502-EMD-001

HDFEOS5 Converted NCEP Stratospheric Analysis Combined Products Specifications for the EMD Project

Technical Paper

February 2004

Prepared Under Contract NAS5-03098

RESPONSIBLE AUTHOR

Abe Taaheri /s/ 2/4/04

Abe Taaheri, Toolkit Task Leader Date
ECS Maintenance and Development Project

RESPONSIBLE OFFICE

Larry Klein /s/ 2/5/04

Larry Klein, Manager Toolkit Date
ECS Maintenance and Development Project

Raytheon Company
Upper Marlboro, Maryland

Abstract

This document will serve as the guide to the HDF-EOS5 format converted NCEP Stratospheric Analysis Combined Data Products. HDF refers to the scientific data format standard selected by NASA as the baseline standard for EOS, and HDF-EOS refers to EOS conventions for using HDF. This document will provide information on the contents of native binary format (in IEEE Little-Endian byte order) NCEP files and the HDF-EOS5 files that are created using data from the binary files.

***Keywords**:* NCEP Stratospheric Analysis Combined Data Products, HDF-EOS, National Meteorological Center, Geopotential Height, Temperature, Wind, Moisture, NMC, Grid

Contents

Abstract

1. Introduction

2. NCEP Native Binary Files

3. Structure of HDFEOS5-Converted NCEP Files

3.1. MOISTURE 4

3.2. TEMPERATURE 5

3.3. WIND 5

3.4. HEIGHT 6

4. Accessing HDFEOS5-Converted NCEP Files

Appendix A: Sample Inventory Metadata

References

1. Introduction

The MLS and HIRDLS instrument teams need the NCEP Stratospheric Analysis Combined Data Products as processing input. The products need to be made available from the ECS archive in HDF-EOS format.

The NCEP Stratospheric Analysis combined Data Products will be delivered daily to the data link server at GDAAC in native binary format (in IEEE Little-Endian byte order). The ECS will convert the native files to HDF-EOS5 format and archive both formats.

2. NCEP Native Binary Files

The NCEP Stratospheric Analysis Data products contain the same data as the UARS Correlative NMC data files, except that NCEP files are produced in IEEE Little-Endian byte order format while NMC files are in VAX/VMS format. The National Meteorological Center (NMC) geopotential heights and temperatures are derived from two analysis systems:

1) tropospheric fields from 1000mb to 100mb, and

2) stratospheric analyses from 70mb to 04.mb

The tropospheric fields from 1000mb to 100mb are the 1200 GMT gridded fields which are part of the Global Daily Assimilation System (GDAS) described by McPherson et al. (1979), where data from radiosondes, aircraft, satellites, ships, buoys, or any other conventional means are assimilated and merged into meteorological fields (heights, temperature, winds). The stratospheric analyses developed by the NMC Climate Analysis Center are 1200 GMT operational analyses at the 70-0.4mb pressure levels, produced from satellite temperature retrievals and RAOBS via a modified Cressman analysis (Gelman et al., 1987), which is a successive iteration method where the fields are iteratively adjusted by weighted data depending on distance from the grid point. Tropospheric temperature analyses use combined NOAA-10 and NOAA-11 data.

The operational analyses now use satellite retrievals based on a minimum variance simultaneous retrieval method described in Goldberg et al. (1988), Fleming et al. (1988), and Fleming et al. (1986). The minimum variance retrieval method is not dependent on week-old regression coeficients of co-located radiosonde data with satellite radiances as was done in the past.

The NMC analyses below 10mb are heavily dependent on RAOBS (particularly over the continents), but less dependent on the RAOBS and more dependent on the TOVS retrievals over ocean areas and ice covered regions.

Moisture (only for Northern hemisphere) and Winds are obtained from the NMC global data assimilation systems (briefly explained above). Analyses from assimilation are on the same gridded fields as temperature and height analyses.

Analyzed fields are produced for the following parameters:

Field Name	***Short Field Name***	***Units***
Temperature	TEMP	K
Geopotential Height	HEIGHT	m
Wind Components:	WINDS	m/s
- u comp. of wind wrt grid	(U_WIND)	
- v comp. of wind wrt grid	(V_WIND)	
Moisture	MOISTURE	%

Analyzed fields will be supplied to the Central Data Handling Facility (CDHF) for 12GMT each day, and transferred to the GSFC DAAC for distribution to the user community and conversion to HDFEOS5 files.

The gridded fields are on the standard 65x65 NMC polar stereographic grid oriented 80W (Grid increment 381km at 60N) and 100E (grid increment 381km at 60S); Pole at (33,33).

Fields are produced at 18 standard meteorological levels from 1000 mb to 0.4 mb for heights and temperatures. For moisture only the 6 lowest altitude pressure levels are used and there is no data for the southern hemisphere. For winds there are both U and V components. The U comp. and V comp. of wind are given with respect to grid in the data files For the northern hemisphere there is wind data only for the 12 lowest altitude pressure levels (up to the 50 mb level). For the southern hemisphere there is wind data only at the 4 levels 1000, 500, 300, and 250 mb.

The 18 Standard pressure levels for data (mb) are: 1000., 850., 700., 500., 400., 300., 250., 200., 150., 100., 70., 50., 30., 10., 5., 2., 1., and 0.4

3. Structure of HDFEOS5-Converted NCEP Files

The NCEP HDFEOS5-converted data products will contain all the data in native binary files and will have the following specifications:

- The files are produced using HDFEOS5 and, therefore, has an extension ".he5".
- Currently the HDFEOS5-converted file names are the same as the BINARY file names, extended by ".he5".
- Each HDFEOS5-converted file is accompanied by a ".met" file, containing INVENTORY metadata. The HDF files themselves contain the same metadata.
- Files with prefixes nmcm, nmct, nmcw, and nmcz contain MOISTURE, TEMPERATURE, WIND, and HEIGHT data respectively.

Sections 3.1 to 3.4 list the structure and content of the products.

3.1. MOISTURE

File names: nmcm_YYMMDD.dat.he5, nmcm_YYMMDD.dat.he5.met

Contents:

- INVENTORY Metadata (see Appendix A for sample)
- Grids: NorthernHemisphere

NorthernHemisphere grid:

Projection: Polar Stereographic
Projection Prameteres:(0.0 0.0 0.0 0.0 -80000000.0 60000000.0 0.0 0.0
0.0 0.0 0.0 0.0 0.0)
XDIM, YDIM: 65, 65
nlevels: 6

Fields in Grid				
Field Name	**Data Type**	**Dimmension List**	**Dimmension Sizes**	**Units**
Moisture	FLOAT	nlevels,YDim,XDim	6,65,65	%
Presure	FLOAT	nlevels	6	Milibars

3.2. TEMPERATURE

File names: nmct_YYMMDD.dat.he5, nmct_YYMMDD.dat.he5.met

Contents:

- INVENTORY Metadata (see Appendix A for sample)
- Grids: NorthernHemisphere, SouthernHemisphere

NorthernHemisphere grid:

Projection: Polar Stereographic
Projection Prameteres:(0.0 0.0 0.0 0.0 -80000000.0 60000000.0 0.0 0.0
0.0 0.0 0.0 0.0 0.0)
XDIM, YDIM: 65, 65
nlevels: 18

Fields in Grid				
Field Name	**Data Type**	**Dimmension List**	**Dimmension Sizes**	**Units**
Temperature	FLOAT	nlevels,YDim,XDim	18,65,65	Degrees Kelvin
Presure	FLOAT	nlevels	18	Milibars

SouthernHemisphere grid:

Projection: Polar Stereographic
Projection Prameteres:(0.0 0.0 0.0 0.0 100000000.0 -60000000.0 0.0 0.0
0.0 0.0 0.0 0.0 0.0)
XDIM, YDIM: 65, 65
nlevels: 18

Fields in Grid				
Field Name	**Data Type**	**Dimmension List**	**Dimmension Sizes**	**Units**
Temperature	FLOAT	nlevels,YDim,XDim	18,65,65	Degrees Kelvin
Presure	FLOAT	nlevels	18	Milibars

3.3. WIND

File names: nmcw_YYMMDD.dat.he5, nmcw_YYMMDD.dat.he5.met

Contents:

- INVENTORY Metadata (see Appendix A for sample)
- Grids: NorthernHemisphere, SouthernHemisphere

NorthernHemisphere grid:
Projection: Polar Stereographic
Projection Prameteres:(0.0 0.0 0.0 0.0 -80000000.0 60000000.0 0.0 0.0
0.0 0.0 0.0 0.0 0.0)
XDIM, YDIM: 65, 65
nlevels: 12

Fields in Grid				
Field Name	**Data Type**	**Dimmension List**	**Dimmension Sizes**	**Units**
U_Wind	FLOAT	nlevels,YDim,XDim	12,65,65	Meters/Sec
V_Wind	FLOAT	nlevels,YDim,XDim	12,65,65	Meters/Sec
Presure	FLOAT	nlevels	12	Milibars

SouthernHemisphere grid:
Projection: Polar Stereographic
Projection Prameteres:(0.0 0.0 0.0 0.0 100000000.0 -60000000.0 0.0 0.0
0.0 0.0 0.0 0.0 0.0)
XDIM, YDIM: 65, 65
nlevels: 4

Fields in Grid				
Field Name	**Data Type**	**Dimmension List**	**Dimmension Sizes**	**Units**
U_Wind	FLOAT	nlevels,YDim,XDim	4,65,65	Meters/Sec
V_Wind	FLOAT	nlevels,YDim,XDim	4,65,65	Meters/Sec
Presure	FLOAT	nlevels	4	Milibars

3.4. HEIGHT

File names: nmcz_YYMMDD.dat.he5, nmcz_YYMMDD.dat.he5.met

Contents:

- INVENTORY Metadata (see Appendix A for sample)
- Grids: NorthernHemisphere, SouthernHemisphere

NorthernHemisphere grid:

Projection: Polar Stereographic

Projection Prameteres:(0.0 0.0 0.0 0.0 -80000000.0 60000000.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0)

XDIM, YDIM: 65, 65

nlevels: 18

Fields in Grid				
Field Name	**Data Type**	**Dimmension List**	**Dimmension Sizes**	**Units**
Height	FLOAT	nlevels,YDim,XDim	18,65,65	Meters
Presure	FLOAT	nlevels	18	Milibars

SouthernHemisphere grid:

Projection: Polar Stereographic

Projection Prameteres:(0.0 0.0 0.0 0.0 100000000.0 -60000000.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0)

XDIM, YDIM: 65, 65

nlevels: 18

Fields in Grid				
Field Name	**Data Type**	**Dimmension List**	**Dimmension Sizes**	**Units**
Height	FLOAT	nlevels,YDim,XDim	18,65,65	Meters
Presure	FLOAT	nlevels	18	Milibars

4. Accessing HDFEOS5-Converted NCEP Files

To access the file contents one should install TOOLKIT, HDF4, HDF5, and HDFEOS5. Also Users must note that the *.he5 files are internally compressed and in order to read them correctly HDF5 must be configured and installed with "zlib". Generally in LINUX platforms HDF5 is configured automatically with system zlib. However, in other platforms one must install HDF4 first and run "./configure" as below before installing HDF5:

"./configure -v --with-zlib=$HDFINC,$HDFLIB --prefix=$install_dir $config_flags"

where HDFINC, HDFLIB, install_dir, and config_flags are environment variables set for HDF4 include directory, HDF4 library directory, directory for installing HDF5, and configuation flags (usually unneccessry), respectively. If one Installs HDF5 during the installation of TOOLKIT, HDF5 installation will be automatically configured with zlib.

The following is a sample FORTRAN code to inquire and extract data from a NCEP HDFEOS5 file:

```
!     program       NCEP_test.f

      implicit      none

      include       'hdfeos5.inc'

      integer       i
      integer       status

      integer       gdfid
      integer       gdid1
      integer       gdid2

      integer       nflds
      integer       rk(32)
      integer       nt(32)

      integer       spherecode
      integer       projcode
      integer       zonecode
      real*8        projparm(13)

      integer       he5_gdopen
      integer       he5_gdattach
      integer       he5_gdrdfld
      integer       he5_gddetach
      integer       he5_gdclose
      integer       he5_gdfldinfo
      integer       he5_gdprojinfo
      integer       he5_gdgridinfo
      integer       he5_gdinqflds
      integer*4     he5_gddiminfo
      integer*4     he5_gdinqdims
      integer*4     he5_gdnentries
      integer*4     he5_gdinqgrid
```

```
      integer*4     start(3)
      integer*4     stride(3)
      integer*4     count(3)
      integer*4     ndim
      integer*4     dims(32)
      integer*4     xdimsize
      integer*4     ydimsize
      integer*4     dimsize
      integer*4     n, strbufsize
      real*8        upleftpt(2)
      real*8        lowrightpt(2)

      character*64  dimname
      character*64  mxdimname
      character*64  fieldlist
      character*64  gridlist
      real*4        temp(65,65,18)

      integer       FAIL
      parameter     (FAIL=-1)
      integer        ngrid

!     Open NCEP's HDF-EOS5 "*.he5" file
!     -----------------------------
      gdfid = he5_gdopen("nmct_030126.dat.he5", HE5F_ACC_RDONLY)
      write(*,*) 'File ID returned by he5_gdopen():  ', gdfid

      if (gdfid .ne. FAIL) then

!     Inquire grids
!     -----------------
         ngrid = he5_gdinqgrid('nmct_030126.dat.he5', gridlist,
     1        strbufsize)
         if(ngrid .lt. 1) then
            write(*,*) 'There is no grid in the file'
            stop
         endif

         write(*,*) 'Grids are:', gridlist

!     Attach to the NorthernHemisphere and SouthernHemisphere grids
!     -------------------------------------------------------------
         gdid1 = he5_gdattach(gdfid, 'NorthernHemisphere')
         write(*,*) 'Grid ID returned by he5_gdattach():  ',gdid1


         gdid2 = he5_gdattach(gdfid, 'SouthernHemisphere')
         write(*,*) 'Grid ID returned by he5_gdattach():  ',gdid2

!     Inquire dimensions
!       -----------------
         if(gdid1 .ne. FAIL) then
            ndim = he5_gdinqdims(gdid1, dimname, dims)
            write(*,*) 'Dimension list (NorthernHemisphere):', dimname
            do i=1,ndim
               write(*,*) 'dim size: ', dims(i)
            enddo
         endif

         if(gdid2 .ne. FAIL) then
            ndim = he5_gdinqdims(gdid2, dimname, dims)
            write(*,*) 'Dimension list (SouthernHemisphere):', dimname
```

```
            do i=1,ndim
               write(*,*) 'dim size: ', dims(i)
            enddo
         endif

!     Get the sizes of certain dimensions
!     -----------------------------------
         if(gdid1 .ne. FAIL) then
            dimsize = he5_gddiminfo(gdid1, 'nLevels')
            write(*,*) 'Size of "nLevels" dim (NorthernHemisphere):',
     1           dimsize
         endif

         if(gdid2 .ne. FAIL) then
            dimsize = he5_gddiminfo(gdid2, 'nLevels')
            write(*,*) 'Size of "nLevels" dim (SouthernHemisphere):',
     1           dimsize
         endif

!     Get grid parameters
!     -------------------
         if(gdid1 .ne. FAIL) then
            status = he5_gdgridinfo(gdid1,xdimsize,ydimsize,
     1           upleftpt,lowrightpt)
            write(*,*)'X dim size, Y dim size (NorthernHemisphere): ',
     2           xdimsize,ydimsize

            write(*,*) 'Up left pt (NorthernHemisphere): ',upleftpt(1),
     1           upleftpt(2)
            write(*,*) 'Low right pt (NorthernHemisphere): ',
     2           lowrightpt(1),lowrightpt(2)
         endif

         if(gdid2 .ne. FAIL) then
            status = he5_gdgridinfo(gdid2,xdimsize,ydimsize,
     1           upleftpt, lowrightpt)
            write(*,*) 'X dim size, Y dim size (SouthernHemisphere): ',
     2           xdimsize,ydimsize

            write(*,*) 'Up left pt (SouthernHemisphere): ',upleftpt(1),
     1           upleftpt(2)
            write(*,*) 'Low right pt (SouthernHemisphere): ',
     2           lowrightpt(1), lowrightpt(2)
         endif

!     Get projection parameters
!     -------------------------
         write(*,*) ' '
         if(gdid1 .ne. FAIL) then
            status = he5_gdprojinfo(gdid1,projcode,zonecode,
     1           spherecode,projparm)
            write(*,*) 'projcode,zonecode (NorthernHemisphere):',
     2           projcode, zonecode
            write(*,*) 'spherecode (NorthernHemisphere): ', spherecode
            do i=1,13
               write(*,*) 'Projection Parameter: ',i,projparm(i)
            enddo
         endif

         write(*,*) ' '
         if(gdid2 .ne. FAIL) then
            status = he5_gdprojinfo(gdid2,projcode,zonecode,
```

```
     1           spherecode,projparm)
            write(*,*) 'projcode (SouthernHemisphere): ', projcode
            write(*,*) 'spherecode (SouthernHemisphere): ', spherecode
            do i=1,13
               write(*,*) 'Projection Parameter: ',i,projparm(i)
            enddo
         endif

!     Get information about fields
!     ----------------------------
         if(gdid1 .ne. FAIL) then
            nflds = he5_gdinqflds(gdid1, fieldlist, rk, nt)
            if (nflds .ne. 0) then
               write(*,*) 'Data fields (NorthernHemisphere):',fieldlist
               do i=1,nflds
                  write(*,*) 'rank type: ',rk(i),nt(i)
               enddo
            endif
         endif

         if(gdid2 .ne. FAIL) then
            nflds = he5_gdinqflds(gdid2, fieldlist, rk, nt)
            if (nflds .ne. 0) then
               write(*,*) 'Data fields (SouthernHemisphere):',fieldlist
               do i=1,nflds
                  write(*,*) 'rank type: ',rk(i),nt(i)
               enddo
            endif
         endif

         if(gdid1 .ne. FAIL) then
            write(*,*) ' '
            write(*,*) 'NorthernHemisphere:'
            status = he5_gdfldinfo(gdid1,'Temperature',rk,dims,nt,
     1           dimname,mxdimname)
            write(*,*) 'Temperature rank dims: ',rk(1)
            write(*,*) 'Temperature dim names: ',dimname
            write(*,*) 'Temperature max. dim names: ',mxdimname
            do i=1,rk(1)
               write(*,*) 'Temperature dims: ',i,dims(i)
            enddo
         endif

         if(gdid2 .ne. FAIL) then
            write(*,*) ' '
            write(*,*) 'NorthernHemisphere:'
            status = he5_gdfldinfo(gdid2,'Temperature',rk,dims,nt,
     1           dimname,mxdimname)
            write(*,*) 'Temperature rank dims: ',rk(1)
            write(*,*) 'Temperature dim names: ',dimname
            write(*,*) 'Temperature max. dim names: ',mxdimname
            do i=1,rk(1)
               write(*,*) 'Temperature dims: ',i,dims(i)
            enddo
         endif

!     Get number of grid dimensions and dim. list length
!     --------------------------------------------------
         if(gdid1 .ne. FAIL) then
            n = he5_gdnentries(gdid1, HE5_HDFE_NENTDIM, strbufsize)
         write(*,*) 'Number of dimension entries (NorthernHemisphere):',
     1           n
```

```
         write(*,*) 'Length of Dimension List (NorthernHemisphere): ',
     1           strbufsize
         endif
!     Get number of data fields and field list length
!     -----------------------------------------------
         if(gdid1 .ne. FAIL) then
            n = he5_gdnentries(gdid1, HE5_HDFE_NENTDFLD, strbufsize)
            write(*,*) 'Number of data fields (NorthernHemisphere): ',
     1           n
            write(*,*) 'Length of Field List (NorthernHemisphere): ',
     1           strbufsize
         endif

         start(1)  = 0
         start(2)  = 0
         start(3)  = 0
         stride(1) = 1
         stride(2) = 1
         stride(3) = 1
         count(1)  = xdimsize
         count(2)  = ydimsize
         count(3)  = dims(3)

!     Read the data from "Temperature" field
!     --------------------------------------
         if (gdid1 .ne. FAIL) then
            status = he5_gdrdfld(gdid1,"Temperature",
     1           start,stride,count,temp)
            write(*,*) 'Status returned by he5_gdrdfld():  ',status

            write(*,*) 'temp(1,1,3) = ',temp(1,1,3),
     1           '      temp(65,65,3) = ',temp(65,65,3)


         endif
!     Detach from the grid
!     --------------------
         if(gdid2 .ne. FAIL) then
            status = he5_gddetach(gdid2)
            write(*,*) 'Status returned by he5_gddetach():  ',status
         endif

         if(gdid1 .ne. FAIL) then
            status = he5_gddetach(gdid1)
            write(*,*) 'Status returned by he5_gddetach():  ',status
         endif

!     Close the file
!     --------------
         status = he5_gdclose(gdfid)
         write(*,*) 'Status returned by he5_gdclose():  ',status
      endif
      stop
      end
```

Appendix A: Sample Inventory Metadata

Followings are the sample INVENTORY metadata for all 4 NCEP products that are written to both output HDF files and ASCII *.met files during the conversion from binary to HDF-EOS5 format.

A.1 MOISTURE

```
GROUP                   = INVENTORYMETADATA
  GROUPTYPE             = MASTERGROUP

  GROUP                   = ECSDATAGRANULE

    OBJECT                  = PRODUCTIONDATETIME
      NUM_VAL               = 1
      VALUE                 = "2003-05-15T17:19:07.000Z"
    END_OBJECT              = PRODUCTIONDATETIME

  END_GROUP               = ECSDATAGRANULE

  GROUP                   = COLLECTIONDESCRIPTIONCLASS

    OBJECT                  = VERSIONID
      NUM_VAL               = 1
      VALUE                 = 1
    END_OBJECT              = VERSIONID

    OBJECT                  = SHORTNAME
      NUM_VAL               = 1
      VALUE                 = "SAMOISTH"
    END_OBJECT              = SHORTNAME

  END_GROUP               = COLLECTIONDESCRIPTIONCLASS

  GROUP                   = SPATIALDOMAINCONTAINER

    GROUP                   = HORIZONTALSPATIALDOMAINCONTAINER

      GROUP                   = BOUNDINGRECTANGLE

        OBJECT                  = EASTBOUNDINGCOORDINATE
          NUM_VAL               = 1
          VALUE                 = 180.0
        END_OBJECT              = EASTBOUNDINGCOORDINATE

        OBJECT                  = WESTBOUNDINGCOORDINATE
          NUM_VAL               = 1
          VALUE                 = -180.0
        END_OBJECT              = WESTBOUNDINGCOORDINATE

        OBJECT                  = SOUTHBOUNDINGCOORDINATE
          NUM_VAL               = 1
```

```
          VALUE                = -90.0
        END_OBJECT             = SOUTHBOUNDINGCOORDINATE

        OBJECT                 = NORTHBOUNDINGCOORDINATE
          NUM_VAL              = 1
          VALUE                = 90.0
        END_OBJECT             = NORTHBOUNDINGCOORDINATE

      END_GROUP              = BOUNDINGRECTANGLE

    END_GROUP              = HORIZONTALSPATIALDOMAINCONTAINER

  END_GROUP              = SPATIALDOMAINCONTAINER

  GROUP                  = SINGLEDATETIME

    OBJECT                 = TIMEOFDAY
      NUM_VAL              = 1
      VALUE                = "12:00:00.00"
    END_OBJECT             = TIMEOFDAY

    OBJECT                 = CALENDARDATE
      NUM_VAL              = 1
      VALUE                = "2003-01-26"
    END_OBJECT             = CALENDARDATE

  END_GROUP              = SINGLEDATETIME

END_GROUP              = INVENTORYMETADATA

END
```

A.2 TEMPERATURE

```
GROUP                  = INVENTORYMETADATA
  GROUPTYPE            = MASTERGROUP

  GROUP                  = ECSDATAGRANULE

    OBJECT                 = PRODUCTIONDATETIME
      NUM_VAL              = 1
      VALUE                = "2003-05-15T17:19:16.000Z"
    END_OBJECT             = PRODUCTIONDATETIME

  END_GROUP              = ECSDATAGRANULE

  GROUP                  = COLLECTIONDESCRIPTIONCLASS

    OBJECT                 = VERSIONID
      NUM_VAL              = 1
      VALUE                = 1
    END_OBJECT             = VERSIONID
```

```
    OBJECT                  = SHORTNAME
      NUM_VAL               = 1
      VALUE                 = "SATEMPH"
    END_OBJECT              = SHORTNAME

  END_GROUP               = COLLECTIONDESCRIPTIONCLASS

  GROUP                   = SPATIALDOMAINCONTAINER

    GROUP                   = HORIZONTALSPATIALDOMAINCONTAINER

      GROUP                   = BOUNDINGRECTANGLE

        OBJECT                  = EASTBOUNDINGCOORDINATE
          NUM_VAL               = 1
          VALUE                 = 180.0
        END_OBJECT              = EASTBOUNDINGCOORDINATE

        OBJECT                  = WESTBOUNDINGCOORDINATE
          NUM_VAL               = 1
          VALUE                 = -180.0
        END_OBJECT              = WESTBOUNDINGCOORDINATE

        OBJECT                  = SOUTHBOUNDINGCOORDINATE
          NUM_VAL               = 1
          VALUE                 = -90.0
        END_OBJECT              = SOUTHBOUNDINGCOORDINATE

        OBJECT                  = NORTHBOUNDINGCOORDINATE
          NUM_VAL               = 1
          VALUE                 = 90.0
        END_OBJECT              = NORTHBOUNDINGCOORDINATE

      END_GROUP               = BOUNDINGRECTANGLE

    END_GROUP               = HORIZONTALSPATIALDOMAINCONTAINER

  END_GROUP               = SPATIALDOMAINCONTAINER

  GROUP                   = SINGLEDATETIME

    OBJECT                  = TIMEOFDAY
      NUM_VAL               = 1
      VALUE                 = "12:00:00.00"
    END_OBJECT              = TIMEOFDAY

    OBJECT                  = CALENDARDATE
      NUM_VAL               = 1
      VALUE                 = "2003-01-26"
    END_OBJECT              = CALENDARDATE

  END_GROUP               = SINGLEDATETIME

END_GROUP               = INVENTORYMETADATA

END
```

A.3 WIND

```
GROUP                   = INVENTORYMETADATA
  GROUPTYPE             = MASTERGROUP

  GROUP                   = ECSDATAGRANULE

    OBJECT                  = PRODUCTIONDATETIME
      NUM_VAL               = 1
      VALUE                 = "2003-05-15T17:19:20.000Z"
    END_OBJECT              = PRODUCTIONDATETIME

  END_GROUP               = ECSDATAGRANULE

  GROUP                   = COLLECTIONDESCRIPTIONCLASS

    OBJECT                  = VERSIONID
      NUM_VAL               = 1
      VALUE                 = 1
    END_OBJECT              = VERSIONID

    OBJECT                  = SHORTNAME
      NUM_VAL               = 1
      VALUE                 = "SAWINDSH"
    END_OBJECT              = SHORTNAME

  END_GROUP               = COLLECTIONDESCRIPTIONCLASS

  GROUP                   = SPATIALDOMAINCONTAINER

    GROUP                   = HORIZONTALSPATIALDOMAINCONTAINER

      GROUP                   = BOUNDINGRECTANGLE

        OBJECT                  = EASTBOUNDINGCOORDINATE
          NUM_VAL               = 1
          VALUE                 = 180.0
        END_OBJECT              = EASTBOUNDINGCOORDINATE

        OBJECT                  = WESTBOUNDINGCOORDINATE
          NUM_VAL               = 1
          VALUE                 = -180.0
        END_OBJECT              = WESTBOUNDINGCOORDINATE

        OBJECT                  = SOUTHBOUNDINGCOORDINATE
          NUM_VAL               = 1
          VALUE                 = -90.0
        END_OBJECT              = SOUTHBOUNDINGCOORDINATE

        OBJECT                  = NORTHBOUNDINGCOORDINATE
          NUM_VAL               = 1
          VALUE                 = 90.0
        END_OBJECT              = NORTHBOUNDINGCOORDINATE

      END_GROUP               = BOUNDINGRECTANGLE
```

```
    END_GROUP               = HORIZONTALSPATIALDOMAINCONTAINER

  END_GROUP               = SPATIALDOMAINCONTAINER

  GROUP                   = SINGLEDATETIME

    OBJECT                  = TIMEOFDAY
      NUM_VAL               = 1
      VALUE                 = "12:00:00.00"
    END_OBJECT              = TIMEOFDAY

    OBJECT                  = CALENDARDATE
      NUM_VAL               = 1
      VALUE                 = "2003-01-26"
    END_OBJECT              = CALENDARDATE

  END_GROUP               = SINGLEDATETIME

END_GROUP               = INVENTORYMETADATA

END
```

A.4 HEIGHT

```
GROUP                   = INVENTORYMETADATA
  GROUPTYPE             = MASTERGROUP

  GROUP                   = ECSDATAGRANULE

    OBJECT                  = PRODUCTIONDATETIME
      NUM_VAL               = 1
      VALUE                 = "2003-05-15T17:19:05.000Z"
    END_OBJECT              = PRODUCTIONDATETIME

  END_GROUP               = ECSDATAGRANULE

  GROUP                   = COLLECTIONDESCRIPTIONCLASS

    OBJECT                  = VERSIONID
      NUM_VAL               = 1
      VALUE                 = 1
    END_OBJECT              = VERSIONID

    OBJECT                  = SHORTNAME
      NUM_VAL               = 1
      VALUE                 = "SAGHGTH"
    END_OBJECT              = SHORTNAME

  END_GROUP               = COLLECTIONDESCRIPTIONCLASS

  GROUP                   = SPATIALDOMAINCONTAINER

    GROUP                   = HORIZONTALSPATIALDOMAINCONTAINER

      GROUP                   = BOUNDINGRECTANGLE
```

```
        OBJECT                  = EASTBOUNDINGCOORDINATE
          NUM_VAL               = 1
          VALUE                 = 180.0
        END_OBJECT              = EASTBOUNDINGCOORDINATE

        OBJECT                  = WESTBOUNDINGCOORDINATE
          NUM_VAL               = 1
          VALUE                 = -180.0
        END_OBJECT              = WESTBOUNDINGCOORDINATE

        OBJECT                  = SOUTHBOUNDINGCOORDINATE
          NUM_VAL               = 1
          VALUE                 = -90.0
        END_OBJECT              = SOUTHBOUNDINGCOORDINATE

        OBJECT                  = NORTHBOUNDINGCOORDINATE
          NUM_VAL               = 1
          VALUE                 = 90.0
        END_OBJECT              = NORTHBOUNDINGCOORDINATE

      END_GROUP               = BOUNDINGRECTANGLE

    END_GROUP               = HORIZONTALSPATIALDOMAINCONTAINER

  END_GROUP               = SPATIALDOMAINCONTAINER

  GROUP                   = SINGLEDATETIME

    OBJECT                  = TIMEOFDAY
      NUM_VAL               = 1
      VALUE                 = "12:00:00.00"
    END_OBJECT              = TIMEOFDAY

    OBJECT                  = CALENDARDATE
      NUM_VAL               = 1
      VALUE                 = "2003-01-26"
    END_OBJECT              = CALENDARDATE

  END_GROUP               = SINGLEDATETIME

END_GROUP               = INVENTORYMETADATA

END
```

References

McPherson, R., D., K. H. Bergman, R. E. Kistler, G. E. Rasch, and D. S. Gordon: "The NMC Operational Global Data Assimilation System," Mon. Wea. Rev., Vol. 107, pp 1445-61.

Fleming, H. E., M. E. Goldberg, and D. S. Crosby, 1988: "Operational Implementation of the Minimum Variance Simultaneous Retrieval Method", Proceedings, AMS Third Conference on Satellite Meteorology and Oceanography, Jan. 31 - Feb. 5, 1988, Anaheim, Calif., pp.16-19.

--,--,--, 1986: "Minimum Variance Simultaneous Retrieval of Temperature and Water Vapor from Satellite Radiance Measurements", Preprint Volume, Second Conference on Satellite Meteorology/Remote Sensing and Applications, May 13-16, 1986, Williamsburg, Va. Published by the American Meteorological Society, Boston, Mass., pp 20-23.

Golberg, M. D. and J. M. Daniels, 1988: "A Method for Obtaining and Improved Approximation for the Temperature/Moisture Retrieval Problem", Proceeding, AMS Third Conference on Satellite Meteorology and Oceanography, Jan. 31 - Feb. 5, 1988, Anaheim, Calif. pp. 20-23.