University of Wyoming King Air/LPVEX Data

Release Notes for {LPVEX_QC13}

1. Introduction

The following documents the release of the data from the University of Wyoming King Air (UWKA) for the Light Precipitation Validation Experiment (LPVEX) conducted in southern Finland in September/October 2010. Details of the experiment can be found on the main LPVEX project webpage:

http://lpvex.atmos.colostate.edu

and/or the UWKA project webpage:

http://flight.uwyo.edu/projects/lpvex10

The UW contact for this data set is:

Jeff French
UWKA Project Manager
Dept. Atmospheric Science
University of Wyoming
jfrench@uwyo.edu
307-766-4143

The Principal Investigator for the UWKA in LPVEX is:

Tristan L'Ecuyer
Dept. Atmospheric Science
University of Wisconsin
tlecuyer@wisc.edu

Questions related to data quality, availability, instrument & measurement limitations, etc., should be addressed to Jeff French, UWKA project manager. Questions related to data usage should be addressed to Tristan L'Ecuyer, UWKA PI for LPVEX.

2. General Information for Research Flights and Data Release

From September 16 to October 20, the UWKA conducted 16 research flights as part of LPVEX. A table listing basic information from each flight, including date, takeoff/landing time, brief description of weather, crew, and any significant instrument issues can be found on the UWKA LPVEX Project webpage:

http://flights.uwyo.edu/projects/lpvex10/

Data released on 10 October, 2011 represents the most recent release of the processed data for the UWKA data set. All of the processed data are in netCDF format and can be accessed through use of standard NCAR/NSF developed tools such as ncplot or aeros, or can be accessed through netcdf library routines in programming languages such as C, FORTRAN, MATLAB, or IDL. To ensure that the user has the most recent/up-to-date version of the UWKA processed data, **he/she should check the netCDF header within the global attributes for the following tag: LPVEX_QC13**. The tag will be listed in several locations within the global attributes, any "QC number" less than 13 indicates an earlier version of the processed data. Such files should be discarded and the user should download an updated version of the data set.

Data Breaks during Flight

During several flights, the data system needed to be restarted during flight. For these cases, this leads to a roughly 2 minute data-gap during which time no data were being collected. For the current release all the data are continuous for a given flight (ie, there is one file per flight and one second per record, records are continuous from start to end {no seconds are "skipped"}). During that time when the data system was being re-initialized in flight all variables (except TIME) will be set to fill-value numbers, to indicate the data is missing for that period.

The following table lists by flight number and day the number of data gaps, for flights not listed, there are no data gaps {Note-the data system restart generally took place between legs, before or after spirals to reduce impact on "usable" data}:

Flight Number/Date	Number of Data Gaps
RF01/16 Sept	2 gaps
RF03/19 Sept	2 gaps
RF04/21 Sept	3 gaps
RF05/23 Sept	1 gap
RF10/10 Oct	1 gap
RF11/12 Oct	1 gap
RF15/19 Oct	1 gap
RF16/20 Oct	1 gap

Specific Flight and Instrument Issues

Of the 16 research flights, 3 had significant data/instrument issues:

- RF02/18 Sept—mission aborted shortly into flight due to mechanical issues with the aircraft. It is unlikely any useful data were collected on this flight.
- RF12/14 Oct—Data drive filled early in the flight (about 45 minutes after takeoff). Following the filling of the drive, no UWKA data are available. Upto the time the drive filled, all data are available. WCR data from this entire flight are available. The WCR data are geo-referenced using separately-recorded GPS. Reflectivity is usable, Doppler velocity is not.
- RF14/17 Oct—WCR modulator failed on startup. No WCR data for this flight. All microphysics/other UWKA data are usable.

Less significant issues relating to specific flight and/or instruments are listed below:

- CIP (DMT Cloud Imaging Probe)—the program used to collect data from the DMT CIP probe would occasionally crash during flight. This in turn caused the computer to reboot. It often happened during periods of extremely high particle concentration as a result of the probe/program being unable to handle extreme data rates. This occurred on 4 flights. These data gaps are not apparent in the CIP processed variables—ie the values are simply 0 for these time periods. The following table lists data gaps (times) from the CIP based on flight day/number.

Flight Number/Date	CIP data gaps (time)
RF04/21 Sept	~1115 to ~1123
RF11/12 Oct	~0954 to ~0956
RF13/16 Oct	~0950 to ~0955 AND ~1035 to ~1039
RF15/19 Oct	~1108 to ~1111

- RF1 – RF11: PCASP data is suspect at high altitudes (cold temperatures). A faulty deice circuit led to spurious signals within the PCASP. Users of the PCASP data should take care interpreting data from this probe for the first 2/3 of the flights conducted. Lower altitudes were generally considered OK, and some of the high altitude legs are also OK (the problem was intermittent). Users should consider using measurements from the CPC to evaluate PCASP measurements. NOTE-PCASP measures are always suspect in-cloud due to shattering on sample tip.
- RF08—CPC and LICOR shutdown for the last ~1/3 of flight due to faulty pump. This effects cpc particle concentration and Licor dewpoint.

3. Information on Specific Measurements/Variables

The UWKA system produces several redundant measurements. Some are based on different processing methods from the same instrument. Others are based on different instruments, making the same measurement, at times with similar technology/methodology, at other times with a very different methodology. It is up to the user to determine under what conditions which measurement may provide the more accurate result. This section aims to provide guidance to the user for determining the appropriate usage of variables for a given measurement type. Below is a brief discussion of variable types and names. The following pages contain information from an aircraft data file header listing the variable names broken into measurement types (and some recommendations on which variables to use).

Time:

There exist several time variables. All should be consistent with one another and are based on UTC time (not GPStime). The user should choose whichever variable suits his/her needs in terms of variable format

Static Pressure:

Static pressure is measured from just one location on the aircraft, but is plumbed into several sensors. Because this is a very basic and crucial measure that is needed for just about every other measurement on the aircraft, there is a lot of built in redundancy. During LPVEX our primary measurement worked the entire project. Users should use ‘ps_hads_a’ for static pressure.

Altitude:

There are several different measures of altitude, including those derived from GPS measurements and those derived from integrating the atmosphere using pressure measurements. Users should choose a variable suitable for his/her purpose.

Position:

Position is provided by GPS and INS. Raw INS is subject to significant drift error. GPS tends to be slow (~ 1 sec). For higher frequency calculations GPS measures are complimentary filtered with INS data.

Airspeed:

The primary airspeed comes from AIAS (A-probe Indicated Airspeed), our secondary measurement is BIAS. BIAS was not working correctly in LPVEX. All data requiring airspeed are based on AIAS. True airspeed (TAS) was determined from AIAS and temperature.

Aircaft State/Attitude:

All of these measures are provided by the INS. No correction is made to the raw output in the files.

Horizontal Wind:

There are several horizontal wind variables to choose from. They are consistent with one another and are provided for the user to choose the appropriate variable for his/her application.

Vertical Wind:

Vertical wind is provided as unfiltered (hw) and filtered (hwf). The filtered methodology is provided in an attempt to remove long term drift (on the order of ~0.4 m/s) due to changes in aircraft fuel load through the flight that lead to changes in upwash and ultimately reduction in accuracy (not precision) of the calculation.

Static Temperature:

Two probes (Rosemount and Reverse Flow) provide measures of temperature. Reverse Flow is faster response and somewhat less susceptible to wetting. Use 'trf'.

Moisture Variables:

Two probes are used to measure water vapor in the atmosphere. The chilled mirror is a direct measure of the dewpoint by maintaining a thin layer of "dew" on a mirror and measuring the temperature of the mirror. This is the most accurate measurement, but tends to be slow response (several seconds, worse when colder). Once frost forms (typically at a temperature between -10 and -20 C), the measure is the frost point, rather than the dew point, and the response tends to slow. The Licor pulls an air sample into the aircraft and measures the number of water vapor molecules in a given volume. The response is much faster than the chilled mirror, but abosolute accuracy is considerably worse, especially at cold temperatures. The LICOR is susceptible to errors due to changing cabin pressure, but these tend to be less than ~1-2 C in dewpoint.

Cloud Droplet and Liquid Water Content

In LPVEX there were 5 different probes providing some sort of measurement of cloud liquid water content. It is expected the user is familiar with these measurements, here we simply highlight some basic limitations, known problems, and issues specific to UWKA processing.

<u>FSSP</u>-The FSSP provides "raw" measures of spectra, liquid water content, and concentration that are calculated by the probe. In addition, UW processing uses a statistical model developed by Brenguier to correct for coincidence and sampling issues. These "improved" measures of total concentration and fssp-derived liquid water contents should be used. Choose methodology #2, these are variables "jlb_lwc2_..." and "jlb_conc2_...". In comparisons with all other probes, the FSSP consistently predicted the high CLWCs throughout LPVEX. It is expected the probe was oversizing roughly 1 bin (3 microns).

CDP-The CDP also provides raw measures of spectra and liquid water content. No corrections are attempted to these data for coincidence. Following the completion of LPVEX the probe was sent to DMT for evaluation at which time it was noted by the DMT engineers that there were potential problems at cold temperatures—leading to baseline drift and miscalculation of droplet concentration. This appeared to be a problem with ALL DMT CDPs. Comparison of CDP with other CLWCs from LPVEX indicated that this problem, if it occurred, was likely only apparent in portions of the last few flights.

Gerber PVM-The Gerber Particle Volume Monitor provides a direct measure of cloud liquid water content and cloud droplet surface area, for particles up to about 50-70 microns. The processed data are corrected for baseline drift. Significant departures from reality can occur (even in the processed data), following descents or ascents in moist conditions due to fogging of the optics.

DMT-LWC100-The DMT LWC-100 is a hot wire probe, similar to the King/CSIRO probe. The data are processed using a reasonably sophisticated mathematical model to remove the significant dry-air contribution to wire cooling. Because of long in-cloud legs, some of the data from LPVEX suffers from an inability to retrieve an accurate estimate of the clear air term. This results in baseline drift and a reduction in accuracy of the measurement. The user should look for regions outside of cloud to ensure there is not significant drift in this measure.

Nevzorov-The Nevzorov probe was provided by Environment Canada for use on the UWKA during LPVEX. The data were processed and made available by Alexei Korolev and Dave Hudak. The data were merged into the latest release of the UWKA data set.

Cloud and Precipitation Particles

During LPVEX the UWKA carried a modified 2DC (similar to a CIP, except in sample volume) and a 2DP. A basic processing of these data are provided in the UWKA data files. We provide measures of total particle concentration and size spectra using 3 different methodologies: (0) using IX for diameter, (1) using IY for diameter {All-in}, and (2) using max(IX,IY) for diameter {All-in}. For each methodology, we attempt to remove artifacts such as streakers and splashes using simple algorithms. We do not correct for particle shattering, etc.

UWKA Processed Variables for LPVEX

Table of Contents:

VARIABLES:

Raw Variables-

Variables that contain **_RAW** in their name can be ignored, these are used for QC

Time Variables-

```
double time(time) ;
     time:units = "seconds since 2011-01-01 00:00:00 +0000" ;
     time:standard_name = "time" ;
     time:strptime_format = "seconds since %F %T %z" ;
     time:MissingValues = 0 ;
float TIME(time) ;
     TIME:long_name = "Time HHMMSS (GMT)" ;
     TIME:units = "HHMMSS" ;
float DATE(time) ;
     DATE:long_name = "Date yymmdd (GMT)" ;
     DATE:units = "yyyymmdd" ;
float HOUR(time) ;
     HOUR:long_name = "Hour from midnight (GMT)" ;
     HOUR:units = "hours" ;
float MINUTE(time) ;
     MINUTE:long_name = "Minute from Beginning of HOUR" ;
     MINUTE:units = "minutes" ;
float SECOND(time) ;
     SECOND:long_name = "SECOND from Beginning of SECOND" ;
     SECOND:units = "seconds" ;
double TIME14D(time) ;
     TIME14D:long_name = "TIME yyyymmddhhmmss GMT" ;
     TIME14D:units = "yyyymmddhhmmss" ;
```

Static Pressure Variables-(use ps_hads_a)

```
float ps_weston(time) ;
     ps_weston:long_name = "Static Pressure (Weston Digital)" ;
     ps_weston:units = "millibar" ;
float ps_hads_a(time) ;
     ps_hads_a:long_name = "Static Pressure (Rosemount 1501 High
           Accuracy Digital Sensing Module A)" ;
     ps_hads_a:units = "millibar" ;
float ps_hads_b(time) ;
     ps_hads_b:long_name = "Static Pressure (Rosemount 1501 High
           Accuracy Digital Sensing Module B)" ;
     ps_hads_b:units = "millibar" ;
```

Altitude Variables- (Take your pick…)

```
float z(time) ;
     z:long_name = "Pressure altitude (Std Atm)" ;
     z:units = "meters" ;
float PALT(time) ;
     PALT:long_name = "Pressure altitude (Std Atm)" ;
     PALT:units = "meters" ;
float ztrue(time) ;
     ztrue:long_name = "Altitude (Hypsometric)" ;
     ztrue:units = "meters" ;
float GALT(time) ;
     GALT:long_name = "Altitude  GPS" ;
     GALT:units = "m" ;
float galt(time) ;
     galt:long_name = "Altitude  GPS" ;
     galt:units = "m" ;
float hi3(time) ;
     hi3:long_name = "Height (inertial-baro)" ;
     hi3:units = "meters" ;
float ralt1(time) ;
     ralt1:long_name = "Radar altitude (King)" ;
     ralt1:units = "meter" ;
float topo(time) ;
     topo:long_name = "Topography from database" ;
     topo:units = "m" ;
     topo:OutputRate = 1 ;
```

Position Variables- (use LATC & LONC)

```
double LATC(time) ;
     LATC:long_name = "Latitude (IRS, gps corrected) " ;
     LATC:units = "degree_N" ;
double LONC(time) ;
     LONC:long_name = "Longitude (IRS gps corrected)" ;
     LONC:units = "degree_E" ;
float LAT(time) ;
     LAT:long_name = "Latitude (not corrected with GPS)" ;
     LAT:units = "degree_N" ;
float LON(time) ;
     LON:long_name = "Longitude (not corrected with GPS)" ;
     LON:units = "degree_E" ;
float hlat(time) ;
     hlat:long_name = "Latitude (uncorrected, IRS)" ;
     hlat:units = "degree_N" ;
float hlon(time) ;
     hlon:long_name = "Longitude (uncorrected, IRS)" ;
     hlon:units = "degree_E" ;
float GLAT(time) ;
     GLAT:long_name = "Latitude  GPS" ;
     GLAT:units = "degree_N" ;
float GLON(time) ;
     GLON:long_name = "Longitude  GPS" ;
     GLON:units = "degree_E" ;
float glat(time) ;
     glat:long_name = "Latitude  GPS" ;
     glat:units = "degree_N" ;
float glon(time) ;
     glon:long_name = "Longitude  GPS" ;
     glon:units = "degree_E" ;
float xdist(time) ;
     xdist:long_name = "Position (east)" ;
     xdist:units = "km" ;
float ydist(time) ;
     ydist:long_name = "Position (north)" ;
     ydist:units = "km" ;
```

Airspeed Variables- (AIAS and BIAS are the basic measurements, we use AIAS and TASX in calculations)

```
float aias(time) ;
     aias:long_name = "Indicated airspeed (boom pitot)" ;
     aias:units = "knots" ;
float bias(time) ;
     bias:long_name = "Indicated airspeed (co-pilot pitot)" ;
     bias:units = "knots" ;
float tas(time) ;
     tas:long_name = "True Airspeed " ;
     tas:units = "m/s" ;
float TASX(time) ;
     TASX:long_name = "True Airspeed (same as tas)" ;
     TASX:units = "m/s" ;
```

Aircraft State/Attitude Variables-

```
float htrk(time) ;
    htrk:long_name = "Track angle (IRS uncorrected)" ;
    htrk:units = "degree_T" ;
float hgs(time) ;
    hgs:long_name = "Ground Speed (IRS, uncorrected)" ;
    hgs:units = "m/s" ;
float hpitch(time) ;
    hpitch:long_name = "Pitch angle " ;
    hpitch:units = "degree" ;
float hroll(time) ;
    hroll:long_name = "Roll angle" ;
    hroll:units = "degree" ;
float hthead(time) ;
    hthead:long_name = "Heading angle (true)" ;
    hthead:units = "degree_T" ;
float hewvel(time) ;
    hewvel:long_name = "Inertial ground speed (E-W component,
          uncorrected)" ;
    hewvel:units = "m/s" ;
float hnsvel(time) ;
    hnsvel:long_name = "Inertial ground speed (N-S component,
          uncorrected)" ;
    hnsvel:units = "m/s" ;
float hwp3(time) ;
    hwp3:long_name = "Vertical speed (inertial-baro)" ;
    hwp3:units = "m/s" ;
float gvns(time) ;
    gvns:long_name = "Ground velocity component North GPS" ;
    gvns:units = "m/s" ;
float gvew(time) ;
    gvew:long_name = "Ground velocity component East GPS" ;
    gvew:units = "m/s" ;
float gvz(time) ;
    gvz:long_name = "Ground velocity component Vertical GPS" ;
    gvz:units = "m/s" ;
```

Horizontal Wind Variables-

```
float hwind_qflag(time) ;
     hwind_qflag:long_name = "Horizontal Wind Correction Quality flag
            (0=accept, -1=reject)" ;
     hwind_qflag:units = "number" ;
float hwdir(time) ;
     hwdir:long_name = "Wind direction (from)" ;
     hwdir:units = "degree_T" ;
float hwmag(time) ;
     hwmag:long_name = "Wind magnitude" ;
     hwmag:units = "m/s" ;
float hu(time) ;
     hu:long_name = "Wind component (East)" ;
     hu:units = "m/s" ;
float hv(time) ;
     hv:long_name = "Wind component (North)" ;
     hv:units = "m/s" ;
float ux(time) ;
     ux:long_name = "Wind component (horizontal longitudinal)" ;
     ux:units = "m/s" ;
float vy(time) ;
     vy:long_name = "Wind component (horizontal lateral)" ;
     vy:units = "m/s" ;
```

Vertical Wind Variables-(hwf is preferred vertical wind—removes long term variation)

```
float hw(time) ;
     hw:long_name = "Wind component (Vertical)" ;
     hw:units = "m/s" ;
float hwf(time) ;
     hwf:long_name = "Wind component (Vertical, high pass filtered)" ;
     hwf:units = "m/s" ;
float turb(time) ;
     turb:long_name = "MRI Eddy dissipation rate ^1/3 u-component
            (corrected)" ;
     turb:units = "MKS" ;
```

Static Temperature Variables-(use trf)

```
float trf(time) ;
     trf:long_name = "Static Temperature (In-house Reverse Flow)" ;
     trf:units = "Celsius" ;
float trose(time) ;
     trose:long_name = "Static Temperature (Rosemount 102)" ;
     trose:units = "Celsius" ;
```

Moisture Variables-(tdp is from chilled mirror (slow); tdp_licor is from licor (fast))

```
float tdp(time) ;
     tdp:long_name = "Dew Point Temperature" ;
     tdp:instrument = "EdgeTech Vigilant model 137" ;
     tdp:units = "Celsius" ;
float tdplicor(time) ;
     tdplicor:long_name = "Dew point temperature from LICOR H2O mixing
            ratio" ;
     tdplicor:units = "Celsius" ;
float mr(time) ;    (Chilled mirror)
     mr:long_name = "Mixing ratio" ;
     mr:units = "gram/kgram" ;
float rh(time) ;     (Chilled mirror)
     rh:long_name = "Relative Humidity" ;
     rh:units = "percent" ;
float h2o1s(time) ;    (Licor)
     h2o1s:long_name = "Licor H2O concentration" ;
     h2o1s:units = "volt" ;
float h2oml(time) ;    (Licor)
     h2oml:long_name = "H2O mole fraction LICOR" ;
     h2oml:units = "mmol/mole" ;
float h2omx(time) ;    (Licor)
     h2omx:long_name = "H2O mixing ratio LICOR" ;
     h2omx:units = "gram/kgram" ;
```

Other Atmospheric State Variables-(Derived from above measurements)

```
float thetad(time) ;
     thetad:long_name = "Potential temperature (dry)" ;
     thetad:units = "K" ;
float thetae(time) ;
     thetae:long_name = "Equivalent potential temperature" ;
     thetae:units = "K" ;
```

CO2 Variables-(all from the licor)

```
float co21s(time) ;
     co21s:long_name = "Licor CO2 concentration" ;
     co21s:units = "volt" ;
float co2ml(time) ;
     co2ml:long_name = "CO2 mole fraction LICOR" ;
     co2ml:units = "umol/mole" ;
float co2mx(time) ;
     co2mx:long_name = "CO2 mixing ratio LICOR" ;
     co2mx:units = "ugram/gram" ;
float co2pbmx(time) ;
     co2pbmx:long_name = "CO2 mixing ratio LICOR (corrected for press
            broading)" ;
     co2pbmx:units = "ugram/gram" ;
float co2pbmld(time) ;
     co2pbmld:long_name = "CO2 mole fraction (dry) LICOR (corrected for
           press broading)" ;
     co2pbmld:units = "umol/mole" ;
```

Cloud Droplet Liquid Water Content-

```
float lwc100(time) ;
     lwc100:long_name = "Liquid water content (DMT100)" ;
     lwc100:units = "gram/m3" ;
float rlwc(time) ;  (ONLY WORKS IN CLOUDS BELOW 0C)
     rlwc:long_name = "Liquid water content from Rosemount 871 icing
           probe" ;
     rlwc:units = "gram/m3" ;
float pvmlwc(time) ;
     pvmlwc:long_name = "PVM-100A (Gerber) liquid water content" ;
     pvmlwc:units = "gram/m3" ;
float jlb_lwc2_IBL(time) ;
     jlb_lwc2_IBL:long_name = "FSSP liquid water content (JLB) method 2"
     jlb_lwc2_IBL:units = "gram/m3" ;
float jlb_lwc3_IBL(time) ;
     jlb_lwc3_IBL:long_name = "FSSP liquid water content (JLB) method 3"
     jlb_lwc3_IBL:units = "gram/m3" ;
float jlb_lwc4_IBL(time) ;
     jlb_lwc4_IBL:long_name = "FSSP liquid water content (JLB) method 4"
     jlb_lwc4_IBL:units = "gram/m3" ;
float cdplwc_NRB(time) ;
     cdplwc_NRB:long_name = "DMT CDP Liquid Water Content" ;
     cdplwc_NRB:units = "g m-3" ;
float PLWCF_IBL(time) ; (RAW FSSP variable)
     PLWCF_IBL:units = "gram/m3" ;
     PLWCF_IBL:long_name = "FSSP-100 Water Content" ;
float nevlwc(time) ;
     nevlwc:long_name = "Nevzorov Liquid Water Content" ;
     nevlwc:Category = "Derived" ;
     nevlwc:units = "gram/m3" ;
     nevlwc:SampleRate = 1000 ;
     nevlwc:type = "analog" ;
     nevlwc:Package = "Supplied by Alexei" ;
     nevlwc:OutputRate = 1 ;
```

```
float nevtwc(time) ;
     nevtwc:long_name = "Nevzorov Total (condensed) Water Content" ;
     nevtwc:Category = "Derived" ;
     nevtwc:units = "gram/m3" ;
     nevtwc:SampleRate = 1000 ;
     nevtwc:type = "analog" ;
     nevtwc:Package = "Supplied by Alexei" ;
     nevtwc:OutputRate = 1 ;
```

Cloud Droplet Concentration-

```
float cdpconc_NRB(time) ;
     cdpconc_NRB:long_name = "DMT CDP Total Concentration" ;
     cdpconc_NRB:units = "cm-3" ;
float jlb_conc2_IBL(time) ;
     jlb_conc2_IBL:long_name = "FSSP droplet concentration (JLB) method
            2 " ;
     jlb_conc2_IBL:units = "cm-3" ;
float jlb_conc3_IBL(time) ;
     jlb_conc3_IBL:long_name = "FSSP droplet concentration (JLB) method
            3 " ;
     jlb_conc3_IBL:units = "cm-3" ;
float jlb_conc4_IBL(time) ;
     jlb_conc4_IBL:long_name = "FSSP droplet concentration (JLB) method
            4 " ;
     jlb_conc4_IBL:units = "cm-3" ;
float CONCF_IBL(time) ;  (RAW FSSP variable)
     CONCF_IBL:units = "cm-3" ;
     CONCF_IBL:long_name = "FSSP-100 Concentration (all cells)" ;
```

Gerber PVM – specific Variables-

```
float pvmre_c(time) ;
     pvmre_c:long_name = "PVM-100A (Gerber) effective radius (computed
            from lwc and psa)" ;
     pvmre_c:units = "micrometer" ;
float pvmpsa(time) ;
     pvmpsa:long_name = "PVM-100A (Gerber) particle surface area" ;
     pvmpsa:units = "cm2/m3" ;
```

DMT CDP – specific Variables-

size spectra {for ACDP and CCDP see full header info for cell sizes}

```
float ACDP_NRB(time, sps1, Vec31) ;
     ACDP_NRB:long_name = "DMT CDP number (per cell)" ;
     ACDP_NRB:units = "number" ;
float CCDP_NRB(time, sps1, Vec31) ;
     CCDP_NRB:long_name = "DMT CDP concentration (per cell)" ;
     CCDP_NRB:units = "cm-3" ;
float cdpacc_NRB(time) ;
     cdpacc_NRB:long_name = "DMT CDP Total number" ;
     cdpacc_NRB:units = "number" ;
float cdpdbar_NRB(time) ;
     cdpdbar_NRB:long_name = "DMT CDP Mean Diameter" ;
     cdpdbar_NRB:units = "um" ;
```

PMS FSSP – specific Variables-

size spectra {for AFSSP and CFSSP see full header info for cell sizes}

```
 float AFSSP_IBL(time, sps1, Vec16) ;
     AFSSP_IBL:units = "counts" ;
     AFSSP_IBL:long_name = "FSSP-100 Raw Accumulation (per cell)" ;
 float CFSSP_IBL(time, sps1, Vec16) ;
     CFSSP_IBL:units = "cm-3" ;
     CFSSP_IBL:long_name = "FSSP-100 Accumulation (per cell)" ;
```

Derived

```
 float REffective_IBL(time) ;
     REffective_IBL:long_name = "FSSP-100  effective radius" ;
     REffective_IBL:units = "um" ;
 float SurfArea_IBL(time) ;
     SurfArea_IBL:long_name = "FSSP-100  Particle Surface Area" ;
     SurfArea_IBL:units = "cm2/m3" ;
 float RBarVol_IBL(time) ;
     RBarVol_IBL:long_name = "FSSP-100  Mean volume radius" ;
     RBarVol_IBL:units = "um" ;
```

Raw (little or no scientific use)

```
 float FRST_IBL(time) ;
     FRST_IBL:units = "counts" ;
     FRST_IBL:long_name = "FSSP-100 Fast Resets" ;
 float DBARF_IBL(time) ;
     DBARF_IBL:units = "um" ;
     DBARF_IBL:long_name = "FSSP-100 Mean Particle Diameter" ;
 float FBMFR_IBL(time) ;
     FBMFR_IBL:units = "fraction" ;
     FBMFR_IBL:long_name = "FSSP-100 Beam Fraction" ;
 float FACT_IBL(time) ;
     FACT_IBL:units = "none" ;
     FACT_IBL:long_name = "FSSP-100 Activity Fraction" ;
 float FSTB_IBL(time) ;
     FSTB_IBL:units = "counts" ;
     FSTB_IBL:long_name = "FSSP-100 Total Strobes" ;
 float FRNG_IBL(time) ;
     FRNG_IBL:units = "none" ;
     FRNG_IBL:long_name = "FSSP-100 Size Range Category" ;
 float FDOFFR_IBL(time) ;
     FDOFFR_IBL:units = "fraction" ;
     FDOFFR_IBL:long_name = "FSSP-100 DOF Fraction" ;
```

PMS/DMT CIP – specific Variables-

```
 float twodcip_IBR(time) ;
     twodcip_IBR:long_name = "DMT CIP *shadow OR* " ;
     twodcip_IBR:units = "#/liter" ;
     twodcip_IBR:Status = "CIP2D Derived Variable" ;
 float npart_cip_IBR(time) ;
     npart_cip_IBR:long_name = "DMT CIP # of particles" ;
     npart_cip_IBR:units = "#" ;
     npart_cip_IBR:Status = "CIP2D Derived Variable" ;
 float CONC0_cip_IBR(time) ;
     CONC0_cip_IBR:long_name = "DMT CIP concentration " ;
     CONC0_cip_IBR:method = "IX, svol corr IX/(64+IX)" ;
     CONC0_cip_IBR:units = "#/liter" ;
 float CONC1_cip_IBR(time) ;
     CONC1_cip_IBR:long_name = "DMT CIP concentration " ;
     CONC1_cip_IBR:method = "IY, all-in, svol corr IY/(62-IY)" ;
     CONC1_cip_IBR:units = "#/liter" ;
 float mass0_cip_IBR(time) ;
     mass0_cip_IBR:long_name = "DMT CIP mass concentration " ;
     mass0_cip_IBR:method = "IX, svol corr IX/(64+IX)" ;
     mass0_cip_IBR:units = "gram/m3" ;
 float mass1_cip_IBR(time) ;
     mass1_cip_IBR:long_name = "DMT CIP mass concentration " ;
     mass1_cip_IBR:method = "IY, all-in, svol corr IY/(62-IY)" ;
     mass1_cip_IBR:units = "gram/m3" ;
 float mass2_cip_IBR(time) ;
     mass2_cip_IBR:long_name = "DMT CIP mass concentration " ;
     mass2_cip_IBR:method = "max(IX,IY), all-in, svol corr IY/(62-IY)" ;
     mass2_cip_IBR:units = "gram/m3" ;
```

size spectra {for see full header info for cell sizes}

```
 float ACIP0_IBR(time, sps1, Vec101) ;
     ACIP0_IBR:long_name = "DMT CIP number per cell" ;
     ACIP0_IBR:method = "IX, no svol correction" ;
     ACIP0_IBR:units = "#" ;
 float CCIP0_IBR(time, sps1, Vec101) ;
     CCIP0_IBR:long_name = "DMT CIP number per cell " ;
     CCIP0_IBR:method = "IX, svol corr IX/(64+IX)" ;
     CCIP0_IBR:units = "#/liter" ;
 float CCIPsz0_IBR(time, sps1, Vec16) ;
     CCIPsz0_IBR:long_name = "DMT CIP number per cell " ;
     CCIPsz0_IBR:method = "IX, svol corr IX/(64+IX)" ;
     CCIPsz0_IBR:units = "#/liter" ;
 float ACIP1_IBR(time, sps1, Vec62) ;
     ACIP1_IBR:long_name = "DMT CIP number per cell" ;
     ACIP1_IBR:method = "IY, all-in, no svol correction" ;
     ACIP1_IBR:units = "#" ;
 float CCIP1_IBR(time, sps1, Vec62) ;
     CCIP1_IBR:long_name = "DMT CIP number per cell " ;
     CCIP1_IBR:method = "IY, all-in, svol corr IY/(62-IY)" ;
     CCIP1_IBR:units = "#/liter" ;
 float CCIPsz1_IBR(time, sps1, Vec14) ;
     CCIPsz1_IBR:long_name = "DMT CIP number per cell " ;
     CCIPsz1_IBR:method = "IY, all-in, svol corr IY/(62-IY)" ;
     CCIPsz1_IBR:units = "#/liter" ;
```

```
float ACIP2_IBR(time, sps1, Vec101) ;
    ACIP2_IBR:long_name = "DMT CIP number per cell" ;
    ACIP2_IBR:method = "max(IX,IY), IY all-in, no svol correction" ;
    ACIP2_IBR:units = "#" ;
float CCIP2_IBR(time, sps1, Vec101) ;
    CCIP2_IBR:long_name = "DMT CIP number per cell " ;
    CCIP2_IBR:method = "max(IX,IY), IY all-in, svol corr IY/(62-IY)" ;
    CCIP2_IBR:units = "#/liter" ;
float ACIPsz2_IBR(time, sps1, Vec16) ;
    ACIPsz2_IBR:long_name = "DMT CIP number per cell " ;
    ACIPsz2_IBR:method = "max(IX,IY), IY all-in, svol corr IY/(62-IY)"
          ;
    ACIPsz2_IBR:units = "#" ;
float CCIPsz2_IBR(time, sps1, Vec16) ;
    CCIPsz2_IBR:long_name = "DMT CIP number per cell " ;
    CCIPsz2_IBR:method = "max(IX,IY), IY all-in, svol corr IY/(62-IY)"
          ;
    CCIPsz2_IBR:units = "#/liter" ;
```

PMS TwoDP – specific Variables-

```
 float twodp(time) ;
      twodp:long_name = "2DP shadow OR concentration" ;
      twodp:units = "liter-1" ;
 float npart_2dp_OBL(time) ;
      npart_2dp_OBL:long_name = "PMS 2D-P # of particles" ;
      npart_2dp_OBL:units = "#" ;
      npart_2dp_OBL:Status = "PMS2D Derived Variable" ;
 float CONC0_2dp_OBL(time) ;
      CONC0_2dp_OBL:long_name = "PMS 2D-P concentration" ;
      CONC0_2dp_OBL:method = "IX, svol corr IX/(32+IX) " ;
      CONC0_2dp_OBL:units = "#/liter" ;
 float CONC1_2dp_OBL(time) ;
      CONC1_2dp_OBL:long_name = "PMS 2D-P concentration" ;
      CONC1_2dp_OBL:method = "IY, all-in, svol corr IY/(30-IY)" ;
      CONC1_2dp_OBL:units = "#/liter" ;
 float mass0_2dp_OBL(time) ;
      mass0_2dp_OBL:long_name = "PMS 2D-P mass concentration" ;
      mass0_2dp_OBL:method = "IX, svol corr IX/(32+IX) " ;
      mass0_2dp_OBL:units = "gram/m3" ;
 float mass1_2dp_OBL(time) ;
      mass1_2dp_OBL:long_name = "PMS 2D-P mass concentration" ;
      mass1_2dp_OBL:method = "IY, all-in, svol corr IY/(30-IY)" ;
      mass1_2dp_OBL:units = "gram/m3" ;
 float mass2_2dp_OBL(time) ;
      mass2_2dp_OBL:long_name = "PMS 2D-P mass concentration" ;
      mass2_2dp_OBL:method = "max(IX,IY), all-in, svol corr IY/(30-IY)" ;
      mass2_2dp_OBL:units = "gram/m3" ;
```

<u>*size spectra {see full header info for cell sizes}*</u>

```
 float A2DP0_OBL(time, sps1, Vec101) ;
      A2DP0_OBL:long_name = "PMS 2D_P number per cell" ;
      A2DP0_OBL:method = "IX, no svol correction" ;
      A2DP0_OBL:units = "#" ;
 float C2DP0_OBL(time, sps1, Vec101) ;
      C2DP0_OBL:long_name = "PMS 2D_P number per cell " ;
      C2DP0_OBL:method = "IX, svol corr IX/(32+IX)" ;
      C2DP0_OBL:units = "#/liter" ;
 float C2DPsz0_OBL(time, sps1, Vec20) ;
      C2DPsz0_OBL:long_name = "PMS 2D_P number per cell " ;
      C2DPsz0_OBL:method = "IX, svol corr IX/(32+IX)" ;
      C2DPsz0_OBL:units = "#/liter" ;
 float A2DP1_OBL(time, sps1, Vec30) ;
      A2DP1_OBL:long_name = "PMS 2D_P number per cell" ;
      A2DP1_OBL:method = "IY (all-in), no svol correction" ;
      A2DP1_OBL:units = "#" ;
 float C2DP1_OBL(time, sps1, Vec30) ;
      C2DP1_OBL:long_name = "PMS 2D_P number per cell " ;
      C2DP1_OBL:method = "IY (all-in), svol corr IY/(30-IY)" ;
      C2DP1_OBL:units = "#/liter" ;
 float C2DPsz1_OBL(time, sps1, Vec17) ;
      C2DPsz1_OBL:long_name = "PMS 2D_P number per cell " ;
      C2DPsz1_OBL:method = "IY (all-in), svol corr IY/(30-IY)" ;
      C2DPsz1_OBL:units = "#/liter" ;
 float A2DP2_OBL(time, sps1, Vec101) ;
```

```
    A2DP2_OBL:long_name = "PMS 2D_P number per cell" ;
    A2DP2_OBL:method = "max(IX,IY), no svol correction" ;
    A2DP2_OBL:units = "#" ;
float C2DP2_OBL(time, sps1, Vec101) ;
    C2DP2_OBL:long_name = "PMS 2D_P number per cell " ;
    C2DP2_OBL:method = "max(IX,IY), IY all-in, svol corr IY/(30-IY)" ;
    C2DP2_OBL:units = "#/liter" ;
float A2DPsz2_OBL(time, sps1, Vec20) ;
    A2DPsz2_OBL:long_name = "PMS 2D_P number per cell " ;
    A2DPsz2_OBL:method = "max(IX,IY), IY all-in, svol corr IY/(30-IY)"
          ;
    A2DPsz2_OBL:units = "#" ;
float C2DPsz2_OBL(time, sps1, Vec20) ;
    C2DPsz2_OBL:long_name = "PMS 2D_P number per cell " ;
    C2DPsz2_OBL:method = "max(IX,IY), IY all-in, svol corr IY/(30-IY)"
          ;
    C2DPsz2_OBL:units = "#/liter" ;
```

Total Aerosol Variables

```
float conc_cpc(time) ;
     conc_cpc:long_name = "Cloud Particle Counter (CPC)" ;
     conc_cpc:units = "cm-3" ;
```

DMT PCASP SPP200-Specific Variables

```
float AS200_OBR(time, sps1, Vec31) ;
     AS200_OBR:long_name = "PMS PCASP (UWYO/NOAA) number (per cell)" ;
     AS200_OBR:units = "number" ;
float CS200_OBR(time, sps1, Vec31) ;
     CS200_OBR:long_name = "PMS PCASP (UWYO/NOAA) concentration (per
            cell)" ;
     CS200_OBR:units = "cm-3" ;
float TCNTP_OBR(time) ;
     TCNTP_OBR:long_name = "PMS PCASP (UWYO/NOAA) number concentration "
            ;
     TCNTP_OBR:units = "number" ;
float CONCP_OBR(time) ;
     CONCP_OBR:long_name = "PMS PCASP (UWYO/NOAA) number concentration "
            ;
     CONCP_OBR:units = "cm-3" ;
float DBARP_OBR(time) ;
     DBARP_OBR:long_name = "PCASP (UWYO/NOAA) Mean Particle Diameter" ;
     DBARP_OBR:units = "um" ;
float PSFCP_OBR(time) ;
     PSFCP_OBR:long_name = "PCASP (UWYO/NOAA) surface area
            concentration" ;
     PSFCP_OBR:units = "um2/cm3" ;
float PVOLP_OBR(time) ;
     PVOLP_OBR:long_name = "PCASP (UWYO/NOAA) volume concentration" ;
     PVOLP_OBR:units = "um3/cm3" ;
float DISPP_OBR(time) ;
     DISPP_OBR:long_name = "PCASP (UWYO/NOAA) size dispersion" ;
     DISPP_OBR:units = " " ;
float PFLW_OBR(time) ;
     PFLW_OBR:long_name = "PCASP (UWYO/NOAA) sample flow (standard)" ;
     PFLW_OBR:units = "cm3/sec" ;
float PFLWC_OBR(time) ;
     PFLWC_OBR:long_name = "PCASP (UWYO/NOAA) sample volume (actual)" ;
     PFLWC_OBR:units = "cm3/sec" ;
float PFLWS_OBR(time) ;
     PFLWS_OBR:long_name = "PCASP (UWYO/NOAA) sheath flow (standard)" ;
     PFLWS_OBR:units = "cm3/sec" ;
```

Variables Used primarily for QC/QA—of little scientific interest

```
float licort(time) ;
    licort:long_name = "Licor Temperature" ;
    licort:units = "Celsius" ;
float licorp(time) ;
    licorp:long_name = "Licor Pressure" ;
    licorp:units = "kPa" ;


float dpr(time) ;
    dpr:long_name = "Rosemount 1332B1" ;
    dpr:units = "millibar" ;
float dpb(time) ;
    dpb:long_name = "Rosemount 1332B1" ;
    dpb:units = "millibar" ;
    dpb:MissingValues = 200 ;
float dpa(time) ;
    dpa:long_name = "Rosemount 1332B1" ;
    dpa:units = "millibar" ;
float boom_pcor(time) ;
    boom_pcor:long_name = "Static pressure correction from boom
          calculation" ;
float alpha(time) ;
    alpha:long_name = "Attack angle (corrected)" ;
    alpha:units = "radians" ;
float beta(time) ;
    beta:long_name = "Sideslip angle (corrected)" ;
    beta:units = "radians" ;


float hpitchr(time) ;
    hpitchr:long_name = "Pitch angle rate " ;
    hpitchr:units = "radian/sec" ;
float hrollr(time) ;
    hrollr:long_name = "Roll angle rate " ;
    hrollr:units = "radian/sec" ;
float hyawr(time) ;
    hyawr:long_name = "Yaw angle rate" ;
    hyawr:units = "radian/sec" ;
float hlata(time) ;
    hlata:long_name = "Lateral acceleration, body axis (IRS)" ;
    hlata:units = "g" ;
float hlonga(time) ;
    hlonga:long_name = "Longitudinal acceleration, body axis (IRS)" ;
    hlonga:units = "g" ;
float hnorma(time) ;
    hnorma:long_name = "Normal acceleration, body axis(IRS)" ;
    hnorma:units = "g" ;
float hivs(time) ;
    hivs:long_name = "Inertial vertical speed (IRS)" ;
    hivs:units = "m/s" ;
float hia(time) ;
    hia:long_name = "Inertial altitude (IRS)" ;
    hia:units = "m" ;
```

```
  float hacz3(time) ;
      hacz3:long_name = "Vertical acceleration (inertial-baro)" ;
      hacz3:units = "m/s2" ;

  float xerr(time) ;
      xerr:long_name = "Position error (x-component)" ;
      xerr:units = "km" ;
  float yerr(time) ;
      yerr:long_name = "Position error (y-component)" ;
      yerr:units = "km" ;
  float uerr(time) ;
      uerr:long_name = "Velocity error (x-component)" ;
      uerr:units = "m/s" ;
  float verr(time) ;
      verr:long_name = "Velocity error (y-component)" ;
      verr:units = "m/s" ;
```

Global Attributes for the file(s)

```
// global attributes:
                :FlightNumber = "RF01" ;
                :HeaderUsed = "tdms" ;
                :DataSystem = "N2UW" ;
                :RadarWing = 1. ;
                :FileTypeVersion = "8.7" ;
                :ProjectNumber = "102" ;
                :Phone = "(307) 766-3246" ;
                :Aircraft = "N2UW" ;
                :ProjectPI = "Tristan L\'Ecuyer" ;
                :Source = "University of Wyoming, Department of Atmospheric Science" ;
                :ProjectLocation = "Turku, Finland" ;
                :ProjectName = "LPVEX10" ;
                :ProjectEndDate = "10/20/2010" ;
                :HeaderCreated = "2010-08-12 20:05:58.765" ;
                :Address = "Dept. 3038, 1000 E. University Ave., Laramie, WY 82071-3038" ;
                :Platform = "N2UW" ;
                :Project = "Light Precipitation Validation Experiment (LPVEX)" ;
                :ProjectStartDate = "09/15/2010" ;
                :tdms_version = "4712" ;
                :FileYear = "2010" ;
                :time_coverage_start = "2010-09-16 07:23:52 +0000" ;
                :time_coverage_end = "2010-09-16 10:52:42 +0000" ;
                :FlightDate = " 9/16/2010" ;
                :TimeInterval = "07:23:52-10:52:42" ;
                :ArincTimeOffset = -0.009 ;
                :CorrectionFile = "corrections_lpvex10" ;
                :CorrectionFileModified = "04-Mar-2011 16:00:03" ;
                :nctdms_revision = "$Id: nctdms_revision.m,v 1.33 2011/02/28 20:24:59 ldoolman
Exp $" ;
                :DateProcessed = "19-Sep-2011 16:50:58" ;
                :Conventions = "NCAR-RAF/nimbus" ;
                :ConventionsURL = "http://www.eol.ucar.edu/raf/Software/netCDF.html" ;
                :ConventionsVersion = "1.3" ;
                :coordinates = "LONC LATC ztrue time" ;
                :geospatial_lat_min = 59.7116966333333 ;
                :geospatial_lat_max = 60.5623382833333 ;
                :latitude_coordinate = "LATC" ;
                :geospatial_lon_min = 22.2404945 ;
                :geospatial_lon_max = 25.6626529666667 ;
                :longitude_coordinate = "LONC" ;
                :geospatial_vertical_min = 73.577 ;
                :geospatial_vertical_max = 4468.5 ;
                :geospatial_vertical_positive = "up" ;
```

```
:geospatial_vertical_units = "m" ;
:zaxis_coordinate = "ztrue" ;
:Categories = "Position" ;
:PROCESSING = "Development" ;
:Version = "1.3" ;
:wind_field = "hwmag hwdir hw" ;
:time_coordinate = "time" ;
:CenterCoordLon0 = -106.744 ;
:CenterCoordLat0 = 40.455 ;
:CenterCoordName = "Storm Peak Laboratory" ;
:landmarks = "41.31336 -105.6728 LAR,40.455 -106.744 SPL" ;
:Winds.TempUsed = "trf" ;
:Winds.PressUsed = "ps_hads_a" ;
:Winds.PitotUsed = "aias" ;
:Winds.UseDPR = "yes" ;
:Winds.ManeuverDate = "20100527" ;
:Winds.QFactor = 0. ;
:Winds.PFactor = 0. ;
:Winds.AttackFactor = 0.239, -0.0132 ;
:Winds.SideslipFactor = 0.2125, -0.0049 ;
:Winds.PitchOffsetRadians = 0.0023 ;
:Winds.RollOffsetRadians = 0.0098 ;
:Winds.HeadOffsetRadians = 0.0027 ;
:TAS.UseHumidity = "yes" ;
:DMT.defaultWireTempCoeffs = -1.4444, 108.9922 ;
:BadValues = "Var: Number of values not physically correct" ;
:MissingValues = "Var: Number of missing values (filled with ~9e36)" ;
:DataType = "Derived variables processed with matlab(TM)" ;
:CVS_data = " File get_nevzarov1.m Tag lpvex10_qc13 Rev 1.1" ;
:CVS_misc = " File get_dmt1.m Tag lpvex10_qc13 Rev 1.8" ;
:CVS_irs = " File get_irs1.m Tag lpvex10_qc13 Rev 1.1" ;
:CVS_gps = " File get_gps1.m Tag lpvex10_qc13 Rev 1.1" ;
:CVS_topo = " File get_topo1.m Tag lpvex10_qc13 Rev 1.2" ;
:CVS_wind = " File cvwind1.m Tag lpvex10_qc13 Rev 1.5" ;
:Winds.PitchOffsetUsed = 0.0023 ;
:Winds.RollOffsetUsed = 0.0098 ;
:Winds.HeadOffsetUsed = 0.0027 ;
:Winds.AttackFactorUsed = 0.239, -0.0132 ;
:Winds.SideslipFactorUsed = 0.2125, -0.0049 ;
:Winds.QFactorUsed = 0. ;
:Winds.PFactorUsed = 0. ;
:CVS_licor = " File get_licor1.m Tag lpvex10_qc13 Rev 1.1" ;
:CVS_icing = " File get_icing1.m Tag lpvex10_qc13 Rev 1.4" ;
:CVS_fssp = " File get_fssp1.m Tag lpvex10_qc13 Rev 1.3" ;
:Nevzorov = "Added" ;
}
```