$^{-1/2}$)	
		Sum	Win	Sum	Win	Sum	Win	Sum	Win	Sum	Win
1	Evergrn. Broadlf.	12	12	50	50	95	95	50	50	0.05	0.05
2	Broadlf, Decids.	16	17	30	60	93	93	50	50	0.04	0.05
3	Decids. Evergrn.	14	14	35	70	95	95	40	40	0.05	0.06
4	Evergrn. Needlf.	12	12	30	60	95	95	50	50	0.04	0.05
5	Decids. Needlf.	16	17	30	60	93	93	50	50	0.04	0.05
6	Grnd. Tree Shrb.	20	20	15	15	92	92	15	15	0.03	0.03
7	Ground only	19	23	15	30	92	92	.12	.10	0.03	0.04
8	Broadlf. Shrb.P.G.	19	23	15	30	92	92	12	10	0.03	0.04
9	Broadlf. Shrb.B.S.	19	23	15	30	92	92	12	10	0.03	0.04
10	Grndcvr. DT. Shrb	15	70	50	90	92	92	10	10	0.05	0.05
11	Bare Soil	25	25	2	5	85	85	10	10	0.02	0.02
12	Agricltr. or C3 Grs	17	23	30	60	92	92	15	5	0.04	0.04
13	Perst. Wetland	14	14	50	75	95	95	20	20	0.06	0.06
14	Dry Coarst. Comp.	19	23	15	30	92	92	12	10	0.03	0.04
15	Water	8	8	100	100	98	98	.01	.01	0.06	0.06
16	Ice cap & Glacier	55	70	95	95	95	95	5	5	0.05	0.05
17	No data										

Table 4.2c Description of 25-category (USGS) and-use categories and physical parameters for N.H. summer (15 April - 15 October) and winter (15 October - 15 April).

Landuse Integer Identification	Landuse Description	Albedo(%)		Moisture Avail. (%)		Emissivity (% at 9 μ m)		Roughness Length (cm)		Thermal Inertia (cal cm^{-2} k^{-1} $s^{-1/2}$)	
		Sum	Win	Sum	Win	Sum	Win	Sum	Win	Sum	Win
1	Urban	18	18	10	10	88	88	50	50	0.03	0.03
2	Drylnd Crop. Past.	17	23	30	60	92	92	15	5	0.04	0.04
3	Irrg. Crop. Past.	18	23	50	50	92	92	15	5	0.04	0.04
4	Mix. Dry/Irrg.C.P.	18	23	25	50	92	92	15	5	0.04	0.04
5	Crop./Grs. Mosaic	18	23	25	40	92	92	14	5	0.04	0.04
6	Crop./Wood Mosc	16	20	35	60	93	93	20	20	0.04	0.04
7	Grassland	19	23	15	30	92	92	.12	.10	0.03	0.04
8	Sgrubland	22	25	10	20	88	88	10	10	0.03	0.04
9	Mix Shrb./Grs.	20	24	15	25	90	90	11	10	0.03	0.04
10	Savanna	20	20	15	15	92	92	15	15	0.03	0.03
11	Decids. Broadlf.	16	17	30	60	93	93	50	50	0.04	0.05
12	Decids. Needlf.	14	15	30	60	94	93	50	50	0.04	0.05
13	Evergrn. Braodlf.	12	12	50	50	95	95	50	50	0.05	0.05
14	Evergrn. Needlf..	12	12	30	60	95	95	50	50	0.04	0.05
15	Mixed Forest	13	14	30	60	94	94	50	50	0.04	0.06
16	Water Bodies	8	8	100	100	98	98	.01	.01	0.06	0.06
17	Herb. Wetland	14	14	60	75	95	95	20	20	0.06	0.06
18	Wooden Tundra	14	14	35	70	95	95	40	40	0.05	0.06
19	Bar. Sparese Veg.	25	25	2	5	85	85	10	10	0.02	0.02
20	Herb. Tundra	15	60	50	90	92	92	10	10	0.05	0.05
21	Wooden Tundra	15	50	50	90	93	93	30	30	0.05	0.05
22	Mixed Tundra	15	55	50	90	92	92	15	15	0.05	0.05
23	Bare Grnd. Tundra	25	70	2	95	85	95	.10	5	0.02	0.05
24	Snow or Ice	55	70	95	95	95	95	5	5	0.05	0.05
25	No data										

4.3 Mesoscale domain information

Because the size of input data files, especially for the high resolution data, is rather large it is impossible and unnecessary to store the whole dataset in the core memory. In order to only store a subset of the source data in the memory, we have to determine a *search area* which covers the desired mesoscale domain. Then the data in the search area is read in the core memory. The search area is determined based on mesoscale domain information provided by users through namelist. Using this information, the program calculates the maximum and minimum latitude/longitude for the search area. The formulas to calculate the latitude/longitude (λ, ϕ) from mesoscale grid indices (I, J) and vice versa for different map projections can be found in the documentation "Terrain and Land Use for the Fifth-Generation Penn State/NCAR Mesoscale Modeling System (MM5): Program TERRAIN", page 10-17.

Mesoscale domain setting information:

- Map projection:
  - Lambert conformal
  - Polar stereographic
  - Mercator

- Coarse domain parameters:
  - Central latitude and longitude
  - Expanded domain information
  - Domain size
  - Grid distance

- Nested domain parameters
  - Location of grid point (1,1) in its mother domain
  - Mother domain ID
  - Domain size
  - Grid distance (must have a ratio of 3 for 2-way runs)

The latitudes and longitudes of mesoscale grids should be in the range of

$$-90° \le \varphi \le 90° \tag{4.1}$$

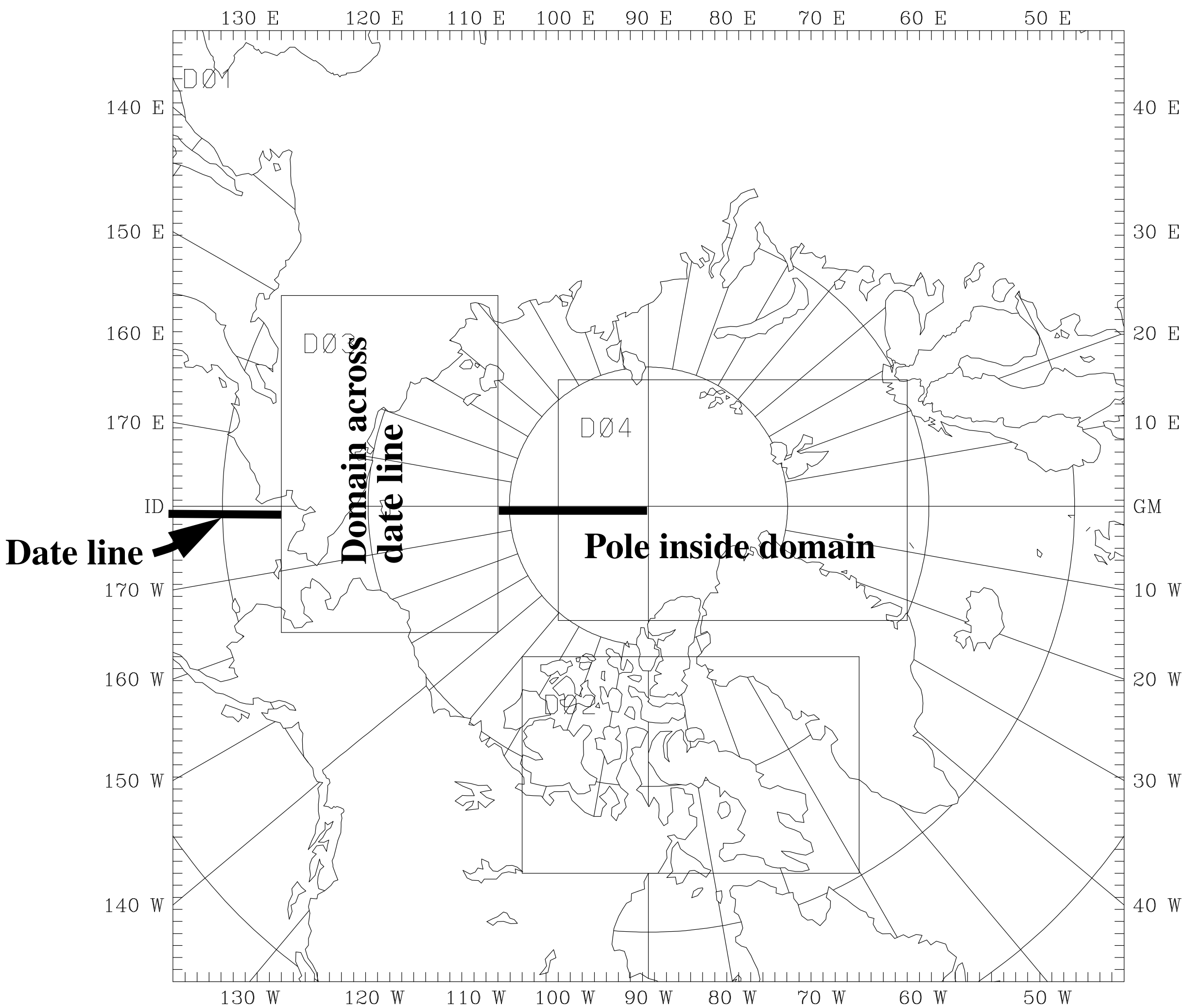
$$-180° \le \lambda \le 180° \tag{4.2}$$

- In general situation, determination of the *search area* is a straightforward job.

- In case of domain across the dateline, the longitudes at some of the points must have a conversion shown below prior the calculation

$$\lambda = \lambda - 360° \tag{4.3}$$

- In case of poles inside the domain, determination of the *search area* is more complicated. User may refer to page 22-25 of the documentation of TERRAIN program (Guo and Chen 1994).

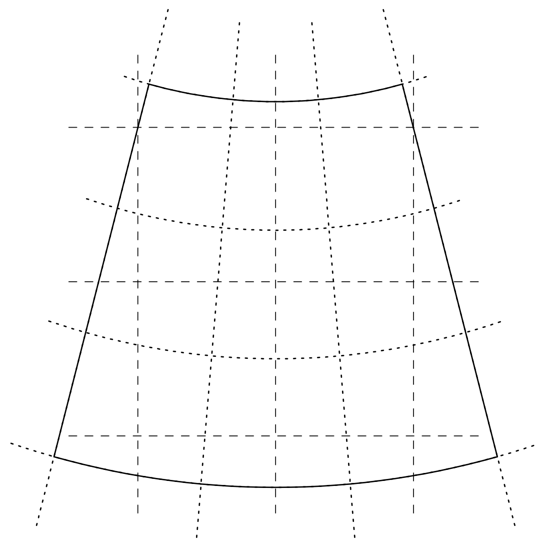


130 E
120 E
110 E
100 E
90 E
80 E
70 E
60 E
50 E
140 E
150 E
160 E
170 E
ID
170 W
160 W
150 W
140 W
40 E
30 E
20 E
10 E
GM
10 W
20 W
30 W
40 W
130 W
120 W
110 W
100 W
90 W
80 W
70 W
60 W
50 W
DØ1
DØ3
DØ4
Domain across date line
Date line
Pole inside domain

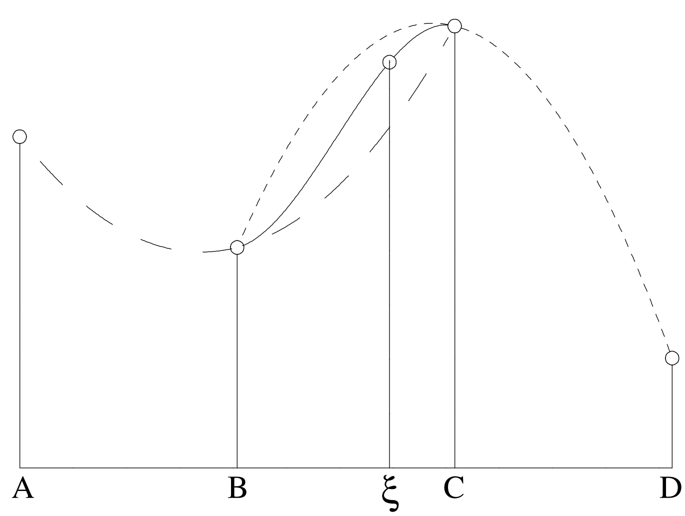
4.4 Interpolation

4.4.1 Overlapping parabolic interpolation

- For both terrain height and land use,
- In spherical (latitude-longitude) coordinate
- 16-point, 2-dimensional parabolic fit (see page 81-82 of Guo and Chen 1994)



16-point, 2-dimension parabolic interpolation



One-dimension overlapping parabolic interpolation

4.4.2 Cressman-type objective analysis

- For terrain height only
- There is no first guess field used
- Only single-pass scan is performed

The weighting function is defined as

$$W_s = \begin{cases} \dfrac{R^2 - r_s^2}{R^2 + r_s^2} & r_s \le R \\ 0 & r_s > R \end{cases}$$

$$r_s^2 = (I - I_{obs})^2 + (J - J_{obs})^2$$

$$HT(I, J) = \frac{\sum_{s=1}^{SN} W_s \times ht_s}{\sum_{s=1}^{SN} W_s}$$

In the TERRAIN program, both of the above methods are available as the interpolation options for users. No systematic comparison with these two methods is performed. They are kept in the current program for historical reason (they are from TERRAIN program of MM4 modeling system). In general, a large radius of influence settings will give a smoother results (with less gradient of the terrain height) while the small radius of influence will cause “no data available” for certain grid boxes if a lower resolution dataset was used. We recommend that user should choose the source dataset with the resolution comparable to the grid distance of the given domain.

4.5 Adjustment

MM5 can be applied to multiple-nest simulation. Each of the nest domains obtain their lateral boundary condition from their mother domain during the integration, and feed the results back to the mother domain in the two-way nested application. After the terrain height and land-use files are produced for each domain, the following procedure,

- reset the nested domain boundary values for both 1-way and 2-way applications, and
- feed back the nest domain information to the mother domain for 2-way application

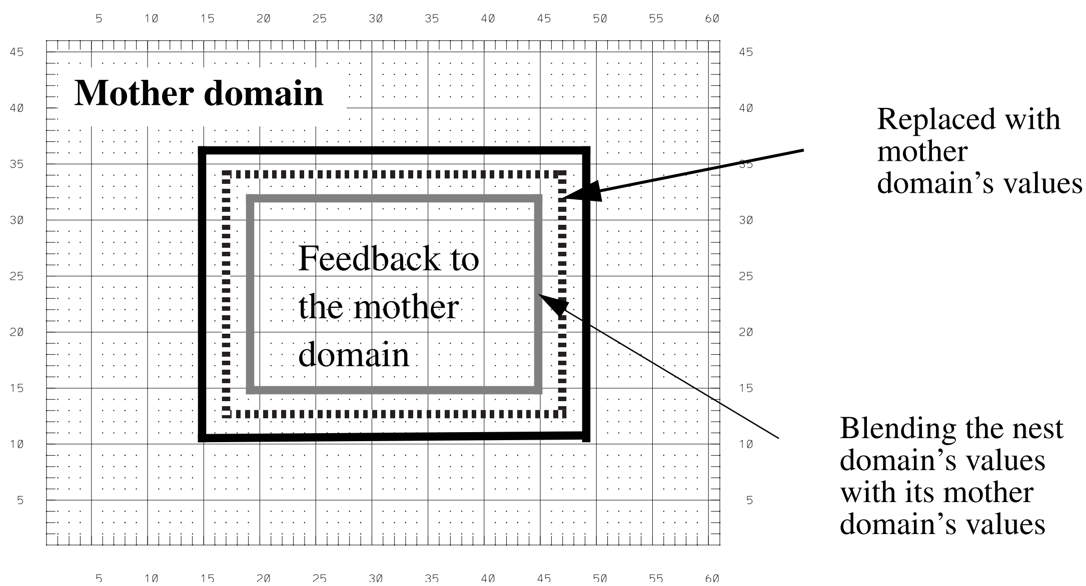
must be completed to make the terrain height and land-use consistent between the domains.

Reset the nested domain boundary values

Interpolate the mother domain's terrain heights to the nest grid by using the monotone interpolation scheme (ratio=3) or bi-parabolic interpolation scheme (ratio≠3)

For rows and columns 1 to 3 (2-way) or 1 to 4 (one-way) along the nest domain boundaries, terrain heights are replaced with mother domain's values

For rows and columns 4 to 6 (2-way) or 5 to 7 (one-way),
Blending the nest domain's values with mother domain's values



User must leave enough space (at least 6 grid-points) between the nest's boundary and its mother domain's boundary so that the interpolation can be completed correctly. If there is not enough space between the boundaries, the program will stop and issue a warning message.

Feedback

The interior values of terrain and land-use in nest-domain are fed back to the mother domains for the two-way nested application. This is done from the highest nest level domain down to the coarsest domain.

4.6 Fudging function

4.6.1 Water body correction

- **Based on land-water mask data files**

When user chose to use new land-use data (NewLandUse = TRUE), the land-water mask files generated based on the vegetation data are used to correct the land-use categories and the height of water bodies. In this case, the EZFUGE function is forced to turn off.

- **Based on the EZMAP from NCAR GRAPHICS**

NCAR graphic package has the capability of identifying the water bodies. The information from a call to ARGTAI can be used to correct the land-use categories and the height of water bodies. When the IFEZFUG = .T., the inland spurious lakes can be eliminated, and the terrain heights are also matched with the coastline better. The heights of the lakes in the US have been defined in the namelist EZFUDGE, user can define more lakes in this namelist. We recommend that users set IFEZFUG = .T. to correct possible errors from the source land-use data if NewLandUse = FALSE. The disadvantage of using IFEZFUG = .T. is that more computer memory and CPU time are needed. Because TERRAIN is not a big job in the MM5 modeling system, the computer requirement usually is not a problem unless you run a job with very large (nearly global) and very high resolution mesh.

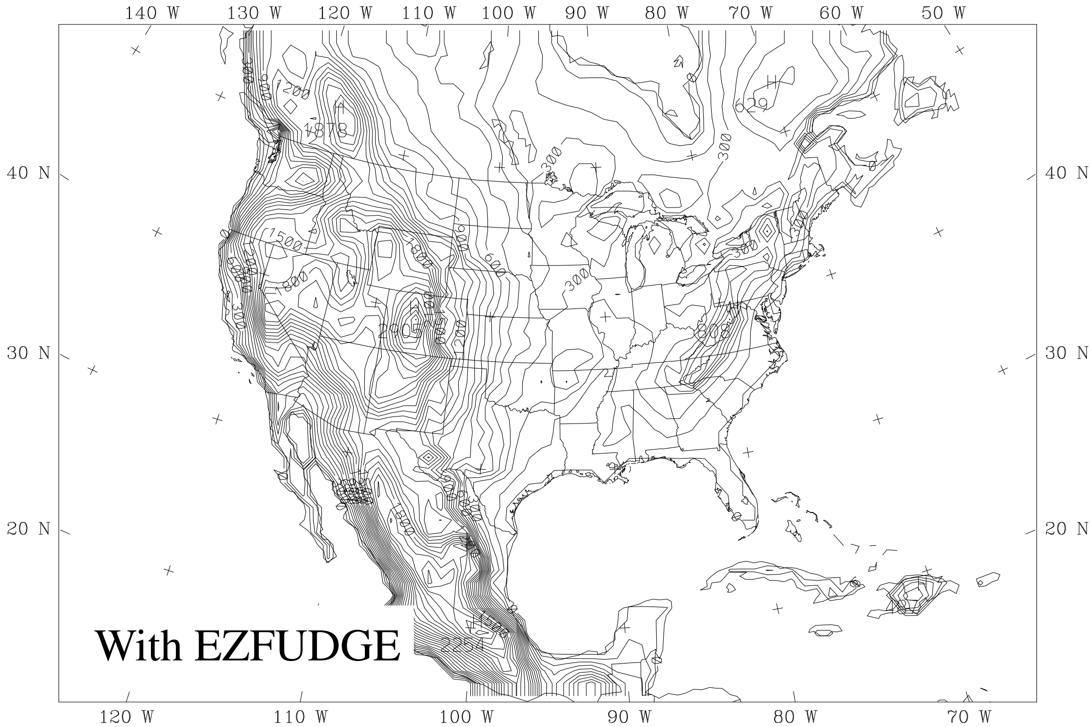
To skip this EZfudging in special areas, turn the switch IFTFUG on and specify the LAT/LON boxes in the namelist FUDGET.

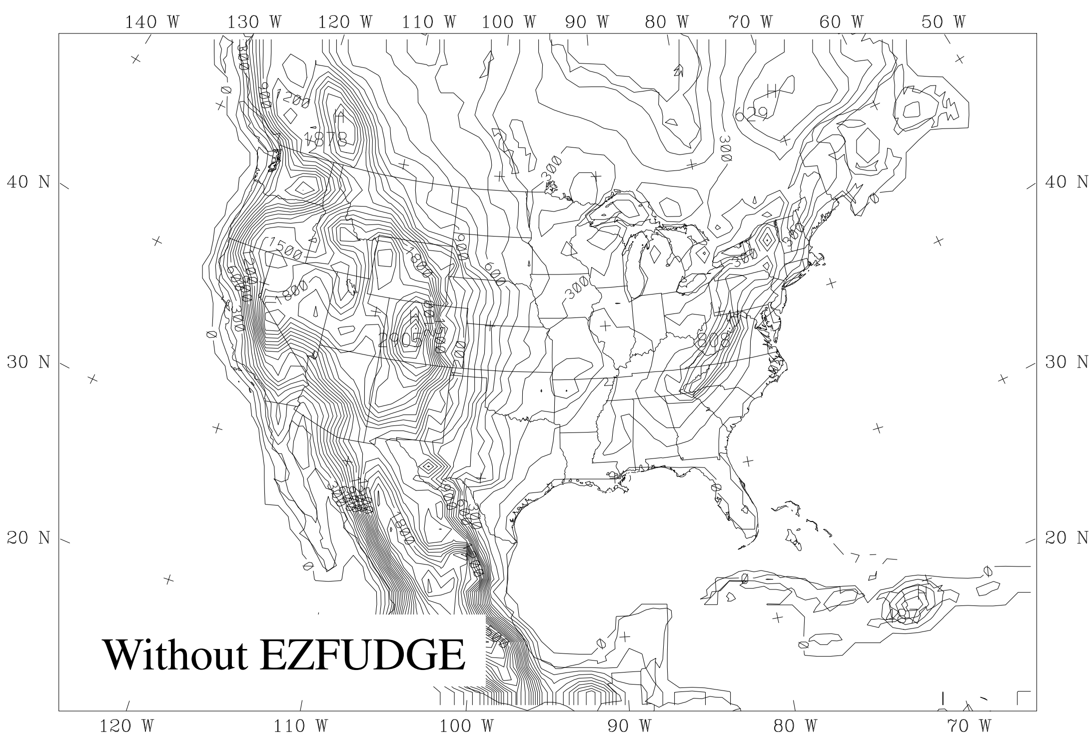
Because of data used in NCAR GRAPHICS are rather old, the outlines for some parts of the world are very rough with no details. Using land-water mask files can make the outlines more realistic over the world. But now there are still some cavities existed in land-water mask files as mentioned before. After fixing the cavities in land-water files, we plan to get rid of the EZFUGE.

4.6.2 Land-use fudge

After the TERRAIN program is finished, a user must look at the results carefully. Sometime the program does not give the correct land-use category at some grid points because of the data error in the original dataset, or sometimes the land-use categories do not meet user's needs in the numerical experiments. TERRAIN gives user another chance to modify the land-use categories at some grid points for each domain. In the namelist, the switch IFFUDG = .T. allows a user to fudge the land-use data point by point. The locations (IFUG, JFUG) and land-use values (LNDFUG) are specified in namelist FUDGE. After the namelist variables IFFUG, NDFUG, IFUG, JFUG, LNDFUG are modified, a user need to submit the TERRAIN job again to get the corrected land-use data output.

Right now the TERRAIN program allows a user to create the terrain height and land-use files for a maximum of 100 domains. For each domain, the maximum number of grid points fudged is 200 (see TERRAIN job deck).


140 W 130 W 120 W 110 W 100 W 90 W 80 W 70 W 60 W 50 W
40 N
30 N
20 N
With EZFUDGE
120 W 110 W 100 W 90 W 80 W 70 W


140 W 130 W 120 W 110 W 100 W 90 W 80 W 70 W 60 W 50 W
40 N
30 N
20 N
Without EZFUDGE
120 W 110 W 100 W 90 W 80 W 70 W

4.7 Script Variables

GLOBAL30S Switch to indicate using global 30-s dataset (T) or not (F).

Data_Dir Set to directory in which the global 30-s data resided, or set to “ftp” to get the data from ftp site when GLOBAL30S = TRUE. Leave it blank if GLOBAL30S = FALSE.

Users Users from MMM/NCAR to set Users = MMM, otherwise leave it blank.

TerPlt Create the map background only (F) or generate the terrain and land-use files (T).

FillCo Switch to indicate the plots are color filled (T) or only black/white (F).

NewLandUse Switch to indicate using new land-use data (T) or not(F).

VegType Switch to indicate which new land use data used: = SiB (Simple Biosphere model), USGS (U.S. Geological Survey) when NewLandUse = TRUE.

4.8 Parameter statement

parame.incl To specify the maximum dimensions (IIMX, JJMX) of any domains (expanded or not expanded).

paramed.incl To specify the maximum dimensions (ITRH, JTRH) of array holding the source data. These may depend upon the shape of mesoscale domain, source data resolution, and map projection, etc.

4.9 Namelist Options

4.9.1 Map Background Options

PHIC Central latitude of the coarse domain in degree.

XLONC Central longitude of the coarse domain in degree.

IEXP Logical flag to use the expended coarse domain (T) or not (F).

AEXP Approximate expansion (km) of the grid on all sides of the coarse domain.

IPROJ Map projection: ‘LAMCON’ for Lambert Conformal, ‘POLSTR’ for Polar Stereographic, and ‘MERCAT’ for Mercator.

4.9.2 Domain Setting Options

MAXNES Maximum number of domains. Now the TERRAIN program allows the maximum number of domains less than or equal to 100.

NESTIX The I(y)-direction dimensions for each of the domains.

NESTJX The J(x)-direction dimensions for each of the domains.

DIS The grid size for each of the domains.

NUMNC The mother domain's ID number for each of the domains. For the coarse domain, always set NUMNC=1.

NESTI The I location in its mother domain of the nest domain's low-left corner --- point (1,1).

NESTJ The J location in its mother domain of the nest domain's low-left corner --- point (1,1).

RID The radius of influence in unit of grid points used only for Cressman type objective analysis (IFANAL=T).

NTYPE The source terrain height and land-use data type for each of the domains:1, one degree; 2, 30 min.; 3, 10min.; 4, 5 min.; 5, 30 sec.

NSTTYP To indicate the nested type: 1, one way nest; 2, two way nest.

4.9.3 Function Options

IFTER Logical flag to indicate to create terrain height and land-use files ($TerPlt=T) or map background only ($TerPlt=F).

DATASW To indicate how to choose the source data files: .T. -- based on the user specification from NTYPE; .F. -- program automatically choose the source data based on the resolution of the each domain.

IFANAL Interpolation method: .T. -- Cressman type objective analysis; .F. -- Overlapping parabolic interpolation.

ISMTHTR To choose smoothing method: 1, 1-2-1 smoother; 2, smoother/desmoother.

IFEZFUG To activate the EZFUDGE function. .T. turns on, .F. is off.

IFFUDG Need to do land-use fudging (T) or not (F).

IFTFUG Need to skip the EZFUDGE function over certain areas (T) or not (F).

IPRINTD Print out the latitude and longitude of the mesoscale grids (T) or not (F).

IPRTHT Print out the terrain height of the mesoscale grids (T) or not (F).

IPRTLU Print out the land-use of the mesoscale grids (T) or not (F).

FIN Contour level (meter) of terrain height plots for each of domains.

TRUELAT1 The first true latitude for the map projection. Default value = 91.0 means the standard values will be used for the projections.

TRUELAT2 The second latitude for the map projection. Default value = 91.0 means the standard value will be used for the projections.

IFILL (=.$FillCo.) Plots are color filled (T) or not (F).

VegeSoil (=.${NewLandUse}.) Switch to indicate creating vegetation and soil files or not in MM5V3. Here for new land use only.

VegOnly (=.${NewLandUse}.) Switch to indicate creating vegetation file only (T) or all the files, vegetation and soil files, etc., requested by land-surface model (F) in MM5V3. Here for new land use only.

G30sD (=.${GLOBAL30S}.) To indicate using global 30-s elevation data (T) or not (F).

4.9.4 Land-use Fudging Options (used when IFFUDG=T)

IFFUG To indicate which domains need to be fudged (T) or not (F).

NDFUG The number of fudge points for each of the domains. The maximum of NDFUG is 200, that means that user can fudge maximum of 200 points for land-use for each of the domains.

IFUG The I location of the fudge points for each of the domains. IFUG is a 2-dimension array IFUG(200,100), the first index is corresponding to points, and the second index corresponding to domains.

JFUG The J location of the fudge points for each of the domains.

LNDFUG The land-use category of the fudge points for each of the domains.

4.9.5 Skip the EZFUDGE over the boxes (used when IFTFUG=T)

Note: The maximum number of boxes is 10. user can use STARTLAT(10),..., to specify the boxes over which no EZFUDGE was done.

STARTLAT The latitudes of the lower-left corner of the area.

ENDLAT The latitudes of the upper-right corner of the area.

STARTLON The longitudes of the lower-left corner of the area.

ENDLON The longitudes of the upper-right corner of the area.

4.10 How to run Terrain

1) Obtain the source code tar file from one of the following places:

Anonymous ftp:
ftp://ftp.ucar.edu/mesouser/MM5V2/Terrain/terrain.tar.gz

On NCAR MSS:
/MESOUSER/MM5V2/TERRAIN/NEW-TERRAIN.TAR.gz

2) gunzip the file, untar it, and edit configure.make file to select appropriate RUNTIME_SYSTEM and compile options. Comment out the ones not appropriate to your machine.

3) Type ‘make terrain.deck’ to create a job deck for your platform.

4) Edit terrain.deck to select script option, set parameter statements, and select namelist options. The Terrain program allows for not exact parameters used in dimensional statements, as long as they are large enough.

5) Type terrain.deck to compile and execute the program. It is usually a good practice to pipe the output to an output file so that if the program fails, you can take a look at the log file. To do so, type: terrain.deck >& terrain.log, for example.

The output files from Terrain program:

TERRAIN_DOMAIN01[,02,03,... if MAXNES > 1]
TER.PLT: plot file if you choose to run with NCAR Graphics option on.

4.11 TERRAIN Didn't Work: What Went Wrong?

When a terrain job completes, always check for 'STOP 99999', which often indicates the program has completed successfully. If the job has failed, check to see if one of the following is a possibility:

- Missing NCAR Graphics environment variable: see if you have included the following line in your .cshrc file on NCAR's Cray:
  setenv NCARG_ROOT /usr/local
  This is needed for plotting maps using NCAR Graphics.

- Program aborted in subroutine SETUP: most likely you didn't provide the map background information correctly. Check the namelist MAPBG and variables TRUELAT1, TRUELAT2.

- The program stopped abnormally, check the terrain.print.out to find the maximum dimensions required. For example, when polar projection specified and the pole inside the domain, the JTRH should be much larger than ITRH, but for other projections, both ITRH and JTRH may be comparable. Also IIMX and JJMX should be the maximum dimensions including the expanded domain.

- When the constant fields (for example, whole domain located over ocean) are generated, the plotting errors will occurred if FillCo = TRUE. Set FillCo = FALSE or reset your domains.

- "The nest 2 is too close to the boundary of the domain 1 ..." and STOP in subroutine TFUDGE: This means there is not enough grid points between domains' boundaries, change the domain settings (e.g. NESTI and NESTJ), and submit the job again.

4.12 TERRAIN Files and Unit Numbers

Table 4.3 List of shell names, fortran unit numbers and their description for TERRAIN.

Shell name	Unit number	Description
mif	fort.15	namelist
raobsta	fort.16	Global raob station list
*.tbl	fort.17	the tables used for plotting
ezids	fort.18	area ID file used by ezmap
landu.60	fort.20	one degree source land-use file
ter.60	fort.21	one degree source terrain file
landu.30	fort.22	30 minutes source land-use file
ter.30	fort.23	30 minutes source terrain file
landu.10	fort.24	10 minutes source land-use file
ter.10	fort.25	10 minutes source terrain file
ter.05	fort.27	5 minutes source terrain file
ter.30s	fort.29	30 seconds source terrain file
TERRAIN_DOMAIN0?	fort.3?	TERRAIN output files for domain ?
lwmask*.60	fort.40	one degree land-water mask file
lwmask*.30	fort.41	30 minutes land-water mask file
lwmask*.10	fort.42	10 minutes land-water mask file
lwmask*.05	fort.43	5 minutes land-water mask file
lwmask*.30s	fort.44	30 seconds land-water mask file
veg-*.60	fort.45	one degree new land use (vegetation) file
veg-*.30	fort.46	30 minutes new land use (vegetation) file
veg-*.10	fort.47	10 minutes new land use (vegetation) file
veg-*.05	fort.48	5 minutes new land use (vegetation) file
veg-*.30s	fort.49	30 seconds new land use (vegetation) file
new_30sdata_info	fort.97	global 30 sec. elevation data information
new_30sdata	fort.98	global 30 sec. elevation data input to TERRAIN

4.13 TERRAIN tar File

The terrain.tar file contains the following files and directory

CHANGES	Description of changes to Terrain program
Makefile	Makefile to create Terrain executable
README	General information about the Terrain directory
Templates/	Job deck directory
con.tbl	Table file for plots
confi.tbl	Table file for plots
configure.make	Rules for the Makefile
configure.make.CRAY	Rules for the Makefile for CRAY
dem_convert.csh	Shell script for converting 30-s tiled-data to sequential file
ezids	File for NCAR Graphics geographic area identifier
luco.tbl	Table file for plots
luv1.tbl	Table file for plots
luv2.tbl	Table file for plots
map.tbl	Table file for plots
maparea.tbl	Table file for plots
mapfi.tbl	Table file for plots
raobsta	Radiosonde locations for plots
src/	Terrain source code

4.14 Terrain Miscellanea

- Input elevation and land-use source data files are available from *anonymous ftp.ucar.edu*:
  under directory: */mesouser/Data/TERRAIN*
- TERRAIN program is available from *anonymous ftp.ucar.edu*:
  /mesouser/MM5V2/Terrain/terrain.tar.Z
- To get the job deck for different architecture, just type '*make terrain.deck*' after modifying the file: *configure.make*.
- The global 30 seconds elevation data are available from *anonymous edcftp.cr.usgs.gov*:
  under directory: */pub/data/gtopo30/global*

4.15 terrain.deck

```
#!/bin/csh
# terrain.csh
#
set echo
#
unalias rm cd ls -l
#
# SET THE FOLLOWING SCRIPT VARIABLES
#
# set GLOBAL30S = TRUE, if you want to use global 30 sec. elevation data:
#
#   set GLOBAL30S = TRUE
    set GLOBAL30S = FALSE
#
if ( $GLOBAL30S == TRUE ) then
#
# (1) .. Specify the directory in which the 30s data was stored:
#        This assumes you have ftp'ed the data down yourself and
#          uncompressed and untared the data.
#
#   set Data_Dir = /usr/tmp/username
#
# (2) .. Use the data from ftp:
#        Leave users blank if you are not using machines from MMM/NCAR
#
    set Data_Dir = ftp
    set users    = MMM
#   set users    =
else
    set Data_Dir =
endif
#
#  set TerPlt = FALSE to create domain maps only
#  set TerPlt = TRUE  to create terrain/landuse output
#
   set TerPlt = TRUE
#  set TerPlt = FALSE
#
#   whether to create color-filled terrain/landuse plots
#
   set FillCo = TRUE
#  set FillCo = FALSE
#
#  set NewLandUse = TRUE if you want to use a better landuse dataset
#      derived from 30 sec vegetation data
#
#  set NewLandUse = TRUE
   set NewLandUse = FALSE
#
#  if NewLandUse = TRUE, select which type of input data you would
#     like to use:
#     USGS = Global coverage, 24 catagories
#      SiB = 0 - 90N, 60W - 180W, 16 catagories (same as used in Eta)
#
if ( $NewLandUse == TRUE ) then
   set VegType = USGS
#  set VegType = SiB
endif
#
if ( $TerPlt == FALSE ) then
   set GLOBAL30S = FALSE
   set NewLandUse = FALSE
endif
#
# --------------------------------------------------------------
#          1. Set up parameter statements
# --------------------------------------------------------------
#
cat > src/parame.incl.tmp << EOF
C       IIMX,JJMX are the maximum size of the domains, NSIZE = IIMX*JJMX
```

```
    PARAMETER (IIMX = 100, JJMX = 100, NSIZE = IIMX*JJMX)
EOF
cat > src/paramed.incl.tmp << EOF
C       ITRH,JTRH are the maximum size of the terrain data.
C       NOBT = ITRH*JTRH, here assuming
C        ITRH= 270 ( 45 deg. in north-south direction, 10 min. resolution)
C        JTRH= 450 ( 75 deg. in north-south direction, 10 min. resolution)
        PARAMETER (ITRH =  500, JTRH =  500, NOBT = ITRH*JTRH)
C       PARAMETER (ITRH = 1500, JTRH = 1800, NOBT = ITRH*JTRH)
EOF
#
# -------------------------------------------------------------
#          2. Set up NAMELIST
# -------------------------------------------------------------
#
if ( -e terrain.namelist ) rm terrain.namelist
cat > terrain.namelist << EOF
 &MAPBG
 PHIC  =   36.0,       ; CENTRAL LATITUDE (minus for southern hemesphere)
 XLONC =  -85.0,       ; CENTRAL LONGITUDE (minus for western hemesphere)
 IEXP  =    .T.,       ; .T. EXPANDED COARSE DOMAIN, .F. NOT EXPANDED.
 AEXP  =   360.,       ; APPROX EXPANSION(KM)
 IPROJ = 'LAMCON',     ; MAP PROJECTION
 ;IPROJ = 'POLSTR',     ; MAP PROJECTION
 ;IPROJ = 'MERCAT',     ; MAP PROJECTION
 &END
 &DOMAINS
 ;
 MAXNES = 2,            ; max no of domains
 ;
 NESTIX=   35, 49,136,247,  1,     ; DOMAIN  IX
 NESTJX=   41, 52,181,247,  1,     ; DOMAIN  JX
 DIS   =   90.,30.,6., 2.,  1.,    ; GRID SIZE(KM)
 NUMNC =    1,  1,  2,  3,  1,     ; MOTHER DOMAIN ID
 NESTI =    1, 10, 28, 28,  1,     ; LOWER LEFT I OF NEST
 NESTJ =    1, 17, 25, 35,  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,  3,  3,  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 terrain and landuse
 ;      5; 30 sec ( ~.9 km) global terrain and landuse
 ;
 NSTTYP=   1,   2,  2,  2,   2,    ;  1 -- ONE WAY NEST, 2 -- TWO WAY NEST
 &END
 &OPTN
 IFTER   = .${TerPlt}.,   ; .T.-- TERRAIN,  .F.-- PLOT DOMAIN MAPS ONLY
 DATASW  = .T.,         ; .T. user specify terrain and landuse resolution (ntype)
 ;                        .F. terrain program choose the data resolution
 IFANAL  = .F.,         ; .T.-- OBJECTIVE ANALYSIS, .F.-- INTERPOLATION
 ISMTHTR =  2 ,         ;  1: 1-2-1 smoother, 2: two pass smoother/desmoother
 IFEZFUG = .T.,         ; .T. USE EZMAP WATER BODY INFO TO FUDGE THE LAND USE
 IFTFUG  = .F.,         ; .T. DON'T DO EZFUDGE WITHIN THE USER-SPECIFIED
 ;                        LAT/LON BOXES, need to define namelist fudget
 IFFUDG  = .F.,         ; .T. POINT-BY-POINT FUDGING OF LANDUSE,
                        ;  need to define namelist fudge
 IPRNTD  = .F.,         ; PRINT OUT LAT. AND LON. ON THE MESH
 IPRTHT  = .F.,         ; PRINT OUT TERRAIN HEIGHT ON THE MESH
 IPRTLU  = .F.,         ; PRINT OUT LANDUSE ON THE MESH
 FIN     = 200., 100., 300., 500., 100.,  ; CONTOUR INTERVAL (meter) FOR TERRAIN HEIGHT
PLOT
 ;TRUELAT1=91.,         ; TRUE LATITUDE 1
 ;TRUELAT2=91.,         ; TRUE LATITUDE 2, use this if IPROJ='LAMCON'
 IFILL    = .${FillCo}.,         ; .TRUE. --- color filled plots
 VegeSoil = .${NewLandUse}.,     ; .TRUE. --- Create the Vegetation-Soil files
  VegOnly = .${NewLandUse}.,     ; .TRUE. --- Vegetation files only
 G30sD    = .${GLOBAL30S}.,      ; .TRUE. --- Use global 30 sec elevation data
 &END
 &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
 ; The example below is for two domains. Add more for domain 3 and up
 ;       if needed. Do not remove 0 values for domain 1 and/or 2 even
 ;       they are not used.
 ;
 IFUG(1,1)=  200*0,     ; I location for fudge points in domain 1
 IFUG(1,2)=  200*0,     ; I location for fudge points in domain 2
 JFUG(1,1)=  200*0,     ; J location for fudge points in domain 1
 JFUG(1,2)=  200*0,     ; J location for fudge points in domain 2
 LNDFUG(1,1)= 200*0,    ; land-use value at fudge points for domain 1
 LNDFUG(1,2)= 200*0,    ; land-use value at fudge points for domain 2
 &END
 &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
 &END
 &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
 &END
EOF
#
# ------------------------------------------------------------------------------
#
#                  END OF USER MODIFICATION
#
# ------------------------------------------------------------------------------
#
#       Create a namelist without comments
#
sed -f Templates/no_comment.sed terrain.namelist | grep "[A-Z,a-z]" > terlif.tmp
mv terlif.tmp terrain.namelist
#
# ------------------------------------------------------------------------------
#
#       Get low-resolution terrain/landuse source data
#
cd Data
if ( $TerPlt == TRUE ) then
  if ( ! -e landu_ieee.30 ) then
     set dhost=ftp.ucar.edu
     echo "About to contact $dhost ..."
     cat >! ftp.tmp << EOF
     user anonymous ${USER}@
     bi
     cd mesouser/Data/TERRAIN
     mget terln.tar.gz
     quit
EOF
     echo "ftping ..."
```

```
      ftp -v -n -i $dhost < ftp.tmp >&! ftp.output
      rm -rf ftp.tmp ftp.output
      echo "untarring and decompressing..."
      gunzip -c terln.tar.gz | tar xvf -
      rm -rf terln.tar.gz
  endif
endif
#
#       Get vegetation/land-water mask data if NewLandUse = TRUE
#
if ( $NewLandUse == TRUE ) then

  if ( $VegType == USGS ) then

      if ( ! -e lwmask-usgs.30 ) then
         set dhost=ftp.ucar.edu
         echo "About to contact $dhost ..."
         cat >! ftp.tmp << EOF
         user anonymous ${USER}@
         bi
         cd mesouser/Data/TERRAIN
         mget landwatermask-usgs.tar.gz
         mget vegetation-usgs.tar.gz
         quit
EOF
         echo "ftping ..."
         ftp -v -n -i $dhost < ftp.tmp >&! ftp.output
         rm -rf ftp.tmp ftp.output
         echo "untarring and decompressing..."
         gunzip -c landwatermask-usgs.tar.gz | tar xvf -
         gunzip -c vegetation-usgs.tar.gz    | tar xvf -
         rm -rf landwatermask-usgs.tar.gz vegetation-usgs.tar.gz
      endif

      if ( ( $GLOBAL30S == TRUE ) && ( ! -e lwmask-usgs.30s ) ) then
         set dhost=ftp.ucar.edu
         echo "About to contact $dhost ..."
         cat >! ftp.tmp << EOF
         user anonymous ${USER}@
         bi
         cd mesouser/Data/TERRAIN
         get landwatermask30s-usgs.gz lwmask-usgs.30s.gz
         get vegetation30s-usgs.gz veg-usgs.30s.gz
         quit
EOF
         echo "ftping ..."
         ftp -v -n -i $dhost < ftp.tmp >&! ftp.output
         rm -rf ftp.tmp ftp.output
         gunzip lwmask-usgs.30s.gz
         gunzip veg-usgs.30s.gz
      endif

  else if ( $VegType == SiB ) then
#
      if ( ! -e lwmask-sib.30 ) then
         set dhost=ftp.ucar.edu
         echo "About to contact $dhost ..."
         cat >! ftp.tmp << EOF
         user anonymous ${USER}@
         bi
         cd mesouser/Data/TERRAIN
         mget landwatermask-sib.tar.gz
         mget vegetation-sib.tar.gz
         quit
EOF
         echo "ftping ..."
         ftp -v -n -i $dhost < ftp.tmp >&! ftp.output
         rm -rf ftp.tmp ftp.output
         echo "untarring and decompressing..."
         gunzip -c landwatermask-sib.tar.gz | tar xf -
         gunzip -c vegetation-sib.tar.gz | tar xf -
         rm -rf landwatermask-sib.tar.gz vegetation-sib.tar.gz
      endif
```

```
  endif

endif
cd ..
#
#      Set local file names
#
if ( $NewLandUse == TRUE ) then

   if ( $VegType == USGS ) then

      set LwmData60m = Data/lwmask-usgs.60
      set LwmData30m = Data/lwmask-usgs.30
      set LwmData10m = Data/lwmask-usgs.10
      set LwmData05m = Data/lwmask-usgs.05
      set VegData60m = Data/veg-usgs.60
      set VegData30m = Data/veg-usgs.30
      set VegData10m = Data/veg-usgs.10
      set VegData05m = Data/veg-usgs.05
      if ( $GLOBAL30S == TRUE ) then
         set LwmData30s = Data/lwmask-usgs.30s
         set VegData30s = Data/veg-usgs.30s
      endif
      cp src/paramesv1.incl src/paramesv.incl.tmp
      cp src/vs_data2.incl  src/vs_data.incl.tmp
      ./Templates/incldiff.sh src/paramesv.incl.tmp src/paramesv.incl
      ./Templates/incldiff.sh src/vs_data.incl.tmp src/vs_data.incl

   else if ( $VegType == SiB ) then

      set LwmData60m = Data/lwmask-sib.60
      set LwmData30m = Data/lwmask-sib.30
      set LwmData10m = Data/lwmask-sib.10
      set LwmData05m = Data/lwmask-sib.05
      set LwmData30s = Data/lwmask-sib.30s
      set VegData60m = Data/veg-sib.60
      set VegData30m = Data/veg-sib.30
      set VegData10m = Data/veg-sib.10
      set VegData05m = Data/veg-sib.05
      set VegData30s = Data/veg-sib.30s
      cp src/paramesv0.incl src/paramesv.incl.tmp
      cp src/vs_data0.incl  src/vs_data.incl.tmp
      ./Templates/incldiff.sh src/paramesv.incl.tmp src/paramesv.incl
      ./Templates/incldiff.sh src/vs_data.incl.tmp src/vs_data.incl

   else

      echo `STOP due to wrong VegType ` $VegType
      exit (1)

   endif

endif
#
#      link to Fortran units
#
set ForUnit = fort.
rm ${ForUnit}9 ${ForUnit}1* ${ForUnit}2* ${ForUnit}4*
#
   ln -s  namelist              ${ForUnit}9
   ln -s  terrain.namelist      ${ForUnit}15
   ln -s  ezids                 ${ForUnit}18
#
   ln -s  raobsta.ieee          ${ForUnit}16
   ln -s  Data/landu.60         ${ForUnit}20
   ln -s  Data/ter.60           ${ForUnit}21
   ln -s  Data/landu.30         ${ForUnit}22
   ln -s  Data/ter.30           ${ForUnit}23
   ln -s  Data/landu.10         ${ForUnit}24
   ln -s  Data/ter.10           ${ForUnit}25
   ln -s  Data/ter.05           ${ForUnit}27
#
   if ( $NewLandUse == TRUE ) then
      ln -s $LwmData60m         ${ForUnit}40
```

```
      ln -s $LwmData30m         ${ForUnit}41
      ln -s $LwmData10m         ${ForUnit}42
      ln -s $LwmData05m         ${ForUnit}43
#
      ln -s $VegData60m         ${ForUnit}45
      ln -s $VegData30m         ${ForUnit}46
      ln -s $VegData10m         ${ForUnit}47
      ln -s $VegData05m         ${ForUnit}48
      if ( ( $GLOBAL30S == TRUE ) || ( $VegType == SiB ) ) then
         ln -s $LwmData30s         ${ForUnit}44
         ln -s $VegData30s         ${ForUnit}49
      endif
   endif
#
# -------------------------------------------------------------------------
#
unlimit
#
#     Check to see if recompilation is needed
#
cd src
../Templates/incldiff.sh parame.incl.tmp parame.incl
../Templates/incldiff.sh paramed.incl.tmp paramed.incl
cd ..
#
#     Obtain 30 sec global elevation data, and preprocess the data
#
if ( $GLOBAL30S == TRUE ) then
#
# (1) Generate the "dem_read" for getting the 30s data:
#
      make data_area.exe
      src/data_area.exe >! data_area.out
      set toast = $status
      if ( $toast != 0 ) then
         echo "error in running data_area, stopping"
         exit(1)
      endif
      rm data_area.out

# if file dem_read is not created, no need to do the following

  if ( -e dem_read ) then

      echo "---------------------------------------------------"
      echo "Global 30-sec data files needed"
      echo "   FileName     MSS_file      FTP_file    FortranUnit"
      cat dem_read
      echo "---------------------------------------------------"
#
# (2) Get the data from MSS or ftp site or "ln" to the specified directory:
#
      mkdir -p Data30s
      cd Data30s ; mv ../dem_read .
        set File30s = `cat dem_read`
        set Nfiles = ${#File30s}
        echo $Nfiles $File30s
#
        set Num0 = 1
        while ( $Num0 <= $Nfiles )
          @ Num1 = $Num0 + 1
          @ Num2 = $Num0 + 2
          @ Num3 = $Num0 + 3

          if ( $Data_Dir == ftp ) then

#  ----- Data from ftp edcftp.cr.usgs.gov:

            if ( ! -e $File30s[$Num0] ) then

               if ( $users == MMM ) then
                  cp ../Templates/ftp_edcdata.csh . ; chmod +x ftp_edcdata.csh
                  ftp_edcdata.csh $File30s[$Num2].gz
                  rm GetEDCFile.pl
```

```
                  else
                  set dhost=edcftp.cr.usgs.gov
                  echo "About to contact $dhost ..."
                  cat >! ftp.tmp << EOF
                  user anonymous ${USER}@
                  cd /pub/data/gtopo30/global
                  mget ${File30s[$Num2]}.gz
                  quit
EOF
                  echo "ftping ..."
                  ftp -v -n -i $dhost < ftp.tmp >&! ftp.output
                  rm ftp.tmp ftp.output
               endif
               if ( ! -e ${File30s[$Num2]}.gz ) then
                  echo "ftp is failed"
                  exit (1)
               endif
               echo "untarring and decompressing..."
               gunzip -c ${File30s[$Num2]}.gz | tar xvf -
               rm *.tar.gz *.DMW *.GIF *.HDR *.PRJ *.SCH *.SRC *.STX

            endif

#  Assuming the data is in direct access format, convert to non-direct-access form

            echo ls -l ${File30s[$Num0]} >! hold
            set test = `source hold`
            if ( ( $test[5] == 57600000 ) || ( $test[5] == 51840000 ) ) then
               cp ../dem_convert.csh . ; chmod +x dem_convert.csh
               dem_convert.csh $File30s[$Num0] >&! dem_convert.out
            endif

            if ( -e ${ForUnit}$File30s[$Num3] ) rm ${ForUnit}$File30s[$Num3]
            ln -s $File30s[$Num0]  ${ForUnit}$File30s[$Num3]

          else

#  ----- link to data file in directory ${Data_Dir}:

            echo "accessing data from specified directory"
            if ( -e ${Data_Dir}/$File30s[$Num0] ) then

#  Assuming the data is in direct access format, convert to non-direct-access form

               cp ${Data_Dir}/$File30s[$Num0] $File30s[$Num0]
               echo ls -l ${File30s[$Num0]} >! hold
               set test = `source hold`
               if ( ( $test[5] == 57600000 ) || ( $test[5] == 51840000 ) ) then
                  cp ../dem_convert.csh . ; chmod +x dem_convert.csh
                  dem_convert.csh $File30s[$Num0] >&! dem_convert.out
               endif

               if ( -e ${ForUnit}$File30s[$Num3] ) rm ${ForUnit}$File30s[$Num3]
               ln -s $File30s[$Num0]  ${ForUnit}$File30s[$Num3]

            else
               echo "File ${Data_Dir}/$File30s[$Num0] does not exist"
               exit(1)
            endif

          endif
          @ Num0 = $Num0 + 4
        end
      cd ..
#
# (3) To reconstruct the data for use of TERRAIN:
#  .. mv some files to directory: Data30s
#
      echo "beginning reconstruct data"
      mv data30sID  Data30s/.
      mv para.incl  src/.
      make rdem.exe
      mv src/rdem.exe Data30s/.
      cd Data30s
```

```
        rm new_*
        rdem.exe > rdem.out
        set toast = $status
        if ( $toast != 0 ) then
           echo "error in running rdem, stopping"
           exit(3)
        endif
        mv new_* ../Data/.
        rm gmeta rdem.exe data30sID ${ForUnit}79 ${ForUnit}8*
      cd ..
  else
     echo "30 SEC ELEVATION DATA WERE NOT REQUESTED. CHECK NTYPE IN NAMELIST"
     set GLOBAL30S = FALSE
  endif

endif
#
if ( $GLOBAL30S == FALSE ) then
# ------ use 48 US 30 sec data:
   ln -s  Data/ter.30s     ${ForUnit}29
endif
#
# ----------------------------------------------------------------------
#
#      Execute terrain
#
make terrain.exe
#
if ( ! -e terrain.exe ) ln -s src/terrain.exe  terrain.exe
#
date
timex terrain.exe >&! terrain.print.out
#
rm ${ForUnit}* namelist
```