Chapter

8

Processing Data with Nimbus

Overview

Nimbus is the software used to accomplish data processing. It provides both post processing and real time data processing with Sequel Query Language (SQL) support both. **Nimbus** produces output netCDF files of data in engineering units suitable for analysis. See “Processing Data with Nimbus” on page 8-1. **Nimbus** is intended to be used by the Project Manager or Scientist responsible for analyzing the project’s data.

Nimbus is written using a Motif-based Graphical User Interface (GUI). Output files are generated through the netCDF API that produces netCDF files. **Nimbus** supports a multi-rate output format (different variables may be output at different rates). **Nimbus** reports an error if the data format on the disk is not correct or if disk media errors are detected.

Processing of non-analog/digital channels (GPS, IRS, PMS 1D) and derived variables is accomplished in **Nimbus** by using a library of routines called **`AMLIB`**. The source code is flexible and implementation of new probes and derived variable calculations can be accomplished.

Understanding Nimbus Features

Nimbus has the following features:

❏ Low rate (LRT)	Processes all data averaged to 1 sample per second (sps).
❏ High rate (HRT)	Processes all data interpolated or decimated to 25 sps.
❏ Time Lags	Describes the variable’s data delay from the reference time in milliseconds.
❏ Despike	Replaces a “defective” data point with one that is reasonable.
❏ Production Mode	Final processing suitable for archiving and distribution.
❏ Non-production Mode	Review the data, perform preliminary checking and set up for production and data analysis.
❏ Real Time Mode	Processing data in real time.
❏ Asynchronous Data Handling	Processes data that are not synchronized with the data system’s internal clock and can come at random times.
❏ Data Modification	Ability to modify calibration coefficients and dependencies.
❏ Sampled Rate Output	Ability to output variables at their sampled rate.
❏ Multi-rate Digital Filter	Converts data sampled at any rate to 25 sps.
❏ Data Input	Ability to input data from disk file.

Using Nimbus in a Non-Production Mode

The instructions in the following sections are specific to using **Nimbus** in a “non-production” mode. For information outlining the differences between “non-production” and “production” instructions, refer to “Understanding the Difference Between Production and Non-Production Mode” on page 8-24. **Nimbus** can run also in real-time when started up by another program, such as **ads_director**. If you are installing **Nimbus** run it in real time by using one of following options:

-h Display “usage” message.

-b Run in batch mode, which is described in “Running Nimbus in Batch Mode” on page 8-26.

-r Run in “raw” mode with no despiking or time-shifting and output at sample rate.

-rt Run in real-time mode for ADS2.

-rt3 Run in real-time mode for ADS3.

-x Run in conjunction with real-time mode, create SQL packets and insert them into the LDM queue for transmission to the ground. The list of variables to place in these packets is defined in **${PROJ_DIR}/*project_name*/*platform*/groundvars.**

-n Does not load any existing Production Setup files.

STARTING NIMBUS

Nimbus is designed for X-windows. If X-windows is not active on your workstation, you can run **Nimbus** in a limited batch mode only. See “Running Nimbus in Batch Mode” on page 8-26.

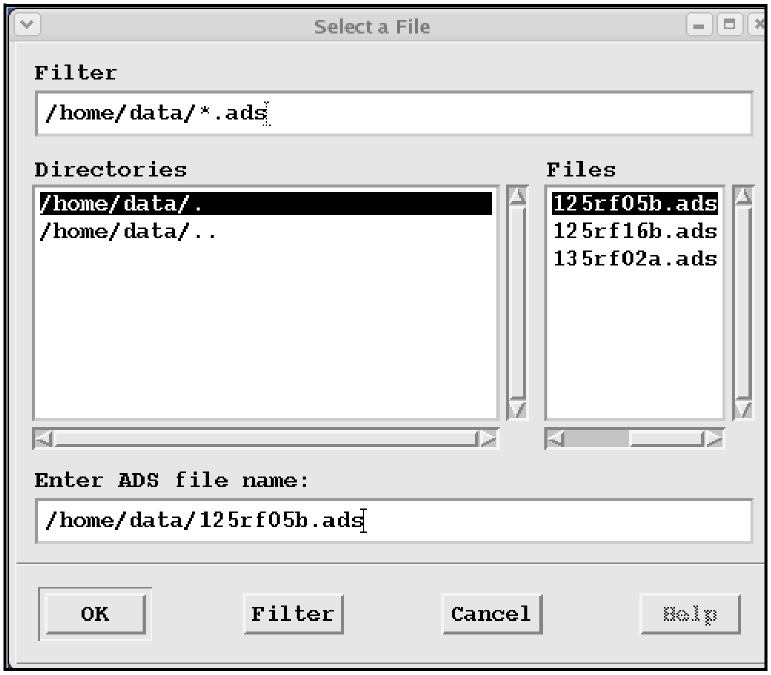
1. Before running **Nimbus**, ensure that the DISPLAY environment variable is set on your workstation. To check whether the variable is present, enter the following command at the system prompt: `echo $DISPLAY`

2. If the environment variable is not present, enter the following command:

   ```
   setenv DISPLAY <your workstation name>:0
   ```

3. Run the **Nimbus** program by entering `nimbus` at the system prompt. If you want to start **Nimbus** without automatically loading the production setup file, enter `nimbus -n` at the system prompt and the Select a File window is displayed as shown in Figure 8-1.

Figure 8-1 Select a File Window



Select a File

Filter: /home/data/*.ads

Directories:
/home/data/.
/home/data/..

Files:
125rf05b.ads
125rf16b.ads
135rf02a.ads

Enter ADS file name: /home/data/125rf05b.ads

NOTE: The ***Help*** *button is not functional.*

4. Select a directory in the **Directories** field or enter the directory path in the **Filters** field. Press the **Enter** key or the **Filter** button to display the ADADS data files in the directory path.

NOTE: If you choose to enter the full path name of the ADADS data file saved on disk. By default, at RAF, ADADS files are normally kept in the **`/home/data`** *directory. You can configure the default path by setting the* **`DATA_DIR`** *environment variable.*

5. Press the **Tab** key to move to the **Files** field, or click **LEFT** on the desired field, and enter a file to select.

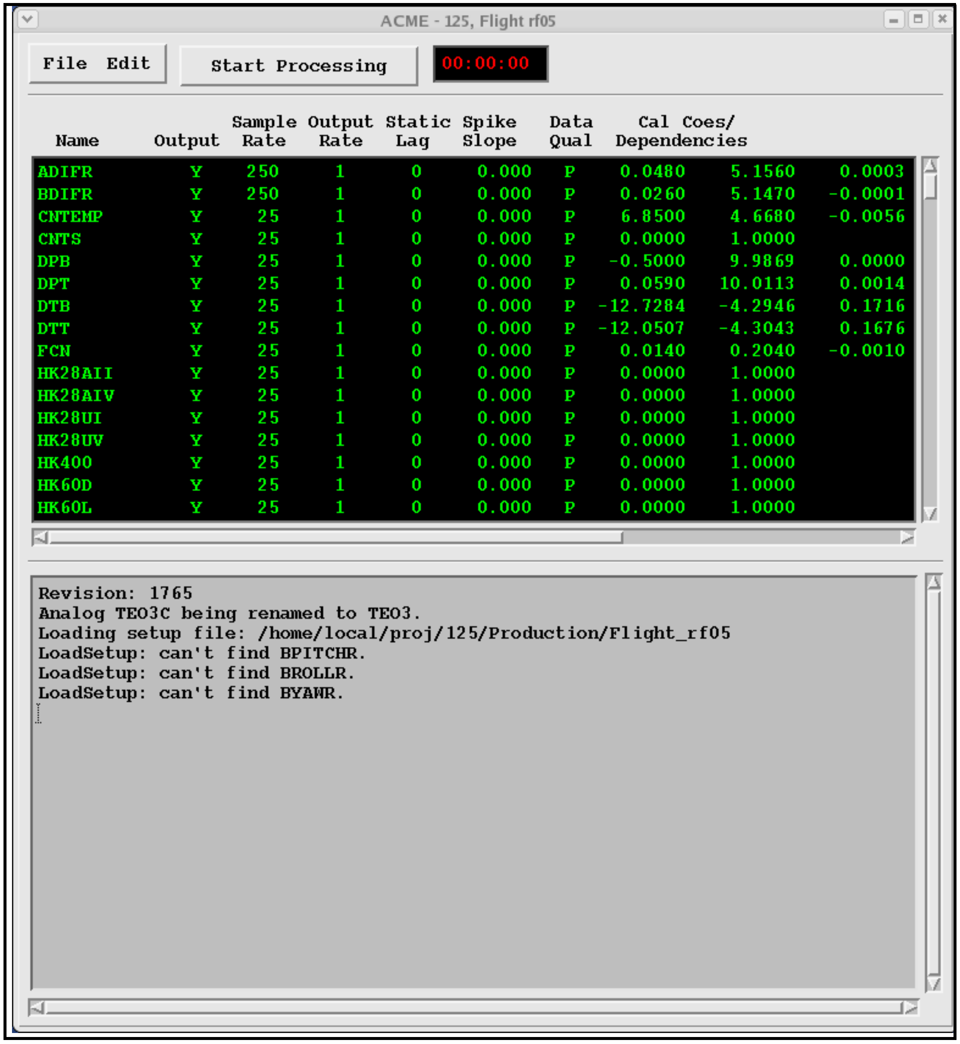
NOTE: ***Nimbus*** *automatically appends the* **`.cdf`** *extension to the file name you enter. By default, the output netCDF file is stored in the* **`/home/data`** *directory. You can configure the default path by setting the* **`DATA_DIR`** *environment variable.*

6. Click on the **OK** button.

Nimbus reads the input data header and obtains the project number and flight number from the input file.

After **Nimbus** completes the preceding steps, and if they are successful, the Setup window is displayed as shown in Figure 8-2 on page 8-6.

Figure 8-2 Setup Window



Name	Output	Sample Rate	Output Rate	Static Lag	Spike Slope	Data Qual	Cal Coes/ Dependencies		
ADIFR	Y	250	1	0	0.000	P	0.0480	5.1560	0.0003
BDIFR	Y	250	1	0	0.000	P	0.0260	5.1470	-0.0001
CNTEMP	Y	25	1	0	0.000	P	6.8500	4.6680	-0.0056
CNTS	Y	25	1	0	0.000	P	0.0000	1.0000	
DPB	Y	25	1	0	0.000	P	-0.5000	9.9869	0.0000
DPT	Y	25	1	0	0.000	P	0.0590	10.0113	0.0014
DTB	Y	25	1	0	0.000	P	-12.7284	-4.2946	0.1716
DTT	Y	25	1	0	0.000	P	-12.0507	-4.3043	0.1676
FCN	Y	25	1	0	0.000	P	0.0140	0.2040	-0.0010
HK28AII	Y	25	1	0	0.000	P	0.0000	1.0000	
HK28AIV	Y	25	1	0	0.000	P	0.0000	1.0000	
HK28UI	Y	25	1	0	0.000	P	0.0000	1.0000	
HK28UV	Y	25	1	0	0.000	P	0.0000	1.0000	
HK400	Y	25	1	0	0.000	P	0.0000	1.0000	
HK60D	Y	25	1	0	0.000	P	0.0000	1.0000	
HK60L	Y	25	1	0	0.000	P	0.0000	1.0000	

```
Revision: 1765
Analog TEO3C being renamed to TEO3.
Loading setup file: /home/local/proj/125/Production/Flight_rf05
LoadSetup: can't find BPITCHR.
LoadSetup: can't find BROLLR.
LoadSetup: can't find BYAWR.
```

USING THE NIMBUS SETUP WINDOW

The **Nimbus** Setup window is divided into the following areas:

❑	Window Manager Border	Displays the Project number and Flight number of the current input file in the top line of the window.
❑	Menu Bar	Displays buttons used to perform **Nimbus** functions below the Window Manager Border.

❑ Time Box	Displays the ADADS time, which is updated after each minute of data has been processed, in the upper right corner of the window, next to the Menu Bar.
❑ Message Box	Displays informational and system messages in the grey box located at the bottom of the window.
❑ Column Headings/ Variable Parameters	Lists the project-specific variables below the Menu Bar.

Table 8-1 shows the Setup window column headings and descriptions.

Table 8-1 Setup Window Headings and Descriptions

Column Heading	**Description**
Name	The name of the variable.
Output	Shows whether or not the variable's data is output to the netCDF file, **Y** (Yes) or **N** (No). ***NOTE:*** *If an asterisk (*) appears before the value of* **Y** *or* **N***, at least one parameter, and perhaps more than one, associated with the variable has been changed from its original value. You can find the original values in the ADADS header or in the project files:* **Defaults***,* **DependTable** *and/or* **LowRatVars***.*
Sample Rate	The input sample rate for each variable in sps.
Output Rate	The output rate for each variable in sps.
Static Lag	The lag for each variable in milliseconds by which its data are delayed from the reference time (range is ± 1000 ms).
Spike Slope	Operates on the sample data rate, which can change between one sample and the next. You can change these values by editing the **despike** file located in the **${PROJ_DIR}/xxx** directory, where **xxx** is the project number
Cal Coes/ Dependencies	This represents two types of variables: raw and derived. For raw variables, the column lists calibration coefficients, if any exist. For derived variables, the column lists dependencies.

The variables in the Setup window are organized into three categories, respectively:

1. Raw Variables

   Raw variables include all data acquired by the data acquisition system and archived to disk. Raw variables include analog, digital and serial data.

2. Derived Variables

   A derived variable is calculated using some combination or raw and other derived variables. For example, **`TASF`**, a true airspeed, is normally calculated using the variables **`QCFC`** (a dynamic pressure), **`PSFDC`** (a static pressure) and **`TTB`** (a total temperature). **Nimbus** includes the **`amlib.fns`** file in the **`/home/local/raf/nimbus/include`** directory that calculates the value of every derived variable.

All the variables are displayed in one continuous list. Each category of variables is sorted alphabetically and displayed.

UNDERSTANDING THE MENU BAR

The Setup window contains a series of buttons which are displayed in an area of the window referred to as the Menu Bar. The buttons, along with their descriptions and sub-menus, are outlined in Table 8-2:.

Table 8-2 Setup Window Menu Bar

Button	Sub-menu	Description
File		Set up processing for the selected data set.
	Load setup **Ctrl-L**	Load an existing setup file.
	Save setup **Ctrl-S**	Save the current setup under a new file name, except when in “production” mode where the file is automatically saved in a predefined file name.
	Print setup **Ctrl-P**	Send the current setup to a designated printer.
	Synthetic Data File	Fixes a raw data value in order to test **`AMLIB`** derived calculations.
	Quit **Ctrl-Q**	Exit the **Nimbus** program.
Edit		Edit the setup parameters.

Table 8-2 Setup Window Menu Bar

Button	Sub-menu	Description
	Edit Defaults **Ctrl-D**	Edit the entries in the **`Defaults`** file. The user can edit values in the file, but it does not edit the file itself. Any changes are stored within the **Nimbus** program unless saved in a setup file.
	Edit Configuration	Change the flight information, time period, processing rate or processing options.
	Toggle output **Ctrl-O**	Determines if the output is sent to a netCDF file (**`Y`**) or not (**`N`**).
	Toggle rate **Ctrl-R**	Toggles the Output Rate to 1 or 250.
	Toggle probe	Pull-down menu to toggle probes (for example, IRS_INS1, GARMIN), Corrected Winds, Fluxes, All On or All Off. It turns on/off all variables for a given probe.
Start Processing/ Pause		Process the data. Interrupt the processing.

EDITING NIMBUS VARIABLES

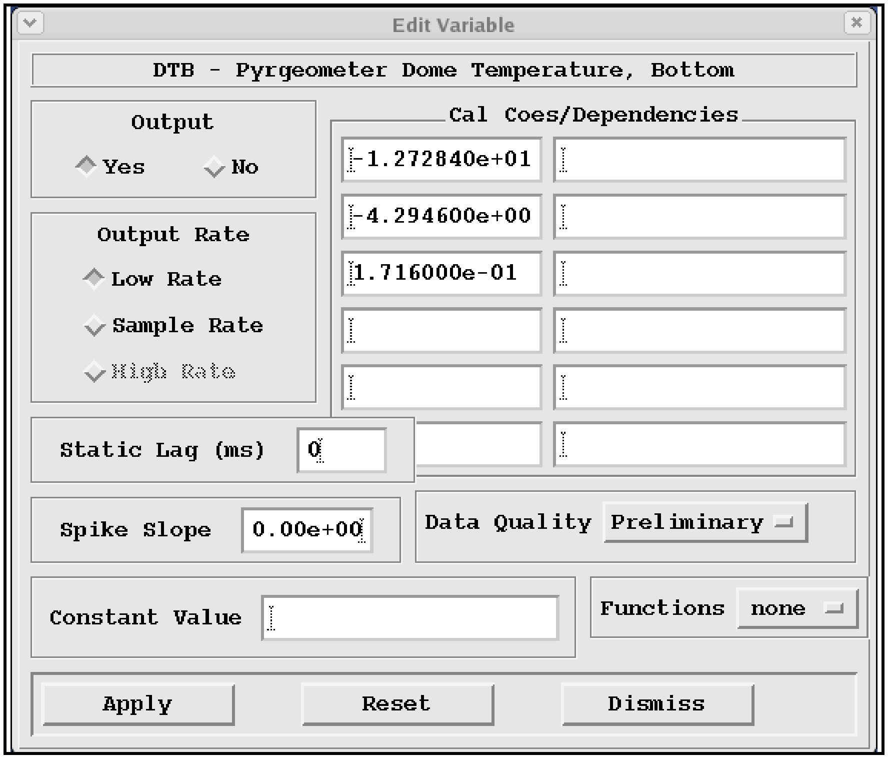
A **Nimbus** variable is any measurement, raw or derived, that has a name defined in the Setup window. The description of the variable cannot be altered, although the parameters associated with the variable (calibration coefficients, dependencies, rates, etc.) can be edited.

Edit the **Nimbus** variables by following the steps below:

1. Double-click **LEFT** on the variable to modify. If double-clicking does not work, highlight the variable (using a single click **LEFT**) and press the **Return** key.

The Edit Variable Window is shown in Figure 8-3.

Figure 8-3 Edit Variable Window



DTB - Pyrgeometer Dome Temperature, Bottom

Output: ● Yes ○ No

Output Rate: ● Low Rate ○ Sample Rate ○ High Rate

Cal Coes/Dependencies: -1.272840e+01, -4.294600e+00, 1.716000e-01

Static Lag (ms): 0

Spike Slope: 0.00e+00

Data Quality: Preliminary

Constant Value:

Functions: none

The selected variable, along with its characteristics, appear in this window. You can change **Output**, **Output Rate**, **Static Lag** and **Spike Slope**, and enter either the **Calibration Coefficients** (cal coes) or **Dependencies**. You can also set the data quality using the **Data Quality** pull-down, apply a function with the **Functions** pull-down or enter a value for the variable for the entire Nimbus run in the **Constant Value** field.

2. Make the desired modifications and click on the **Apply** button to update the Setup window with modifications.

NOTE: Clicking on the ***Reset*** *button prior to the* ***Apply*** *button restores the original value(s).*

NOTE: An asterisk (*) now appears next to the ***Output*** *flag, alerting you that the variable has been changed from its original value.*

EDITING DEFAULTS

You may wish to make changes to the values of parameters in the `Defaults` file for a specific project and flight. Reasons for modifying the default configuration include the following:

❑ To experiment with the default settings and investigate the consequences of the changes.

❑ To make necessary adjustments to ensure that the output data set is optimal.

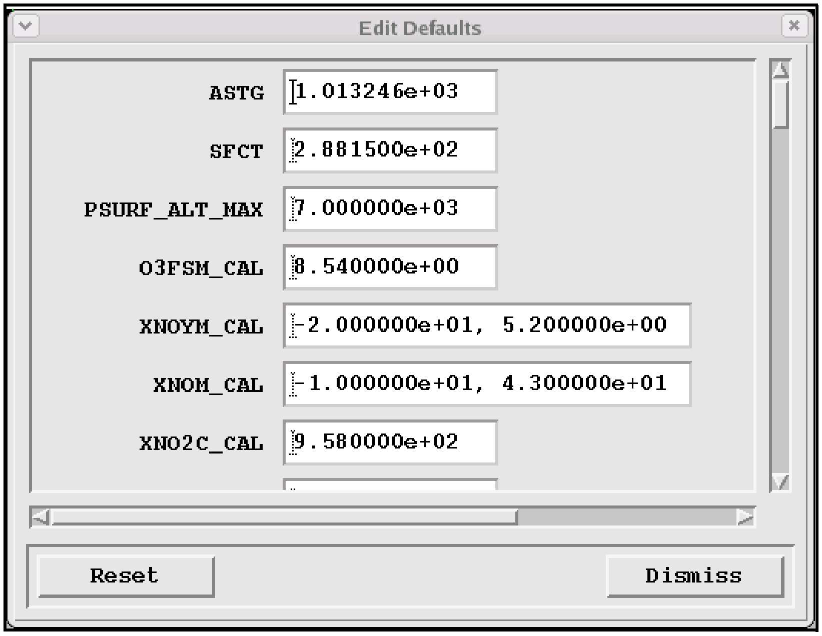
NOTE: If modifications must be made for many or all flights, edit the Defaults file as described in "Using the Defaults File" on page 8-28.

To make modifications do the following:

1. Click **LEFT** on the **Edit** button in the Setup window and select the **Edit Defaults** option, or press **Ctrl-D**.

The Setup window stays intact, and a new Edit Defaults window is displayed as shown in Figure 8-4 on page 8-12.

Figure 8-4 Edit Defaults Window



ASTG 1.013246e+03

SFCT 2.881500e+02

PSURF_ALT_MAX 7.000000e+03

O3FSM_CAL 8.540000e+00

XNOYM_CAL -2.000000e+01, 5.200000e+00

XNOM_CAL -1.000000e+01, 4.300000e+01

XNO2C_CAL 9.580000e+02

2. Make the desired changes by simply entering the new value(s) in the desired dialog box(es).

NOTE: Values entered must agree in number (for example, if 3 values exist prior to the modification, 3 values must exist after the modification). If not, ***Nimbus*** *expects the same number of data values and zero fills at the end of the record, possibly changing your default values.*

If you change your mind and wish to reinstate the original value(s), click on the **Reset** button. All values are reset, not just the last value you changed.

3. Make the desired modifications and click on the **Dismiss** button to update the Setup window with modifications.

NOTE: Clicking on the ***Reset*** *button prior to the* ***Dismiss*** *button restores the original value(s).*

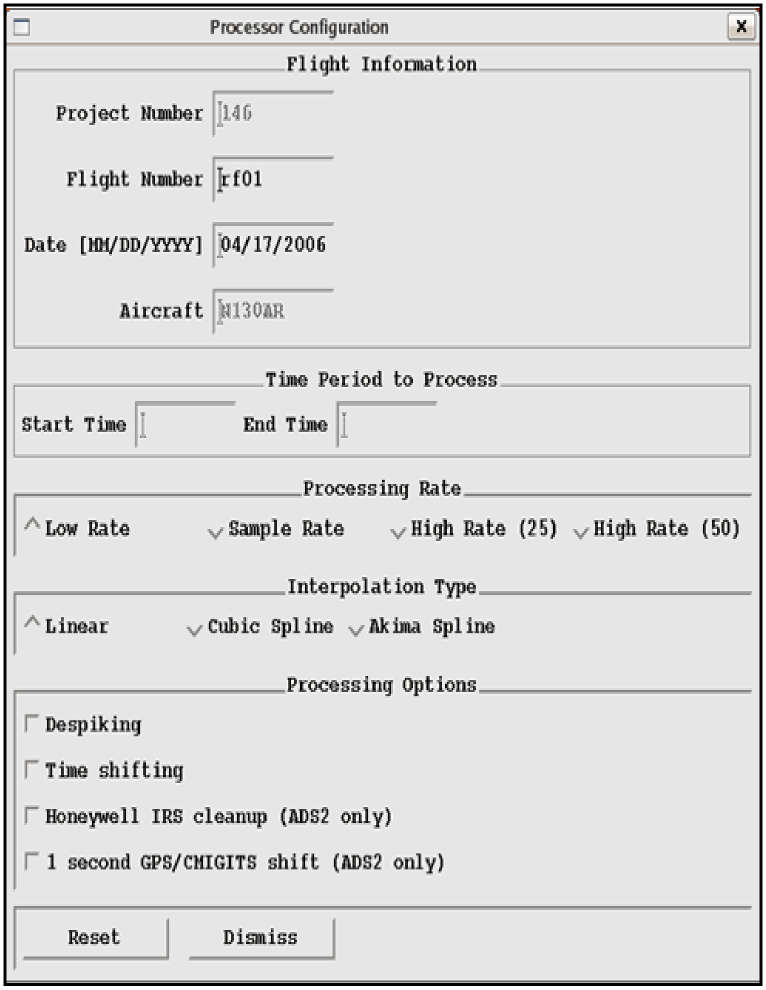
NOTE: *It is not necessary to save after each modification, you can make <u>all</u> desired changes and then save the setup file as described in “Saving or Creating a Setup File” on page 8-21.*

NOTE: *An asterisk (*) now appears next to the **Output** flag, alerting you that the variable has been changed from its original value.*

EDITING THE CONFIGURATION

You can change flight information, time period, processing rate and processing options. To do this click on the **<u>Edit</u>** button in the Setup window and select the **Edit Configuration** option or, press **Ctrl+C**. The Processor Configuration window is displayed in Figure 8-5.

Figure 8-5 Processor Configuration Window



Processor Configuration

Flight Information

Project Number 146

Flight Number rf01

Date [MM/DD/YYYY] 04/17/2006

Aircraft N130AR

Time Period to Process

Start Time | End Time

Processing Rate

● Low Rate ○ Sample Rate ○ High Rate (25) ○ High Rate (50)

Interpolation Type

● Linear ○ Cubic Spline ○ Akima Spline

Processing Options

□ Despiking

□ Time shifting

□ Honeywell IRS cleanup (ADS2 only)

□ 1 second GPS/CMIGITS shift (ADS2 only)

Editing Flight Information and Time Period

You can change flight information and time period by following the steps below:

1. In the Flight Information section of the Processor Configuration window, you can change the flight number and date.

*NOTE: Use the **Tab** and **Shift-Tab** keys to move from one box to another.*

2. In the Time Period to Process section of the window, you can select a time period for data processing. Enter the **Start Time** and **Finish Time** in 24-hour format. The colon delimiter is not required. Make sure to enter 6 digits in order for **Nimbus** to properly interpret the times. For each box, use either all colon delimiters or none at all (for example, 1004:00 does not work).

NOTE: When crossing Midnight, the value in the Finish field needs to be + 24 hours and greater than the Start field. You must manually add 24 hours to the Finish filed when crossing Midnight. For example, 1:00 AM would be entered as 25:00:00.

Changing the Processing Rate

In the Processing Rate section you can toggle the data sampling rate to **Low Rate Sample Rate** or **High Rate**.

All output data values can be set to the Low Rate Turbulence (LRT), defined as one sample per second (sps), by clicking **LEFT** on the **Low Rate** button. The **Low Rate** button is depressed when selected. All output data values are an average (or sum, if appropriate) of all the sample rate data values. “Low Rate” is the default setting of the Master Processing Rate (MPR).

All output data values except derived values can be set to the Sample Rate by clicking **LEFT** on the **Sample Rate** button. The derived value outputs the same as Low Rate, which is 1 sps.

Output data values can be set to the High Rate Turbulence (HRT), defined as 25 or 50 sps, by clicking **LEFT** on one of the High Rate buttons. The **Nimbus** processor is set to output the variable data at two rates, LRT and HRT. If a variable has a sample rate lower than HRT, output data values default to LRT using an average (or sum, if appropriate), of all the sample rate data values. If a variable has a sample rate greater than or equal to the HRT, input data values is put through a digital, multi-rate filter to produce HRT output data. The default rate for all derived variables is HRT, with the exception of entries in the **`LowRateVars`** file described in “Using Nimbus Support Files” on page 8-28.

Toggling Interpolation Type

In the Interpolation Type section of the Processor Configuration window, you can indicate whether you what type of curve fitting you want to use by clicking **LEFT** on the **Linear**, **Cubic** **Spline** or clicking **Akima** **Spline** button. This affects the interpolator used for despiking and time-shifting data.

Toggling Processing Options

In the Processing Options section of the Processor Configuration window, you can indicate which option you want to use: **Despiking**, **Time shifting**, **Honeywell IRS cleanup** or **1 second GPS/CMIGITS shift**.

More information about despiking can be found in “Using the Despike File” on page 8-32.

When you are done editing, click on the **Dismiss** button.

NOTE: *Clicking on the **Reset** button prior to the **Dismiss** button restores the original value(s).*

TOGGLING OUTPUT

You can define whether or not the variable’s data is output to the netCDF file by toggling the output flag to **`Y`** or **`N`**. To toggle the output you can use the Edit Variable window described on page 8-10 or highlight the variable(s) you wish to toggle, click on the **Edit** button and select the **Toggle output** option (or press **Ctrl-O**). The selected variables toggle to the opposite of their previous values.

NOTE: *An asterisk (*) appears next to the **Output** field, indicating that the original value has been changed.*

TOGGLING OUTPUT RATE

To toggle the output rate you can use the Edit Variable window described on page 8-10 or highlight the variable(s) you wish to toggle, click on the **Edit** button and select the **Toggle rate** option, or press **Ctrl-R**.

The selected variables toggle to the next available value. The output rates available for each type of variable are defined below:

❏ SDI Variables - LRT, HRT and the Sample Rate.

❏ Raw Variables (or Block Probe Variables) - LRT, HRT, and the Sample Rate.

❏ Derived Variables - LRT and HRT.

*NOTE: An asterisk (*) appears next to the **Output** field, indicating that the original value has been changed.*

NOTE: The toggle option to HRT is only available when the MRP is set to HRT.

Multiple Toggle

You can toggle the Output flag or Output Rate on or off for more than one variable at a time by simply holding down the **LEFT** mouse button and dragging to highlight a block of variables. Once the block is highlighted, press **Ctrl-O** to toggle the Output flag or **Ctrl-R** to toggle the Output Rate.

*NOTE: You can select multiple groups of variables throughout the Setup window by simply clicking **LEFT** while holding down the **Ctrl** key.*

TOGGLING PROBES

You can toggle the group of variables associated with an instrument or category of instruments on or off by following the steps below:

1. Click **LEFT** on the **Edit** button and select the **Toggle Probe** option. A drop down menu of probes, as well as the following options appear: **Corrected Winds**, **Fluxes**, **All On** and **All Off**.

2. Select one of the following options to toggle.

   - **All Off**: Toggles the probes "off" for all the variables.

   - **All On**: Toggles the probes "on" for all the variables.

   - **Corrected Winds**: This blends the IRS and GPS data together to produce a second set of wind calculations.

   - **Fluxes**: This multiplies the wind times the ozone, temperature, CO_2, or of whatever you want to determine the flux.

   - **Named Probe**: When you select a listed probe it toggles the variables associates with the probe on or off.

NOTE: An asterisk (*) appears to indicate that the value has changed.

Understanding Nimbus Setup Files

Nimbus creates a setup file to save its current internal processing configuration, usually to be associated later with a specific flight (or group of flights). In this context, a "flight" is defined as the data from a single input file. **Nimbus** uses the flight number as a processing key, therefore, great care must be taken when assigning flight numbers during data collection. If a single aircraft flight produces more than one file, each file must be considered as a separate flight. **Nimbus** processing can be problematic when flight numbers are duplicated, therefore, duplication should always be avoided.

When **Nimbus** begins, it reads the header record from the input data tape or file and uses this information to prepare for processing. It also prepares to output the data for the variable names it finds in that header record. It uses reference files (for example, **`Defaults`**, **`SumVars`**, **`ModVars`**) to determine how to process the data, selects which derived calculations to perform, and sets the values of various flags (for example, Output and Output Rate). **Nimbus** creates this default configuration before the user ever sees the Setup window. If you want to make modifications to the default configuration see “Editing Defaults” on page 8-11.

Once the modifications have been made, save the new configuration so it can be recalled at a later time. At any time during the setup process you can save the current configuration to a file as described in “Saving or Creating a Setup File” on page 8-21.

Nimbus uses setup files to alter the default processing configuration for a flight. The only way to prevent the automatic loading of production setup files is to start **Nimbus** with the “`-n`” option. When the “`-n`” option is not used, setup files are loaded using the following precedent:

1. `Flight_`***`ft###`*** (where ***`ft###`*** refers to a flight type and number) files assigned by the `` ` ``**`FlightGroups`**’ file (in the order entries appear in the file).
2. `Flight_`***`ft###`***, if it exists.
3. Manually-loaded setup files.

Configuration changes made to the same variable by loading any setup file overrides any previous change. Non-production setup files can be manually loaded to make one-time changes.

A setup file contains information for only those variables whose settings have been changed from the reference or default condition. Because the setup file entries are variable related and not parameter related, a “changed” variable’s entire configuration is saved, not just the individual settings that were changed. This is data in a variable record is described in “Editing Setup File Data” on page 8-19. For example, if you decide to output a raw variable at its sample rate instead of its default output rate and that change is saved to a setup file, its calibration coefficients are also saved. This is important to remember, because setup files often are shared among different flights. When that setup file is loaded, the calibration coefficients for the variable are used, overriding what was read from the input header.

There are times when the above style of behavior is not desired. (By observing some simple rules, you can carefully edit a setup file (an ordinary text file) to delete the calibration coefficients or any other setting that you do not wish to change.)

NOTE: *Although it is not recommended, you can edit a setup file. Ensure that extreme caution is taken, as* ***Nimbus*** *may malfunction and possibly crash if the file is not edited properly.*

EDITING SETUP FILE DATA

The following data entries appear in **Nimbus** setup files and can be modified by the user. See Figure 8-22 on page E-40 for an example of data entries in a setup file.

❑ Global data entries (appearing once in the setup file):

- Project Number (**`ProjNum`**).
- Master Processing Rate (**`PRate`**).

❑ Variable data entries

- Type of variable, followed by the variable name (**`SDI`**, **`RAW`**, **`DERIVED`**).
- Spike Slope (**`SS`**) in "E" format.
- Output Flag (**`O`**: **`0`**=Off, **`1`**=On)
- Output Rate (**`OR`**: sps)
- Static Lag (**`SL`**: ms)
- When variable type is SDI or RAW, the number of calibration coefficients is followed by the coefficients themselves, in "E" format (**`nCOEFF`**).
- When the variable type is DERIVED, the number of dependencies if followed by the dependencies themselves (**`nDEP`**).

❑ Configuration data entries (items that are not saved as part of the setup file).

- Input file name
- Output file name
- Time Slices

DEFINING SETUP FILE TYPES

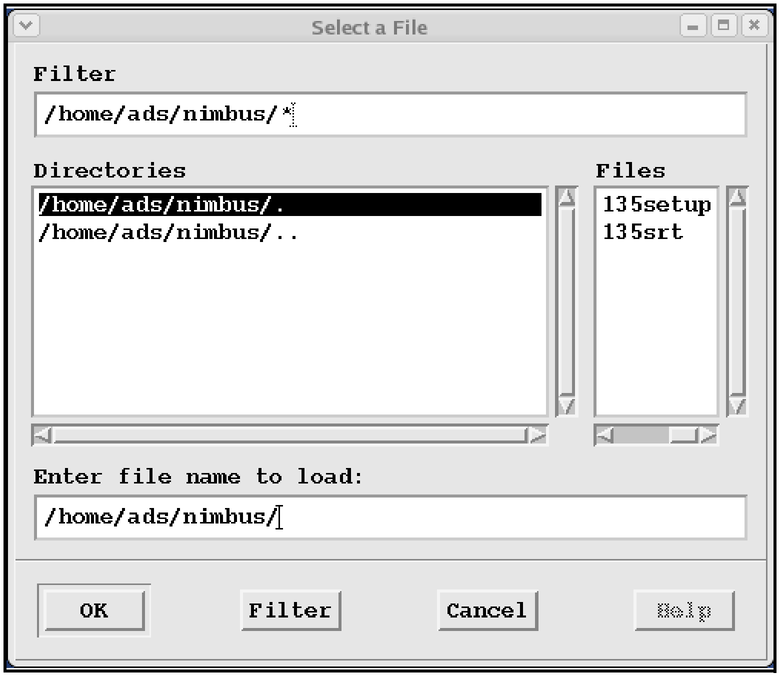
There are two basic types of setup files: production and non-production. If a production setup file exists for a given data set, it is automatically loaded. You use a Production setup file to ensure that needed changes from the default configuration are incorporated into the processing whenever a data set is accessed (production or non-production). Non-production setup files must be loaded manually, except when using batch mode. (See "Running Nimbus in Batch Mode" on page 8-26).

LOADING AN EXISTING SETUP FILE

To load an existing setup file, follow the instructions below:

1. Click **LEFT** on the **<u>File</u>** Button and select the **Load Setup** option, or press **Ctrl-L**. The Select a File window is displayed as shown in Figure 8-6.

Figure 8-6 Select a File Window



Filter: /home/ads/nimbus/*

Directories:
/home/ads/nimbus/.
/home/ads/nimbus/..

Files:
135setup
135srt

Enter file name to load: /home/ads/nimbus/

2. Select the desired file by highlighting the filename in the Files window or by typing the filename in the **Enter File name to load:** field. You can use the scroll bars in the **Directories** and **Files** windows to view all available files.

   You may also “filter” through a directory by highlighting the desired directory and clicking on the **<u>Filter</u>** button or by typing the desired directory in the **Filter** field. The **Files** window is updated with the files from the selected directory.

3. Click on **<u>OK</u>**.

   The grey box at the bottom of the Setup window displays a message indicating that the selected setup file is being loaded. When the file has been loaded, the Setup window is updated with the parameters specific to the selected setup file.

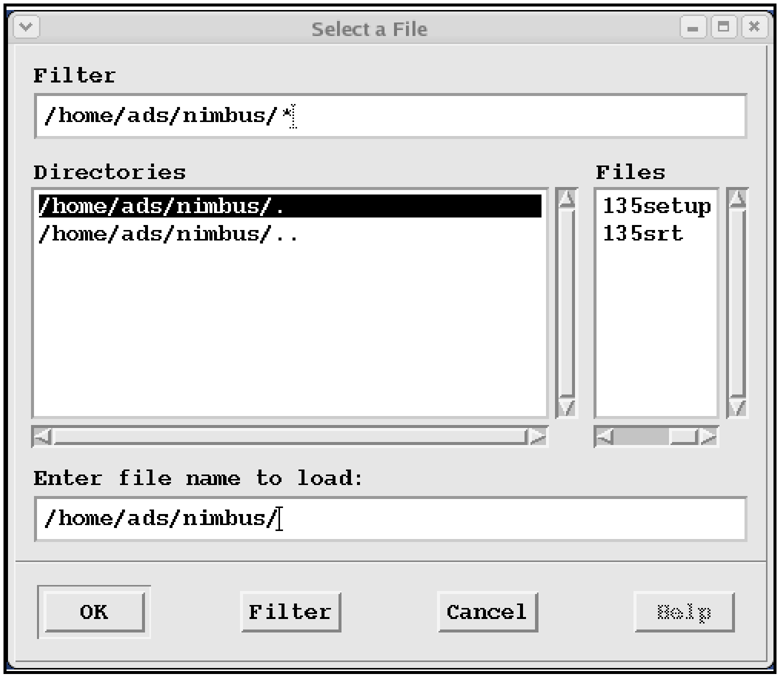
SAVING OR CREATING A SETUP FILE

When saving the current setup, the **Nimbus** processor creates or replaces (if the file already exists) a setup file. Follow the steps below to save a current setup:

1. Click **LEFT** on the **File** button and select the **Save Setup** option, or press **Ctrl-S**, which displays the Select a File window is displayed in Figure 8-7.

NOTE: It is not necessary to save after each modification, you can make all desired changes and then save.

Figure 8-7 Select a File Window



Filter: /home/ads/nimbus/*

Directories:
/home/ads/nimbus/.
/home/ads/nimbus/..

Files:
135setup
135srt

Enter file name to load: /home/ads/nimbus/

2. Select the desired file by highlighting the filename in the **Files** window or by typing the filename in the **Enter File name to save:** field. Use the scroll bars in the **Directories** and **Files** windows to view all available files.

   You may also "filter" through a directory by highlighting the desired directory and clicking on the **Filter** button or by typing the desired directory in the **Filter** field. The **Files** window is updated with the files from the selected directory.

3. Click **LEFT** on the **OK** button to save the setup.

PRINTING THE CURRENT SETUP

You can print the current setup by clicking **LEFT** on the **File** button and selecting the **Print Setup** option. The current setup is sent to the printer defined by your **PRINTER** environment variable with the following statement:

```
setenv $PRINTER <printer name>
```

USING A SYNTHETIC DATA FILE

You can fix a raw data value so that you can test **AMLIB** derived calculations by creating a file with many variables of synthetic data. In the following example, the variables list (**DPB, FCN, GLAT** & **GSF**) is substituted for the actual raw data. This differs from setting the variables in the Edit Variables window (see “Edit Variable Window” on page 8-10) in that the values are not fixed for the entire flight. You can enter whatever values you want when you use a synthetic data file. The following is a sample file.

```
UTC DPB FCN GLAT GSF
21:50:23 23.5329 1.37044 15.7808 60.3291
21:50:24 23.8249 1.47202 15.7908 61.0029
```

QUITTING THE SETUP WINDOW

You can quit **Nimbus** by clicking **LEFT** on the **File** button and selecting the **Quit** option, or by pressing **Ctrl-Q**.

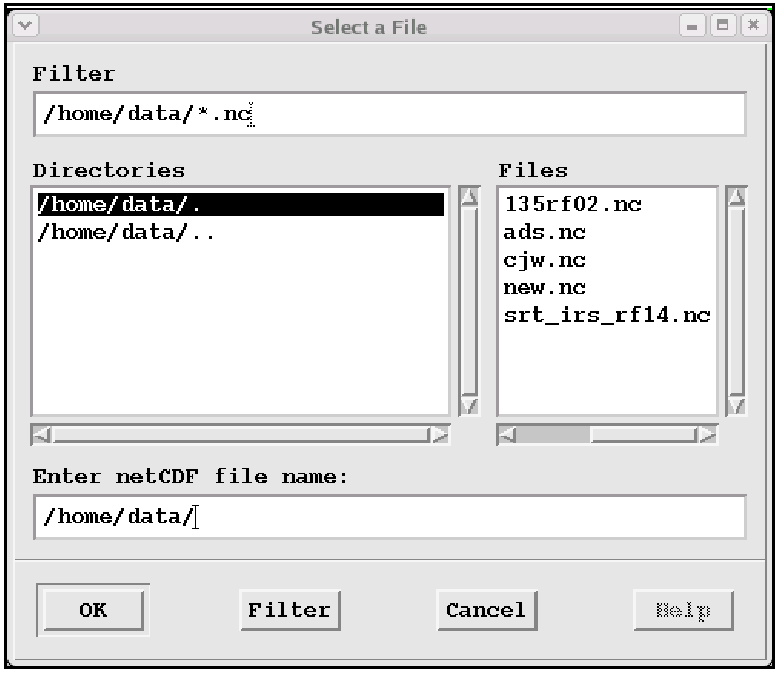
NOTE: Make sure to save any modifications prior to quitting the Setup window, or the modifications are not saved in the setup file.

Processing Data

To process data do the following:

1. Click **LEFT** on the **Go** button in the Setup window and the Select a File window is displayed as shown in Figure 8-8.

Figure 8-8 Select a File Window



Filter
/home/data/*.nc

Directories
/home/data/.
/home/data/..

Files
135rf02.nc
ads.nc
cjw.nc
new.nc
srt_irs_rf14.nc

Enter netCDF file name:
/home/data/

2. Select a directory in the **Directories** field or enter the directory path in the **Filters** field. Press the **Enter** key or the **Filter** button to display the ADADS data files in the directory path.

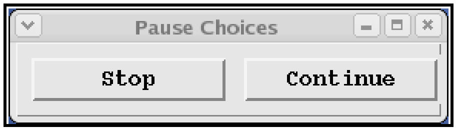
3. Press the **Tab** key to move to the **Files** field, or click **LEFT** on the desired field, and enter a file to select.

4. Click on the **OK** button.

   Data are read from the ADADS input file, processed and output to the specified netCDF output file.The Time Box located in the upper right corner of the Setup window displays the ADADS time, which is updated after each minute of data has been processed. When **Nimbus** is processing data, the **Go** button changes to **Pause**. The Time Box in the upper right corner of the Setup window updates for each minute of data processed.

Click **LEFT** on the **Pause** button to interrupt data processing and a Pause Choice window is displayed as shown in Figure 8-9.

Figure 8-9 Pause Choice Window



You can choose to halt processing by clicking **LEFT** on **Stop**, or continue processing from the point of interruption by clicking **LEFT** on **Continue**.

NOTE: Once you stop the processing, the only option available to you is to Quit ***Nimbus****.*

Understanding the Difference Between Production and Non-Production Mode

There are differences in the instructions for using **Nimbus** if you plan on performing Production Mode processing as opposed to Non-Production Mode processing. The differences are described in the following sections.

USING PRODUCTION MODE

Production processing is designed to produce an archival data set. It is assumed that the raw variables produce the highest quality data set obtainable. In some cases, this involves adjustments to the default processing configuration. **Nimbus** uses several support files when configuring itself for processing, some of which may require changes. This is described in “Using Nimbus Support Files” on page 8-28.

Nimbus uses the “`nimbus`” login as a mechanism to control production processing. When logged onto the system with the user name “`nimbus`,” production mode is assumed, and processing rules change. The system assigns specific output and setup file paths and names. By using these reference files, the archive environment can consistently be reproduced. The output `netCDF` file is stored in the `/home/productiondata` directory, and production setup files are stored in (and used from) the project’s `Production` subdirectory with the name `Flight_`***`ft###`***, where ***`ft###`*** is a string that denotes the flight number. If the file already exists, it is overwritten (unless it is write protected).

NOTE: *Make sure that each data set has a unique flight number, so that your files are not overwritten. You cannot automatically override this feature unless you run **Nimbus** in batch mode using the "`-n`" option and a batch-file entry to load a specific setup file.*

Once archival processing has occurred, we strongly recommended that the following files be protected from future change (unless required to fix a problem discovered later):

❏ **`Defaults, DependTable, LowRateVars, PMSspecs, ProjectName, VarDB`** in the project's directory **`${PROJ_DIR}/nnn`**, where **`nnn`** is the project number.

❏ **`FlightGroups`**, all **`Flight_`*`ft###`*** (where ***`ft###`*** refers to a flight type and number) in the project's **`Production`** directory.

USING NON-PRODUCTION MODE

Non-production processing is used for data perusal before production processing or data analysis afterwards. Prior to production, setup files normally do not exist. You can peruse the output data to find measurement anomalies and to find where modifications are required (for example, a revised calibration or a dependency substitution for a failed instrument).

In non-production mode do the following:

1. Select an output file name at the Select a File window.

2. Save the setup, as described in"Saving or Creating a Setup File" on page 8-21.

The values for the following parameters are ignored for any production setup files that are automatically loaded. (If you want these parameter settings to take effect, you must manually load the setup file again.)

- Master Processing Rate is set to **`LRT`**.
- Each variable's Output is set to **`Y`**.
- Each variable's output rate is set to **`LRT`**.

Non-production setup files are created by logins other than "`nimbus`". When a setup configuration is saved, **Nimbus** displays a window allowing the user to specify a path and file name under which to save the setup. This process is described in "Saving or Creating a Setup File" on page 8-21. You can use this file for production by moving it to the project's `Production` directory with the appropriate file name. All non-production setup files must be manually loaded (or loaded from a batch file).

Running Nimbus in Batch Mode

Nimbus can be run in "batch" mode using existing setup files. Batch mode allows the **Nimbus** processor to run at hours when a user would not be present and frees **Nimbus** from the window environment and all user interaction. Batch mode is only available by using a command line option.

A batch file for each flight must exist for each flight segment to be processed. The batch file must be created as follows:

```
if input_file

of output_file

sf setup_file

pr=n

ti start-end
```

Where:

input_file	Full name of raw ADADS data file*
output_file	Full name of output netCDF file*
setup_file	Full name of setup file to load**
`n`	Processing rate (1 or 25)**
`start`	Start time, in "hh:mm:ss" format**
`end`	Finish time, in "hh:mm:ss" format**

* Mandatory entries.

**Optional entries.

DELAYING BATCH PROCESSING

To have **Nimbus** run at a later time, you can add a sleep process prior to executing **Nimbus**. For example, if you want to process this file at 10:00 p.m., and the current time is 4:00 p.m. (6 hours =21600 seconds earlier), you would enter the following:

```
(sleep 21600; nimbus -b batch_file > log_file)&
```

Where:

batch_file	Full name of batch file
log_file	Full name of log file

*NOTE: The computer must remain up between the time the command is issued and when it is scheduled to run. The ampersand (`&`), instructs **Nimbus** to run in the background.*

SCHEDULING A BATCH RUN

You can also schedule **Nimbus** to run in batch mode by using the `at` command. You this this as follows:

1. Create a ***`script file`*** which includes the following line:

   ```
   nimbus -b batch_file log_file&
   ```

2. Issue the following command:

   ```
   at -m 10:00pm today script_file
   ```

   where:

   `-m` Send mail message after the job has completed. This option may be omitted if you do not want a mail message.

*NOTE: The **script_file** can contain more than one batch request if multiple jobs are being submitted. Specify a unique **log_file** for each job if you choose to submit multiple jobs.*

Using Nimbus Support Files

Nimbus uses a number of support files to perform its operations. You can modify the **Nimbus** support files (except when indicated) using the text editor of your choice. Pay special attention to the guidelines provided for each individual file. A list of the support files, their location, description and an excerpt of the actual file is provided in the following sections.

The support files are broken into two groups (or locations on disk). The first group are the files that pertain to all projects for all time and they reside in the **`${PROJ_DIR}/Configuration/raf`** directory. These files include **`AircraftSpecs`**, **`DerivedNames`**, **`MetaDataTable`**, **`SumVars`**, **`ModVars`** and **`RawNames`**.

The Second group are files that are copied into each project directory and are then modified to support that directory (the default version of this second group also comes from the **`${PROJ_DIR}/Configuration/raf`** directory). These files include **`DependTable`**, **`Defaults`**, **`PMSspecs`**, **`Despike`**, **`Lags`**, **`LowRateVars`**, **`VarDB`** (with Categories and StandardNames), landmarks and **`ProjectName`**.

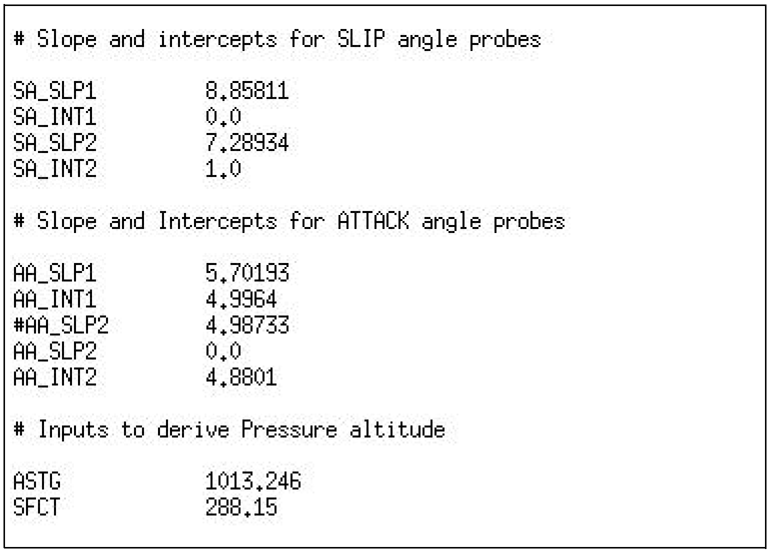
USING THE DEFAULTS FILE

`Defaults` is an ASCII file that contains values for derived variables. The **`Defaults`** file values are usually constants used by one or more variables, such as the wire temperature setting for the King Liquid Water Content probe and alignment offsets for the inertial system. The **`Defaults`** file is located in the project's home directory (**`${PROJ_DIR}/nnn`**), where **`nnn`** is the 3-digit project number. An excerpt from a **`Defaults`** file is shown in Figure 8-10 on page 8-29.

CAUTION: *The* **`Defaults`** *file is project-specific and mandatory. The information in the right-hand column of the file can be modified,* ***do not*** *alter the information in the left-hand column.*

NOTE: *A* **`#`** *in column 1 denotes the line as a comment.* ***Nimbus*** *ignores any comment lines in this file.*

Figure 8-10 **`Defaults`** File Excerpt



```
# Slope and intercepts for SLIP angle probes

SA_SLP1         8.85811
SA_INT1         0.0
SA_SLP2         7.28934
SA_INT2         1.0

# Slope and Intercepts for ATTACK angle probes

AA_SLP1         5.70193
AA_INT1         4.9964
#AA_SLP2        4.98733
AA_SLP2         0.0
AA_INT2         4.8801

# Inputs to derive Pressure altitude

ASTG            1013.246
SFCT            288.15
```

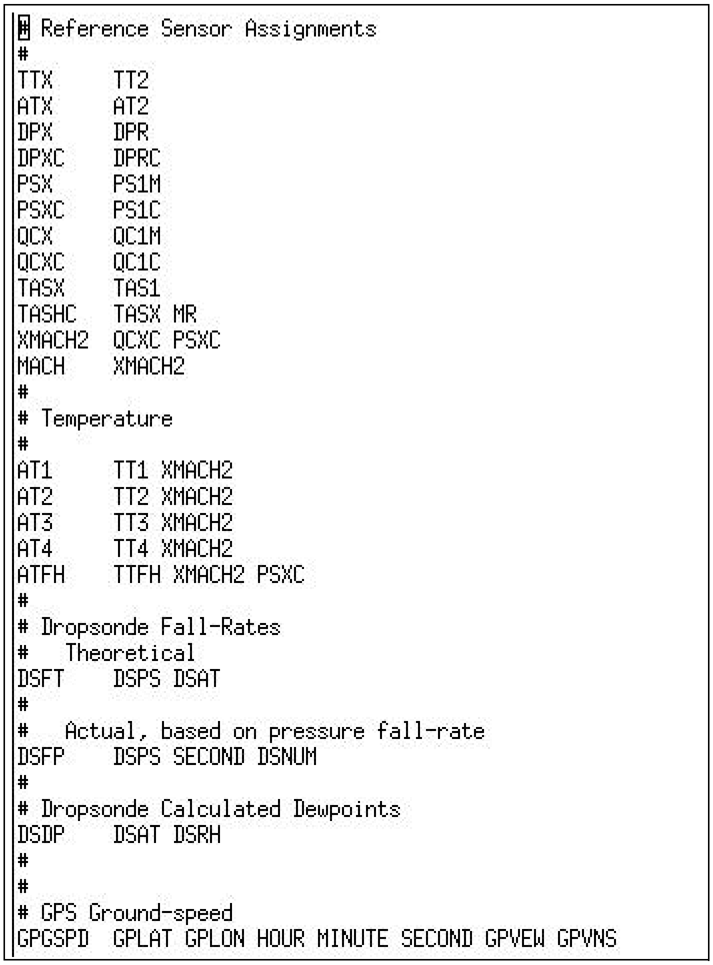
USING THE DEPENDTABLE FILE

`DependTable` is an ASCII file that contains the dependency list for the derived variables. The DependTable file is located in the project's home directory (**`${PROJ_DIR}/nnn`**), where **`nnn`** is the 3-digit project number. An excerpt from a **`DependTable`** file is shown in Figure 8-11 on page 8-30.

CAUTION: *The* **`DependTable`** *file is project-specific and mandatory. The information in the right-hand column of the file can be modified,* ***do not*** *alter the information in the left-hand column. When modifying variables, ensure that the new variables are of the same type as the previous variables. Ensure that the same number of dependencies exist after the modification as did before the modification.*

NOTE: *A* **`#`** *in column 1 denotes the line as a comment.* ***Nimbus*** *ignores any comment lines in this file.*

Figure 8-11 **`DependTable`** File Excerpt

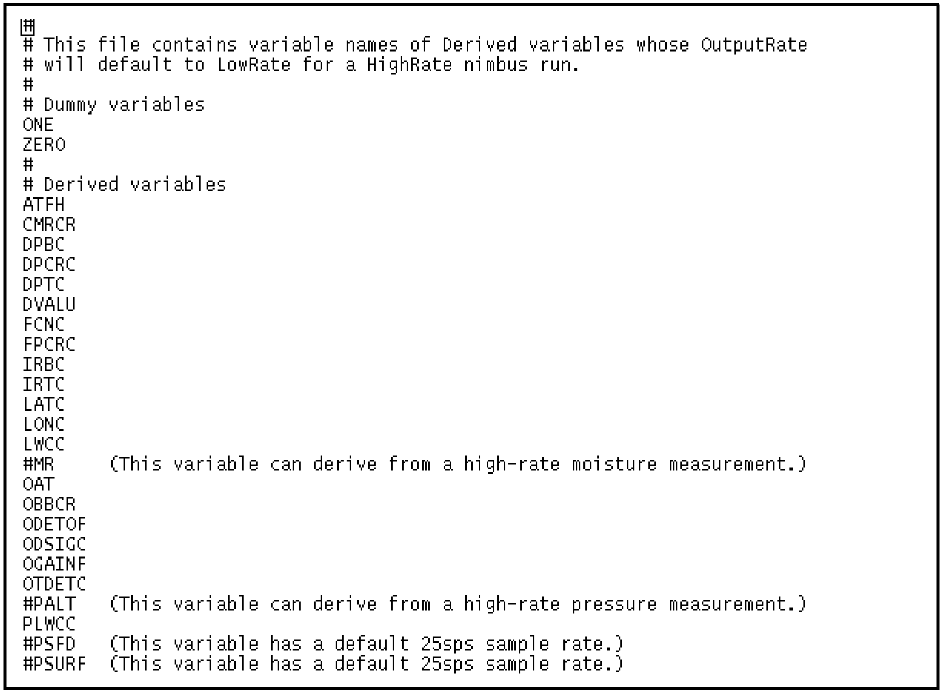


```
# Reference Sensor Assignments
#
TTX     TT2
ATX     AT2
DPX     DPR
DPXC    DPRC
PSX     PS1M
PSXC    PS1C
QCX     QC1M
QCXC    QC1C
TASX    TAS1
TASHC   TASX MR
XMACH2  QCXC PSXC
MACH    XMACH2
#
# Temperature
#
AT1     TT1 XMACH2
AT2     TT2 XMACH2
AT3     TT3 XMACH2
AT4     TT4 XMACH2
ATFH    TTFH XMACH2 PSXC
#
# Dropsonde Fall-Rates
#   Theoretical
DSFT    DSPS DSAT
#
#   Actual, based on pressure fall-rate
DSFP    DSPS SECOND DSNUM
#
# Dropsonde Calculated Dewpoints
DSDP    DSAT DSRH
#
#
# GPS Ground-speed
GPGSPD  GPLAT GPLON HOUR MINUTE SECOND GPVEW GPVNS
```

USING THE `LOWRATEVARS` FILE

`LowRateVars` is an ASCII file containing a list of derived variables for which **Nimbus** presets the output rate to LRT when the MPR is set to **`HRT`**. This file is ignored when the MPR is set to **`LRT`**. By default, **Nimbus** sets the output rate for all derived variables to MPR. LowRateVars is located in the project's home directory (**`${PROJ_DIR}/nnn`**), where **`nnn`** is the project number. An excerpt from a **`LowRateVars`** file is shown in Figure 8-12 on page 8-31.

Figure 8-12 `LowRateVars` File Excerpt



```
#
# This file contains variable names of Derived variables whose OutputRate
# will default to LowRate for a HighRate nimbus run.
#
# Dummy variables
ONE
ZERO
#
# Derived variables
ATFH
CMRCR
DPBC
DPCRC
DPTC
DVALU
FCNC
FPCRC
IRBC
IRTC
LATC
LONC
LWCC
#MR     (This variable can derive from a high-rate moisture measurement.)
OAT
OBBCR
ODETOF
ODSIGC
OGAINF
OTDETC
#PALT   (This variable can derive from a high-rate pressure measurement.)
PLWCC
#PSFD   (This variable has a default 25sps sample rate.)
#PSURF  (This variable has a default 25sps sample rate.)
```

NOTE: *The* **`LowRateVars`** *file is optional and may be edited by the user.*

NOTE: *A* **`#`** *in column 1 denotes the line as a comment.* ***Nimbus*** *ignores any comment lines in this file.*

USING THE PROJECTNAME FILE

ProjectName is an ASCII file containing a single line with the project's name.

NOTE: The **ProjectName** *file is optional.*

USING THE VARDB FILE

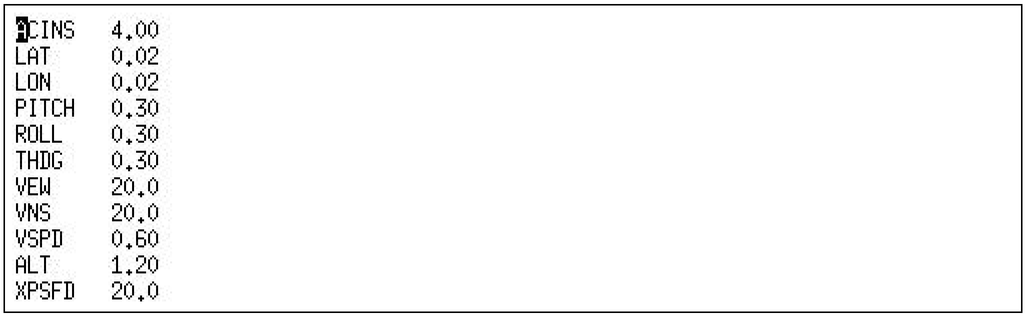
VarDB is a binary file containing each variable's name, measurement units, and a n extended description. Entries in this file can be viewed or modified using the "**vared**" program. Refer to "Using the FlightGroups File" on page 8-39. **VarDB** is located in the project's home directory: (**/home/local/proj/nnn**), where **nnn** is the 3-digit project number.

NOTE: The **VarDB** *file is project-specific.*

USING THE DESPIKE FILE

The **Despike** file contains SDI or Raw variables and their corresponding spike slope values to despike. This file is found in the directory **${PROJ_DIR}${PROJ_DIR}/nnn**, where **nnn** is the project number. An excerpt from a **Despike** file is shown in Figure 8-13.

Figure 8-13 **Despike** File Excerpt



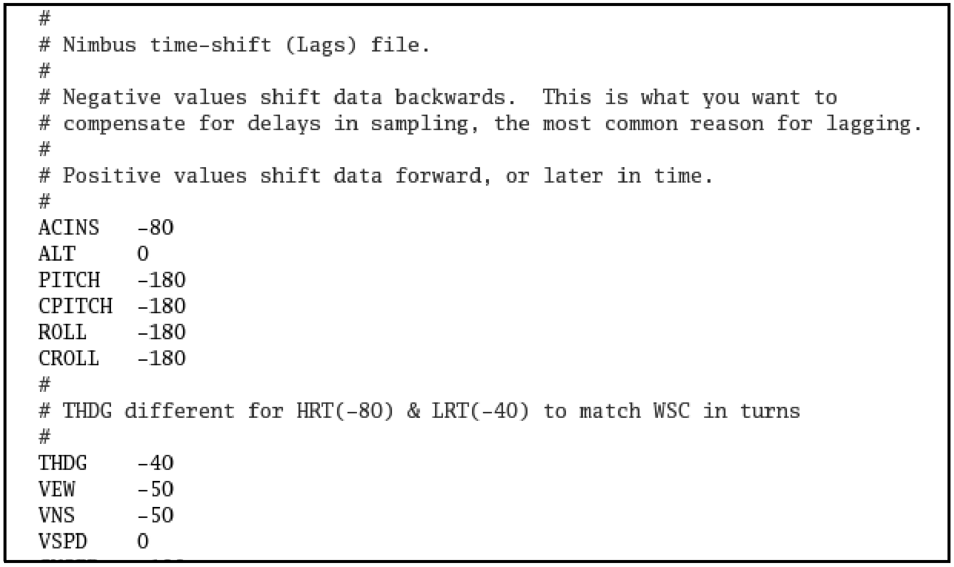
```
ACINS   4.00
LAT     0.02
LON     0.02
PITCH   0.30
ROLL    0.30
THDG    0.30
VEW     20.0
VNS     20.0
VSPD    0.60
ALT     1.20
XPSFD   20.0
```

NOTE: The **Despike** *file is optional and project-specific if it exists.*

USING THE **LAGS** FILE

The **Lags** file contains SDI or Raw variables to lag (up to +/- 1000ms). This file is located in the directory **${PROJ_DIR}/nnn**, where **nnn** is the project number. An excerpt from a **Lags** file is shown in Figure 8-14.

Figure 8-14 **Lags** File Excerpt



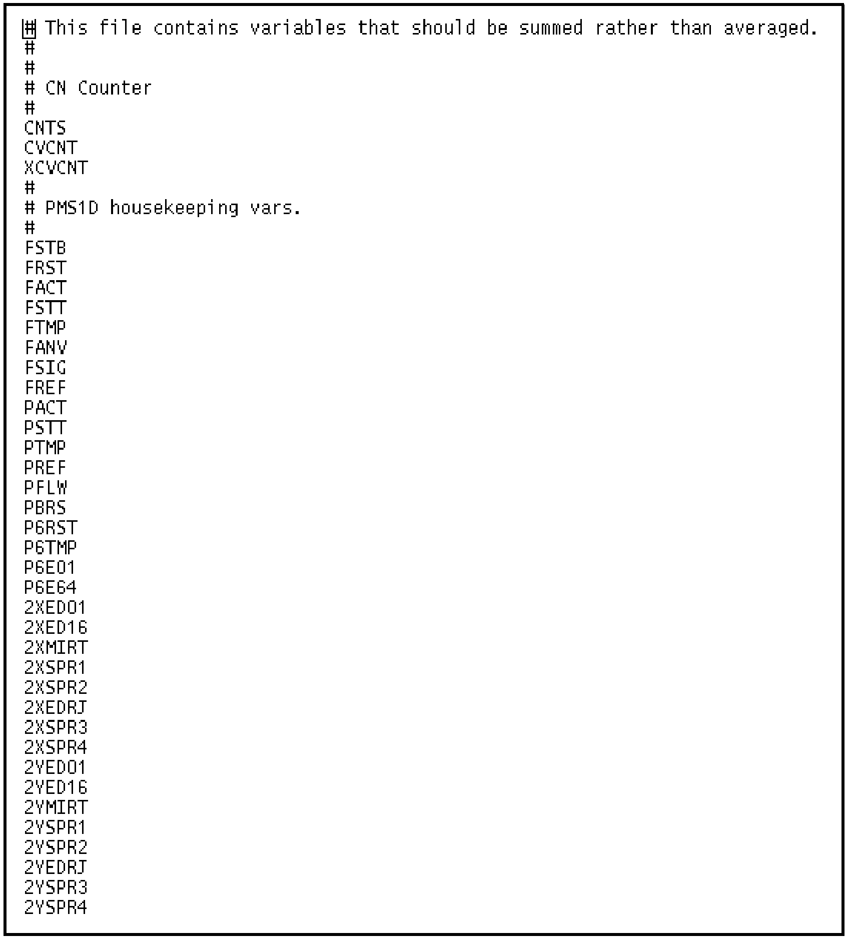
```
#
# Nimbus time-shift (Lags) file.
#
# Negative values shift data backwards.  This is what you want to
# compensate for delays in sampling, the most common reason for lagging.
#
# Positive values shift data forward, or later in time.
#
ACINS   -80
ALT     0
PITCH   -180
CPITCH  -180
ROLL    -180
CROLL   -180
#
# THDG different for HRT(-80) & LRT(-40) to match WSC in turns
#
THDG    -40
VEW     -50
VNS     -50
VSPD    0
```

NOTE: The **lags** *file is optional and project-specific if it exists.*

USING THE **SUMVARS** FILE

The **SumVars** file provides a list of variable names. This file is not project dependent, but is used globally by all projects. This file is located in directory **${PROJ_DIR}/Configuration/raf.** This file tells **Nimbus** which variable should be summed rather than averaged when changing from sample rate to output rate. An excerpt from a **SumVars** file is shown in Figure 8-15 on page 8-34.

Figure 8-15 **`SumVars`** File Excerpt



```
# This file contains variables that should be summed rather than averaged.
#
#
# CN Counter
#
CNTS
CVCNT
XCVCNT
#
# PMS1D housekeeping vars.
#
FSTB
FRST
FACT
FSTT
FTMP
FANV
FSIG
FREF
PACT
PSTT
PTMP
PREF
PFLW
PBRS
P6RST
P6TMP
P6E01
P6E64
2XED01
2XED16
2XMIRT
2XSPR1
2XSPR2
2XEDRJ
2XSPR3
2XSPR4
2YED01
2YED16
2YMIRT
2YSPR1
2YSPR2
2YEDRJ
2YSPR3
2YSPR4
```

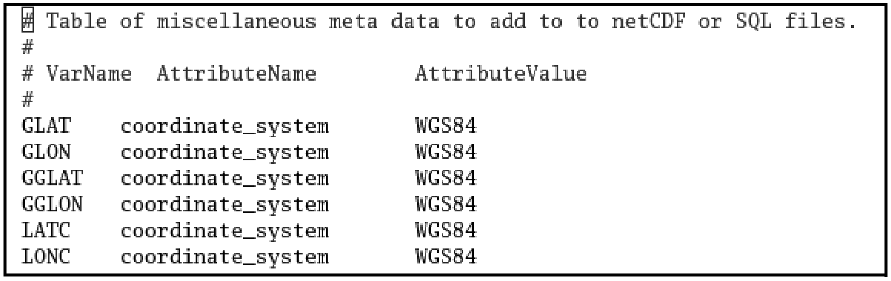
NOTE: *The* **`sumvars`** *file is an optional file.*

NOTE: *A* **`#`** *in column 1 denotes the line as a comment.* ***Nimbus*** *ignores any comment line in this file.*

USING THE METADATATABLE FILE

The **MetaDataTable** file provides a list of attributes or metadata to add to individual variables. This file is not project dependent, but is used globally by all projects. This file is located in directory **${PROJ_DIR}/Configuration/raf**. An excerpt from a **MetaDataTable** file is shown in Figure 8-16.

Figure 8-16 **MetaDataTable** File Excerpt



```
# Table of miscellaneous meta data to add to to netCDF or SQL files.
#
# VarName  AttributeName        AttributeValue
#
GLAT    coordinate_system       WGS84
GLON    coordinate_system       WGS84
GGLAT   coordinate_system       WGS84
GGLON   coordinate_system       WGS84
LATC    coordinate_system       WGS84
LONC    coordinate_system       WGS84
```

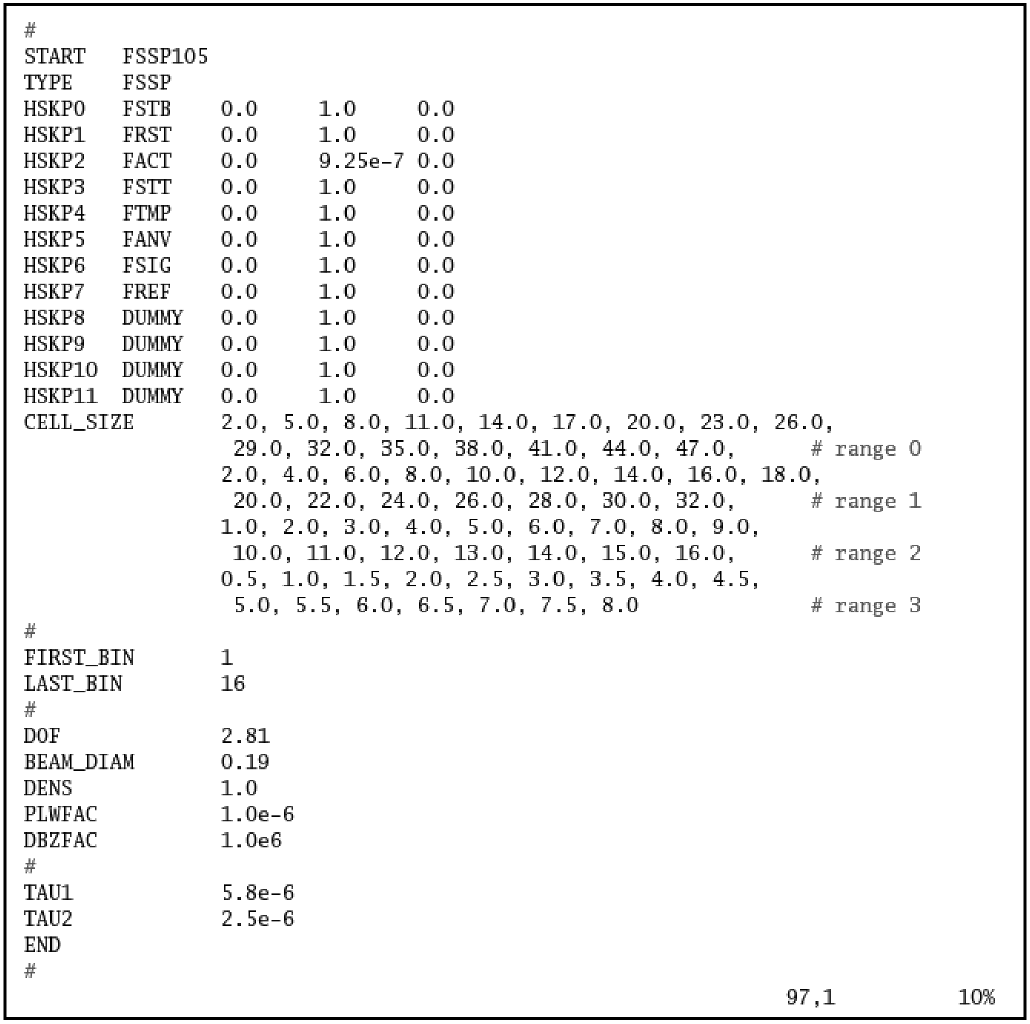
The following lines will be added to **netCDF** file as (last line):

```
float GGLAT(Time) ;
GGLAT:_FillValue = -32767.f ;
GGLAT:units = "degree_N" ;
GGLAT:long_name = "GPS Latitude" ;
GGLAT:valid_range = -90.f, 90.f ;
GGLAT:Category = "Position" ;
GGLAT:standard_name = "latitude" ;
GGLAT:SampledRate = 1 ;
GGLAT:TimeLag = -500 ;
GGLAT:TimeLagUnits = "milliseconds" ;
GGLAT:DataQuality = "Preliminary" ;
GGLAT:coordinate_system = "WGS84" ;
```

USING THE **PMSSPECS** FILE

The **PMSspecs** file contains all information required to process and display size-distribution probe data. Data is stored by probe serial number so that multiple probes of the same type can have entries in the file. This file originally only covered Particle Measuring Systems (PMS) probes, but its use has been expanded to cover any size-distribution probes. An excerpt from a **PMSspecs** file is shown in Figure 8-17.

Figure 8-17 PMSspecs File Excerpt

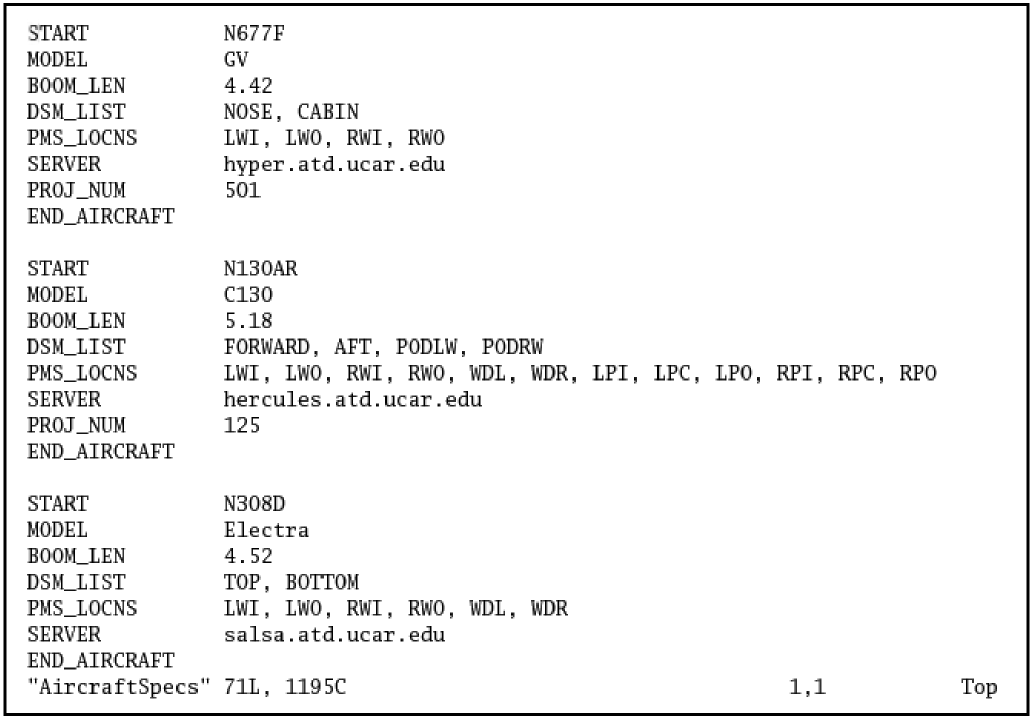


```
#
START   FSSP105
TYPE    FSSP
HSKP0   FSTB    0.0     1.0     0.0
HSKP1   FRST    0.0     1.0     0.0
HSKP2   FACT    0.0     9.25e-7 0.0
HSKP3   FSTT    0.0     1.0     0.0
HSKP4   FTMP    0.0     1.0     0.0
HSKP5   FANV    0.0     1.0     0.0
HSKP6   FSIG    0.0     1.0     0.0
HSKP7   FREF    0.0     1.0     0.0
HSKP8   DUMMY   0.0     1.0     0.0
HSKP9   DUMMY   0.0     1.0     0.0
HSKP10  DUMMY   0.0     1.0     0.0
HSKP11  DUMMY   0.0     1.0     0.0
CELL_SIZE       2.0, 5.0, 8.0, 11.0, 14.0, 17.0, 20.0, 23.0, 26.0,
                 29.0, 32.0, 35.0, 38.0, 41.0, 44.0, 47.0,      # range 0
                2.0, 4.0, 6.0, 8.0, 10.0, 12.0, 14.0, 16.0, 18.0,
                 20.0, 22.0, 24.0, 26.0, 28.0, 30.0, 32.0,      # range 1
                1.0, 2.0, 3.0, 4.0, 5.0, 6.0, 7.0, 8.0, 9.0,
                 10.0, 11.0, 12.0, 13.0, 14.0, 15.0, 16.0,      # range 2
                0.5, 1.0, 1.5, 2.0, 2.5, 3.0, 3.5, 4.0, 4.5,
                 5.0, 5.5, 6.0, 6.5, 7.0, 7.5, 8.0              # range 3
#
FIRST_BIN       1
LAST_BIN        16
#
DOF             2.81
BEAM_DIAM       0.19
DENS            1.0
PLWFAC          1.0e-6
DBZFAC          1.0e6
#
TAU1            5.8e-6
TAU2            2.5e-6
END
#
                                                   97,1          10%
```

USING THE **AIRCRAFTSPECS** FILE

The **AircraftSpecs** file contains configuration data on a per aircraft basis, which is stored based on Tail Number similar to how PMSspecs stores information is based on Serial Number. An excerpt from a **PMSspecs** file is shown in Figure 8-18.

Figure 8-18 **AircraftSpecs** File Excerpt

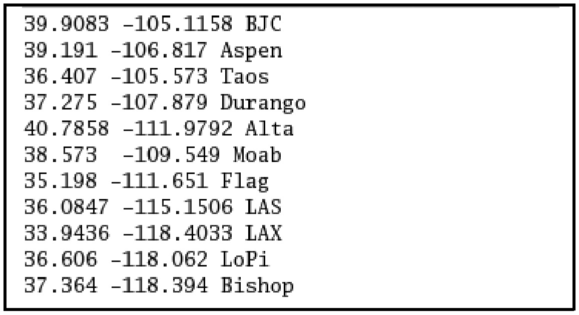


```
START           N677F
MODEL           GV
BOOM_LEN        4.42
DSM_LIST        NOSE, CABIN
PMS_LOCNS       LWI, LWO, RWI, RWO
SERVER          hyper.atd.ucar.edu
PROJ_NUM        501
END_AIRCRAFT

START           N130AR
MODEL           C130
BOOM_LEN        5.18
DSM_LIST        FORWARD, AFT, PODLW, PODRW
PMS_LOCNS       LWI, LWO, RWI, RWO, WDL, WDR, LPI, LPC, LPO, RPI, RPC, RPO
SERVER          hercules.atd.ucar.edu
PROJ_NUM        125
END_AIRCRAFT

START           N308D
MODEL           Electra
BOOM_LEN        4.52
DSM_LIST        TOP, BOTTOM
PMS_LOCNS       LWI, LWO, RWI, RWO, WDL, WDR
SERVER          salsa.atd.ucar.edu
END_AIRCRAFT
"AircraftSpecs" 71L, 1195C                                    1,1          Top
```

USING THE **LANDMARKS** FILE

landmarks is a metadata file listing latitude/longitude coordinates and a text label. This data is transferred to the output netCDF and/or the SQL data files as global attributes. An excerpt from a **landmarks** file is shown in Figure 8-19.

Figure 8-19 **landmarks** File Excerpt

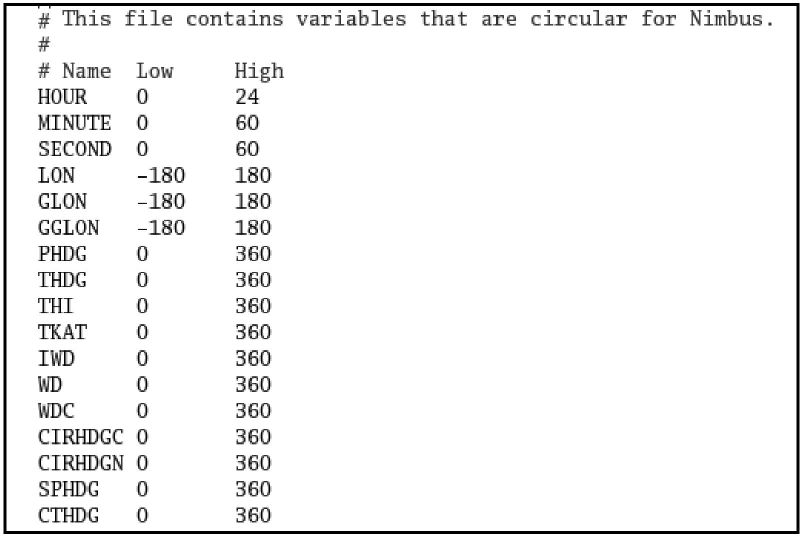


```
39.9083 -105.1158 BJC
39.191 -106.817 Aspen
36.407 -105.573 Taos
37.275 -107.879 Durango
40.7858 -111.9792 Alta
38.573  -109.549 Moab
35.198 -111.651 Flag
36.0847 -115.1506 LAS
33.9436 -118.4033 LAX
36.606 -118.062 LoPi
37.364 -118.394 Bishop
```

USING THE MODVARS FILE

The **ModvVars** file provides a list of variable names. This file is not project dependent, but is used globally by all projects.This file is located in **${PROJ_DIR}/Configuration/raf.** Its purpose is to define the minimum and maximum values of a variable, specifically, angles. An excerpt from a **ModVars** file is shown in Figure 8-20.

Figure 8-20 ModVars File Excerpt



```
# This file contains variables that are circular for Nimbus.
#
# Name  Low     High
HOUR    0       24
MINUTE  0       60
SECOND  0       60
LON     -180    180
GLON    -180    180
GGLON   -180    180
PHDG    0       360
THDG    0       360
THI     0       360
TKAT    0       360
IWD     0       360
WD      0       360
WDC     0       360
CIRHDGC 0       360
CIRHDGN 0       360
SPHDG   0       360
CTHDG   0       360
```

NOTE: *The **modvars** file is not project-dependent and is an optional file.*

NOTE: *A **#** in column 1 denotes the line as a comment. **Nimbus** ignores any comment line in this file.*

USING PRODUCTION DIRECTORY SUPPORT FILES

There are three files in the **Production** subdirectory that once created are automatically loaded each time Nimbus runs. They are **FlightGroups**, **Flight_*ft*###**, and **BlankOut_*ft*###** where:

ft is the flight type, which must be one of the following:

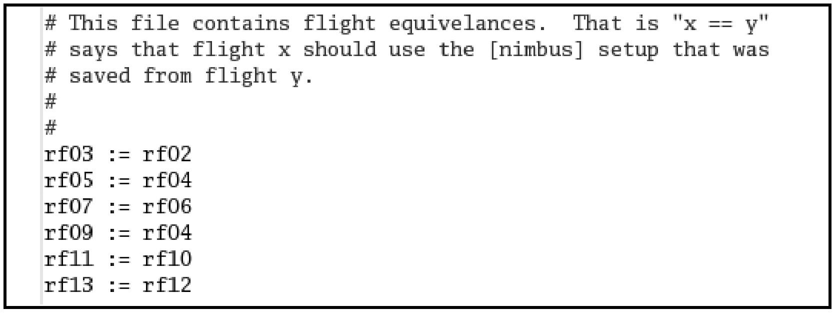
- **ff** (Ferry Flight)
- **tf** (Test Flight)
- **rf** (Research Flight)

and ***###*** is the flight number.

Using the FlightGroups File

FlightGroups is an ASCII file used to designate which flight setup file(s) are to be loaded when processing a particular flight. **FlightGroups** is located in the project's **Production** subdirectory (${PROJ_DIR}${PROJ_DIR}**/nnn/Production**). The file works with the **Flight_ft###** files. Each line in this file associates the setup file for a specific flight with another. For example, a line with the entry **rf003 == rf002** tells **Nimbus** to load the setup file **Flight_rf00** when setting up the processing for flight rf003. If multiple lines exist for a flight (for example, they start with the same number) multiple setup files are loaded for it. You must set up these interrelationships before production processing by editing this file. An excerpt from a **FlightGroups** file is shown in Figure 8-21.

Figure 8-21 **FlightGroups** File Excerpt



```
# This file contains flight equivelances.  That is "x == y"
# says that flight x should use the [nimbus] setup that was
# saved from flight y.
#
#
rf03 := rf02
rf05 := rf04
rf07 := rf06
rf09 := rf04
rf11 := rf10
rf13 := rf12
```

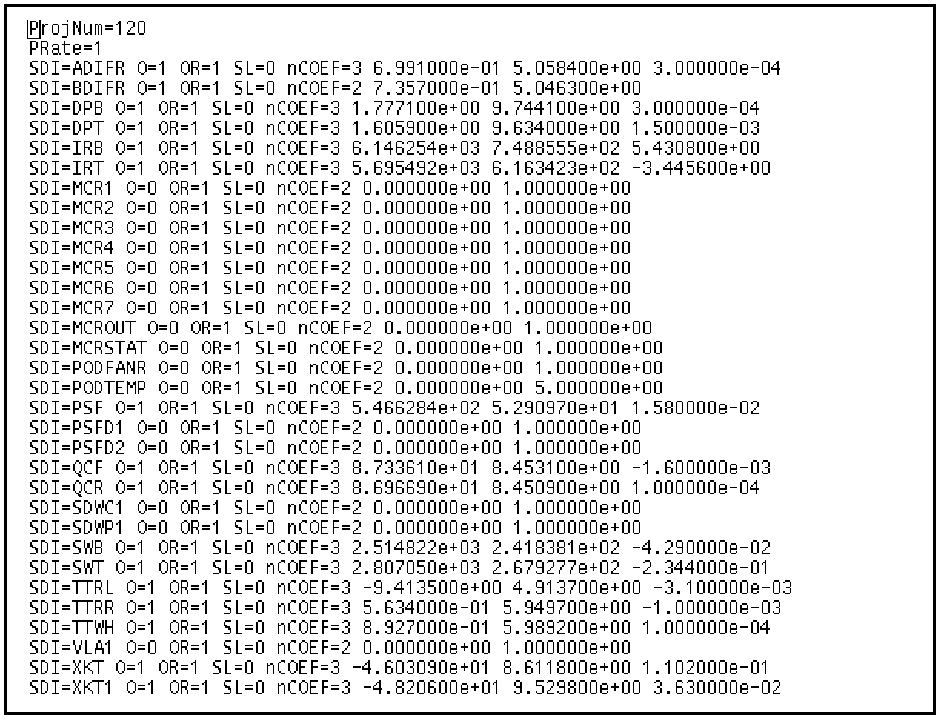
NOTE: *The* **FlightGroups** *file is optional. You can add or change the flight numbers in this file.*

NOTE: A **#** *in column 1 denotes the line as a comment.* ***Nimbus*** *ignores any comment lines in this file.*

Using the `Flight_ft###` File

The **`Flight_ft###`** file (where ***`ft###`*** refers to a flight type and number) is an ASCII flight Production setup file. **`Flight_ft###`** is located in the project's **`Production`** directory. An excerpt from a **`Flight_ft###`** file is shown in Figure 8-22.

Figure 8-22 Flight_*`ft###`* File Excerpt



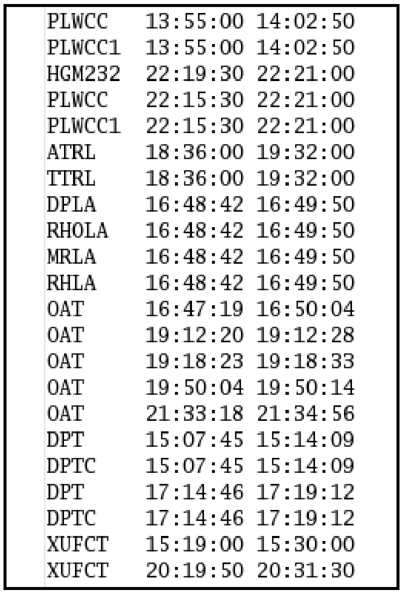
```
ProjNum=120
PRate=1
SDI=ADIFR O=1 OR=1 SL=0 nCOEF=3 6.991000e-01 5.058400e+00 3.000000e-04
SDI=BDIFR O=1 OR=1 SL=0 nCOEF=2 7.357000e-01 5.046300e+00
SDI=DPB O=1 OR=1 SL=0 nCOEF=3 1.777100e+00 9.744100e+00 3.000000e-04
SDI=DPT O=1 OR=1 SL=0 nCOEF=3 1.605900e+00 9.634000e+00 1.500000e-03
SDI=IRB O=1 OR=1 SL=0 nCOEF=3 6.146254e+03 7.488555e+02 5.430800e+00
SDI=IRT O=1 OR=1 SL=0 nCOEF=3 5.695492e+03 6.163423e+02 -3.445600e+00
SDI=MCR1 O=0 OR=1 SL=0 nCOEF=2 0.000000e+00 1.000000e+00
SDI=MCR2 O=0 OR=1 SL=0 nCOEF=2 0.000000e+00 1.000000e+00
SDI=MCR3 O=0 OR=1 SL=0 nCOEF=2 0.000000e+00 1.000000e+00
SDI=MCR4 O=0 OR=1 SL=0 nCOEF=2 0.000000e+00 1.000000e+00
SDI=MCR5 O=0 OR=1 SL=0 nCOEF=2 0.000000e+00 1.000000e+00
SDI=MCR6 O=0 OR=1 SL=0 nCOEF=2 0.000000e+00 1.000000e+00
SDI=MCR7 O=0 OR=1 SL=0 nCOEF=2 0.000000e+00 1.000000e+00
SDI=MCROUT O=0 OR=1 SL=0 nCOEF=2 0.000000e+00 1.000000e+00
SDI=MCRSTAT O=0 OR=1 SL=0 nCOEF=2 0.000000e+00 1.000000e+00
SDI=PODFANR O=0 OR=1 SL=0 nCOEF=2 0.000000e+00 1.000000e+00
SDI=PODTEMP O=0 OR=1 SL=0 nCOEF=2 0.000000e+00 5.000000e+00
SDI=PSF O=1 OR=1 SL=0 nCOEF=3 5.466284e+02 5.290970e+01 1.580000e-02
SDI=PSFD1 O=0 OR=1 SL=0 nCOEF=2 0.000000e+00 1.000000e+00
SDI=PSFD2 O=0 OR=1 SL=0 nCOEF=2 0.000000e+00 1.000000e+00
SDI=QCF O=1 OR=1 SL=0 nCOEF=3 8.733610e+01 8.453100e+00 -1.600000e-03
SDI=QCR O=1 OR=1 SL=0 nCOEF=3 8.696690e+01 8.450900e+00 1.000000e-04
SDI=SDWC1 O=0 OR=1 SL=0 nCOEF=2 0.000000e+00 1.000000e+00
SDI=SDWP1 O=0 OR=1 SL=0 nCOEF=2 0.000000e+00 1.000000e+00
SDI=SWB O=1 OR=1 SL=0 nCOEF=3 2.514822e+03 2.418381e+02 -4.290000e-02
SDI=SWT O=1 OR=1 SL=0 nCOEF=3 2.807050e+03 2.679277e+02 -2.344000e-01
SDI=TTRL O=1 OR=1 SL=0 nCOEF=3 -9.413500e+00 4.913700e+00 -3.100000e-03
SDI=TTRR O=1 OR=1 SL=0 nCOEF=3 5.634000e-01 5.949700e+00 -1.000000e-03
SDI=TTWH O=1 OR=1 SL=0 nCOEF=3 8.927000e-01 5.989200e+00 1.000000e-04
SDI=VLA1 O=0 OR=1 SL=0 nCOEF=2 0.000000e+00 1.000000e+00
SDI=XKT O=1 OR=1 SL=0 nCOEF=3 -4.603090e+01 8.611800e+00 1.102000e-01
SDI=XKT1 O=1 OR=1 SL=0 nCOEF=3 -4.820600e+01 9.529800e+00 3.630000e-02
```

CAUTION: The **`Flight_ft###`** *file is optional and* ***should not*** *be modified by the user.*

Using the `BlankOuts_ft###` File

The **`BlankOuts_ft###`** file (where ***`ft###`*** refers to a flight type and number) is an ASCII flight Production setup file. **`BlankOuts_ft###`** is located in the project's **`Production`** directory. This file allows you to set **`start_time`** and **`end_time`**, in hh:mm:ss format, for the selected variable, **`VarName`**. The system then replaces **`VarName`** with **`MISSING_VALUE`** for the indicated time period. An excerpt from a **`BlankOuts_`*`ft###`*** file is shown in Figure 8-23.

Figure 8-23 **`BlankOuts_`*`ft###`*** File Excerpt



```
PLWCC   13:55:00 14:02:50
PLWCC1  13:55:00 14:02:50
HGM232  22:19:30 22:21:00
PLWCC   22:15:30 22:21:00
PLWCC1  22:15:30 22:21:00
ATRL    18:36:00 19:32:00
TTRL    18:36:00 19:32:00
DPLA    16:48:42 16:49:50
RHOLA   16:48:42 16:49:50
MRLA    16:48:42 16:49:50
RHLA    16:48:42 16:49:50
OAT     16:47:19 16:50:04
OAT     19:12:20 19:12:28
OAT     19:18:23 19:18:33
OAT     19:50:04 19:50:14
OAT     21:33:18 21:34:56
DPT     15:07:45 15:14:09
DPTC    15:07:45 15:14:09
DPT     17:14:46 17:19:12
DPTC    17:14:46 17:19:12
XUFCT   15:19:00 15:30:00
XUFCT   20:19:50 20:31:30
```

Understanding the `vared` Support Program

Nimbus includes a support program called **`vared`**. **`vared`** is used to edit the variable database. The variable database can be edited for a specific file or, can be edited to affect all files. Depending on where you are and which file(s) you want to be affected, access **`vared`** using one of the instructions provided below:

❑ If you are currently in a project's home directory, simply enter **`vared VarDB`** at the system prompt. The **`VarDB`** file contains a list of all variables along with their associated metadata. The metadata includes units, titles, minimum and maximum values, category and standard name (see Unidata CF-metadata for the description of standard names).

- The Variable Database Editor window displays all variables associated with the current project file.

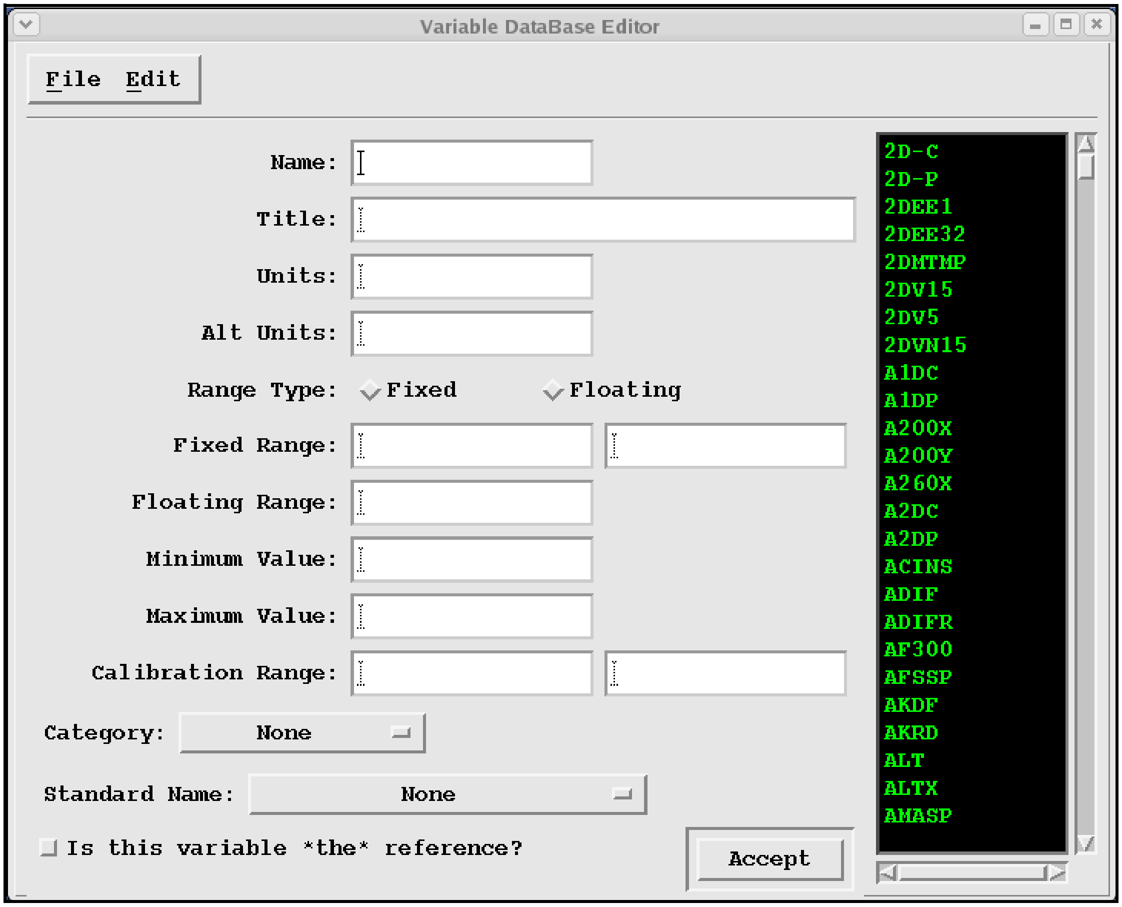
❑ If you are not currently in the project's home directory, access vared by entering **`${PROJ_DIR}/nnn/VarDB`** at the system prompt, where **`nnn`** is the project number.

- The Variable Database Editor window displays all variables associated with that project.

❑ If you want the edit the master file, therefore having the edits affect all files, ensure that you are in the **`${PROJ_DIR}/Configuration/raf`** directory and enter **`vared VarDB.`**

- The Variable Database Editor window shown in Figure 8-24 displays all default variables.

Figure 8-24 Variable Database Editor Window



Category: None

Standard Name: None

□ Is this variable *the* reference?

2D-C
2D-P
2DEE1
2DEE32
2DMTMP
2DV15
2DV5
2DVN15
A1DC
A1DP
A200X
A200Y
A260X
A2DC
A2DP
ACINS
ADIF
ADIFR
AF300
AFSSP
AKDF
AKRD
ALT
ALTX
AMASP

MODIFYING DATABASE VARIABLES

You can modify existing database variables by following the instructions below:

1. Double-click on the desired variable located in the right-most portion of the window.

The parameters associated with the selected variable are displayed in the associated fields.

2. Position the cursor in the desired field and enter the new value.

3. Select the desired option in the **Category** pull-down.

4. Select the desired options in the **Standard Name** pull-down.

---

*NOTE: The lists displayed in the **Category** and **Standard Name** pull-downs are configurable by editing the* **`Categories`** *or* **`StandardNames`** *files in the* **`/home/local/proj/ff###`** *directory.*

---

5. If the **Standard Name** is the best of a group of variables, check the box in the **Is this variable "the" reference?** field.

Being the best of a group of variables means that if you assign **`latitude`** as the **Standard Name**, the system assigns **`latitude`** to all of the latitude variables in the data set (LAT, GLAT, CLAT, LATC).

---

*NOTE: If you change your mind and wish to reset the variable to it's original state, press **Ctrl-R** or choose the **Reset Variable** option found under the **Edit** button. This action must be taken **prior** to clicking on the **Accept** button.*

---

6. When all modifications have been made, click on the **Accept** button to update the database with the new variable parameters.

ADDING DATABASE VARIABLES

You can add a new variable to the database by following the instructions below:

1. Clear the current variable, if one exists, by pressing **Ctrl-C** or selecting the **Clear Variable** option found under the **Edit** button.

2. Enter a new variable name and values in the applicable fields of the window and click on the **Accept** button.

   The new variable is added to the list in the right-most portion of the window.

DELETING DATABASE VARIABLES

Delete variables from the database by following the instructions below:

1. Click on the variable you want to delete.

2. Press **Ctrl-D** or select the **Delete Variable** option under the **Edit** button.

   The Variable is removed from the variable list at the right of the window.

SAVING DATABASE VARIABLES

When you have completed the changes to the variable database, save the file under its original name or a new name by following the instructions below:

1. To save the file under it's original name, press **Ctrl-S** or select the **Save** option found under the **File** button.

2. To save the file under a new name, press **Ctrl-A** or select the **Save As** option under the **File** button.

   The Select a File window appears.

3. Enter the new file name and click on **OK**.

OPENING A NEW VARIABLE DATABASE

You can open a new project file variable database by following the instructions below:

1. Press **Ctrl-O** or select the **Open New File** option under the **File** button.

   The Select a File window pops up.

2. Select the desired directory in the Directory Dialog, select the VarDB file from the Files Dialog and click on **OK**.

The Variable DataBase Editor window is updated with variables associated with the new file.

INTENTIONALLY
LEFT
BLANK