III. DATAGRID

This section discusses the DATAGRID program. Section III.A presents the general purpose and methods of the program, with no discussion of the source code. Section III.B contains some helpful instructions for executing DATAGRID, with detailed discussions of parameters and options which the user may need to set. Section III.C focuses on the DATAGRID code itself, and includes discussions of all major subprograms. Finally, Section III.D briefly discusses an auxiliary program, called D_ECMWF, which may need to be run in conjunction with DATAGRID.

A. Purpose and Methods of DATAGRID

The main purpose of program DATAGRID is to interpolate coarse-resolution global or hemispheric meteorological analyses horizontally to the mesoscale grid. In its most general description, DATAGRID breaks down into three primary tasks:

1) Reading the coarse-resolution input analyses.
2) Interpolating the input analyses to the mesoscale grid.
3) Writing the interpolated analyses to an output file.

In the first task, DATAGRID retrieves the coarse-resolution analyses as specified by the user. The input analyses may currently come from one of four sources: The NMC global analyses, with 2.5° latitude-longitude resolution; the ECMWF northern-hemisphere analyses (for the years 1980 through 1989), with 2.5° latitude-longitude resolution; the ECMWF Basic Level III Consolidated dataset global analyses for the Tropical Ocean and Global Atmosphere program (beginning with 1985) with 2.5° latitude-longitude resolution (hereafter referred to as the ECMWF/TOGA dataset); and real-time global NMC MRF analyses, available from Unidata, with 2.5° latitude and 5.0° longitude resolution. For any particular level at which data are not available, fields are either derived from other existing fields or generated by vertical or temporal interpolation. If an entire NMC data set from one time period is missing, DATAGRID can interpolate temporally from earlier and later fields to create analyses at the missing time.

The second task is the horizontal interpolation from the coarse-resolution analyses to the expanded mesoscale grid. First, the terrestrial fields (latitude, longitude, Coriolis parameter, and map scale factor) are calculated at all the points of the mesoscale grid. Then, each coarse-resolution meteorological field is interpolated to the mesoscale grid using a 16-point, two-dimensional overlapping-parabolic interpolation method (discussed below). The fields of terrain height and land-use characteristics have already been interpolated to the mesoscale grid by program TERRAIN.

Once all meteorological fields from one time period have been interpolated to the mesoscale grid, the third task is executed, which writes the interpolated fields and several derived fields to an output file. If virtual temperature rather than temperature was accessed, it is converted to temperature. If surface temperature is not available, it is derived from pressure-level temperatures, assuming a lapse rate of 6.5°C km^{-1}. Surface geostrophic u and v, adjusted for friction, are calculated. Output fields are written in a standard format and order, so that program RAWINS can easily read them.

1. Overlapping Parabolic Interpolation

The main purpose of DATAGRID is the horizontal interpolation of analyses. The method of interpolation which DATAGRID uses is a 16-point, two-dimensional overlapping parabolic fit, as illustrated in Figs. 3.1 and 3.2. The task is split into a series of one-dimensional interpolations, performed by using four successive data points (A, B, C, and D in Fig. 3.1), two on either side of the point ξ under consideration. For the one-dimensional interpolations, two parabolas are fitted to the three-point sets A, B, C (long dashes) and B, C, D (short dashes). The values at ξ on each parabola are determined, and a weighted average between the two is calculated as the interpolated value at ξ.

The procedure for the two-dimensional interpolation is illustrated in Fig. 3.2. The point (ξ, η), where the value is calculated, should be surrounded by 16 data points. First, the one-dimensional interpolation is used to compute values at A, B, C, and D, along the line $y = \eta$. Then, the value at (ξ, η) is calculated by interpolating along the line $y = \eta$, using the values at A, B, C, and D. The result is not altered if values are first interpolated to points E, F, G, and H; and then from those values to point (ξ, η). Near the boundaries of the input analyses, where fewer than 16 surrounding points exist, a slightly modified procedure is used.

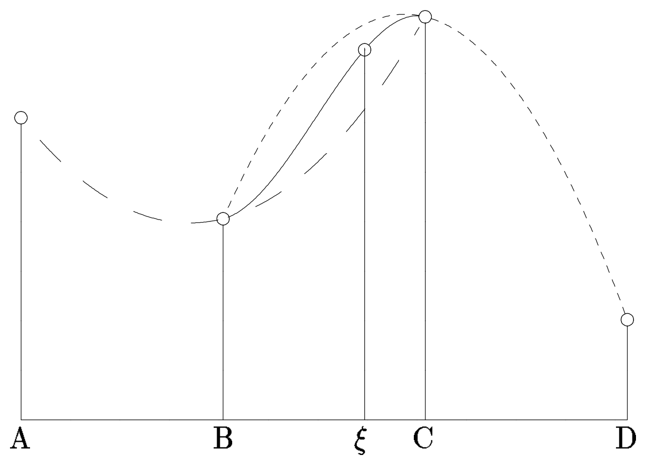

Fig. 3.1: One-dimensional horizontal interpolation used in DATAGRID. A biparabolic fit to point ξ from four points A, B, C, and D.

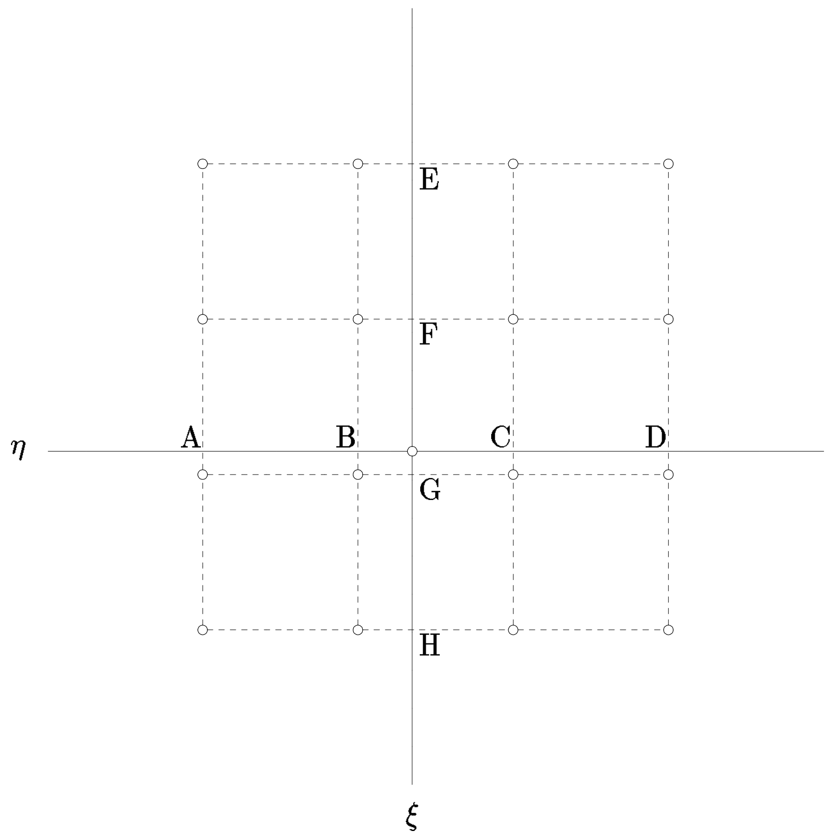

Fig. 3.2: Two-dimensional horizontal interpolation used in DATAGRID, from a field of 16 points to the point (ξ, η).

B. How to use DATAGRID

This section contains information which the user will need in order to run DATAGRID successfully. The discussion is specific to running DATAGRID on NCAR's Cray Y-MP (shavano), and assumes some familiarity with the UNICOS operating system and the C-shell script language. An example of a shell script used to submit program DATAGRID to shavano is shown in Appendix B, and can be found in the file /u1/mesouser/Decks/Datagrid/datagrid.deck on shavano.

The PSU/NCAR modeling system is undergoing continuous change and improvement. With respect to program DATAGRID, this ongoing process means that certain new capabilities are periodically added, and old capabilities may be altered or removed. Thus, the procedures to follow when using DATAGRID will also change periodically. Any detailed instructions to the user therefore have a limited period of usefulness.

Program DATAGRID can use NMC, ECMWF, ECMWF/TOGA, or Unidata analyses as input. However, DATAGRID initially did not have the capability to use ECMWF, ECMWF/TOGA, or Unidata analyses as input. When the ECMWF and Unidata capabilities were added, the reading and some processing of the input analyses were put into separate programs in order to minimize changes to the DATAGRID code itself. The reading and processing of the Unidata input have subsequently been incorporated into the DATAGRID code itself. The reading and processing of the ECMWF input is still performed by a separate program, D_ECMWF. Program D_ECMWF, automatically executed by the DATAGRID script, reads and unpacks the ECMWF analyses and writes them to a file in a format which DATAGRID can read. D_ECMWF is discussed in further detail in Section III.D. The reading and processing of the ECMWF/TOGA analyses is a quite recent addition to DATAGRID, and is not discussed in the same detail as the other capabilities of DATAGRID.

1. Data input to DATAGRID

Before submitting DATAGRID, the user must first make a number of important decisions regarding input options. Since the output from RAWINS, the improved mesoscale analyses, is highly sensitive to the first-guess fields, the options activated for DATAGRID may significantly affect the simulations. Options are activated by setting appropriate variables in the DATAGRID script and the DATAGRID Local Master Input File. It is convenient to separate the discussion into two main categories of submittals for two types of simulations: historical simulations and real-time forecasts.

a. Historical simulations

For historical simulations, several decisions must be made regarding the input of pressure-level analyses, sea-surface temperature analyses, and snow-cover analyses.

Pressure-level Analyses

For historical simulations, the user must first decide which source of input to use for the large-scale pressure-level analyses. The options currently available are NMC, ECMWF, and ECMWF/TOGA analyses, each archived on the NCAR Mass Storage System (MSS). Catalogs listing the MSS files which contain the NMC, ECMWF,and ECMWF/TOGA analyses may be found on shavano in the /u1/mesouser/catalog directory, and are updated periodically

The NMC files include global analyses, at mandatory levels (up to 50 mb), of virtual temperature (K), geopotential height (gpm), u and v wind components (m s^{-1}), and relative humidity (%, up to 300 mb), on a 2.5° latitude-longitude grid, archived every 12 hours. Analyses of surface virtual temperature (K) and sea-level pressure (mb) are also included. Sea-surface temperature is archived every 24 hours, and snow-cover data are archived intermittently. Each data record holds all of the data for one variable, one level, one hemisphere, and one time. These data are available beginning with July, 1976. The fields accessed from the NMC archives are shown in Table 3.1.

The ECMWF files include global mandatory-level analyses (up to 100 mb, except for the 400, 250, and 150 mb levels) of temperature (K), geopotential height (gpm), u and v wind components (m s^{-1}), and relative humidity (%, up to 300 mb), on a 2.5° latitude-longitude grid, archived every 12 hours. No sea-surface temperature, sea-level pressure,

Table 3.1 Input Data Availability

Level	Snow cover (0 or 1)	Sea-sfc temp (K)	Pres (mb)	Temp (K)	Virtual temp (K)	Geop. ht (gpm)	Wind u-comp (m s^{-1})	Wind v-comp (m s^{-1})	Rel. hum %
Sea Lvl		N,U	N,T,U						
Surface	N			T	N		T	T	
1000 mb				E,T	N,U	N,E,T,U	N,E,T,U	N,E,T,U	N,E,T
850 mb				E,T	N,U	N,E,T,U	N,E,T,U	N,E,T,U	N,E,T,U
700 mb				E,T	N,U	N,E,T,U	N,E,T,U	N,E,T,U	N,E,T,U
500 mb				E,T	N,U	N,E,T,U	N,E,T,U	N,E,T,U	N,E,T,U
400 mb				T	N,U	N,T,U	N,T,U	N,T,U	N,T
300 mb				E,T	N,U	N,E,T,U	N,E,T,U	N,E,T,U	N,E,T
250 mb				T	N,U	N,T,U	N,T,U	N,T,U	
200 mb				E,T	N,U	N,E,T,U	N,E,T,U	N,E,T,U	
150 mb				T	N,U	N,T,U	N,T,U	N,T,U	
100 mb				E,T	N,U	N,E,T,U	N,E,T,U	N,E,T,U	
70 mb				T	N	N,T	N,T	N,T	
50 mb				T	N	N,T	N,T	N,T	
30 mb				T		T	T	T	
10 mb				T		T	T	T	

Input data availability for program DATAGRID: Meteorological input data provided from the NMC archives (denoted by "N"), from the ECMWF archives (denoted by "E"), from the ECMWF/TOGA archives (denoted by "T"), and from Unidata (denoted by "U"). The horizontal resolution for NMC, ECMWF, and ECMWF/TOGA data is 2.5° latitude and longitude; the horizontal resolution for Unidata data is 2.5° latitude and 5.0° longitude.

For ECMWF input data, DATAGRID derives sea-level pressure and surface temperature. DATAGRID vertically interpolates to create fields at 400, 250 and 150 mb.

For Unidata input data, DATAGRID derives surface temperature, and approximates relative humidity at 1000, 400, and 300 mb.

or snow-cover data are available. Each data record holds all the data for one variable, one level, and one time. These data are available for the years 1980 through 1989. Although global data are available, DATAGRID accesses only the Northern Hemisphere data. For a model domain which crosses the equator or is entirely in the Southern Hemisphere, ECMWF analyses may not be used as input. The fields accessed from the ECMWF archives are shown in Table 3.1.

The ECMWF/TOGA files include global mandatory level-analyses (up to 10 mb) of temperature (K), geopotential (gpm), u and v wind components (m s^{-1}), and relative (%, up to 300 mb), on a 2.5° latitude-longitude grid, archived every 12 hours. Analyses of surface temperature (K), sea-level pressure (Pa), and surface u and v (m s^{-1}) are also included. Each data record in the file holds all of the data for one variable, one level, one hemisphere, and one time period. These data are available beginning with the year 1985. The fields accessed from the ECMWF/TOGA archives are shown in Table 3.1.

Originally, DATAGRID could access only NMC files. However, it had been noticed that the NMC data set was missing entire time periods of data more frequently than the ECMWF data set. Although DATAGRID can temporally interpolate to generate data for missing NMC time periods, this procedure did not always produce fields of acceptable quality. The capability to access the ECMWF data set was added to DATAGRID to give the user an alternative if time periods were missing from the NMC data set. Unfortunately, the ECMWF archives do not include data at the 400, 250, and 150 mb mandatory levels. Vertical interpolation is thus necessary for DATAGRID to use the ECMWF analyses. The ECMWF/TOGA dataset is more complete than the ECMWF dataset, and also has fields not included in the NMC dataset. If all three data sets are complete, it is unclear whether one produces better first-guess fields than the others. Such differences probably depend on the particular case. If an examination of the NMC data set reveals that time periods are missing, the user may want to select the ECMWF or ECMWF/TOGA dataset as input, rather than temporally interpolate the NMC analyses.

The option to use mandatory-level NMC, ECMWF, or ECMWF/TOGA analyses as input to DATAGRID is activated by setting the script variable `Analysis` to `NMC`, `ECMWF`, or `ECMTOGA`. For NMC input, the MSS filenames which contain the data for the simulation period must be listed by the DATAGRID script variable `InAnly`. The script retrieves the files from the MSS. Program DATAGRID subsequently accesses these files as FORTRAN units 20 and (if necessary) 21. For ECMWF input, the MSS filenames which contain the data for the simulation period must be listed by the script variables `InAnly00` and `InAnly12` (for 0000 and 1200 UTC analyses, respectively.) The DATAGRID script retrieves the files from the MSS. The files are preprocessed by program D_ECMWF (automatically executed by the DATAGRID script) and written to an output file. This file of preprocessed fields is subsequently accessed by program DATAGRID as FORTRAN unit 20. For ECMWF/TOGA input, the MSS filenames which contain the data for the simulation period must be listed by the DATAGRID script variable `InAnly`. The script retrieves the files from the MSS. Program DATAGRID subsequently accesses these files as FORTRAN unit 31 and (if necessary) 32.

Sea-Surface Temperatures

Once the choice of input for the pressure-level analyses has been made, the source of the sea-surface temperatures must be specified. The options available for historical sea-surface temperatures are data from the NMC archives, from Navy archives, or from climatological tables. The NMC archives contain global sea-surface temperatures, on a 2.5° latitude-longitude grid, updated every 24 hours. Navy archives contain Northern Hemisphere sea-surface temperatures, on a 63×63 octagonal grid, updated every 24 hours. The climatological tables have monthly mean values of sea-surface temperature on a global 2° latitude-longitude grid.

If NMC input for the pressure-level analyses has been selected, then the most convenient option for sea-surface temperatures is to use the data from the NMC archives, since no further files need to be accessed. If, however, the sea-surface temperatures are missing from the NMC files, the data from the Navy archives should be accessed. If sea-surface temperatures are missing from both the NMC and the Navy archives, then climatological values should be used.

If ECMWF or ECMWF/TOGA input for the pressure-level analyses has been selected, then the only options available for sea-surface temperatures are the Navy data and the climatological values, since the ECMWF and ECMWF/TOGA data sets do not include sea-surface temperatures. Again, the climatological values should not be used unless the Navy data are not available.

The selection of the source of sea-surface temperatures is made by setting the script variable `SST` to `NMC`, `Navy`, or `Clim`. If Navy sea-surface temperatures are selected (limited to the Northern Hemisphere), the Local Master Input File option IHEMIS must be set to 3.

Snow-Cover Data

For historical submittals (with NMC, ECMWF, or ECMWF/TOGA analyses used as pressure-level input), snow-cover data are available only from the NMC archives. The variables which must be set (in both the DATAGRID script and the Local Master Input File) depend on the selection of pressure-level input.

The snow-cover data are archived intermittently. A catalog listing the dates for which snow-cover data are available may be found on shavano in the /u1/mesouser/catalog directory.

For NMC pressure-level input, the user may specify (with the Local Master Input File options ISNOW1 and ISNOW2) up to two dates for which snow-cover data are available. The first option (ISNOW1) is generally set to the latest date for which snow-cover data are available prior to the simulation period. The second option (ISNOW2) is set to the next date for which snow-cover data are available, if the snow-cover data have been updated during the simulation period. If the snow-cover data have not been updated during the simulation period, set ISNOW2 to 99999999.

Because snow-cover data are archived only intermittently, the first snow date ISNOW1 may refer to a file other than those files accessed for the NMC pressure-level analyses (listed in the script variable `InAnly`). In this case, the snow-cover data must be accessed from an additional NMC file. The MSS filename of that additional file which contains the snow-cover data must be specified by the script variable `InSnow`. The user must also set the Local Master Input File option ISNOW3 to .TRUE., to indicate that an earlier file must be accessed for snow-cover data. This earlier file is accessed by DATAGRID as FORTRAN unit 23.

As an example, consider a 48 h simulation beginning at 1200 UTC on 1 February 1991. The NMC catalog provides the MSS filenames for the NMC data: file /DSS/Y06627 holds the data for 0000 UTC, 16 January through 1200 UTC, 31 January; and file /DSS/Y06666 holds the data for 0000 UTC, 1 February through 1200 UTC, 15 February. The snow-cover catalog provides the dates for which the snow-cover data are available. The latest snow-cover date prior to the simulation period is 1200 UTC, 26 January. The snow-cover data were updated during the simulation period, on 1200 UTC, 2 February.

In this case, the script variable `InAnly` must be set to /DSS/Y06666. Local Master Input File option ISNOW1 must be set to 91012612, and option ISNOW2 must be set to 91020212. File Y06666 does not include the snow-cover data prior to the simulation period; that data would be in file Y06627. So the script variable `InSnow` must be set to /DSS/Y06627. To indicate that RAWINS must access an additional file for snow-cover data, the Local Master Input File option ISNOW3 must be set to .TRUE.

The ECMWF and ECMWF/TOGA archives do not include snow-cover data. If the ECMWF or ECMWF/TOGA data are used as pressure-level input, the user may choose to access one snow-cover analysis from the NMC archives. The time period for the snow-cover data may be either prior to or during the simulation period. The date for which snow-cover data are available must be specified in the Local Master Input File, this time with option SNOECM. Additionally, the MSS filename of the NMC file which contains the snow-cover data must be specified by the script variable `InSnow`. This file is accessed by DATAGRID as FORTRAN unit 23.

The user also has the option to input subjectively generated snow-cover fields. The fields must already be on the mesoscale grid, in FORTRAN unit 2. In this case, ISNOW1 and ISNOW2 for NMC input, and SNOECM for ECMWF or ECMWF/TOGA input, must be set to 88888888.

There is an additional option for not using snow-cover data at all. In this case, ISNOW1 and ISNOW2 for NMC input, and SNOECM for ECMWF and ECMWF/TOGA input, must be set to 99999999.

b. Real-time Forecasts

For real-time forecasts, DATAGRID accesses the NMC MRF analyses available from Unidata. Global analyses and forecasts of sea-level pressure and global mandatory-level analyses and forecasts (up to 100 mb) of virtual temperature, geopotential height, u and v wind components, and relative humidity, on a 2.5° latitude by 5.0° longitude grid, are available from the 00, 06, 12, 18, 24, 30, 36, 42, 48, and 60 hour forecasts. Observed sea-surface temperatures at 0000 UTC are also available. Snow-cover data are available, but are not accessed by DATAGRID. The fields accessed from Unidata are shown in Table 3.1. The option to access the real-time analyses is activated by setting the DATAGRID script variable `Analysis` to `UNIDATA`. Two additional script variables, `unidate` and `sstdate`, must also be set. Script variable `unidate` is the eight-digit date/time at which the NMC MRF forecast analyses were prepared (*i.e.*, the time for which the 00 hr forecast is valid). Script variable `sstdate` is the eight-digit date/time of the Unidata sea-surface temperatures to use (usually the 0000 UTC time one day earlier than `unidate`).

For real-time submittals, the sea-surface temperatures from the previous day are generally available from Unidata. However, if the sea-surface temperatures are not available, climatological values may be used. In either case, the DATAGRID script variable `SST` must be set to `clim`. The Local Master Input File options ISST and UNISST must also be set appropriately to select the source of sea-surface temperatures for real-time forecasts.

c. Input terrain and land-use data

In all cases, terrain and land-use data come from the MSS file identified by the DATAGRID script variable `InTerr`. This file contains the output from the program TERRAIN, written to the MSS; the data have already been interpolated to the mesoscale grid by program TERRAIN.

d. Summary of input

As a convenient reference, all input files to DATAGRID are listed and briefly described in Table 3.2. DATAGRID script variables which must generally be modified by the user are listed in Table 3.3.

Table 3.2 Input files for program DATAGRID

Input File	Description
Unit 2:	User-specified snow-cover file
Unit 3:	Navy sea-surface temperature file
Unit 9:	Terrain and land-use file
Unit 10:	Common Master Input File
Unit 12:	Climatological sea-surface temperature file
Unit 20 (21):	Input analyses (NMC or ECMWF)
Unit 23:	Additional snow-cover file
Unit 29:	DATAGRID Local Master Input File
Unit 31 (32):	Input ECMWF/TOGA analyses

Table 3.3 DATAGRID Script Variables

Variable	Options (if any) and description	

`Analysis`	`NMC`
	`ECMWF`
	`ECMTOGA`
	`UNIDATA`
		Use NMC analyses as pressure-level input
		Use ECMWF analyses as pressure-level input
		Use ECMWF/TOGA analyses as pressure-level input
		Use real-time analyses from Unidata as pressure-level input

`SST`	`NMC`
	`Navy`
	`Clim`
		Use NMC sea-surface temperatures
		Use Navy sea-surface temperatures
		Use climatological sea-surface temperatures (for historical simulations); or, use sea-surface temperatures from Unidata (for real-time forecasts)

`InTerr`	MSS filename of the output from program TERRAIN (*i.e.* terrain and land-use fields)	
`unidate`	Eight-digit date/time of the Unidata input analyses	
`sstdate`	Eight-digit date/time of the Unidata sea-surface temperatures	
`InAnly`	MSS filename(s) of NMC, ECMWF/TOGA, or Unidata input fields	
`InAnly00`	MSS filename(s) of ECMWF input fields at 0000 UTC	
`InAnly12`	MSS filename(s) of ECMWF input fields at 1200 UTC	
`InSnow`	For NMC input, `InSnow` is the MSS filename of the NMC file which holds the snow-cover data, if the snow-cover data is not in one of the NMC files listed in `InAnly`. For ECMWF input, `InSnow` is the MSS filename of the NMC file which holds the snow-cover data	
`InSST`	The MSS filename of the sea-surface temperature files, if climatological or Navy sea-surface temperatures are used.	
`OutDatg`	MSS filename to which the output data are to be written.	

2. Parameter Statements

The sample script contains a FORTRAN PARAMETER statement which replaces a PARAMETER statements in the source code. This statement is:

$$\text{PARAMETER (IMX} = ii\text{, JMX} = jj\text{)}$$

The parameters in this statement are common to many of the programs in the modeling system, and should be set by the user for each specific case. The parameters defined are:

IMX: The I (y) dimension of the expanded coarse grid.

JMX: The J (x) dimension of the expanded coarse grid.

3. Master Input Files

DATAGRID accesses many user-defined options from two input files: the Common Master Input File, and the Local Master Input File. The options in the Common Master Input File are used for several of the programs in the modeling system (hence the designation "Common"). However, the Common Master Input File itself is accessed only by programs TERRAIN and DATAGRID; its contents are passed to other components of the modeling system through record headers written to the output files. The file is composed of four input records communicated to the program through NAMELIST statements. The file contains input variables used to select options regarding date, geographical location, grid resolution, map projection, and print output; these options are discussed in detail in Table 3.4.

The Local Master Input File contains parameters used to select options regarding the source of input data. The options in the Local Master Input File are discussed in detail in Table 3.5.

4. Output from DATAGRID

DATAGRID writes two forms of output: output to FORTRAN unit 4 (the main output file) and print output. DATAGRID also writes a number of temporary scratch files, which will not be discussed here.

Table 3.4 Common Master Input File

Record	Variable	Variable description
&DATIME	IFILES	Number of time periods to process (up to 18)
	JYR	The year (two digits) of each time period to process. Each entry in JYR, JMO, JDY, and JHR corresponds to one date/time (up to 18 entries)
	JMO	The month (two digits) of each time period to process.
	JDY	The day (two digits) of each time period to process.
	JHR	The hour (two digits) of each time period to process (in UTC, usually 00 or 12).
&GRID	IMAX	The number of coarse-grid gridpoints, not including expansion (see options IEXP and AEXP, below), in the I (y) direction
	JMAX	The number of coarse-grid gridpoints, not including expansion, in the J (x) direction
	DS	The gridpoint separation (km) on the coarse grid.
	PHIC	The latitude (degrees) of the center of the coarse grid (negative in southern hemisphere)
	XLONC	The longitude (degrees) of the center of the coarse grid ($-180°$ to $180°$, negative in the western hemisphere)
	IFNEST	T: Use a nest F: Do not use a nest
	IMAXN	Number of nested-grid gridpoints in the I (y) direction
	JMAXN	Number of nested-grid gridpoints in the J (x) direction
	DSN	Gridpoint separation (km) of the nested grid. The nest ratio between the coarse grid and the nested grid must be 3:1
	ICNS1	I location in the coarse-grid domain of the nested point (1,1)
	JCNS1	J location in the coarse-grid domain of the nested point (1,1)
	NSTTYP	Type of nest: 1: one-way interactive nest 2: two-way interactive nest
	IPROJ	Map projection for the mesoscale grid. A Hollerith variable: 6HLAMCON: Lambert Conformal 6HPOLSTR: Polar Stereographic 6HMERCAT: Mercator
	IEXP	Option to use the expanded grid for DATAGRID and RAWINS. The expanded grid is used to expand the area over which the objective analyses are performed, so as to obtain better analyses near the boundaries T: Use the expanded grid F: Do not use the expanded grid
	AEXP	Approximate expansion (km) of the grid on all sides of the domain
&DATLEV	PTOP	Pressure level (mb) defining the top of the model
	NNEWPL	Number of new nonmandatory pressure levels. These nonmandatory levels are used by RAWINS to introduce data at a greater vertical resolution, thus allowing INTERP to make more accurate vertical interpolations to sigma levels
	GNLVL	Array of pressures (mb) at the new nonmandatory pressure levels
&PRINT	INY	Print output interval (gridpoints) in the I (y) direction
	JNX	Print output interval (gridpoints) in the J (x) direction
	KSIGT	Number of significant digits for print output

Table 3.5 **DATAGRID Local Master Input File.**

Variable	Options (if any) and variable description
IFIRST	Source of the input analyses. A Hollerith variable. Used in both DATAGRID and RAWINS: 3HNMC : NMC archived analyses 5HECMWF : ECMWF archived analyses 4HTOGA : ECMWF/TOGA archived analyses 6HUNIDAT : Real-time analyses from Unidata
ISST	T : Use climatological SSTs F : Do not use climatological SSTs
UNISST	T : Use Unidata SSTs F : Do not use Unidata SSTs
IHEMIS	Hemisphere from which data are needed 0 : Northern Hemisphere 1 : Southern Hemisphere 2 : Both hemispheres 3 : Northern Hemisphere, with SSTs from Navy archives
IMSG	Option for interpolating missing data sets in the NMC archives. Used for both DATAGRID and RAWINS T : Interpolate temporally to generate missing data sets F : Leave missing data sets missing
ISEQ	The sequence number of the first missing NMC data set
JSEQ	The number of consecutive missing NMC data sets
IFILE1	The number of desired time periods stored on the first input file. For example, if DATAGRID accesses data for 4 and 5 January at both 0000 and 1200 UTC, and the first input file holds data from 1 January through 4 January, then IFILE1 = 2 (*i.e.* 0000 and 1200 UTC, 4 January)
ISNOW1	The date (YYMMDDHH), prior to the forecast, for which snow-cover data are available in the NMC archives. If snow-cover data are not needed, or if the domain is in the southern hemisphere, set ISNOW1 = 99999999. Option ISNOW1 is for NMC input analyses only
ISNOW2	The date (YYMMDDHH), during the forecast, for which snow-cover data are available in the NMC archives. If snow-cover data are not updated during the simulation period, set ISNOW2 = 99999999. Option ISNOW2 is for NMC input analyses only
ISNOW3	Logical flag to indicate whether the earliest snow-cover data are in the first regular input file (F) or in an earlier input file (T). This option is for NMC input analyses only
SNOECM	The date (YYMMDDHH) for which snow-cover data are available (either prior to or during the simulation) in the NMC archives. This option is for ECMWF or ECMWF/TOGA input analyses only
SNOFIL	The input file name for the user-supplied snow-cover data, if any
MISNV1	First date during the simulation for which Navy SSTs are not available. If no Navy SSTs are missing, set MISNV1 = 99999999. Option MISNV1 is only for Navy SSTs
MISNV2	Second date during the simulation for which Navy SSTs are not available. If no Navy SSTS are missing, set MISNV2 = 99999999. Option MISNV2 is only for Navy SSTs

a) Main Output File

For each time period processed, output is written to FORTRAN unit 4. This output consists of a header record followed by the data records for that time period. The header record contains information about the format of the output data, as well as many other variables from the Common Master Input File and the DATAGRID Local Master Input File. Each of the data records holds all of the data for one variable, at one level, and at one time. The records are written out for each time period in the following sequence.

- Record header.
- Terrain (at cross points).
- Land use (at cross points).
- Map-scale factor (at cross points).
- Map-scale factor (at dot points).
- Coriolis parameter (at dot points).
- Latitude (at cross points).
- Longitude (at cross points).
- Latitude (at dot points).
- Longitude (at dot points).
- Snow cover (at cross points).
- Sea-level pressure (at dot points).
- Sea-level pressure (at cross points).
- Surface temperature (at cross points).
- Sea-surface temperature (at cross points).
- Pressure-level temperature (at cross points); one record for each mandatory level from 1000 mb to PTOP.
- Surface geostrophic u (at dot points), adjusted for friction.
- Surface geostrophic v (at dot points), adjusted for friction.
- Geopotential height, u, and v at pressure levels (at dot points); three records for each mandatory level from 1000 mb to PTOP.
- Relative humidity at pressure levels (at cross points); one record for each mandatory level from 1000 mb to 300 mb.

b) Print Output

The primary print output when special print options are not activated is summarized as follows:

i) An inventory of all input records accessed and processed by DATAGRID. Pressure level 1001 indicates surface values (surface temperature and sea-surface temperature); pressure level 1013 indicates sea-level pressure; pressure level 1023 indicates snow cover. The following variable numbers identify the variables listed.

1	geopotential height.
8	sea-level pressure
16	temperature
48	u wind component
49	v wind component
88	relative humidity
93	snow cover
384	sea-surface temperature

ii) If IMSG = .T. in the Local Master Input File (option to temporally interpolate missing NMC data sets), interpolated values at two selected latitude-longitude gridpoints for each output record.

iii) Values of the input data on a latitude-longitude grid (2.5° resolution for a 30° latitude and 40° longitude window centered near the middle of the mesoscale grid):

- At the surface: snow cover, sea-level pressure, surface temperature, and sea-surface temperature.
- At selected mandatory levels: ϕ, T, u, v, and RH.

iv) Values of the output fields at the surface and at selected mandatory levels, for mesoscale gridpoints determined by INY and JNX in the Common Master Input File. The order of the printed fields is the same as the order that the fields are written to FORTRAN unit 4 (see Section III.B.4.a, above). For each output field, either a sample of the field is printed or a brief message is printed, indicating the variable and level. The u and v wind components are oriented relative to the mesoscale grid map.

C. DATAGRID Code

The DATAGRID code consists of the main program, called GETDAT, and over twenty-five FUNCTION and SUBROUTINE subprograms. These various components of the code share many large arrays during the program execution. The memory structure of the code is therefore discussed first. The main program is then described. Finally, all major subprograms are discussed in detail. The auxiliary program D_ECMWF is discussed briefly in Section III.D.

1. Memory Structure

This section discusses the memory structure of the program. Most of the memory is partitioned among several key arrays, many of which use variable parameterized dimensions. The parameters and arrays are discussed below.

Parameters

One FORTRAN PARAMETER statement is common to several programs in the PSU/NCAR modeling system. The parameters set by this statement define the dimensions of many arrays that require variable dimension. The parameters must generally be changed by the user for different cases. The PARAMETER statement has the form:

$$\text{PARAMETER (IMX} = ii \text{ , JMX} = jj)$$

The parameters defined in this statement are the y and x dimensions of the expanded grid, as discussed in Section III.B.2.

DATAGRID also has a local PARAMETER statement for variable parameterized dimensions. It is:

$$\text{PARAMETER (NPX} = 17\text{, IWX} = 5 * \text{(NPX} - 3) + 10)$$

where NPX is at least the number of mandatory levels plus 3. This statement does not need to be changed by the user during a standard run.

Arrays

A large portion of the in-core storage needed by DATAGRID is used by two large equivalenced arrays called IWZ(675000) and V1(145, 37, 125). IWZ holds packed and unsorted NMC input data, and V1 holds NMC input data that have been unpacked and sorted.

Most of the remaining in-core storage is used by sixteen arrays in a single blank common block. Three of the arrays have fixed dimensions, the others use the parameterized dimensions IMX, JMX, and IWX. Portions of the blank common storage are made available to DATAGRID subprograms by passing the array names and dimensions through the argument lists and common statements. The sixteen arrays are described below:

(a) A1, A2, both dimensioned by (145, 37) – These arrays hold the meteorological input fields (Northern and Southern Hemispheres, respectively, one field at a time), and are gridded to the latitude-longitude mesh of the input fields. A1 and A2 are temporary arrays; the fields are interpolated to the mesoscale grid as they are accessed.

(b) DATUNI(73, 73) – This array holds the real-time input fields from Unidata on a global latitude-longitude grid, one field at a time. DATUNI is a temporary array; the fields are interpolated to the mesoscale grid as they are accessed.

(c) E1, E2, E3, E4, all dimensioned by (IMX, JMX) – These arrays hold meteorological fields that have been horizontally interpolated to the mesoscale grid. E1, E2, E3, and E4 are temporary arrays, used to transfer data into the large 3D storage array WORK.

(d) WORK(IMX, JMX, IWX) – This storage array holds all of the fields on the mesoscale grid that are derived in DATAGRID. IWX = 5 * (NPX - 3) + 10, where NPX is at least the number of mandatory levels plus three. WORK is a permanent array.

(e) LAND, FLND, TER, SNW, TS, P0, F, SSTUNI, all dimensioned by (IMX, JMX) – These arrays are land-use (integer), land-use (floating-point), terrain, snow cover, surface temperature, sea-level pressure, Coriolis parameter, and Unidata sea-surface temperature, respectively. These are permanent arrays.

Common Blocks

DATAGRID has nine named common blocks, holding various arrays and variables which must be shared among portions of the code. They are:

(a) /MAP/ – holds variables related to the mesoscale grid.
(b) /OPTION/ – holds variables defining certain options for the data input.
(c) /OUT/ – holds variables related to the output.
(d) /CMIO/ – holds a single variable, ICMIO, used in storing data in array WORK.
(e) /CON84/ – holds variables related to the identification of the NMC input data.
(f) /VARNAM/ – used to equivalence variable names with numerical values, for locating fields within the third index of array WORK.
(g) /INT/ – holds variables initialized in subroutine INIT.
(h) /HEDMIF/ – holds record header information that is communicated among the programs of the modeling system through MSS output records.
(i) /SNOWDATE/ – holds the Local Master Input File option SNOECM, for use by the routines retrieving ECMWF/TOGA data.

Some peripheral storage is also used in DATAGRID. In the main program, data are stored on units 1 and 11; in subroutine SORTNM, on units 1, 7, 8, and 10; in subroutine MSGFIL, on units 1, 7, and 11; and in subroutine GETUNI, on units 31 and 41.

2. Main Program

The main program of DATAGRID, called GETDAT, manages the three major tasks of DATAGRID: reading the input analyses, interpolating the input analyses to the mesoscale grid, and writing the interpolated analyses to an output file. The flow structure of GETDAT is depicted in Fig. 3.3. The sections of code that accomplish the three main tasks are discussed briefly in this section. For ECMWF/TOGA data, the three tasks are all accomplished within subroutine TOGA2DG.

Input

The reading of the input data is accomplished primarily by subroutines SORTNM, MSGFIL, NAVY, RNAVY, UNAVY, RON84, UON84, GETUNI, SFGPRS, SFGRID, and UNIMSG.

The initial reading of the input analyses, if the NMC analyses have been selected as input, is performed in subroutine SORTNM, which reads packed data records, unpacks identifying variables, and sorts the fields. SORTNM then writes the fields, still in a packed format, in a standard order to FORTRAN unit 1. The reading and unpacking of the records is performed by subroutines RON84 and UON84.

Fig. 3.3 Flow structure for program GETDAT

PROGRAM GETDAT

- Initialize miscellaneous variables.
- Read Common and Local Master Input Files.
- Call Subroutine **HEDREC** to generate a record header with input variables provided by the Common and Local Master Input Files.
- If ECMWF/TOGA data are *not* used, then
  - Call Subroutine **SORTNM** to access the NMC input analyses and temporally interpolate to recreate any missing data sets. If the NMC analyses are the input analyses, SORTNM accesses all surface and pressure-level analyses fields. If the input analyses are from ECMWF or Unidata, SORTNM accesses snow-cover data only, as it is the only NMC analysis field.
- Endif
- If IEXP = .TRUE. then
  - Expand the mesoscale grid for the objective analysis.
- Endif.
- Calculate indices for printout of samples of the latitude and longitude grid.
- If real-time analyses from Unidata are to be used, then
  - Call GETUNI to retrieve real-time analyses and forecasts.
- Endif.
- If ECMWF/TOGA analyses are used, then
  - Call TOGA2DG to retrieve ECMWF/TOGA analyses, interpolate them to the mesoscale grid, and write them to the output file.
- Else
  - If using Unidata SSTs with the Unidata input analyses, then
    - Call Subroutine **NMCLAT** to interpolate Unidata SSTs to the mesoscale grid. The SSTs on the mesoscale grid are put into array SSTUNI(IMAX, JMAX).
  - Endif.
  - Do 100 IFILE = 1, IFILES (Loop over all the time periods).
    - Do 80 NPR = 1, NPRH (Loop over all pressure levels).
      - Do 70 ICY = 1, ICYC (Loop over both hemispheres, if data from both hemispheres are required).
        - If using ECMWF or Unidata input, then
          - Retrieve date, hemisphere, and variable type of the input field.
        - Else if using NMC input, then
          - Call Subroutine **RON84** to retrieve date, hemisphere, variable type of the input field.
        - Endif.

          - Check date of input file against the date in the Common Master Input File currently being processed by loop 100. If the date of the input file is earlier than the current search date, start loop 70 again to read another file. If the date of the input file is later than the current search date, pop out of loop 80.
          - Check hemisphere of input data against the hemisphere needed. If they do not match, stop the program.
          - Check the pressure level of the input data against the pressure level currently processed (NPR). If they do not match, stop the program.
          - Check the variable type of the input data against the variable types needed by the program. If the variable type is not one of the types needed, stop the program.
          - Read a field from one of a number of sources:
            - If the field is from Unidata, it is read from FORTRAN unit 41 (temporary file created in subroutine GETUNI).
            - If the field is from ECMWF, it is read from FORTRAN unit 21 (file created by the ECMWF preprocessor).
            - If the field was originally missing from the NMC dataset, and recreated by temporal interpolation, it is read from FORTRAN unit 11 (file created by subroutine MSGFIL).
            - If the field is from NMC, it is unpacked by subroutine **UON84**.
          - In any case, the field is put in A1 (Northern Hemisphere NMC or ECMWF analysis), A2 (Southern Hemisphere NMC analyses), or DATUNI (global Unidata Analysis).
          - If it is a snow-cover field, check it.
          - Convert Kelvins to °C, if necessary.
          - Print a sample of the input field.
        - Endloop 70 (hemisphere loop). At this point, the input field is either in DATUNI (for Unidata input), A1 (for Northern-Hemisphere NMC or ECMWF input), or both A1 and A2 (for Northern and Southern Hemisphere, respectively, NMC input).
        - Call Subroutine **NMCLAT** to interpolate field to mesoscale grid.
        - Call Subroutine **LCMIO** to write mesoscale field to array WORK.
        - If there are more fields at the current pressure level, start loop 70 again to read another field.
      - Endloop 80 (pressure loop).
      - Print inventory of fields accessed.
      - Check for missing data. Stop if there's a problem.
      - Call Subroutine **OUTAP** to write the interpolated analyses to the MSS output file.
    - Endloop 100 (time period loop).
  - Endif

End Program GETDAT.

Fig. 3.3: Continued.

If Unidata analyses have been selected as input, the initial reading and sorting of the analyses is performed in subroutine GETUNI. GETUNI calls SFGRID and SFGPRS to read and unpack the records, and UNIMSG to recreate missing fields by vertical or temporal interpolation. The fields are written in a standard format and order to FORTRAN unit 41.

If the input fields are ECMWF analyses, then the auxiliary program D_ECMWF has already performed the initial reading of the analyses, and has written the data to an output file in the same order which SORTNM would write for NMC input.

The sorted input data are subsequently read from the external file written by the auxiliary program, or from the files written by subroutine SORTNM or GETUNI. Additional reading occurs in subroutines NAVY (called by OUTAP) and MSGFIL (called by SORTNM). NAVY, which reads the Navy or climatological sea-surface temperatures, calls subroutine RNAVY to unpack header information from the Navy archives, and subroutine UNAVY to unpack the actual Navy data. MSGFIL unpacks data in the process of interpolating between time periods to replace missing fields.

Horizontal interpolation

The horizontal interpolations are accomplished by subprograms LBCLAT, NMCLAT, BINT, ONED, CLMSST, MAPNAV, and NAVY.

The interpolation procedure begins with calls from the main program to subroutine LBCLAT, which calculates latitude, longitude, Coriolis parameter, and map scale factor for each point of the mesoscale grid. Then for each field to be interpolated, the main program calls NMCLAT, which performs the interpolation, using functions BINT and ONED.

Other interpolations are performed in subroutines CLMSST and MAPNAV, both called by subroutine NAVY (called by OUTAP). CLMSST interpolates the climatological sea-surface temperature field, and MAPNAV interpolates the Navy sea-surface temperature field.

Output

The output from DATAGRID is generated primarily by subroutines OUTAP and OUTPT.

The output of the interpolated analyses is performed by subroutine OUTAP, called once for each time period. OUTAP retrieves the interpolated fields from various locations (*e.g.* the storage array WORK, or the temporary files), writes the fields to the output file, and calls subroutine OUTPT, which prints a sample of the field. OUTAP converts virtual temperature to temperature (as needed), and creates fields of surface geostrophic u and v (adjusted for friction).

3. DATAGRID Subprograms

This section discusses each of the main SUBROUTINE and FUNCTION subprograms of DATAGRID, in alphabetical order. For each subprogram, important input and output variables and arrays (from both the argument list and the common blocks) are discussed.

Several variables deserve special mention here, because they are used as arguments to many subprograms.

- WORK is the storage array (discussed in section III.C.1), with dimensions IX, JX, and IWMX.
- IX is the I dimension of the expanded grid, and thus the first dimension of many arrays holding fields on the mesoscale grid. IX is equal to the parameter IMX.
- JX is the J dimension of the mesoscale grid, and thus the second dimension of many arrays holding fields on the mesoscale grid. JX is equal to the parameter JMX.
- IWMX is the third dimension of WORK. IWMX is equal to the parameter IWX.

Because they are used identically in so many subprograms, WORK, IX, JX, and IWMX will not be discussed as arguments to individual subprograms.

Each subprogram is discussed according to the following pattern:

Purpose: A brief statement summarizing the main purpose or purposes of the subprogram.

On Entry: Descriptions of the state of arguments and certain common variables on entry to the subprogram. All important input to the subroutine is listed in this category.

On Exit: Descriptions of the state of important arguments and common variables on completion of the subprogram. All important output from the subroutine is listed in this category.

Calls: A list of any subprograms which are called by the subprogram being discussed.

Called by: A list of any routines which call the subprogram.

Comments: A more detailed discussion than was provided in the **Purpose** section. The comments may include a discussion of the methods which the subprogram uses to achieve the purpose as well as more detailed descriptions of the subprogram's code itself.

Function **BINT**(XX, YY, LIST, III, JJJ)

Purpose: Perform the overlapping-quadratic interpolation to the mesoscale gridpoint (XX, YY) from 16 surrounding gridpoints (see discussion in Section III.A).

On Entry:

- LIST(III,JJJ): The input analysis, *not* on the mesoscale grid.
- XX,YY: X and Y index values (interpolation location) of the mesoscale gridpoint in the coordinate system of the input grid.

On Exit:

- BINT: The interpolated value of the variable in LIST, at point (XX, YY).

Calls: Function ONED.

Called by: Subroutines CLMSST, MAPNAV, and NMCLAT.

Comments: Subroutine BINT is called for each mesoscale gridpoint in each field. For each mesoscale gridpoint, BINT constructs the array STL(4,4), which contains the values at the sixteen surrounding input gridpoints. Next, BINT calls function ONED four times, to calculate four interpolated values along the line $x =$ XX. Finally, BINT calls function ONED one more time, to interpolate from the four new values to the point (XX,YY).

If the point (XX,YY) is near the boundary of the input field and is not surrounded by 16 input gridpoints, the missing edge points are not included in the interpolation procedure. In this case, BINT repeats the process, this time interpolating first to the four points along line $y =$ YY, and then from those four points to point (XX,YY). BINT then returns the average of the two methods as the value at the interpolated point.

Subroutine **CLMSST**(SST, C, IX, JX, IWMX, WORK, YLAT, XLON, LMO, IR)

Purpose: Interpolate (using function BINT) climatological sea-surface temperatures from a latitude-longitude grid (2° resolution) to the mesoscale grid.

On Entry:

- SST(181,91): Climatological sea-surface temperatures on a latitude-longitude grid, with 2° (global) resolution.
- IR: Integer flag (IR = 2) indicating that the call to subroutine LCMIO will retrieve data from the storage array WORK.

YLAT(IX,JX) and XLON(IX,JX) are passed to CLMSST as working space only. LMO is unused in CLMSST.

On Exit: C(IX,JX): The climatological sea-surface temperatures interpolated to the mesoscale grid.

Calls: Subroutine LCMIO and function BINT.

Called by: Subroutine NAVY.

Comments: Subroutine CLMSST calls LCMIO twice to retrieve the arrays XLAT(IX,JX) and XLON(IX,JX), the latitudes and longitudes of the mesoscale gridpoints (cross), from the storage array WORK. Then, for each point in the mesoscale grid, CLMSST calculates XX and YY, the index values (interpolation location) of the mesoscale gridpoint in the coordinate system of the latitude-longitude grid. Finally, CLMSST calls BINT to interpolate the sea-surface temperature field to the point (XX,YY).

Subroutine GETUNI(UNIT41)

Purpose: Access and unpack Unidata input fields, and write them to FORTRAN unit 41.

On Exit: UNIT41: The FORTRAN unit number to which Unidata analyses, both original and interpolated (temporally and vertically), are written.

All Unidata data have been written to FORTRAN unit 41.

Calls: Subroutines SFGPRS, SFGRID, and UNIMSG.

Called by: Main program.

Comments: Subroutine GETUNI accesses Unidata input fields, and writes them to FORTRAN unit 41. First, GETUNI enters a loop over all forecast periods. For each forecast period, GETUNI calls subroutine SFGPRS, which accesses and unpacks the fields of sea-level pressure and sea-surface temperature (returned to GETUNI through Common/GRDATA/). Then, GETUNI enters a loop over the pressure levels. For each pressure level processed, SFGRID is to access and unpack the pressure-level fields of u, v, T, RH, and height (returned to GETUNI through Common/GRDATA/). Header records and fields of sea-surface temperature, sea-level pressure, and 1000 mb temperature are written to FORTRAN unit 31. Relative humidities less than 10% or greater than 100% are set to 10% or 100%, respectively. Pressure-level fields are then written to unit 31 as well.

After GETUNI has finished looping through the pressure levels and forecast periods, it calls UNIMSG to check for and recreate any missing fields. A one-point sample of each field is then printed out. At the completion of GETUNI, all fields, either original or interpolated, have been written to FORTRAN unit 41.

Subroutine HEDREC(I1, I2, I3, I4, I5, I6, I7, I8, I9, J1, J2, J3, J4, J5, J6, J7, J8, R1, R2, R3, R4, R5, R6, L1, L2, L3, L4, IPRES, NPRH)

Purpose: Generate arrays that hold input variables provided by the Common and Local Master Input Files. These arrays are later written out in the header records to the output file.

On Entry:

- I1, ..., I9: Integer values from the Common Master Input File.
- J1, ..., J8: Integer values from the Common and Local Master Input files.
- R1, ..., R6: Real values from the Common Master Input File.
- L1, ..., L4: Logical values from the Common and Local Master Input Files.
- IPRES(NPRH): Array holding pressures 1023, 1013, 1001, and pressures at the mandatory pressure levels from 1000 to 50 mb.
- NPRH: The number of pressure levels in IPRES (*i.e.* 15).

On Exit:

- MIF(30): Array holding integer values from the argument list. Common/HEDMIF/.
- MRF(10): Array holding real values from the argument list. Common/HEDMIF/.
- MLF(10): Array holding logical values from the argument list. Common/HEDMIF/.
- IHEDPR(20): Array holding mandatory pressure levels from 1000 mb to PTOP. Common/HEDMIF/.
- NSTLEV: Number of mandatory pressure levels from 1000 mb to PTOP. Common/HEDMIF/.

Calls: Subroutine HEDREC calls no other subprograms.

Called by: Main program.

Comments: NSTLEV, IHEDPR, MIF, MRF, and MLF are among the variables and arrays written out in the header record to FORTRAN unit 4, the final output file. Through this header record, these values are passed to the other programs in the modeling system. The output file is written by subroutine OUTAP.

Subroutine INIT(IHEMIS, INAVY, JW, IFILES, KF, KG, KKDAT, KSSFC)

Purpose: Initialize a number of miscellaneous variables.

On Entry:

IHEMIS: Integer flag specifying hemisphere. From the DATAGRID Local Master Input File.

INAVY: Logical flag: .TRUE. indicates that Navy, Unidata, or climatological sea-surface temperatures are to be used; .FALSE. indicates that NMC sea-surface temperatures are to be used.

IFILES: The number of time periods for DATAGRID to process. From the Common Master Input File.

On Exit: A number of variables in the argument list (JW, KF, KG, KKDAT, and KSSFC) and the common block /INT/ have been initialized.

Calls: Subroutine INIT calls no other subprograms.

Called by: Subroutine SORTNM.

Subroutine LBCLAT(IX, JX, IWMX, WORK, F, SMAP, XLON, XLAT, POINT)

Purpose: Calculate latitude, longitude, Coriolis parameter, and map scale factor at each mesoscale gridpoint. Store those fields in the storage array WORK.

On Entry:

POINT: Flag specifying whether calculations are for cross points or dot points. A Hollerith variable, equal to either 5HCROSS or 5HDOT.

IPROJ: Projection of mesoscale map. Common/OPTION/. From the Common Master Input File.

The arrays F, SMAP, XLON, and XLAT, all dimensioned by (IX,JX), are passed to LBCLAT as working space only.

Calls: Subroutine LCMIO.

Called by: Main program.

Comments: LBCLAT calculates the values of latitude (XLAT), longitude (XLON), Coriolis parameter (F), and map-scale factor (SMAP) at each gridpoint. LBCLAT then calls subroutine LCMIO four times, to transfer the four arrays to the storage array WORK, for later retrieval.

LBCLAT is called twice by the main program; the first time for calculations at dot points, the second for calculations at cross points.

Subroutine LCMIO(L, IV, B, IX, JX, IWMX, WORK, IND)

Purpose: Transfer horizontal slabs of data to and from the storage array WORK. LCMIO either puts the array B(IX,JX), containing one variable field at one level, into the storage array WORK, or retrieves array B from array WORK.

On Entry:

- L: Level number (1 through 15) of the field in array B.
- IV: Variable number (1 through 8) of the field in array B.
- B(IX,JX): Array containing the horizontal field to be transferred to or retrieved from array WORK.
- IND: Integer flag indicating whether a field should be transferred to or retrieved from array WORK.
  IND = 1: Transfer array B to array WORK.
  IND = 2: Retrieve array B from array WORK.

Calls: Subroutine LCMIO calls no other routines.

Called by: Main program and subroutines CLMSST, LBCLAT, MAPNAV, NMCLAT, and OUTAP.

Comments: Subroutine LCMIO first calculates the value NAME, a unique value for each variable number at each level, according to

$$\mathrm{NAME} = (\mathrm{IV} - 1) \times \mathrm{ICMIO} + \mathrm{L} \ ,$$

where IV is the variable number (1 through 8), L is the level number (1 through 15), and ICMIO is a constant (ICMIO = 12) representing the number of levels of one variable which can be stored.

The value NAME is used as the third index of array WORK, defining the location of the data for storage or retrieval. For example, variable number 1, at levels 1 through 12, will be stored in WORK(i,j,1) through WORK(i,j,12). Variable number 2, at levels 1 through 12, will be stored in WORK(i,j,13) through WORK(i,j,24).

Once LCMIO calculates NAME, the data are simply transferred from B to WORK, or from WORK to B, depending on the value of the argument IND.

Subroutine MAPNAV(A, C, IX, JX, IWMX, WORK, E1, E2, LMO, LDY)

Purpose: Interpolate (using function BINT) Navy sea-surface temperatures from the octagonal grid to the mesoscale grid.

On Entry: A(63,63): Navy sea-surface temperatures (northern hemisphere) on a 63 × 63 octagonal grid.

LMO, LDY: Month and day of the sea-surface temperature observations.

The arrays E1(IX,JX) and E2(IX,JX) are passed to MAPNAV as working space only.

On Exit: C(IX,JX): Navy Sea-surface temperatures, interpolated to the mesoscale grid.

Calls: Subroutine LCMIO and function BINT.

Called by: Subroutine NAVY.

Comments: MAPNAV first calls subroutine LCMIO twice to retrieve the longitudes and latitudes of the mesoscale gridpoints (cross) from the storage array WORK. Then for each mesoscale gridpoint, MAPNAV calculates XX and YY, the index values (interpolation location) of the gridpoint in the coordinate system of the octagonal grid. MAPNAV then calls BINT to interpolate sea-surface temperatures to that point. Middle and late summer sea-surface temperatures less than 0.5°C are set to 0.5°C. The analysis interpolated to the mesoscale grid is returned in array C to subroutine NAVY.

Subroutine MSGFIL(IWZ, V1)

Purpose: Interpolate temporally to generate a missing time period from the NMC input analyses.

On Entry:

ISEQ: Sequence number of the first missing data set. Common/INT/. From the DATAGRID Local Master Input File.

JSEQ: Number of consecutive missing data sets. Common/INT/. From the DATAGRID Local Master Input File.

JHEMIS: Hemisphere from which data are needed. Common/INT/.
0: Northern Hemisphere.
1: Southern Hemisphere.
2: Both hemispheres.

PTOP: Pressure level defining top of the model. Common/INT/. From the Common Master Input File.

IWZ(675000): Working space.

V1(145,37,125): Working space to hold gridded data.

Calls: Subroutines RON84 and UON84.

Called by: Subroutine SORTNM.

Comments: Subroutine MSGFIL interpolates from data already sorted and written to FORTRAN unit 1 (but not unpacked) by subroutine SORTNM. MSGFIL reads the packed data from unit 1 using subroutine RON84, and unpacks the data using subroutine UON84. ISEQ and JSEQ define the missing data sets. The temporal interpolation is performed, and the interpolated data are written temporarily to FORTRAN unit 7. The fields are reordered, and finally written to unit 11 for later retrieval.

Subroutine NAVY(C, IX, JX, IWMX, WORK, E1, E2)

Purpose: Access Navy or climatological sea-surface temperatures and interpolate them to the mesoscale grid.

On Entry:

ISST: Logical flag indicating whether climatological sea-surface temperatures are to be used. Common/OPTION/. From the DATAGRID Local Master Input File.

	JMOCLM(18):	Month for which sea-surface temperatures are needed. Common/OPTION/. JMOCLM = JMO from the Common Master Input File.
	IFILE:	The sequence number of the particular date/time being processed. Common/OPTION/.
	MDATE(18):	Date of the time periods being processed. Common/OPTION/.
	MISNV1:	Time of the first missing Navy dataset. Common/OPTION/. From the DATAGRID Local Master Input File.
	MISNV2:	Time of the second missing Navy dataset. Common/OPTION/. From the DATAGRID Local Master Input File.
	Arrays E1(IX,JX) and E2(IX,JX) are passed to NAVY as working space.	
On Exit:	C(IX,JX):	Navy or climatological sea-surface temperatures interpolated to the mesoscale grid.
Calls:	Subroutines CLMSST, RNAVY, UNAVY, and MAPNAV.	
Called by:	Subroutine OUTAP.	
Comments:	Unit 3 contains the Navy sea-surface temperatures for the Northern Hemisphere, on a 63×63 octagonal grid.	

Unit 12 contains the global climatological sea-surface temperatures, with $2°$ latitude-longitude resolution.

If the user requests climatological sea-surface temperatures, NAVY reads the data directly from unit 12 and calls subroutine CLMSST to interpolate them to the mesoscale grid. The interpolated fields are then returned in array C to subroutine OUTAP.

If the user requests Navy sea-surface temperatures, NAVY first calls RNAVY to read (from unit 3) the NAVY sea-surface temperature file and extract the header information. NAVY then calls UNAVY to unpack the analysis. Finally, NAVY calls MAPNAV to interpolate the data to the mesoscale grid. The interpolated fields are then returned in array C to subroutine OUTAP.

If Navy data for one or two dates are missing (as indicated by the user with options MISNV1 and MISNV2 in the DATAGRID Local Master Input File), subroutine NAVY uses the data from one day earlier.

Subroutine NMCLAT(IX, JX, IWMX, WORK, XLON, XLAT, E, POINT, LONBNT, LATBNT, IFIRST)

Purpose: Interpolate input analyses from the latitude-longitude grid to the mesoscale grid, using function BINT.

On Entry:

A1(145,37)	Input analysis from NMC or ECMWF, Northern Hemisphere. Blank common.
A2(145,37):	Input analysis from NMC, Southern Hemisphere. Blank common.
DATUNI(73,73):	Input analysis from Unidata, global. Blank common.
POINT:	Hollerith flag indicating cross or dot point calculations, 5HCROSS or 5HDOT.
LONBNT:	Longitudinal dimension of the original analysis.
LATBNT:	Latitudinal dimension of the original analyses.
IFIRST:	Source of input fields; IFIRST may be 3HNMC, 5HECMWF, or 6HUNIDAT.

Arrays XLON(IX,JX) and YLAT(IX,JX) are passed to NMCLAT as working space only.

On Exit: E(IX,JX): Interpolated field on the mesoscale grid.

Calls: Subroutine LCMIO and function BINT.

Called by: Main program.

Comments: Subroutine NMCLAT is called for each variable, at each level, and at each time period.

First, NMCLAT calls subroutine LCMIO twice to retrieve the latitudes and longitudes of the mesoscale gridpoints. Then NMCLAT enters two DO-loops that cycle through all mesoscale gridpoints.

For each gridpoint, NMCLAT calculates XX and YY, the index values (interpolation location) of the mesoscale gridpoint in the coordinate system of the input analysis, and calls function BINT to interpolate to the gridpoint.

Once the interpolation has been completed for all points on the mesoscale grid, the interpolated field is returned in array E to the main program.

Function **ONED**(X, A, B, C, D)

Purpose: Perform a one-dimensional interpolation to a point from four flanking points, two on either side (see discussion in Section III.A).

On Entry:

X:	Distance of interpolation location from the point B. See Fig. 3.1.
A,B,C,D:	Values at the four flanking points, from which the fifth value is interpolated.

On Exit:

ONED:	The interpolated value at the interpolation location.

Calls: Function ONED calls no other subprograms.

Called by: Function BINT.

Subroutine **OUTAP**(IX, JX, IWMX, WORK, XLON, E1, E2, G2, TER, LAND, FLND, SNW, KSNGEN, JYR, JMO, JDY, JHR, LPR, IVL, NPRH, IFIRST, TS, P0, F, IFCST, SSTUNI)

Purpose: Retrieve the final interpolated fields from array WORK and from temporary storage arrays, and write them to FORTRAN unit 4 (main output file). The output is discussed in greater detail in Section III.B.4.

On Entry:

KSNGEN:	Logical flag to indicate whether a subjectively generated snow-cover field is used.
JYR(18):	JYR from the Common Master Input File.
JMO(18):	JMO from the Common Master Input File.
JDY(18):	JDY from the Common Master Input File.
JHR(18):	JHR from the Common Master Input File.
NPRH:	Number of pressure levels (*i.e.* 15).
LPR(NPRH):	Array of pressures.
IVL(10,NPRH):	Flag indicating whether a field exists (IVL = 1) or not (IVL = 0).
IFIRST:	IFIRST from the DATAGRID Local Master Input File.
IFCST:	Forecast hour of Unidata input fields.
SSTUNI(IX,JX):	Unidata sea-surface temperatures on the mesoscale grid.

Arrays XLON, E1, E2, G2, TER, LAND, FLND, SNW, TS, P0, and F (all dimensioned by IX,JX) are passed to the array as working space only.

Calls: Subroutines OUTPT, LCMIO, TV2T, NAVY, SFCGEO, and VECT.

Called by: Main program, once for each time period processed.

Comments: Most of the fields written out are retrieved from array WORK, where they have been stored.

Terrain and land-use fields are read from unit 9 (output from program TERRAIN).

Snow-cover is either retrieved from the storage array WORK or from unit 2 (the user-supplied snow-cover data).

Subroutine NAVY is called to access and interpolate Navy or climatological sea-surface temperatures. If Unidata sea-surface temperatures are used, they are passed to OUTAP through the array SSTUNI.

Surface temperature (for ECMWF or Unidata input sources) is interpolated from pressure-level temperatures.

Virtual temperatures (from NMC or Unidata input) is converted to temperature with subroutine TV2T.

Surface geostrophic u and v, adjusted for friction, are calculated in subroutine SFCGEO.

Upper-level u and v are converted from meteorological coordinates to mesoscale-map coordinates by subroutine VECT.

For each field written to unit 4, subroutine OUTPT is called to print a sample of the field or a message that the field has been written.

Subroutine OUTPT(FLD, IYY, IA, IB, INY, JXX, JA, JB, JNX, KSIGT, NAME, LPRES)

Purpose: Print out a sample of the interpolated fields (mesoscale grid.)

On Entry: LPRES: Pressure level index of the data to print.

NAME: Hollerith identifier of variable to print.

FLD(IYY,JXX): Array holding the field on the mesoscale grid.

KSIGT: Number of significant digits to be printed for each data point.

IA, IB: Initial and final sampling points in the I direction.

JA, JB: Initial and final sampling points in the J direction.

INY, JNX: Sampling intervals in the I and J directions.

Calls: Subroutine VTRAN.

Called by: Subroutine OUTAP.

Comments: The number of digits used to represent each printed value is assigned from KSIGT. The field is automatically scaled so that the largest value (absolute) will have KSIGT digits. Values more than KSIGT orders of magnitude less than the largest absolute value will be printed as a positive or negative zero. Data points may be excluded from the scaling process by setting the absolute values greater than 1.0×10^{30}. As the scaled data values are printed, labels showing the I and J indices are printed at the boundaries of the field. A caption is written above the data to identify the variable with a character string and to give the scaling power applied to the field. Subroutine VTRAN (not discussed) is used to transpose the data field array FLD, before and after the scaling and printing operations of OUTPT.

Subroutine RNAVY(NUN, NYR, NMO, NDY, NHR, NPR, NFC, NFT, NMS, NSC, NST, BSE, NFD, NER)

Purpose: Read a record containing a navy sea-surface temperature analysis, and unpack header information (year, month, day, hour).

On Entry: NUN: FORTRAN unit number of the navy files.

On Exit: NFD(1063): Buffer holding the packed record, read from unit NUN.

NYR: The year of the Navy data (two digits).

NMO: The month of the Navy data (two digits).

NDY: The day of the Navy data (two digits).

NHR: The UTC hour of the Navy data (two digits).

NER: An integer flag indicating an error in reading the record. NER is not used outside of RNAVY.

Calls: Subroutine N76SUM (not discussed).

NCAR subroutines RDTAPE, IOWAIT, and GBYTE.

NCAR function SCONV.

Called by: Subroutine NAVY.

Comments: RNAVY reads the record from unit NUN using NCAR subroutines RDTAPE and IOWAIT. RNAVY unpacks the header variables using NCAR subroutine GBYTE.

Variables NPR, NFC, NFT, NMS, NSC, NST, BSE, and NER are returned to the calling routine, but are not used outside of subroutine RNAVY.

Subroutine RON84(IUN, NBF, MXB, IST, IWDS)

Purpose: Read a packed NMC input record and unpack identifying variables preceding the data. The data are not unpacked.

On Entry: IUN: FORTRAN unit number of the NMC file.

On Exit: NBF(MXB): Buffer holding packed record.

IST: Read status. 0=OK, 1=EOF, 2=Error.

IWDS: Number of words in NBF.

/CON84/: All variables in the common block /CON84/ are returned to the calling routine. These variables identify the data in the record, and describe the data's format.

Calls: NCAR subroutines RDTAPE, RDVBS, IOWAIT, and GBYTES.

Called by: Main program, and subroutines MSGFIL and SORTNM.

Comments: Each record in the NMC archive file consists of a 12-word label (32 bits per word) followed by the the packed data. The label contains identifying information such as level, time, meteorological variable, grid type, and grid size. Other variables, such as NBSN, NBEX, NBFR, and NSCL describe unpacking parameters.

RON84 reads one record (using NCAR subroutine RDTAPE) into buffer array NBF and extracts the variables in the 12-word label. For details concerning the contents of the label and the unpacking procedures, see NMC Office Note 84.

The data are later unpacked by subroutine UON84.

Subroutine SFCGEO(IX, JX, UG, VG, HT, TS, P0, F)

Purpose: Calculate the surface geostrophic u and v, adjusted for friction.

On Entry:

- P0(IX,JX): Sea-level pressure (mb) on the mesoscale grid (cross).
- TS(IX,JX): Surface temperature (°C) on the mesoscale grid (cross).
- HT(IX,JX): Terrain height (m) on the mesoscale grid (cross).
- F(IX,JX): Coriolis parameter (s^{-1}) on the mesoscale grid (dot).

On Exit:

- UG(IX,JX): Surface geostrophic u (m s^{-1}), adjusted for friction, on the mesoscale grid (dot).
- VG(IX,JX): Surface geostrophic v (m s^{-1}), adjusted for friction, on the mesoscale grid (dot).

Calls: Subroutine SFCGEO calls no other subprograms.

Called by: Subroutine OUTAP.

Comments: The geostrophic u and v are calculated by

$$u_g = -\frac{1}{\rho_s f}\frac{\partial p_o}{\partial y} \quad \text{and} \quad v_g = \frac{1}{\rho_s f}\frac{\partial p_o}{\partial x}, \tag{3.1}$$

where f is the Coriolis parameter (s^{-1}); p_o is the sea-level pressure (mb); and surface density ρ_s ($\times 10^{-2}$ kg m^{-3}) is approximated by

$$\rho_s = \frac{p_o}{R_d T_s}\left(\frac{44330 - z_s}{44330}\right), \tag{3.2}$$

where R_d is the gas constant of dry air, T_s is the surface temperature (K), and z_s is the terrain height (m).

The geostrophic wind is then adjusted for friction over land by rotating it cyclonically 25° and reducing the speed to 50 %, and over ocean by rotating it 10° and reducing the speed to 70 %.

Subroutine SFGPRS(IFH, FH, FT)

Purpose:	Access Unidata fields of sea-level pressure and sea-surface temperature.	
On Entry:	IFH:	Index for forecast hour (1 = 00; 2 = 6; 3 = 12; ...; 9 = 48; 10 = 60).
On Exit:	FH:	The forecast hour of the fields.
	FT:	The forecast time (YYMMDDHHmm).
	P(73,73):	Sea-level pressure (mb). Common/GRDATA/.
	SST(73,73):	Sea-surface temperature (°C). Common/GRDATA/.
Calls:	Subroutines SORT (not discussed) and DTIME (not discussed).	
	NetCDF subroutines NCVGT, NCVGTC, NCVGT1 and NCCLOS.	
Called by:	Subroutine GETUNI.	
Comments:	Subroutine SFGPRS retrieves the fields of sea-level pressure and sea-surface temperature for a single forecast time from the Unidata files. SFGPRS is called by GETUNI for each forecast time to process. SFGPRS makes use of an number of netCDF subroutines to retrieve the data, as well as two other DATAGRID subroutines, SORT (to sort a list of forecast hours) and DTIME (to create integers representing dates). These two subroutines are not discussed.	

Subroutine SFGRID(IFH, IP, PRS)

Purpose:	Access Unidata analyses at pressure levels.	
On Entry:	IFH:	Index indicating forecast hour (1 = 00; 2 = 06; 3 = 12; ... 9 = 48; 10 = 60).
	IP:	Index indicating pressure level (1 through 10, representing mandatory levels from 1000 mb to 100 mb).
On Exit:	PRS:	The pressure level (mb) of the analyses.
	U(73,73):	Analysis of u-component (m s^{-1}). Common/GRDATA/.
	V(73,73):	Analysis of v-component (m s^{-1}). Common/GRDATA/.
	T(73,73):	Analysis of T (°C). Common/GRDATA/.

RH(73,73):	Analysis of RH (%). Common/GRDATA/.
H(73,73):	Analysis of height (m). Common/GRDATA/.

Calls: NetCDF subroutines NCVGT, NCVGT1, and NCCLOS.

Called by: Subroutine GETUNI.

Comments: Subroutine SFGRID is called (by GETUNI) for each forecast time and each pressure level, to retrieve the pressure-level fields of T, u, v, RH, and height. It makes use of several netCDF subroutines, not discussed in this document.

Subroutine SORTNM(INAVY, IHEMIS, IFILES, IFILE1, IMSG, ISNOW1, ISNOW2, ISNOW3, IUNEC, SNOECM, SARRAY)

Purpose: Access NMC data, write the desired fields to temporary files for later retrieval, and interpolate to generate missing NMC data sets (subroutine MSGFIL). The fields are sorted according to time period, level, and variable type, and written temporarily to FORTRAN unit 1.

On Entry:

INAVY:	Logical flag, .TRUE. indicates that Navy or climatological sea-surface temperatures are to be used; .FALSE. indicates that NMC sea-surface temperatures are to be used.
IHEMIS:	Integer flag indicating which hemispheres are needed. From the DATAGRID Local Master Input File.
IFILES:	Number of time periods to be processed. From the Common Master Input File.
IFILE1:	Hollerith flag indicating source of the input analyses: 3HNMC, 5HECMWF, or 6HUNIDAT. From the DATAGRID Local Master Input File.
IMSG:	Logical flag; .TRUE. indicates that missing NMC data are to be interpolated, .FALSE. indicates that missing NMC data remain missing. From the DATAGRID Local Master Input File.
ISNOW1:	First NMC snow-cover date. From the DATAGRID Local Master Input File.
ISNOW2:	Second NMC snow-cover date. From the DATAGRID Local Master Input File.

ISNOW3:	Logical flag indicating whether first NMC snow-cover date is on a separate file. From the DATAGRID Local Master Input File.
IUNEC:	Flag indicating whether input is from NMC (IUNEC = 0) or from either Unidata or ECMWF (IUNEC = 1).
SNOECM:	Snow-cover date for ECMWF data. From the DATAGRID Local Master Input File.

On Exit: SARRAY(145,37): The NMC snow-cover data for ECMWF or Unidata input.

Calls: Subroutines RON84, INIT, and MSGFIL.

NCAR subroutines WRTAPE and IOWAIT.

Called by: Main program.

Comments: For ECMWF or Unidata input using NMC snow-cover data, subroutine SORTNM accesses the snow-cover field and puts it into array SARRAY(145,37). SORTNM then returns processing to the main program.

For NMC input, the first snow-cover field is written temporarily to unit 7; the second to unit 8. Packed data records, not including the snow-cover data, are read, sorted, and written to FORTRAN unit 20. Snow-cover and sea-surface temperature data are inserted into the final sequence. All sorted data records, still in packed format, are written to temporary unit 1 for later retrieval.

For ECMWF input, the external program D_ECMWF performs tasks similar to those in SORTNM.

Subroutine TOGA2DG(WORK, IMX, JMX, IWX, IUNIT, IUNIT2, IPRINT)

Purpose: Subroutine TOGA2DG serves as a driver program which calls routines which read the ECMWF/TOGA analyses, interpolates them to the mesoscale grid, and writes them to the DATAGRID output file.

On Entry:

WORK (IMX,JMX,IWX):	Working space to hold the data on the mesoscale grid.
IUNIT:	FORTRAN unit number of the file from which ECMWF/TOGA data are read. IUNIT = 31.

IUNIT2: FORTRAN unit number of the file to which the interpolated analyses (on the mesoscale grid) are written. IUNIT2 = 4.

IPRINT: Flag to activate print statements

Calls: Subroutines TOGA2XY, SORTTOGA, XYGRID, PRTDATA, and OUTTOGA.

Called by: Main program

Comments: Subroutine TOGA2DG performs for ECMWF/TOGA data the tasks of reading the analyses, interpolating them to the mesoscale grid, and writing them to the DATAGRID output file. It is called by the main program only if ECMWF/TOGA analyses are requested. If TOGA2DG is called, much of the main program is bypassed, since nearly all of the necessary steps are performed by the routines called from TOGA2DG.

TOGA2DG first calls TOGA2XY to compute the index values of each of the mesoscale gridpoints in the coordinate system of the ECMWF/TOGA data, for both dot points and cross points. TOGA2DG then enters a loop which cycles through the total number of time periods

For each time period, TOGA2DG calls four subroutines:

- SORTTOGA, to retrieve the ECMWF/TOGA analyses from the input file;
- XYGRID, to interpolate the ECMWF/TOGA analyses to the mesoscale grid;
- PRTDATA, to print samples of the input ECMWF/TOGA analyses on the latitude/longitude grid;
- OUTTOGA, to write the interpolated analyses to the DATAGRID output file, FORTRAN unit 4.

Once the loop has processed all time periods, TOGA2DG ends and processing returns to the main program.

Subroutine TV2T(TV, RH, P, T, IX, JX, PMAN)

Purpose: Converts virtual temperature (TV) to temperature (T).

On Entry: RH(IX,JX): Relative humidity (%).

	TV(IX,JX):	Virtual temperature (°C).
	P(IX,JX):	Pressure (mb).
	PMAN:	For surface fields, PMAN = 0.0; for mandatory-level fields, PMAN is the pressure at the mandatory level.
On Exit:	T(IX,JX):	Temperature (°C).

Calls: Subroutine TV2T calls no other subprograms.

Called by: Subroutine OUTAP.

Comments: Since the ECMWF files archive temperature and not virtual temperature, TV2T is called only for NMC or Unidata input.

Subroutine TV2T performs the conversion for each point on the mesoscale grid through an iterative procedure. Virtual temperature is first used as a first guess ($T_1 = T_v$). Saturation vapor pressure is estimated by

$$e_s = 6.112 \times \exp\left[\frac{17.67(T_1 - 273.15)}{T_1 - 29.65}\right]. \tag{3.3}$$

Saturation mixing ratio q_s is then estimated by

$$q_s = \frac{\varepsilon e_s}{p - e_s}. \tag{3.4}$$

where ε is equal to 0.622, and p is pressure (mb). Mixing ratio w is estimated by

$$w = RH \times q_s, \tag{3.5}$$

where RH is relative humidity. Finally, a second guess temperature T_2 is calculated by :

$$T_2 = \frac{T_1}{1 + 0.61w}. \tag{3.6}$$

The process is then repeated, using T_2 as the current guess rather than T_v, until the absolute difference between two successive iterations is less than 0.1°C.

Subroutine UNAVY(NFD, AR)

Purpose: Unpack the Navy sea-surface temperature record.

On Entry: NFD(1063): Buffer array holding data to be unpacked.

On Exit: AR(3969): Array holding unpacked data (63 $\times$ 63 grid).

Calls: NCAR subroutines GBYTES and GBYTE.

Called by: Subroutine NAVY.

Comments: UNAVY calls NCAR subroutine GBYTE twice to extract from the buffer array NFD two unpacking parameters IBI and ISC. UNAVY calls NCAR subroutine GBYTES to extract the packed data from NFD and store it in array AR, still in packed format. The packed data in array AR are unpacked and restored to AR. AR returns the unpacked data to the calling routine (subroutine NAVY).

Subroutine UNIMSG(IU, JU, IUU, JUU, NREC, NTIMES, NMSG, DUM2, UNIT41, UNIT31, GFLAG, NPRINT)

Purpose: Examine the Unidata analyses for missing data, and interpolate missing data either temporally or vertically. The final set of Unidata analyses are written to FORTRAN unit 41.

On Entry:

IU, JU: Parameters dimensioning the Unidata analyses. IU and JU are both equal to 73.

IUU,JUU: I and J indices of the center point of the Unidata analyses. IUU and JUU are both equal to 37.

NREC: Number of records in the set of Unidata analyses for each time period. NREC = 48 (10 fields each of u, v, T, and height; 6 fields of RH; and 1 field each of sea-level pressure and sea-surface temperature).

NTIMES: Number of forecast times of Unidata analyses. NTIMES = 10 (for the 00, 06, 12, 18, 24, 30, 36, 42, 48, 60 hour forecasts).

NMSG: The maximum number of "bad" fields allowed before the program execution is halted. NMSG = 30.

DUM2(IU,JU): Working space to hold individual analyses.

UNIT41: The FORTRAN unit number of the file to which the final set of Unidata analyses is written. UNIT41 = 41.

UNIT31: The FORTRAN unit number of the file to which the Unidata analyses were written (by subroutines SFGPRS and SFGRID), and from which UNIMSG reads the analyses. UNIT31 = 31.

GFLAG: Integer flag which indicates bad data. GFLAG = −999.

NPRINT: Logical flag to activate printout of Unidata information. The value set in the program is .FALSE.

On Exit: FORTRAN unit UNIT41 holds the final set of Unidata analyses.

Calls: Subroutine UNIMSG calls no other subprograms.

Called by: Subroutine GETUNI.

Comments: UNIMSG first examines each analyses for bad data. If it finds one bad data value in the western half of the northern hemisphere, UNIMSG labels the field as bad. Then for each bad field, an attempt is made to create a new field by temporal interpolation from bracketing fields. If this interpolation cannot be performed, an attempt is made to create a new field by vertical interpolation. Once all necessary fields have been interpolated, UNIMSG cycles through all fields (48 fields for each of 10 forecast periods). Each field is read from unit 31, and if the field was not previously found to be bad, it is written to unit 41. If, however, the field was previously found to be bad, the new interpolated field is written to unit 41. Thus, when UNIMSG has finished, all the fields which DATAGRID expects from subroutine GETUNI have been written to FORTRAN unit 41.

Subroutine UON84(NBF, DATA, NERR)

Purpose: Unpack data from the buffer array NBF, returned by RON84.

On Entry: NBF: Buffer array as read by RON84; NBF holds the data to be unpacked.

NIDIM: I dimension of gridded data in NBF. Common/CON84/.

NJDIM: J dimension of gridded data in NBF. Common/CON84/.

NJJ: Total number of gridpoints. Common/CON84/.

	NK:	Integer flag identifying the grid type. Common/CON84/.
On Exit:	DATA:	Array containing the unpacked data.
	NERR:	An integer flag, returned with a value of 0.

Calls: NCAR Subroutine GBYTES.

Called by: Main program and subroutines MSGFIL and SORTNM.

Comments: The packed data in buffer array NBF are unpacked, using the values BASE and NSCL (both in common/CON84/). For information concerning the packing and unpacking of the data, see NMC Office Note 84: "Packing and identification of NMC grid-point data."

Subroutine VECT(IX, JX, XLON, E1, E2).

Purpose: Convert u and v in meteorological coordinates to u and v in the coordinate system of the mesoscale map.

On Entry:	XLON(IX,JX):	Longitudes of the points of the mesoscale grid.
	E1(IX,JX):	u, in meteorological coordinates.
	E2(IX,JX):	v, in meteorological coordinates.
On Exit:	E1(IX,JX):	u, in mesoscale-map coordinates.
	E2(IX,JX):	v, in mesoscale-map coordinates.

Calls: Subroutine VECT calls no other subprograms.

Called by: Subroutine OUTAP.

D. Program D_ECMWF

The program D_ECMWF performs for the ECMWF input files many of the same operations that subroutine SORTNM in program DATAGRID does for NMC files: it accesses the input analyses, sorts them, and writes them in a standard format to a separate file.

Program D_ECMWF and the script that executes it, D_ECMWF.DECK, probably do not require any user modification. Both are executed automatically by the DATAGRID script as needed. Program D_ECMWF takes as input the Common Master Input File (See Section III.B.3) and the 0000 and 1200 UTC ECMWF files (up to two of each). These ECMWF files must be listed in the DATAGRID script by shell variables `InAnly00` and `InAnly12`. D_ECMWF searches through the input files, extracts the times needed for DATAGRID, and writes them to FORTRAN unit 3, (called `firstgA` by the DATAGRID script). This file is later used as input to program DATAGRID.

D_ECMWF code

For each date/time period processed, D_ECMWF calls subroutine RDECM to read the packed ECMWF record and unpack identification variables to be returned in common block /CMECM/. If the identifiers match the required conditions of time, date, variable type, etc., subroutine UPECM is called to unpack the data record. The data field is returned in the array D(144,73), containing the global analysis with 2.5° latitude-longitude resolution. The Northern-Hemisphere portions of the data are transferred to three-dimensional arrays, which hold the data for one variable and one time period at all levels. Subroutine INTERP then interpolates each variable (ϕ, T, u, v, and RH) to the 400, 250, and 150 mb pressure levels, which DATAGRID expects but the ECMWF archives do not supply. Sea-level pressure is calculated from temperature and geopotential height data. The 1000 mb temperature is substituted for surfaced temperature. Finally, the output is written to FORTRAN unit 3, in the order and format expected by DATAGRID.

For each variable, a header record is written first, followed by the data records (written in subroutine WRTDK). The data records written are sea-level pressure and the 1000 mb temperature, followed by each level of ϕ, T, u, v, and RH. Above 300 mb, RH is not written.