Input documentation for SBDART

Paul Ricchiazzi <paul@icess.ucsb.edu>
Institute for Computational Earth System Science
University of California, Santa Barbara

This file documents input parameters for SBDART, (Santa Barbara DISORT Atmospheric Radiative Transfer). SBDART is a software tool that computes plane-parallel radiative transfer in clear and cloudy conditions within the Earth's atmosphere and at the surface. For a general description and review of the program please refer to Ricchiazzi et al 1998. (Bulletin of the American Meteorological Society, October 1998).

SBDART's main input file is called INPUT. This file contains a single NAMELIST input block also named INPUT. A significant advantage of NAMELIST input is that not all elements of an input block need be specified by the user. Since most of the code inputs have been initialized with reasonable default values, new users can start by specifying just a few interesting input parameters. The default state of input parameters may be determined by removing INPUT from the current working directory. When SBDART detects the absense of file INPUT, it will print the default settings of all input parameters. This output may be redirected to a file for editing.

The default configuration of INPUT is as follows:

```
==========================================================================

 $INPUT
 idatm   =  4           ,  amix    =  0.0        ,  isat    =  0          ,
 wlinf   =  0.550       ,  wlsup   =  0.550      ,  wlinc   =  0.0        ,
 sza     =  0.0         ,  csza    =  -1.0       ,  solfac  =  1.0        ,
 nf      =  2           ,  iday    =  0          ,  time    =  16.0       ,
 alat    =  -64.7670    ,  alon    =  -64.0670   ,  zpres   =  -1.0       ,
 pbar    =  -1.0        ,  sclh2o  =  -1.0       ,  uw      =  -1.0       ,
 uo3     =  -1.0        ,  o3trp   =  -1.0       ,  ztrp    =  0.0        ,
 xrsc    =  1.0         ,  xn2     =  -1.0       ,  xo2     =  -1.0       ,
 xco2    =  -1.0        ,  xch4    =  -1.0       ,  xn2o    =  -1.0       ,
 xco     =  -1.0        ,  xno2    =  -1.0       ,  xso2    =  -1.0       ,
 xnh3    =  -1.0        ,  xno     =  -1.0       ,  xhno3   =  -1.0       ,
 xo4     =  1.0         ,  isalb   =  0          ,  albcon  =  0.0        ,
 sc      =  1.0,3*0.0   ,  zcloud  =  5*0.0      ,  tcloud  =  5*0.0      ,
 lwp     =  5*0.0       ,  nre     =  5*8.0      ,  rhcld   =  -1.0       ,
 krhclr  =  0           ,  jaer    =  5*0        ,  zaer    =  5*0.0      ,
 taerst  =  5*0.0       ,  iaer    =  0          ,  vis     =  23.0       ,
 rhaer   =  -1.0        ,  wlbaer  =  47*0.0     ,  tbaer   =  47*0.0     ,
 abaer   =  -1.0        ,  wbaer   =  47*0.950   ,  gbaer   =  47*0.70    ,
 pmaer   =  940*0.0     ,  zbaer   =  50*-1.0    ,  dbaer   =  50*-1.0    ,
 nothrm  =  -1          ,  nosct   =  0          ,  kdist   =  3          ,
 zgrid1  =  0.0         ,  zgrid2  =  30.0       ,  ngrid   =  50         ,
 ickp    =  0           ,  zout    =  0.0,100.0  ,  iout    =  10         ,
 deltam  =  t           ,  lamber  =  t          ,  ibcnd   =  0          ,
 phi0    =  0.0         ,  prnt    =  7*f        ,  ipth    =  1          ,
 fisot   =  0.0         ,  temis   =  0.0        ,  nstr    =  4          ,
 nzen    =  0           ,  uzen    =  20*-1.0    ,  nphi    =  0          ,
 phi     =  20*-1.0     ,  imom    =  0          ,  negprn  =  0          ,
```

```
 ttemp   =  -1.0         ,  btemp   =  -1.0
 $END
```

==========

NOTE: Unfortunately, many fortran compilers produce rather cryptic error messages in response to improper NAMELIST input files. Here are three common NAMELIST error messages and their meaning:

1. ERROR MESSAGE: invalid reference to variable in NAMELIST input
   MEANING: you misspelled one of the NAMELIST variable names

2. ERROR MESSAGE: end-of-file during read
   MEANING: you didn't include a NAMELIST block specifier (INPUT, DINPUT or END) or you misspelled it.

3. ERROR MESSAGE: too many values for NAMELIST variable
   MEANING: you specified too many values for a variable, most likely because you separated variables by more than one comma.

General options (NAMELIST $INPUT):

WAVELENGTH LIMITS, FILTER FUNCTION SPECIFICATION

NF: SOLAR SPECTRUM SELECTOR

-1 = read from file solar.dat (user supplied)

```
read(13,*) nw,wmin,wmax,(sun(i),i=1,nw)
```

where nw is the number of spectral entries
wmin is the minimum wavenumber (cm-1)
wmax is the maximum wavenumber (cm-1)
sun is the solar irradiance at the top of the atmosphere (w/m2/um) at equal increments of wavenumber between wmin and wmax.

0 = spectrally uniform

1 = 5s solar spectrum
0.005 micron resolution, .25 to 4 micron

2 = LOWTRAN_7 solar spectrum (default)
20 cm-1 resolution, 0. to 28780 cm-1
10 cm-1 resolution, 28780. to 57490 cm-1

3 = MODTRAN_3 solar spectrum
20 cm-1 resolution, 100 - 49960 cm-1

ISAT: FILTER FUNCTION TYPES

-4	Guassian filter, WLINF-2*WLSUP to WLINF+2*WLSUP
-3	Trianglar filter, WLINF-WLSUP to WLINF+WLSUP
-2	Flat filter, WLINF-.5*WLSUP to WLINF+.5*WLSUP
-1	USER DEFINED, read from filter.dat
0	WLINF TO WLSUP WITH FILTER FUNCTION = 1 (default)
1	METEO
2	GOES(EAST)
3	GOES(WEST)
4	AVHRR1(NOAA8)
5	AVHRR2(NOAA8)
6	AVHRR1(NOAA9)
7	AVHRR2(NOAA9)
8	AVHRR1(NOAA10)
9	AVHRR2(NOAA10)
10	AVHRR1(NOAA11)
11	AVHRR2(NOAA11)
12	GTR-100 ch1
13	GTR-100 ch2
14	GTR-100 410nm channel
15	GTR-100 936nm channel
16	MFRSR 415nm channel
17	MFRSR 500nm channel
18	MFRSR 610nm channel
19	MFRSR 665nm channel
20	MFRSR 862nm channel
21	MFRSR 940nm channel
22	AVHRR3 (nominal)
23	AVHRR4 (nominal)
24	AVHRR5 (nominal)
25	Biological action spectra for DNA damage by UVB radiation

NOTE: If ISAT=-1 a user supplied filter data file "filter.dat" is read from the current working directory. This ASCII file is read with the following free format read (numbers may be separated by spaces, commas or carriage returns);

```
read(13,*) wmin,wmax,dww,nnf,(srr(i),i=1,nnf)
```

where, wmin = minimum wavelength (microns)
wmax = maximum wavelength (microns)
dww = wavelength increment (microns)
nnf = number of wavelengths (<= 200)
srr = filter function value

WLINF: lower wavelength limit when ISAT=0 (WLINF > .250 microns)
central wavelength when ISAT=-2,-3,-4

WLSUP: upper wavelength limit when ISAT=0 (WLSUP < 100.0 microns)
equivalent width when ISAT=-2,-3,-4

NOTE:

If ISAT eq -2, a rectangular filter (constant with wavelength) is used, with central wavelength at WLINF and an equivalent width of WLSUP (full width = WLSUP)

If ISAT eq -3, a triangular filter function is used with the central wavelength at WLINF and an equivalent width of WLSUP (full width = 2*WLSUP) (filter function is zero at end points, and one at WLINF).

If ISAT eq -4, a gaussian filter function is used with the central wavelength at WLINF and an equivalent width of WLSUP (full width = 4*WLSUP)

If output is desired at a single wavelength, set WLINF=WLSUP and ISAT=0. In this case, SBDART will set WLINC=1 (the user specified value of WLINC is ignored) and the output will be in units of (W/m2/um) for irradiance and (W/m2/um/sr) for radiance.

WLINC: This parameter specifies the spectral resolution of the SBDART run. Though the spectral limits of the calculation are always input in terms of wavelength, the spectral step size can be specified in terms of constant increments of wavelength, log(wavelength) [same as constant increment of log(wavenumber)] or wavenumber. Which one to choose depends on where in the spectral bandpass you want to place the most resolution.

Since SBDART is based on LOWTRAN7 band models, which have a spectral resolution of 20 cm-1, it would be extreme overkill to allow spectral step size less than 1 cm-1. On the other hand a spectral resolution coarser than 1 um is also pretty useless. Therefore the way WLINC is interpreted depends on whether it is less than zero, between zero and one, or greater than 1.

* WLINC = 0 (the default) => wavelength increment is equal to 0.005 um or 1/10 the wavelength range, which ever is smaller.

* WLINC < 0 => wavelength increment is a constant fraction of the current wavelength. WLINC is interpreted as a specified value of delta(lambda)/lambda and the wavelength steps are adjusted so that wavelength step is approximately the product of the current wavelength and WLINC.

Specifying the wavelength increment as a fractional step size is useful when the wavelength range extends over more than an decade of wavelength. For example, if the wavelength range is 0.5 to 20.0, specifying a constant wavelength increment of .01 microns tends to under-resolve the low wavelengths and over-resolve the long wavelengths. Setting WLINC = -.01 causes the

code to use a wavelength increment of about .005 microns in the visible and about .2 micron in the thermal infrared, which is a better compromise of resolution and computer time.

* 1 >= WLINC > 0 => WLINC is the wavelength step size (um)

if WLINC > 1 then WLINC is the step size in inverse centimeters. If maximum fidelity is required and gaseous absorption is the primary influence on the output, then WLINC should be set to 20, which is the wavenumber resolution of the LOWTRAN7 band models.

The total number of wavelength steps, nwl, is given by

nwl = 1+ln(wlsup/wlinf)/|wlinc| wlinc < 0

nwl = 1+(wlsup-wlinf)/wlinc 1 >= wlinc > 0

nwl = 1+10000*(1/wlinf-1/wlsup)/wlinc wlinc > 1

SOLAR GEOMETRY
==========

SZA: solar zenith angle (degrees) (default = 0.)
SZA is ignored if CSZA is non-negative or IDAY is non-zero.

CSZA: Cosine of solar zenith angle. If CSZA > 0, solar zenith angle is set to acos(CSZA) (default = -1.)

IDAY: If IDAY > 0, the solar illumination angles (SZA, PHI0) are computed from the specified time and geographic coordinates using an internal solar ephemeris algorithm (see subroutine zensun). IDAY is the number of days into a standard "year", assumed to consist of 365 days. if IDAY > 365, IDAY is replaced internally by mod(IDAY-1,365)+1.

If IDAY < 0, the code writes the values of abs(iday),time, alat,alon,sza,azm, and solfac to standard output and exits.

TIME: UTC time (Grenwich) in decimal hours
ALAT: latitude of point on earth's surface
ALON: east longitude of point on earth's surface

NOTE: TIME, ALAT and ALON are ignored if IDAY .eq. 0

SOLFAC: solar distance factor. Use this factor to account for seasonal variations of the earth-sun distance. If R is the earth-sun distance in Astronomical Units then SOLFAC=1./R**2. SOLFAC is set internally when the solar geometry is set through IDAY, TIME, ALAT and ALON. In this case SOLFAC is set to,

SOLFAC = (1.-eps*cos(2*pi*(IDAY-perh)/365))

where eps = orbital eccentricity = 0.01673
and perh = day of perihelion = 2 (jan 2)

NOTE: seasonal variations in earth-sun distance produce a +/-3.4% perturbation in the TOA solar flux. This factor should be included when making detailed comparisons to surface measurements.

NOSCT: If set to 1, compute radiative flux due to thermal sources only. Solar direct or scattered radiation is not included. Thermal sources are not computed for wavelengths less than 2um unless NOTHRM=0.

SURFACE REFLECTANCE PROPERTIES

ISALB: SURFACE ALBEDO FEATURE

-1 -spectral surface albedo read from "albedo.dat"
0 -user specified, spectrally uniform, surface albedo
1 -snow
2 -clear water
3 -lake water
4 -sea water
5 -sand
6 -vegetation
10 -combination of snow, seawater, sand and vegetation

NOTE: If ISALB=-1 a user supplied spectral reflectance data file, "albedo.dat" is read from the current working directory. This ASCII file is read with the following free format read statements:

```
read(13,*) nn,wmin,wmax
read(13,*,end=900) (r(i),i=1,nn)
```

where, nn = number of sample points (nn .le. 751)
wmin = minimum wavelength
wmax = maximum wavelength
r = spectral reflectivity
such that r(i) is at wavelength
wmin+(i-1)*(wmax-wmin)/(nn-1)

The user specified reflectance may cover any wavelength range and have arbitrarily high resolution. This contrasts with the standard reflectance models (sand, vegetation, lake water and sea water) which are only specified in in the range .25 to 4 um at 5nm resolution.

ALBCON: User specified, spectrally uniform, surface albedo (applies only when ISALB=0)

SC: Composite albedo fractions (applies only when ISALB=10)
SC(1) = fraction of snow
SC(2) = fraction of ocean
SC(3) = fraction of sand
SC(4) = fraction of vegetation

NOTE: SC(1)+SC(2)+SC(3)+SC(4) need not sum to 1. Thus, it is possible to use the SC factor to boost the overall reflectance of a given surface type. For example, SC=0,0,2,0 would compute the effects over a surface whose spectral reflectivity were twice that of sand. But be careful, SC=2,0,0,0 will produce a surface reflectance twice that of snow, which will produce a reflectance greater than unity at visible wavelengths.

MODEL ATMOSPHERES

IDATM: ATMOSPHERIC PROFILE:	default water vapor (g/cm2)	ozone(atm-cm) total	ozone(atm-cm) below_10km
0 User Specified			
1 TROPICAL	4.117	0.253	.0216
2 MID-LATITUDE SUMMER	2.924	0.324	.0325
3 MID-LATITUDE WINTER	0.854	0.403	.0336
4 SUB-ARCTIC SUMMER	2.085	0.350	.0346
5 SUB-ARCTIC WINTER	0.418	0.486	.0340
6 US62	1.418	0.349	.0252
-n List to standard out			

If IDATM = 0, a user supplied atmospheric profile, "atms.dat", is read from the current working directory. This ASCII file is read with the following free format read statements (input values may be separated by spaces, commas or carriage returns);

```
     read(13,*) nn
     do 10 i=1,nn
       read(13,*) z(i),p(i),t(i),wh(i),wo(i)
10   continue
```

where nn is the number atmospheric layers (nn .le. 50)
z is the layer altitude in km
(z must be monotonically decreasing)
p is the pressure in millibars
t is the temperature is Kelvin
wh water vapor density g/m3
wo ozone density g/m3

If IDATM is set to a negative number in the range -1 to -6 SBDART prints the atmospheric model corresponding to abs(idatm) to standard out, and then quits.

AMIX: weighting factor, when positive this factor controls

how much of the atms.dat atmospheric profile to mix in with one of the standard internal profiles selected by IDATM. For example IDATM=1 and AMIX=.7 specifies a 70% weighting of atms.dat and a 30% weighting of profile TROPIC. No (default=-1)

UW: integrated water vapor amount (G/CM2)

UO3: integrated ozone concentration (ATM-CM) above the level ZTRP. The default value of ZTRP=0, so UO3 usally specifies the total ozone column. (1 atm-cm = 1000 Dobson Units)

NOTE: Use UW or UO3 to set the integrated amounts of water vapor or ozone in the model atmosphere. Aside from multiplicative factors the vertical profile will be that of the original model atmosphere set by IDATM. The original unmodified density profile is used when UW or UO3 is negative.

O3TRP: integrated ozone concentration (ATM-CM) in troposphere. i.e., for z.lt.ZTRP. The original tropospheric density is used when O3TRP is negative. (default=-1.)

ZTRP: The altitude of the tropopause. The parameters UO3 and O3TRP sets the total column ozone in the stratosphere and troposphere, respectively. Note: since the default value of ZTRP is zero, UO3 normally sets the integrated ozone amount of the entire atmosphere (default=0).

XN2:	volume mixing ratio of N2	(PPM,	default =	781000.00)
XO2:	volume mixing ratio of O2	(PPM,	default =	209000.00)
XCO2:	volume mixing ratio of CO2	(PPM,	default =	360.00)
XCH4:	volume mixing ratio of CH4	(PPM,	default =	1.74)
XN2O:	volume mixing ratio of N2O	(PPM,	default =	0.32)
XCO:	volume mixing ratio of CO	(PPM,	default =	0.15)
XNH3:	volume mixing ratio of NH3	(PPM,	default =	5.0e-4)
XSO2:	volume mixing ratio of SO2	(PPM,	default =	3.0e-4)
XNO:	volume mixing ratio of NO	(PPM,	default =	3.0e-4)
XHNO3:	volume mixing ratio of HNO3	(PPM,	default =	5.0e-5)
XNO2:	volume mixing ratio of NO2	(PPM,	default =	2.3e-5)

NOTE: Setting any of these factors to -1 causes that atmospheric component to retain its nominal mixing ratio defined in the US62 atmosphere (as listed above).

The volume mixing ratio (VMR) of a given species is adjusted by specifying the surface value of its VMR in PPM. The entire altitude profile is multiplied by the ratio of the user specified VMR and the nominal surface VMR.

There are no further re-normalizations of the VMR. Thus, the total of all the VMRs may be

greater or less than 10^6. By the way, the default set of VMRs do not add up to 10^6 because of the exclusion of the noble gases which do not have any radiative effects.

XRSC: sensitivity factor for Rayleigh scattering (default=1)
This factor varies the strength of Rayleigh scattering for sensitivity studies.

PBAR: surface pressure in mbar.
If PBAR .gt. 0 then each pressure is multiplied by the factor (PBAR/P0) where P0 is the surface pressure of the original atmosphere.
If PBAR .le. 0, the original pressure profile is used.

ZPRES: Surface altitude in kilometers.
This parameter is just an alternate way of setting the surface pressure, and should not be set when PBAR is specified. When ZPRES is set PBAR is obtained by logarithmic interpolation on the current model's atmosphere pressure and altitude arrays. Changing ZPRES does not alter other parameters in the atmospheric model in any way. Note that setting a large value of ZPRES may push the tropopause (where dT/dz=0) to an unrealistically high altitude.

SCLH2O: Water vapor scale height in km.

If SCLH2O .gt. 0, then water vapor is vertically distributed as exp(-z/SCLH2O)

If SCLH2O .le. 0, then the original vertical profile is used. Changing SCLH2O has no effect on the total water vapor amount.

CLOUD PARAMETERS
==========

ZCLOUD: Altitude of cloud layers (km) (up to 5 layers), Cloud layers may be specified in two ways. To specify separate cloud layers, set ZCLOUD to a sequence of monotonically increasing altitudes. Each value of ZCLOUD will set the altitude (above the surface) of the corresponding optical depth in the TCLOUD array.

To specify a range of altitudes which will be filled by cloud, tag the second element of the range with a minus sign. Consider,

```
zcloud=1,-3,10,-15
tcloud=4,0,8,0,0
nre=6,6,8,9,10
```

In this example two continuous cloud layers are defined, the lower one extends from 1 to 3 km and has a total optical depth of 4 and an effective radius of

6um. The upper cloud layer extends from 10 to 15 km, has a total optical thickness of 8 and a sliding value of effective radius which starts 8um at the bottom of the cloud and ramps up to 9um at 15km. However, beware that the actual location of the cloud layers is determined by the resolution and placement of vertical grid points in SBDART, as explained below.

SBDART puts the i'th cloud layer at the highest vertical grid point, k, such that

```
z(k) .le. abs(ZCLOUD(i)+.001)
```

NOTE: A cloud with a nominal altitude equal to that of one of the computational layer altitudes, Z(K), actually extends from Z(k) to the next higher grid point. For example, a cloud layer at Z(k) will not affect the direct beam flux at Z(k-1) (one layer above) but will strongly affect it at Z(k). (You can check this out your self by setting IOUT=10 and ZCLOUD=1 and messing around with ZOUT to get outputs just above or below the cloud).

Suppose the bottom of your computational grid looks like

k	z(k)
...	...
30	2.5
31	2.0
32	1.5
33	1.0
34	0.5
35	0.0

If you want a cloud to extend from 0.5 to 1.5 km, then set ZCLOUD= .5, -1.5. Actually the same result would be obtained by setting the second element of ZCLOUD to anything between -(1.0+epsilon) and -1.5.

Consider,

```
TCLOUD=[6., 0.0,  12., 0.0, 0]
ZCLOUD=[1.0,-6.0, 4.0,-9.0, 0]
```

Here two overlapping cloud decks are specified, one extending from 1 to 6 km with a total optical thickness of 6, and the other from 4 to 9 km with a total thickness of 10. Since the total optical thickness is spread over the total altitude range we would have 1 optical depth per km for the lower cloud deck and 2 optical depths per km for the second. The code adds the effects of both cloud decks in the region of overlap. So the above specification would yield 1 optical depth per km between 1 and 4 km, 3 optical depths per km between 4 and 6 km and 2 optical depths per km between 4 and 9 km for a total optical depth of 18.

If you have any doubt about where the code is putting the cloud, set ICKP=1 (see below) and check the diagnostic print out in rtinfo.00

NOTE: do not try to put an ice cloud (NRE < 0) in a cloud layer range which includes water cloud (2 le NRE le 128). In other words this specification won't work:

```
ZCLOUD = 1,-4
TCLOUD = 1, 0
NRE    = 8,-1
```

TCLOUD: Optical thickness of cloud layer, (up to 5 values)

TCLOUD specifies the cloud optical depth at a wavelength of 0.55um. The rt codes compute cloud optical depth at other wavelengths using the relation,

tau = TCLOUD*Q(wl)/Q(0.55um),

where Q is the extinction efficiency which is a function of effective radius and wavelength (see discussion of LWP for a definition of Q). The codes contain look-up tables of Q that cover effective radii in the range 2 to 128um for water clouds and for a single effective radius of 106um for ice clouds. The wavelengths range is 0.29 to 333.33 um for water clouds and .29 to 20 um for ice clouds.

When specifying an optical depth for a range of grid levels, the second TCLOUD entry corresponding to the cloud top altitude is usually set to zero. This produces a uniform distribution of opacity over the altitude range.

For example,

```
ZCLOUD=[1 ,-5,0,0,0]   # uniformly distributed opacity
TCLOUD=[10, 0,0,0,0]   # for a cloud of extent 4 km
                       # 2.5 optical depths per km
```

A linearly varying opacity distribution can be obtained by setting the second TCLOUD entry to a factor which represents the ratio of the opacity in the highest layer to that in the lowest layer

For example,

```
ZCLOUD=[1 ,-5,0,0,0]   # linearly distributed opacity
TCLOUD=[10, 4,0,0,0]   # for a cloud of extent 4 km
                       # between tau(1-2km)=1
                       # between tau(2-3km)=2
                       # between tau(3-4km)=3
                       # between tau(4-5km)=4
                       #
```

```
                        # tau(total)=10
                        # tau(4-5km)/tau(1-2km)=4
```

NOTE: if r is the ratio of the top to bottom and t is the average opacity per level then,

```
tau(top_level)=t*2r/(1+r)

tau(bot_level)=t*2/(1+r)
```

NOTE: a linear increase in opacity, starting from zero at the cloud bottom, is obtained by setting,

```
r=1 + 2*zdiff/dz
```

where dz is the grid spacing and zdiff is the total altitude range over which the cloud extends. This formula assumes constant grid spacing over the cloud altitude range. Thus, if dz=1 then ZCLOUD=[1,-5] and TCLOUD=[10,7] yeilds a linear increase from zero.

If the first element of TCLOUD is negative, cloud specification records are read from file usrcld.dat. The first record in this file corresponds to the lowest layer in the atmosphere, that is between the surface and the lowest cell boundary altitude. Each following record sets values for the next higher atmospheric layer in model atmosphere. usrcld.dat is read with the following fortran statements:

```
do i=1,nz-1
  read(13,*,end=100) lwp(i),re(i),fwp(i),rei(i),cldfrac(i)
enddo
100 continue
```

where lwp	liquid water path in layer i.	(g/m2)
re	effective radius of liquid water in layer i. supersedes value of nre	(um)
fwp	frozen water path in layer i.	(g/m2)
rei	effective radius of frozen water in layer i. if negative, CCM3 ice model is used to set effective radius of ice	(um)
cldfrac	cloud fraction in layer. this parameter reduces cloud optical depth by factor cldfrac**1.5	

It is not necessary to provide input records for layers above the highest cloud. In addition, a forward slash terminates interpretation of data values

in a record. For example, the following records in usrcld.dat specify a cloud that extends from 2 to 4 km (assuming idatm>0 and no regridding):

```
/         # lwp=0 between 0-1 km
/         # lwp=0 between 1-2 km
20 /      # lwp=45, re=10 between 2-3 km
60 20 /   # lwp=60, re=20 between 3-4 km
```

Any input quantities that are left unspecified will retain their initial default values of lwp=0. reff=10, fwp=0, reice=-1, and cldfrac=1. The radiative properties of ice are computed from a CCM3 model

NRE: Cloud drop effective radius (microns). (up to 5 values)

If NRE is specified as a floating point number in the range 2.0 to 128.0, a liquid water cloud of given effective radius is selected.

For liquid water clouds, we assume the drop size distribution is specified by a gamma distribution,

$$N(r) = C * (r/Ro)^{(p-1)} e^{(-r/Ro)}$$

where C is a normalization constant [C=1./(Ro*gamma(p))], p=7, and Ro=NRE/(p+2)

The factor (p+2) relating Ro to NRE follows from the defining equation of NRE:

$$NRE = < r^3 N(r) > / < r^2 N(r) >,$$

where the angle brackets indicate integration over all drop radii.

Another frequently used parameter to describe the size distribution is the mode radius, Rm, which is defined as the radius at which N(r) is maximized. For our drop size distribution Rm=(p-1)*Ro. Using the relation between Ro and NRE we find that, Rm=(p-1)*NRE/(p+2)

If instead, NRE < 0, a cirrus cloud composed of spherical ice particles is selected. Ice parameters are for a fixed effective radius of 106um. The ice particle size distribution is given by an empirical fit to in situ cirrus cloud data. When NRE is negative, the magnitude of this parameter is used to modify the absorption strength of the photon interaction. That is, the single scattering albedo (SSA) is modified as,

```
SSA=1.-abs(nre)*(1-SSA)
```

NOTE: Some detailed scattering calculations indicate that the single scattering co-albedo (1-SSA) for complex crystal shapes can be much smaller than the co-albedo predicted using Mie theory for an equal area spherical crystal. NRE in the range -1<NRE<0 will reduce the co-albedo, possibly bringing it into closer agreement with the detailed calculations.

LWP: The liquid water path of a cloud is specified in units of g/m2. This is another way to specify cloud optical depth.

A linearly varying opacity distribution can be obtained by setting the second LWP entry to a factor which represents the ratio of the opacity in the highest layer to that in the lowest layer For more details see the discussion of TCLOUD.

NOTE: a 1 mm column of liquid water = 1000 g/m2,

NOTE: LWP and TCLOUD cannot be used at the same time

NOTE: The cloud optical depth is related to LWP by

$$\tau = \frac{3\, Q(wl) * LWP}{4\ RHO * NRE}$$

where Q is the scattering efficiency and RHO is the density of liquid water (1 g/cm3). The value of Q that applies to a distribution of cloud droplets can be expressed in terms of the extinction cross-section at a given wavelength and liquid drop radius.

Let sigma = extinction cross-section at a given wavelength and drop radius

q = sigma/(pi*r^2) (dimensionless)

where (pi*r^2) is the geometrical cross-section of the cloud drop

then Q is a weighted average over drop radius, given by:

$$Q = \langle r^2\, q\, N(r) \rangle / \langle r^2\ N(r) \rangle$$

for visible light Q is typically about 2 (dimensionless).

For example:
NRE = 10um and LWP= 200g/m2 = 0.2mm => tau = 30

RHCLD: The relative humidity within a cloud layer (a floating point value between 0.0 and 1.0). RHCLD<0 disables the adjustment of relative humidity, in which case the relative humidity in the cloud layer follows solely from the temperature and water vapor density of the initial model atmosphere.

KRHCLR: If zero, water vapor mixing ratio in clear layers is proportionately reduced to maintain the water vapor path specified by WH. This option has no effect if RHCLD is negative or TCLOUD is zero. (default)

if 1, the relative humidity in clear layers is unchanged.

NOTE: if KRHCLR=1 and clouds are present, the actual water vapor path will differ from that specified by WH. On the other hand, if KRHCLR=0, the normalization procedure may drive the water vapor in clear layers to zero and still be unable produce a given WVP.

STRATOSPHERIC AEROSOLS (LOWTRAN 7 model)
==========

JAER: 5 element array of stratospheric aerosol types

0-no aerosol
1-background stratospheric
2-aged volcanic
3-fresh volcanic
4-meteor dust

ZAER: altitudes (above the surface) of stratospheric aerosol layers (km) Up to 5 layer altitudes may be specified. NOTE: even though these models are for stratospheric aerosols, the scattering layer may be placed anywhere within the numerical grid. See ZCLOUD for a discussion of how aerosol (cloud) layers are positioned within SBDART's computational grid.

TAERST: optical depth (at 0.55 microns) of each stratospheric aerosol layer. Up to 5 layer optical depths may be specified.

BOUNDARY LAYER AEROSOLS (BLA)
==========

IAER: Boundary layer aerosol type selector

0-no boundary layer aerosols (all BLA parameters ignored)

1-rural

2-urban

3-oceanic

4-tropospheric

5-user defined spectral dependence of BLA

The wavelength dependence of the aerosol scattering parameters are replaced by those read in from input parameters wlbaer, tbaer, wbaer and gbaer. Between 1 and 47 spectral values may be specified.

NOTE: the spectral dependence of the boundary layer aerosol models (IAER=1,2,3,4) vary with relative humidity. See SUBROUTINE AEROSOL for details.

NOTE: Don't be mislead by the term "boundary layer aerosol". The BLA models, IAER=1,2,3,4 were originally developed to describe aerosols in the lower atmosphere. However in SBDART, the default vertical density of BLA falls off exponentially, and affects regions above the normal extent of the boundary layer. The vertical influence of these aerosols may be confined to a specified boundary layer altitude with the optional parameters ZBAER and DBAER.

RHAER: The spectral dependence of the boundary layer aerosol scattering parameters are sensitive to relative humidity. Use input parameter RHAER to set the relative humidity used in the boundary layer aerosol model. Set RHAER=-1 (the default value) to use the ambient surface relative humidity. RHAER has no effect when IAER = 5.

VIS: (Horizontal Path) Visibility (km) at 0.55 microns due to boundary layer aerosols. This parameter does not set the optical depth for the user defined aerosol model (IAER=5), but does affect that model through the vertical structure (see below).

NOTE: unlike the stratospheric aerosols, the boundary layer aerosols have predefined vertical density distributions. These vertical structure models vary with visibility. (see discussion of ZBAER and DBAER)

NOTE: The boundary layer aerosol optical depth (absorption + scattering) at 0.55 microns is given by

tauaero(0.55um) = 3.912 * integral (n(z)/n(0) dz) / VIS

where n(z) is the vertical profile of aerosol density. For the 5 and 23 km visibility models the indicated integral is 1.05 and 1.51 km, respectively. So,

tauaero(0.55um) = 3.912*(1.05*w+1.51*(1-w))/vis

where w is a weighting factor between the two extremes and is given by

$$w = \frac{(1/vis-1/23)}{(1/5-1/23)}, \quad 5 < vis < 23$$

$$w = 1, \quad vis < 5$$

$$w = 0, \quad vis > 23$$

NOTE: Visibility is defined as the horizontal distance in km at which a beam of light at 0.55um is attenuated by a factor of 0.02.

n(0)*sigma*VIS = -ln(.02), or

VIS = 3.912/(n(0)*sigma)

where sigma is the aerosol absorption+scattering cross-section at 0.55 microns. See Glossary of Meteorology, American Meteorology Society, 1959

ZBAER: Altitude grid for custom aerosol vertical profile (km) Up to 50 altitude points may be specified. ZBAER is active for all values of IAER.

DBAER: Aerosol density at ZBAER altitude grid points, active for all values of IAER. Up to 50 density values may be specified. The number of density values must match the number of ZBAER. The units used to specify aerosol density is arbitrary, since the overall profile is scaled by the user specified total vertical optical depth. The aerosol density at all computational grid points is found through logarithmic interpolation on the ZBAER and DBAER values. The normal vertical profile from 5s is used when DBAER is unset.

For example

```
ZBAER=0,1,100
DBAER=1000,500,1
```

specifies a aerosol density profile that drops by a factor 2 (exponential fall off) between 0 and 1km altitude and then by a factor of 500 between 1 and 100 km.

If DBAER is set but ZBAER is not set, then the elements of DBAER are used to set the aerosol density for each computational layer, starting from the bottom layer.

For example,

```
DBAER=10,0,1,0
```

puts aerosol in the first and third layer.

If neither ZBAER or DBAER are set, the boundary layer aerosols are assumed to follow a pre-defined vertical distribution which drops off exponentially with a scale height between 1.05 and 1.51 km depending visibility (see VIS). Thus, even if visibility is not used to set the vertical optical depth it can affect the result through the vertical profile. Note that ZBAER and DBAER do not affect the total vertical optical depth of aerosols. (See discussion for VIS).

TBAER: Vertical optical depth of boundary layer aerosols at 0.55 um. TBAER input is significant for all values of IAER. When IAER=1,2,3,4 the specified value of TBAER supersedes the aerosol optical depth derived from input parameter VIS (but VIS still controls vertical structure model unless DBAER and ZBAER are set).

QBAER QBAER is the extinction efficiency. QBAER is only active when IAER=5. When TBAER is set, QBAER sets the spectral dependence of the extinction optical depth as,

```
   tau= tbaer * Qext(wave_length)/Qext(0.55um)
```

where Qext(wave_length) = QBAER interpolated to wavel_length

If TBAER is not set, then the values of QBAER are interpreted as extinction optical depths at each wavelength WLBAER.

For example, the Multi Filter Rotating Shadowband Radiometer (MFRSR) installed at the Southern Great Plains ARM site is able to retrieve aerosol optical depth in 6 SW spectral channels. This information may be supplied to SBDART by setting,

```
  wlbaer= .414,  .499,  .609,  .665,  .860,  .938
  qbaer= 0.109, 0.083, 0.062, 0.053, 0.044, 0.041
  wbaer=6*.9
  gbaer=6*0.8
```

This spectral information is iterpolated or extrapolated to all wavelengths using logarithmic fitting on QBAER and linear fitting on WBAER and GBAER. Many aerosol types display a power law dependence of extinction efficiency on wavelength. The logarithmic interpolation/extrapolation on QBAER will reproduce this behavior if it exists in the input data.

WLBAER Wavelengths points (um) for user defined aerosol spectral dependence. Only used when IAER=5. WLBAER (and QBAER) need not be specified if a single spectral point is set. In this case the aerosol optical depth is extrapolated to other wavelengths using a power law (see ABAER)

WBAER: Single scattering albedo used with IAER=5.

WBAER represents the single scattering albedo of boundary layer aerosols at wavelengths WLBAER.

GBAER: Asymmetry factor used with IAER=5

GBAER represents the asymmetry factor of boundary layer aerosols at wavelengths WLBAER. Number of values must match the number of WLBAER.

GBAER is ignored when parameter PMAER is set.

PMAER: Legendre moments of the scattering phase function of boundary layer aerosols, only active for IAER=5. The Legendre moments of the phase function are defined as the following integral over the scattering phase function, f:

$$pmaer(i) = \frac{\int f(mu)\, P(i,mu)\, d\,mu}{\int f(mu)\, d\,mu}$$

where P(i,mu) is the Legendre polynomial, mu is the cosine of the scattering angle, and the range of the integrals are from -1 to 1. The Legendre moment for i=0 is always one. Hence, the zero'th moment is assumed by SBDART and should not be specified.

Unlike the previous boundary layer aerosol parameters, you need to specify at least NSTR values for each wavelength point, for a total of NSTR*NAER values, where NAER is the number of wavelength points supplied. The order of specification should be such that wavelength variation is most rapid. For example, here is a case with 4 wavelengths and 6 streams:

```
nstr=6
wlbaer=.400,.500,.600,.700
pmaer= 0.80,0.70,0.60,0.50,
       0.64,0.49,0.36,0.25,
       0.51,0.34,0.22,0.12,
       0.41,0.24,0.13,0.06,
       0.33,0.17,0.08,0.03,
       0.26,0.12,0.05,0.02
```

ABAER: Wavelength (Angstrom model) exponent used to extrapolate BLA extinction efficiency to wavelengths outside the range of WLBAER [Qext ~ (lambda)^(-abaer)]. This parameter is only operative when IAER=5.

If ABAER is set to a positive number, then that value is used as a power-law wavelength dependence to extrapolate the extinction efficiency for wavelengths less than WLBAER(1) or greater than WLBAER(nn) (where nn is the number of specified values). If ABAER is not set, the wavelength extrapolation is based on the

last two specified points (wlbaer(1),wlbaer(2) or wlbaer(nn-1),wlbaer(nn)). If ABAER is not set and a single wavelength is set, then a spectrally constant extinction efficiency is used.

==========

NOTHRM: nothrm=-1 => Thermal emission turned on only for wavelengths greater than 2.0 um (default)

(Note: During daylight hours solar radiation is a factor of about 1.e5 greater than thermal radiation at 2.0um)

nothrm=0 => Thermal emission turned on for all wavelengths

nothrm=1 => No thermal emission

NOTE: If thermal emission is desired, be sure that the temperature steps in the atmospheric model are small enough to resolve changes in the Planck function. The original version of the DISORT radiative transfer module issued a warning message if the temperature difference between successive levels in the atmosphere exceeded 20 K. All the standard atmospheres violate this condition for at least 1 stratospheric layer. To avoid clutter in SBDART's standard output, I have disabled the warning message. If near-IR thermal emission from the stratosphere is important to your application, you should supply SBDART with a new model atmosphere with higher resolution in the stratosphere. (see ZGRID1, ZGRID2, an NGRID)

KDIST: KDIST=0 causes the optical depth due to molecular absorption is set to the negative log of the LOWTRAN transmission function. This approximation is not appropriate for cases in which multiple scattering is important, but is not very wrong when the molecular absorption is weak or the scattering optical depth is small.

KDIST=1 causes SBDART to use the LOWTRAN7 k-distribution model of absorption by atmospheric gases. Since a three term exponential fit is used, SBDART execution times are up to 3 times longer when KDIST > 0.

KDIST=2 causes the k-fit transmissions to exactly match the LOWTRAN transmission along the solar beam direction. This option may be useful when computing surface irradiance under clouds of optical thickness less than about 10. This is because in this thin cloud case much of the radiation which reaches the surface propagates along the direct beam direction.

KDIST=3 causes the k-fit transmission to exactly match the LOWTRAN transmission along the solar beam direction for parts of the atmosphere above a scattering layer. As the scattering optical depth increases above 1 the k-fit factors are ramped back to there original LOWTRAN values. This is the default.

ZGRID1:
ZGRID2:
NGRID:

These three parameters can be used to change the grid resolution of the model atmosphere. ZGRID1 controls the resolution near the bottom of the grid while ZGRID2 sets the maximum permissible step size (at the top of the grid). NGRID sets the number of grid points. For example ZGRID1=.5, ZGRID2=30, NGRID=45 specifies a 45 element grid with a resolution of .5 km throughout the lower part of the grid and a largest step of 30 km.

The regridding is performed after the call to subroutine ATMS. This allows regridding of the standard internal atmospheres as well as user specified atmospheres (read with IDATM=0). No matter how many grid points were used to specify the original atmosphere, the new regridded atmosphere will contain NGRID vertical array elements. The default value of ZGRID2 and NGRID are currently set to 30km and 45, respectively. The internal parameter, mxly, sets the maximum number of levels allowed. Currently, a maximum of 50 layers are allowed.

Setting ZGRID1=0 (the default) causes the initial (un-modified) atmospheric model to be used.

If ZGRID1 is negative SBDART terminates execution after printing out the regridded values of Z,P,T,WH,WO to standard out. This option can be used to preview the effect of a given set of abs(ZGRID1),ZGRID2 and NGRID values.

OUTPUT OPTIONS
==============

ICKP:

DIAGNOSTIC OUTPUT SELECTOR

The ICKP print flag is set up to control a large variety of diagnostic output. Different options can be independently selected by packing into ICKP the binary sequence which represents the print option settings. The diagnostic output is sent to files rtinfo.nn where nn is given below,

nn	n 2	
00	1	print atmospheric profile and absorption diagnostic for a single wavelength

01	2	cloud parameters, extinction efficiency, asymmetry factor and single scatter albedo
02	4	gaseous absorption integrals and optical depth
03	8	Optical depth due to Rayleigh, aerosols, cloud molecular continuum and line, single scattering albedo and asymmetry factor. Additional printouts for each term in the k-fit are produced if KDIST=1.
04	16	print flux divergence at each level in the atmosphere

For example to obtain gaseous absorption and cloud parameter info on the same run set ICKP = 2+4 = 6. To obtain all diagnostic information set ICKP = 31. About a page worth of data is written for each wavelength in the calculation.

NOTE: Diagnostic output is overwritten for each new run of the rt code.

ZOUT: 2 element array specifying BOT and TOP altitude points (km) for IOUT output. For example ZOUT=0,50 specifies output information for 0 and 50 km. The surface is always set at zero. Note that the actual layers for which output is generated is determined by finding the atmospheric layers nearest the chosen value of ZOUT(1) and ZOUT(2). (default = 0,100)

IOUT: STANDARD OUTPUT SELECTOR

value

1. one output record for each wavelength, output quantities are,

   WL,FFV,TOPDN,TOPUP,TOPDIR,BOTDN,BOTUP,BOTDIR

WL	= wavelength	(microns)
FFV	= filter function value	
TOPDN	= total downward flux at ZOUT(2) km	(w/m2/micron)
TOPUP	= total upward flux at ZOUT(2) km	(w/m2/micron)
TOPDIR	= direct downward flux at ZOUT(2) km	(w/m2/micron)
BOTDN	= total downward flux at ZOUT(1) km	(w/m2/micron)
BOTUP	= total upward flux at ZOUT(1) km	(w/m2/micron)
BOTDIR	= direct downward flux at ZOUT(1) km	(w/m2/micron)

   NOTE: When ISAT ne 1 these radiometric quantities are each multiplied by the filter function, To get the actual specific irradiance divide by FFV(WL).

2. one output record per wavelength

output quantities are,

```
WL,TXH2O,TXCO2,TXO3,TXN2O,TXCO,TXCH4,TXO2N2,TXTRC,TXTOT,TXMOL
```

```
WL     = wavelength
TXH2O  = -log transmission due to water vapor
TXCO2  = -log transmission due to co2
TXO3   = -log transmission due to ozone
TXN2O  = -log transmission due to n2o
TXCO   = -log transmission due to co
TXCH4  = -log transmission due to ch4
TXO2N2 = -log transmission due to o2 and n2
TXTRC  = -log transmission due to trace gases
TXTOT  = -log transmission due to all gases
TXMOL  = optical depth due to rayleigh scattering

NOTE: if you define the optical depth as
      transmission = exp(-tau) then
      -log transmission = tau
```

3. Averaged gas absorption over solar spectrum and filter function. Output format:

```
write(*,'(5x,11a13)') 'z','airmass','h2o','co2','o3',
&             'n2o','co','ch4','o2+n2','trace','total'
do j=nz,1,-1
   write(*,'(i5,1p11e13.5)') j,z(j),airmass(j),
&    (-log(eps+trnsgas(i,j)/phidw),i=1,nta)
```

```
where j is the level index
      z is the level height (km)
      airmass = g * integral(rho dz/mu) / Pzero
          where g=9.8m/s2, pzero 1013.25mb
          rho is the mass desity of air, and
          mu is the cosine of the solar zenith angle (SZA)
      trnsgas is the transmission due to the species
          listed in the title line
```

the output quantity is the negative log of the transmission which, aside from non-Beer's law behaviour, is like optical depth. If the input quantity NF is non-zero then the transmission is averaged over the solar spectrum. If NF=0 the average is over the filter function. Remember to set NF=0 and SZA=0 when dealing with LW radiation.

5. nzen+3) records for each wavelength. Output format:

```
write(*,*) '"tbf'     ; Block id (used in postprocessors)

do m=1,nw
  write(*,*)
&    wl,ffv,topdn,topup,topdir,botdn,botup,botdir
  write(*,*)  nphi,nzen
  write(*,*) (phi(j),j=1,nphi)
  write(*,*) (uzen(j),j=1,nzen)
  do i=nzen,1,-1
```

```
          write(*,*) (uurs(i,k),k=1,nphi)
        enddo
      enddo
```

where,

```
   WL    = wavelength                            (microns)
   FFV   = filter function value
   TOPDN = total downward flux at ZOUT(2) km  (w/m2/micron)
   TOPUP = total upward flux at ZOUT(2) km    (w/m2/micron)
   TOPDIR= direct downward flux at ZOUT(2) km (w/m2/micron)
   BOTDN = total downward flux at ZOUT(1) km  (w/m2/micron)
   BOTUP = total upward flux at  ZOUT(1) km   (w/m2/micron)
   BOTDIR= direct downward flux at ZOUT(1) km (w/m2/micron)
   NPHI  = number of user azimuth angles
   NZEN  = number of user zenith angles
   PHI   = user specified azimuth angles      (degrees)
   UZEN  = user specified zenith angles       (degrees)
   UURS  = radiance at user angles at         (w/m2/um/str)
           altitude ZOUT(2) (top)
```

NOTE: The radiance output from SBDART represents scattered radiation. It does not include the solar direct beam. Also, keep in mind that UURS represents the radiance at the user specified sample directions. Hence, computing the irradiance by an angular integration of UURS will not yield BOTDN because of the neglect of the direct beam, and it will probably not yield (BOTDN-BOTDIR) because of under-sampling.

6. same as IOUT=5 except radiance is for ZOUT(1) altitude (bottom)

7. radiative flux at each layer for each wavelength. This output option can produce a huge amount of output if many wavelength sample points are used

```
   write(*,*) '"fzw'       ; block id (used in postprocessors)
   write(*,*) nz           ; number of z levels
   write(*,*) nw           ; number of wavelengths

   do j=1,nw
     write(*,*) wl
     write(*,*)
  &    (Z(i),i=nz,1,-1),   ; altitude              (km)
  &    (fdird(i),i=1,nz),  ; downward direct flux  (w/m2/um)
  &    (fdifd(i),i=1,nz),  ; downward diffuse flux (w/m2/um)
  &    (flxdn(i),i=1,nz),  ; total downward flux   (w/m2/um)
  &    (flxup(i),i=1,nz)   ; total upward flux     (w/m2/um)
   enddo
```

10. one output record per run, integrated over wavelength. output quantities are, (integrations by trapezoid rule)

```
   WLINF,WLSUP,FFEW,TOPDN,TOPUP,TOPDIR,BOTDN,BOTUP,BOTDIR

   WLINF = lower wavelength limit             (microns)
   WLSUP = upper wavelength limit             (microns)
   FFEW  = filter function equivalent width   (microns)
   TOPDN = total downward flux at ZOUT(2) km  (w/m2)
   TOPUP = total upward flux at ZOUT(2) km    (w/m2)
   TOPDIR= direct downward flux at ZOUT(2) km (w/m2)
   BOTDN = total downward flux at ZOUT(1) km  (w/m2)
   BOTUP = total upward flux at  ZOUT(1) km   (w/m2)
   BOTDIR= direct downward flux at ZOUT(1) km (w/m2)
```

11. radiant fluxes at each atmospheric layer integrated over wavelength. Output format:

```
   write(*,*) nz,phidw
   do i=1,nz
     write(*,*) zz,pp,fxdn(i),fxup(i),fxdir(i),dfdz,heat
   enddo
```

```
 where,  nz    = number of atmospheric layers
         ffew  = filter function equivalent width(um)
         zz    = level altitudes                  (km)
         pp    = level pressure                   (mb)
         fxdn  = downward flux (direct+diffuse)   (W/m2)
         fxup  = upward flux                      (W/m2)
         fxdir = downward flux, direct beam only  (W/m2)
         dfdz  = radiant energy flux divergence   (mW/m3)
         heat  = heating rate                     (K/day)
```

NOTE: dfdz(i) and heat(i) are defined at the layer centers, i.e., halfway between level i-1 and level i.

20. radiance output at ZOUT(2) km.

Output format:

```
   write(*,*)  wlinf,wlsup,ffew,topdn,topup,topdir,
  &            botdn,botup,botdir
   write(*,*)  nphi,nzen
   write(*,*)  (phi(i),i=1,nphi)
   write(*,*)  (uzen(j),j=1,nzen)
   write(*,*)  ((r(i,j),i=1,nphi),j=1,nzen)
```

The first record of output is the same as format IOUT=10 (WLINF,WLSUP,FFEW,TOPDN,TOPUP,TOPDIR,BOTDN,BOTUP,BOTDIR) addition records contain:

```
   NPHI  = number of user azimuth angles
   NZEN  = number of user zenith angles
   PHI   = user azimuth angles (nphi values)
   UZEN  = user zenith angles (nzen values)
   R     = radiance array (nphi,nzen) (W/m2/sr)
```

21. same as IOUT=20 except radiance output at ZOUT(1) km.

22. radiance and flux at each atmospheric layer integrated over wavelength.

Output format:

```
write(*,*) nphi,nzen,nz,ffew
write(*,*) (phi(i),i=1,nphi)
write(*,*) (uzen(j),j=1,nzen)
write(*,*) (z(k),k=nz,1,-1)
write(*,*) (fxdn(k),k=1,nz)
write(*,*) (fxup(k),k=1,nz)
write(*,*) (fxdir(k),k=1,nz)
write(*,*) (((uurl(i,j,k),i=1,nphi),j=1,nzen),k=1,nz)
```

where,

nphi	= number of user specified azimuth angles	
nzen	= number of user specified zenith angles	
nz	= number of atmospheric levels	
ffew	= filter function equivalent width	(um)
phi	= user specified anizmuth angles	(degrees)
uzen	= user specified zenith angles	(degrees)
z	= altitudes of atmospheric layers	(km)
fxdn	= downward flux (direct+diffuse)	(W/m2)
fxup	= upward flux	(W/m2)
fxdir	= downward flux, direct beam only	(W/m2)
uurl	= radiance at each layer	(W/m2/str)

23. same as IOUT=20 except
lower hemisphere radiance output corresponds to ZOUT(1)
upper hemisphere radiance output corresponds to ZOUT(2)
Use this output format to determine radiance above and and below a scattering layer. For example, if ZCLOUD=1 and TCLOUD=10, you can get the scattered radiation field above and below the cloud with, IOUT=23, ZOUT=1,2.

==========

DISORT options (NAMELIST $DINPUT):

DELTAM: if set to true, use delta-m method (see Wiscombe, 1977). This method is essentially a delta-Eddington approximation applied to multiple radiation streams.

In general, for a given number of streams, intensities and fluxes will be more accurate for phase functions with a large forward peak if 'DELTAM' is set TRUE. Intensities within 10 degrees or so of the forward scattering direction will often be less accurate, however, so when primary interest centers in this so-called 'aureole region', DELTAM should be set FALSE.(default=true)

NSTR: number of computational zenith angles used. NSTR must be divisible by 2. Using NSTR=4 reduces the time required for flux calculations by about a factor of 5 compared to NSTR=16, with very little penalty in accuracy (about 0.5% difference when DELTAM is set true).

IMOM: Setting IMOM gt 0 causes SBDART to read a direct access file, pmom.dat, which contains cloud scattering parameters. If IMOM=1 the code uses the Legendre expansion coefficients of the phase function read from pmom.dat. If IMOM=2 Q, omega and g parameters are used but the phase function coefficients are not. Instead, the phase function is derived using the Henyey-Greenstein approximation. The default (IMOM=0) is to use the Q, omega and g parameters computed from the Wiscombe's Mie scattering code, which are hard-coded into SBDART. The file format of pmom.dat is described in subroutine cloudqwp (module cloudpar).

Radiance output

NZEN: Number of user zenith angles. If this parameter is specified SBDART will output radiance values at NZEN zenith angles, evenly spaced between the first two values of input array UZEN. For example,

```
 nzen=9,
 uzen=0,80
```

will cause output at zenith angles 0,10,20,30,40,50,60,70,80.

UZEN: User zenith angles. If NZEN is specified then UZEN is interpreted as the limits of the zenith angle range, and only the first two elements are required. If NZEN is not specified then up to NSTR values of UZEN may be specified. (i.e., the zenith angle at which the radiation propagates)

* UZEN = 0 => radiation propagates directly up
* UZEN < 90 => radiation in upper hemisphere
* UZEN > 90 => radiation in lower hemisphere
* UZEN = 180 => radiation propagates directly down

NPHI: Number of user azimuth angles. If this parameter is specified SBDART will output radiance values at NPHI azimuth angles, evenly spaced between the first two values of input array PHI. For example, nphi=7, phi=0,180 will cause output at zenith angles 0,30,60,90,120,150,180

PHI: User azimuth angles. If NPHI is specified then PHI is interpreted as the limits of the azimuth angle range, and only the first two elements are required. If NPHI is not specified then up to NSTR values of PHI may be specified.

NOTE: Azimuth increases clockwise looking down on the Earth's surface. If PHI0=0, PHI is interpreted as a relative azimuth angle from the forward scattering direction. See discussion of PHI0

* PHI-PHI0 < 90 => forward scattered radiation
* PHI-PHI0 > 90 => backward scattered radiation

For example, if the sun is setting in the West, radiation propagating to the South-East has a relative azimuth of 45 degrees.

NOTE: A radiance calculation is performed if and only if some values of UZEN are specified and a radiance output format is selected (see IOUT). If UZEN is specified and PHI is not, then an azimuth avereaged calculation is performed. In any case, the number of values specified for UZEN and PHI need not match.

NOTE: SBDART is currently configured to model radiation with at most 20 computational zenith angles and 20 azimuthal modes. While these limits may be expanded, be aware that running SBDART with a much larger number will significantly increase running time and memory requirements. In tests performed on a DEC Alpha, the execution time scaled roughly with NSTR^2, for NSTR less than 20. The code's memory usage also scales roughly as NSTR^2.

PHI0: azimuth angle of incident beam. Use this parameter to relate the radiance output to fixed navigational headings. For example if the sun is positioned at zenith=10 and azimuth=110 degrees (measured clockwise from due north) then setting PHI0=-70 degrees will cause PHI to be interpreted as a compass direction. In this example the forward scattering peak will be found at uzen=170, phi=-70. Otherwise if PHI0 is zero (the default value), PHI is interpreted as a relative azimuth angle (i.e., relative to the forward scattering direction).

If the solar zenith and azimuth are set using IDAY, TIME, ALAT, and ALON, then PHI0 is automatically set to the computed solar azimuth plus 180 degrees. In this case the input value of PHI0 is ignored.

radiation boundary conditions

```
 IBCND:
       = 0 : general case: boundary conditions any combination of:
              * beam illumination from the top      ( see FBEAM)
              * isotropic illumination from the top ( see FISOT)
              * thermal emission from the top       ( see TEMIS,TTEMP)
              * internal thermal emission sources   ( see TEMPER)
              * reflection at the bottom            ( see LAMBER,ALBEDO,HL)
              * thermal emission from the bottom    ( see BTEMP)
       = 1 : isotropic illumination from top and bottom, in order to get
             ALBEDO and transmissivity of the entire medium vs. incident
             beam angle;
             The only input variables considered in this case are
             NLYR,DTAUC,SSALB,PMOM,NSTR,USRANG,NUMU,UMU,ALBEDO,DELTAM,
             PRNT,HEADER, and the array dimensions.
             NOPLNK,LAMBER are assumed TRUE, the bottom boundary can have
             any ALBEDO.  the sole output is ALBMED,TRNMED.  UMU is
             interpreted as the array of beam angles in this case.
             If USRANG = TRUE they must be positive and in increasing
             order, and will be returned this way; internally, however, the
             negatives of the UMU's are added, so MAXUMU must be at least
             2*NUMU.
             If USRANG = FALSE, UMU is returned as the NSTR/2 positive
             quadrature angle cosines, in increasing order.



 FISOT:      intensity of top-boundary isotropic illumination.  (units
             w/sq m if thermal sources active, otherwise arbitrary
             units).  corresponding incident flux is pi (3.14159...)
             times 'FISOT'.


 TEMIS:      emissivity of top layer, needed only if NOTHRM=0



           DISORT specific output options:
           ===================================
PRNT:
PRNT(1)    print input information
PRNT(2)    print layer-by-layer fluxes
PRNT(3)    print azimuthally averaged intensities at quadrature angles
PRNT(4)    print azimuthally averaged intensities at user angles
PRNT(5)    print intensities at user angles
PRNT(6)    print planar albedo and transmissivity of medium
                   as a function of incident beam angle
PRNT(7)    print phase function moments (iff prnt(1) true)


=========================================================================
```