2017 GSI Community Tutorial
July 11-14, 2017. College Park, MD

GSI Fundamentals (5): PrepBUFR and BUFR format

Guoqing Ge,

Kathryn Newman, Ming Hu & Ruifang Li

Developmental Testbed Center (DTC)

Developmental Testbed Center

Topics covered

- NCEP observational data
  - Observa
  - Operation BUFR/PrepBUFR types and data servers

Kathryn Newman's talk on 2015 GSI tutorial
http://www.dtcenter.org/com-GSI/users/docs/presentations/2015_tutorial/20150813_L12_Newman_Obs_processing.pdf

- BUFR/PrepBUFR format
  - BUFR/PrepBUFR file structure
  - Encode, decode and append a simple BUFR file
- NCEP DX BUFR table
  - DX BUFR table structure and examples
  - Understand a mnemonic by referring to the BUFR table

What is BUFR?

- **BUFR** is a binary data format maintained by WMO.
  Binary **U**niversal **F**orm for the **R**epresentation of meteorological data

- Lots of operational observation data are stored in BUFR format

- The power of BUFR - **self-descriptive**:
  A BUFR message(or record) not only contains obs data also contains a complete description of what those data are (e.g. lat,lon,units,…)

- Advantages of BUFR:
  - Flexibility
  - Compact Data Storage
  - WMO Standard

NCEP BUFR/PrepBUFR

- NCEP developed BUFRLIB to interface with BUFR files
- Features of NCEP BUFR/PrepBUFR:
  > BUFR table encoded into "dictionary" messages at top of file.
  >Files are self defined, no need for external BUFR table to decode
- PrepBUFR is "prepared" or "QC'ed" BUFR data.
  Obs data has been added QC flags and corresponding obs errors.
- PrepBUFR file is still a BUFR file

Operational BUFR/PrepBUFR files

- File name convention
  - gdas1.t00z.prepbufr.nr
  - gfs.t00z.gpsro.tm00.bufr_d
  - ndas.t18z.lbamub.tm03.bufr_d
  - nam.t00z.aircar.tm00.bufr_d.nr

MODEL.t*cyc*z.*TYPE*.tm*MM*.bufr_d

MODEL: operation system
cyc: cycling time
TYPE: data type
MM: 00 for all types except for *NDAS*, in which MM indicate the NDAS catch up cycle analysis time:

=0 analysis time = cyc
>0 analysis time = cyc - MM
For example:
ndas.t18z.lbamub.tm03.bufr_d
has analysis time=18z-03z=15z

nr: non-restricted data
bufr_d: bufr format

See BUFR User's Guide Chapter 5.2

- Data coverage and cut off time
  - GDAS (Global Data Assimilation System):
    Covers global, latest 6 hours data
  - GFS (Global Forecast System):
    Covers global, 2:45 hours data
  - NDAS (NAM Data Assimilation System):
    Covers North America, longer cut off time than NAM
  - NAM (North American Model):
    Covers North America, shorter cut off time comparing to others

Operational BUFR/PrepBUFR data

Data listed in BUFR User's Guide Chapter 5.3

- NCEP NOMADS Site:
  - BUFR/PrepBufr for GDAS (Global) - 1 month buffer:
    - http://nomads.ncep.noaa.gov/pub/data/nccf/com/gfs/prod/
  - BUFR/PrepBufr for NDAS (North America) - 1 month buffer:
    - http://nomads.ncep.noaa.gov/pub/data/nccf/com/nam/prod/

- NCDC NOMADS Site:
  - BUFR/PrepBufr for GDAS (Global) - archive starting May 2007:
    - http://nomads.ncdc.noaa.gov/data/gdas/

- NCAR/CISL Research Data Archive (RDA) Site:
  - **DS337.0**: NCEP ADP Global Upper Air and Surface Observations (PrepBUFR and NetCDF PB2NC Output) - archive starting May 1997:
    - http://dss.ucar.edu/datasets/ds337.0/

BUFR/PrepBUFR Format

- BUFR/PrepBUFR file structure
- Encode, decode and append a simple BUFR file

Examples/tools are provided in GSI util/bufr_tools;
Detailed information is in BUFR User's Guide.

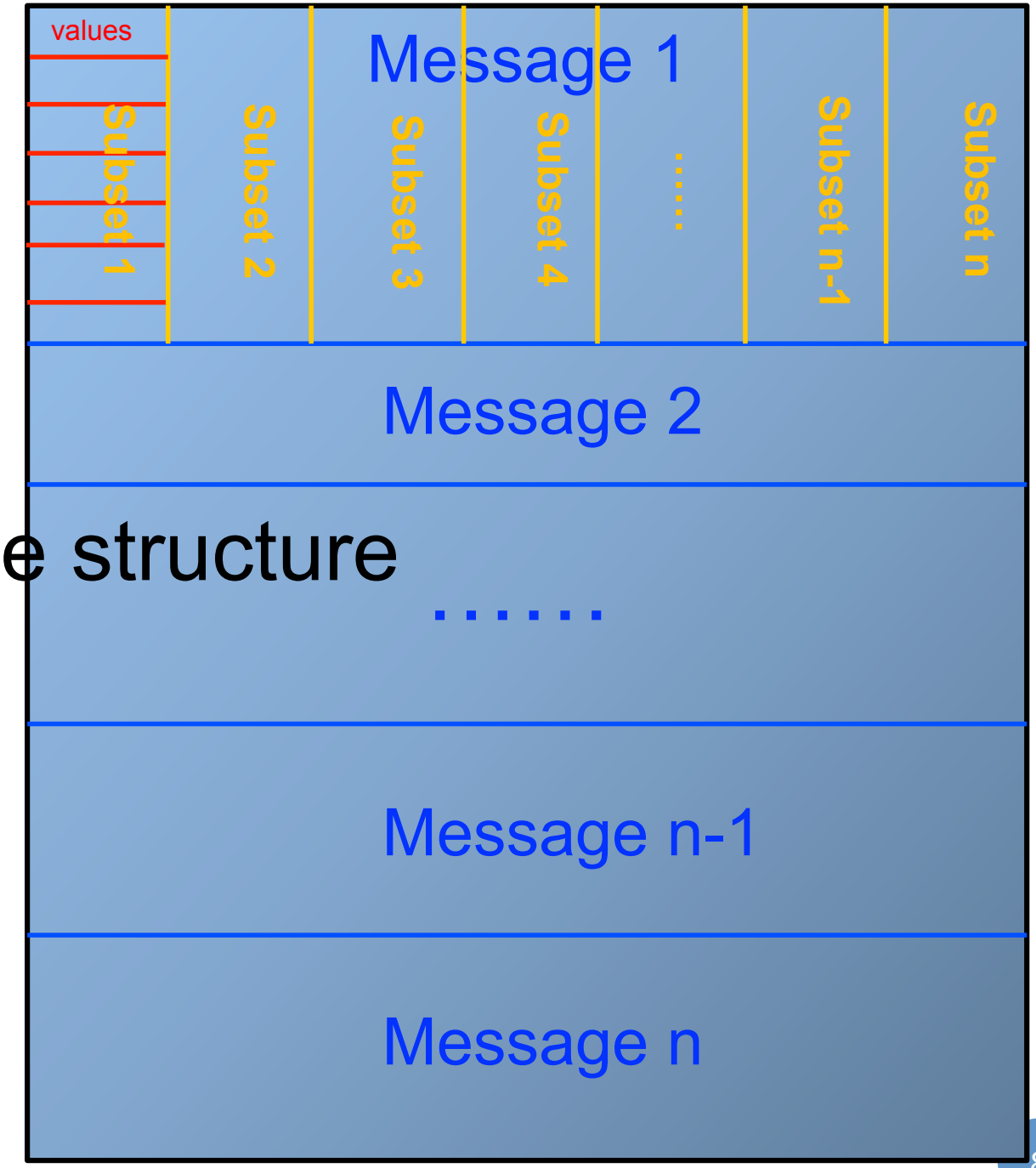
http://www.dtcenter.org/com-GSI/BUFR/docs/users_guide/BUFR_PrepBUFR_User_Guide_v1.pdf

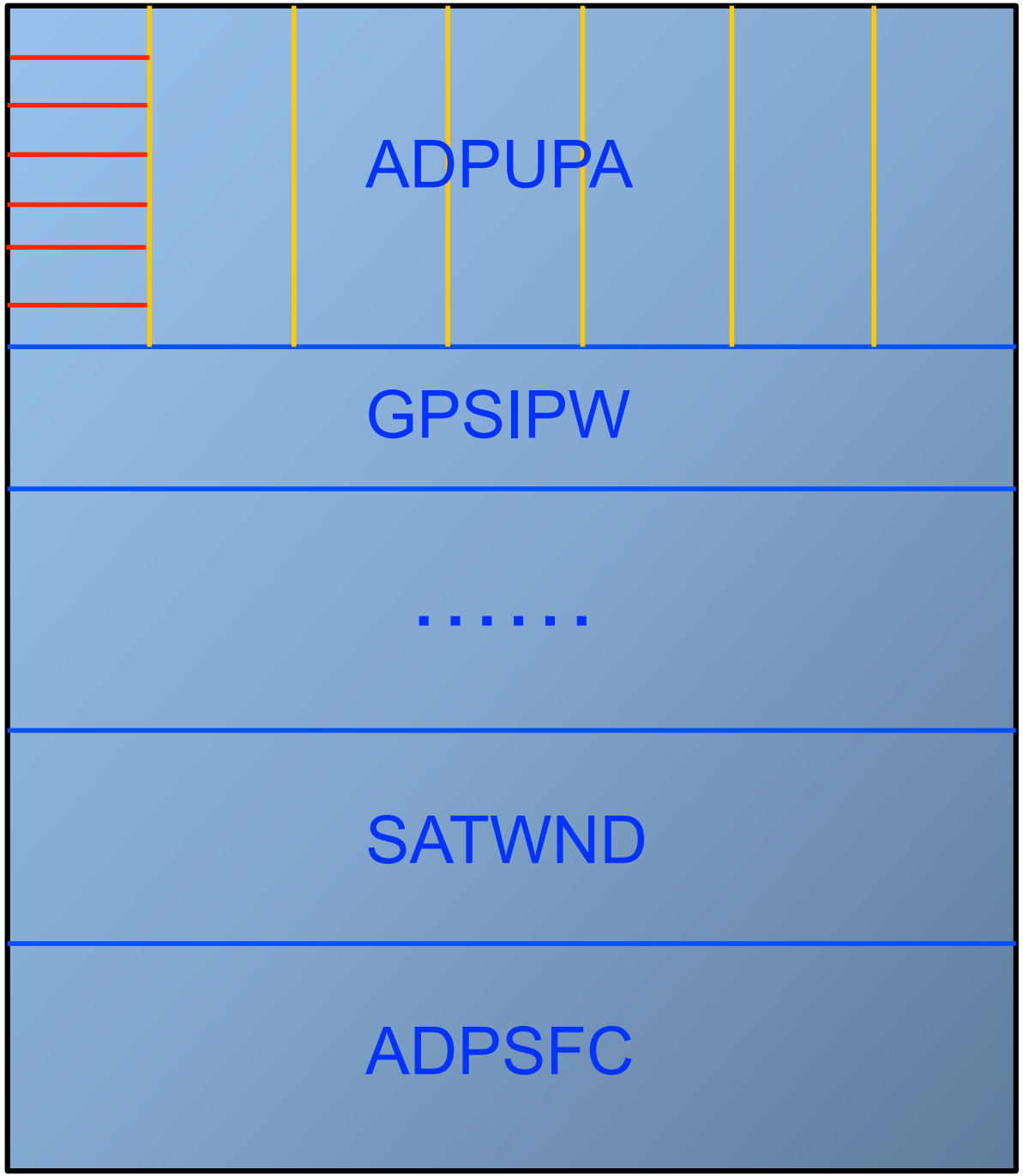
BUFR file structure

A BUFR files is a collection of **messages**

A message is a collection of same type **subsets**

A subset is a collection of data values (time,lat,lon,T,P,U,V,Q, OE, QM,-repeat for K levels-) a piece of obs




ADPUPA
GPSIPW
.....
SATWND
ADPSFC

Encode/Decode a BUFR file

- **Encode**:
  - **Write** observations into a new BUFR file.
- **Decode**:
  - **Read** observations from a BUFR file.
- **Append**:
  - **Add** observations to the end of an existing BUFR file.

NCEP BUFRLIB:

http://www.nco.ncep.noaa.gov/sib/decoders/BUFRLIB/

(10-20 out of 250 subs/functions are often used)

Encode BUFR file

- Write the observation into a BUFR file

```fortran
program bufr_encode_sample
!
!  example of writing one value into a bufr file
!
 implicit none

 character(80):: hdstr='XOB YOB DHR'
 character(80):: obstr='TOB'
 real(8) :: hdr(3),obs(1,1)

 character(8) msgtyp
 integer :: unit_out=10,unit_table=20
 integer :: idate,iret

! set data values
 hdr(1)=75.;hdr(2)=30.;hdr(3)=-0.1
 obs(1,1)=287.15
 idate=2008120100  ! YYYYMMDDHH
 msgtyp='ADPUPA'   ! upper-air reports

! encode
 open(unit_table,file='prepobs_prep.bufrtable')
 open(unit_out,file='sample.bufr',action='write',form='unformatte
 call datelen(10)
 call openbf(unit_out,'OUT',unit_table)
    call openmb(unit_out,msgtyp,idate)
       call ufbint(unit_out,hdr,3,1,iret,hdstr)
       call ufbint(unit_out,obs,1,1,iret,obstr)
       call writsb(unit_out)
    call closmg(unit_out)
 call closbf(unit_out)

end program
```

Define variables and assign obs values

Encode data using BUFRLIB subroutines and functions:

openbf, closbf,
openmb, closmg,
ufbint, writsb,
datelen

These subroutines/functions are very often used to read/write BUFR file. Understanding usage of them will be very helpful in users's own application.

```
program bufr_encode_sample
+--   3 lines: ----------------------------------------------------------
+-- 16 lines: implicit none------------------------------------------------
! encode
 open(unit_table,file='prepobs_prep.bufrtable')
 open(unit_out,file='sample.bufr',action='write',form='unforma
 call datelen(10)
 call openbf(unit_out,'OUT',unit_table)
    call openmb(unit_out,msgtyp,idate)
      call ufbint(unit_out,hdr,3,1,iret,hdstr)
      call ufbint(unit_out,obs,1,1,iret,obstr)
       call writsb(unit_out)
    call closmg(unit_out)
 call closbf(unit_out)

end program
```

Open the BUFR table

Open a binary file

DX BUFR table

```
program bufr_encode_sample   (34 Lines)
!
!  example of writing one value into a bufr file
!
 implicit none

 character(80):: hdstr='XOB YOB DHR'
 character(80):: obstr='TOB'
 real(8) :: hdr(3),obs(1,1)

 character(8) msgtype
 integer :: unit_out=10,unit_table=20
 integer :: idate,iret

! set data values
 hdr(1)=75.;hdr(2)=30.;hdr(3)=-0.1
 obs(1,1)=17.15
 idate=2008120100  ! YYYYMMDDHH
 msgtype='ADPUPA'   ! upper-air reports

! encode
 open(unit_table,file='prepobs_prep.bufrtable')
 open(unit_out,file='sample.bufr',action='write' ,form='unform
 call datelen(10)
 call openbf(unit_out,'OUT',unit_table)
   call openmb(unit_out, msgtype,idate)
      call ufbint(unit_out,hdr,3,1,iret,hdstr)
      call ufbint(unit_out,obs,1,1,iret,obstr)
      call writsb(unit_out)
   call closmg(unit_out)
 call closbf(unit_out)

end program
```

BUFR table is in released GSI version ./util/bufr_tools directory.

BUFR table defines the content for each message type. It is embedded in the file as the first few BUFR messages.

Understanding BUFR table will be very helpful in users own application.

```
program bufr_encode_sample   (34 Lines)
!
!  example of writing one value into a bufr file
!
 implicit none

 character(80):: hdstr='XOB YOB DHR'
 character(80):: obstr='TOB'
 real(8) :: hdr(3),obs(1,1)

 character(8) msgtype
 integer :: unit_out=10,unit_table=20
 integer :: idate,iret

! set data values
 hdr(1)=75.;hdr(2)=30.;hdr(3)=-0.1
 obs(1,1)=17.15
 idate=2008120100  ! YYYYMMDDHH
 msgtype='ADPUPA'  ! upper-air reports
! encode
 open(unit_table,file='prepobs_prep.bufrtable')
 open(unit_out,file='sample.bufr',action='write',form='unformatted')
 call datelen(10)
 call openbf(unit_out,'OUT',unit_table)
   call openmb(unit_out, msgtype,idate)
      call ufbint(unit_out,hdr,3,1,iret,hdstr)
      call ufbint(unit_out,obs,1,1,iret,obstr)
      call writsb(unit_out)
   call closmg(unit_out)
 call closbf(unit_out)

end program
```

Define BUFR table mnemonics
Assign data values

A mnemonic is a descriptive, alphanumeric name for an data value.

XOB: Longtitude (006240)
YOB: Latitude
DHR: obs time – cycle time
TOB: temperature
ADPUPA: UPPER-AIR (RAOB, PIBAL, RECCO, DROPS) REPORTS

Setup data

Data written to subset.

Match mnemonics and data

hdstr= XOB	YOB	DHR	obstr= TOB	
hdr = (1)75.	(2)30.	(3)-0.1	obs(1)= 17.51	

BUFR file

```
program bufr_encode_sample   (34 Lines)
!
!  example of writing one value into a bufr file
!
 implicit none

 character(80):: hdstr='XOB YOB DHR'
 character(80):: obstr='TOB'
 real(8) :: hdr(3),obs(1,1)

 character(8) msgtype
 integer :: unit_out=10,unit_table=20
 integer :: idate,iret

! set data values
 hdr(1)=75.;hdr(2)=30.;hdr(3)=-0.1
 obs(1,1)=17.15
 idate=2008120100  ! YYYYMMDDHH
 msgtype='ADPUPA'   ! upper-air reports
! encode
 open(unit_table,file='prepobs_prep.bufrtable')
 open(unit_out,file='sample.bufr',action='write' ,form='unformatted')
 call datelen(10)
 call openbf(unit_out,'OUT',unit_table)
   call openmb(unit_out, msgtype,idate)
     call ufbint(unit_out,hdr,3,1,iret,hdstr)
     call ufbint(unit_out,obs,1,1,iret,obstr)
     call writsb(unit_out)
   call closmg(unit_out)
 call closbf(unit_out)

end program
```

Fortran 'open' command to open an unformatted binary file for write.

***OPENBF* (LUBFR, CIO, LUNDX)**

Purpose: Identifies to the BUFRLIB a BUFR file of logical unit *LUBFR.*

***CLOSBF* (LUBFR)**

Purpose: close the connection between logical unit *LUBFR* and the BUFRLIB.

Set cycle date

```
program bufr_encode_sample   (34 Lines)
!
!  example of writing one value into a bufr file
 implicit none

 character(80):: hdstr='XOB YOB DHR'
 character(80):: obstr='TOB'
 real(8) :: hdr(3),obs(1,1)

 character(8) msgtype
 integer :: unit_out=10,unit_table=20
 integer :: idate,iret

! set data values
 hdr(1)=75.;hdr(2)=30.;hdr(3)=-0.1
 obs(1,1)=17.15
 idate=2008120100  ! YYYYMMDDHH
 msgtype='ADPUPA'   ! upper-air reports

! encode
 open(unit_table,file='prepobs_prep.bufrtable')
 open(unit_out,file='sample.bufr',action='write',form='unformatted')

 call datelen(10)
 call openbf(unit_out,'OUT',unit_table)
   call openmb(unit_out, msgtype,idate)
      call ufbint(unit_out,hdr,3,1,iret,hdstr)
      call ufbint(unit_out,obs,1,1,iret,obstr)
      call writsb(unit_out)
   call closmg(unit_out)
 call closbf(unit_out)

end program
```

***DATELEN* (LEN)**

Purpose: specify the format *IDATE.*

Input arguments:

LEN INTEGER

Length of Section 1 date-time values

8 = YYMMDDHH (2-digit year)

10 = YYYYMMDDHH (4-digit year)

Message

```
program bufr_encode_sample   (34 Lines)
!
!  example of writing one value into a bufr file
!
 implicit none

 character(80):: hdstr='XOB YOB DHR'
 character(80):: obstr='TOB'
 real(8) :: hdr(3),obs(1,1)

 character(8) msgtype
 integer :: unit_out=10,unit_table=20
 integer :: idate,iret

! set data values
 hdr(1)=75.;hdr(2)=30.;hdr(3)=-0.1
 obs(1,1)=17.15
 idate=2008120100  ! YYYYMMDDHH
 msgtype='ADPUPA'   ! upper-air reports

! encode
 open(unit_table,file='prepobs_prep.bufrtable')
 open(unit_out,file='sample.bufr',action='write',form='unformatted')
 call datelen(10)
 call openbf(unit_out,'OUT',unit_table)
  call openmb(unit_out, msgtype,idate)
    call ufbint(unit_out,hdr,3,1,iret,hdstr)
   call ufbint(unit_out,obs,1,1,iret,obstr)
   call writsb(unit_out)
  call closmg(unit_out)
 call closbf(unit_out)

end program
```

***OPENMB* (LUBFR, CSUBSET, IDATE)**
Purpose: Open and initialize, within internal arrays, a new BUFR message for eventual output to *LUBFR*, using *CSUBSET* as message type, *IDATE* as date.

***CLOSMG* (LUBFR)**
Purpose: Close existing internal BUFR message (if any) and write it to output.

Data subsets

```
program bufr_encode_sample   (34 Lines)
!
!  example of writing one value into a bufr file
!
 implicit none

 character(80):: hdstr='XOB YOB DHR'
 character(80):: obstr='TOB'
 real(8) :: hdr(3),obs(1,1)

 character(8) msgtype
 integer :: unit_out=10,unit_table=20
 integer :: idate,iret

! set data values
 hdr(1)=75.;hdr(2)=30.;hdr(3)=-0.1
 obs(1,1)=17.15
 idate=2008120100  ! YYYYMMDDHH
 msgtype='ADPUPA'   ! upper-air reports

! encode
 open(unit_table,file='prepobs_prep.bufrtable')
 open(unit_out,file='sample.bufr',action='write',form='unformatted')
 call datelen(10)
 call openbf(unit_out,'OUT',unit_table)
   call openmb(unit_out, msgtype,idate)
   call ufbint(unit_out,hdr,3,1,iret,hdstr)
   call ufbint(unit_out,obs,1,1,iret,obstr)
   call writsb(unit_out)
   call closmg(unit_out)
 call closbf(unit_out)

end program
```

hdstr= XOB	YOB	DHR
hdr = (1)75.	(2)30.	(3)-0.1

***UFBINT* (LUBFR, R8ARR, MXMN, MXLV, iret, CMNSTR)**

Purpose: writes or reads specified values to or from the current BUFR data subset within the internal arrays.

obstr= TOB
obs(1)= 17.51

Data subsets

```
program bufr_encode_sample   (34 Lines)
!  example of writing one value into a bufr file
 implicit none

 character(80):: hdstr='XOB YOB DHR'
 character(80):: obstr='TOB'
 real(8) :: hdr(3),obs(1,1)

 character(8) msgtype
 integer :: unit_out=10,unit_table=20
 integer :: idate,iret

! set data values
 hdr(1)=75.;hdr(2)=30.;hdr(3)=-0.1
 obs(1,1)=17.15
 idate=2008120100  ! YYYYMMDDHH
 msgtype='ADPUPA'   ! upper-air reports

! encode
 open(unit_table,file='prepobs_prep.bufrtable')
 open(unit_out,file='sample.bufr',action='write',form='unformatted')
 call datelen(10)
 call openbf(unit_out,'OUT',unit_table)
   call openmb(unit_out, msgtype,idate)
      call ufbint(unit_out,hdr,3,1,iret,hdstr)
      call ufbint(unit_out,obs,1,1,iret,obstr)
      call writsb(unit_out)
   call closmg(unit_out)
 call closbf(unit_out)

end program
```

***WRITSB* (LUBFR)**

Purpose: Indicates to BUFRLIB that the subset is ready to be encoded into the current message for the BUFR file.

Mnemonics and data array

```
program bufr_encode_sample   (34 Lines)
!
!  example of writing one value into a bufr file
!
 implicit none

 character(80):: hdstr='XOB YOB DHR'
 character(80):: obstr='TOB'
 real(8) :: hdr(3),obs(1,1)

 character(8) msgtype
 integer :: unit_out=10,unit_table=20
 integer :: idate,iret

! set data values
 hdr(1)=75.;hdr(2)=30.;hdr(3)=-0.1
 obs(1,1)=17.15
 idate=2008120100  ! YYYYMMDDHH
 msgtype='ADPUPA'   ! upper-air reports

! encode
 open(unit_table,file='prepobs_prep.bufrtable')
 open(unit_out,file='sample.bufr',action='write',form='unformatted')
 call datelen(10)
 call openbf(unit_out,'OUT',unit_table)
   call openmb(unit_out, msgtype,idate)

   call ufbint(unit_out,hdr,3,1,iret,hdstr)
   call ufbint(unit_out,obs,1,1,iret,obstr)
    call writsb(unit_out)
   call closmg(unit_out)
 call closbf(unit_out)

end program
```

hdstr= XOB YOB DHR obstr= TOB

hdr = (1)75. (2)30. (3)-0.1 obs(1)= 17.51

Write to bufr file sample.bufr

Message type: ADPUPA

Section 3	Section 4
XOB YOB DHR TOB	75. 30. -0.1 17.51

Decode BUFR file

- Read the observation out of BUFR file

```fortran
program bufr_decode_sample
!
! example of reading observations from bufr
!
 implicit none

 character(80):: hdstr='XOB YOB DHR'
 character(80):: obstr='TOB'
 real(8) :: hdr(3),obs(1,10)

 integer :: ireadmg,ireadsb
 character(8) msgtyp
 integer :: unit_in=10
 integer :: idate,iret,num_message,num_subset

! decode
 open(unit_in,file='sample.bufr',action='read',form='unformatted')
 call openbf(unit_in,'IN',unit_in)
    call datelen(10)
    num_message=0
    msg_report: do while (ireadmg(unit_in,msgtyp,idate) == 0)
       num_message=num_message+1
       num_subset = 0
       write(*,'(I10,I4,a10)') idate,num_message,msgtyp
       sb_report: do while (ireadsb(unit_in) == 0)
          num_subset = num_subset+1
          call ufbint(unit_in,hdr,3,1 ,iret,hdstr)
          call ufbint(unit_in,obs,1,10,iret,obstr)
          write(*,'(2I5,4f8.1)') num_subset,iret,hdr,obs(1,1)
       enddo sb_report
    enddo msg_report
 call closbf(unit_in)

end program
```

Encode

```
program bufr_encode_sample
!
!  example of writing one value into a bufr
!
 implicit none

 character(80):: hdstr='XOB YOB DHR'
 character(80):: obstr='TOB'
 real(8) :: hdr(3),obs(1,1)

 character(8) msgtype
 integer :: unit_out=10,unit_table=20
 integer :: idate,iret

! set data values
 hdr(1)=75.;hdr(2)=30.;hdr(3)=-0.1
 obs(1,1)=17.15
 idate=2008120100  ! YYYYMMDDHH
 msgtype='ADPUPA'   ! upper-air reports

! encode
 open(unit_table,file='prepobs_prep.bufrtable')
 open(unit_out,file='sample.bufr',action='write' &
            ,form='unformatted')
 call datelen(10)
 call openbf(unit_out,'OUT',unit_table)
  call openmb(unit_out,msgtype,idate)
    call ufbint(unit_out,hdr,3,1,iret,hdstr)
    call ufbint(unit_out,obs,1,1,iret,obstr)
      call writsb(unit_out)
  call closmg(unit_out)
 call closbf(unit_out)

end program
```

Decode

```
rogram bufr_decode_sample
!
! example of reading observations from bufr
!
 implicit none

 character(80):: hdstr='XOB YOB DHR'
 character(80):: obstr='TOB'
 real(8) :: hdr(3),obs(1,10)

 integer :: ireadmg,ireadsb
 character(8) msgtype
 integer :: unit_in=10
 integer :: idate,iret,num_message,num_subset

! decode
 open(unit_in,file='sample.bufr',action='read',form='unformatted')
 call openbf(unit_in,'IN',unit_in)
 call datelen(10)
  num_message=0
 msg_report: do while (ireadmg(unit_in,msgtype,idate) == 0)
    num_message=num_message+1
    num_subset = 0
    write(*,'(I10,I4,a10)') idate,num_message,msgtype
    sb_report: do while (ireadsb(unit_in) == 0)
      num_subset = num_subset+1
      call ufbint(unit_in,hdr,3,1 ,iret,hdstr)
      call ufbint(unit_in,obs,1,10,iret,obstr)
      write(*,'(2I5,4f8.1)') num_subset,iret,hdr,obs(1,1)
    enddo sb_report
  enddo msg_report
 call closbf(unit_in)

end program
```

Read message, data subset, and data values

```
program bufr_decode_sample
!
! example of reading observations from bufr
!
 implicit none

 character(80):: hdstr='XOB YOB DHR'
 character(80):: obstr='TOB'
 real(8) :: hdr(3),obs(1,10)

 integer :: ireadmg,ireadsb
 character(8) msgtype
 integer :: unit_in=10
 integer :: idate,iret,num_message,num_subset

! decode
 open(unit_in,file='sample.bufr',action='read',form='unformatted')
 call openbf(unit_in,'IN',unit_in)
 call datelen(10)
   num_message=0
msg_report: do while (ireadmg(unit_in,msgtype,idate) == 0)
    num_message=num_message+1
    num_subset = 0
    write(*,'(I10,I4,a10)') idate,num_message,msgtype
   sb_report: do while (ireadsb(unit_in) == 0)
     num_subset = num_subset+1
     call ufbint(unit_in,hdr,3,1 ,iret,hdstr)
     call ufbint(unit_in,obs,1,10,iret,obstr)
     write(*,'(2I5,4f8.1)') num_subset,iret,hdr,obs(1,1)
   enddo sb_report
  enddo msg_report
 call closbf(unit_in)

end program
```

IRET = *IREADMG* (LUBFR, CSUBSET, IDATE)
Purpose: reads the next BUFR message from the given BUFR file pointed to by *LUBFR*.

IRET = *IREADSB* (LUBFR)
Purpose: reads a subset from that internal message.

***UFBINT* (LUBFR, R8ARR, MXMN, MXLV, iret, CMNSTR)**
Purpose: writes or reads specified values to or from the current BUFR data subset within the internal arrays.

Mnemonics and data array

```
program bufr_decode_sample
!
! example of reading observations from bufr
!
 implicit none

 character(80):: hdstr='XOB YOB DHR'
 character(80):: obstr='TOB'
 real(8) :: hdr(3),obs(1,10)

 integer :: ireadmg,ireadsb
 character(8) msgtype
 integer :: unit_in=10
 integer :: idate,iret,num_message,num_subset

! decode
 open(unit_in,file='sample.bufr',action='read',form='unformatted')
 call openbf(unit_in,'IN',unit_in)
 call datelen(10)
   num_message=0
   msg_report: do while (ireadmg(unit_in,msgtype,idate) == 0)
     num_message=num_message+1
     num_subset = 0
     write(*,'(I10,I4,a10)') idate,num_message,msgtype
     sb_report: do while (ireadsb(unit_in) == 0)
       num_subset = num_subset+1
        call ufbint(unit_in,hdr,3,1 ,iret,hdstr)
        call ufbint(unit_in,obs,1,10,iret,obstr)
       write(*,'(2I5,4f8.1)') num_subset,iret,hdr,obs(1,1)
     enddo sb_report
   enddo msg_report
 call closbf(unit_in)

end program
```

hdstr= XOB YOB DHR obstr= TOB

Message type: ADPUPA

Section 3	Section 4
XOB YOB DHR TOB	75. 30. -0.1 17.51

read from sample.bufr

hdr = (1)75. (2)30. (3)-0.1 obs(1)= 17.51

Append to BUFR file

- Append the observation to existing BUFR

Encode

```
program bufr_encode_sample
!
!  example of writing one value into a bufr file
!
 implicit none

 character(80):: hdstr='XOB YOB DHR'
 character(80):: obstr='TOB'
 real(8) :: hdr(3),obs(1,1)

 character(8) msgtype
 integer :: unit_out=10,unit_table=20
 integer :: idate,iret

! set data values
 hdr(1)=75.;hdr(2)=30.;hdr(3)=-0.1
 obs(1,1)=17.15
 idate=2008120100  ! YYYYMMDDHH
 msgtype='ADPUPA'   ! upper-air reports

! encode
 open(unit_table,file='prepobs_prep.bufrtable')
 open(unit_out,file='sample.bufr',action='write' &
            ,form='unformatted')
 call datelen(10)
 call openbf(unit_out,'OUT',unit_table)
   call openmb(unit_out, msgtype,idate)
      call ufbint(unit_out,hdr,3,1,iret,hdstr)
      call ufbint(unit_out,obs,1,1,iret,obstr)
      call writsb(unit_out)
   call closmg(unit_out)
 call closbf(unit_out)
end program
```

Append

```
program bufr_append_sample

!sample of appending one observation into bufr file
 implicit none
 character(80):: hdstr='XOB YOB DHR'
 character(80):: obstr='TOB'
 real(8) :: hdr(3),obs(1,1)

 character(8) msgtype
 integer :: unit_out=10,unit_table=20
 integer :: idate,iret

! set data values
 hdr(1)=85.0;hdr(2)=50.0;hdr(3)=0.2
 obs(1,1)=15.0
 idate=2008120101  ! YYYYMMDDHH
 msgtype='ADPSFC'   ! surface land reports

! get bufr table from existing bufr file
 open(unit_table,file='prepobs_prep_app.bufrtable')
 open(unit_out,file='sample.bufr',status='old',form='unformatted')
 call openbf(unit_out,'IN',unit_out)
 call dxdump(unit_out,unit_table)
 call closbf(unit_out)

! append
 open(unit_out,file='sample.bufr',status='old',form='unformatted')
 call datelen(10)
 call openbf(unit_out,'APN',unit_table)
   call openmb(unit_out, msgtype,idate)
      call ufbint(unit_out,hdr,3,1,iret,hdstr)
      call ufbint(unit_out,obs,1,1,iret,obstr)
      call writsb(unit_out)
   call closmg(unit_out)
 call closbf(unit_out)
end program
```

Extract BUFR table from existing BUFR file

```
program bufr_append_sample
!
! sample of appending one observation into bufr file
 implicit none
 character(80):: hdstr='XOB YOB DHR'
 character(80):: obstr='TOB'
 real(8) :: hdr(3),obs(1,1)

 character(8) msgtype
 integer :: unit_out=10,unit_table=20
 integer :: idate,iret

! set data values
 hdr(1)=85.0;hdr(2)=50.0;hdr(3)=0.2
 obs(1,1)=15.0
 idate=2008120101  ! YYYYMMDDHH
 msgtype='ADPSFC'   ! surface land reports

! get bufr table from existing bufr file
 open(unit_table,file='prepobs_prep_app.bufrtable')
 open(unit_out,file='sample.bufr',status='old',form='unformatted')
 call openbf(unit_out,'IN',unit_out)
 call dxdump(unit_out,unit_table)
 call closbf(unit_out)

! append
 open(unit_out,file='sample.bufr',status='old',form='unformatted')
 call datelen(10)
 call openbf(unit_out,'APN',unit_table)
   call openmb(unit_out, msgtype,idate)
      call ufbint(unit_out,hdr,3,1,iret,hdstr)
      call ufbint(unit_out,obs,1,1,iret,obstr)
      call writsb(unit_out)
   call closmg(unit_out)
 call closbf(unit_out)
end program
```

Appending requires the report structure (BUFR table) of the new data subset fits the report structure in the existing file. So we use the following subroutine to retrieve BUFR table from the existing BUFR file:

***DXDUMP* (LUBFR, LDXOT)**

Purpose: reads the embedded tables information in the BUFR file and write it out to the ASCII format file.

Append

```
program bufr_append_sample
! sample of appending one observation into bufr file
 implicit none
 character(80):: hdstr='XOB YOB DHR'
 character(80):: obstr='TOB'
 real(8) :: hdr(3),obs(1,1)

 character(8) msgtype
 integer :: unit_out=10,unit_table=20
 integer :: idate,iret

! set data values
 hdr(1)=85.0;hdr(2)=50.0;hdr(3)=0.2
 obs(1,1)=15.0
 idate=2008120101  ! YYYYMMDDHH
 msgtype='ADPSFC'   ! surface land reports

! get bufr table from existing bufr file
 open(unit_table,file='prepobs_prep_app.bufrtable')
 open(unit_out,file='sample.bufr',status='old',form='unformatted')
 call openbf(unit_out,'IN',unit_out)
 call dxdump(unit_out,unit_table)
 call closbf(unit_out)

! append
 open(unit_out,file='sample.bufr',status='old',form='unformatted')
 call datelen(10)
 call openbf(unit_out,'APN',unit_table)
 call openmb(unit_out, msgtype,idate)
    call ufbint(unit_out,hdr,3,1,iret,hdstr)
    call ufbint(unit_out,obs,1,1,iret,obstr)
    call writsb(unit_out)
 call closmg(unit_out)
 call closbf(unit_out)
end program
```

Write the new message type and data subset to the existing BUFR file.

Basic Practice Cases - 5

./bufr_encode_sample.exe

This generates a new bufr file *sample.bufr*

./bufr_decode_sample.exe

This reads one observation from *sample.bufr*, and write result on screen:

```
2008120100    1  ADPUPA
    1     1     75.0     30.0     -0.1    287.1
```

./bufr_append_sample.exe

Now, append a new observation to *sample.bufr*.

./decode_sample.exe

Read *sample.bufr* and show two observations in it:

```
2008120100    1  ADPUPA
    1     1     75.0     30.0     -0.1    287.1
2008120101    2  ADPSFC
    1     1     85.0     50.0      0.2    300.0
```

Basic Practice Cases – 5 (cont'd)

```
[geguo@yslogin2 bufr_tools]$ ls prepbufr*.exe
prepbufr_append_retrieve.exe  prepbufr_encode_surface.exe
prepbufr_append_surface.exe   prepbufr_encode_upperair.exe
prepbufr_append_upperair.exe  prepbufr_inventory.exe
prepbufr_decode_all.exe
```

- These examples have the same structure and call the same BUFRLIB subroutines/functions as those three simple examples
- The only difference is the mnemonic lists used in these examples are much longer

MNEMONIC in prepbufr_decode_all.f90

In GSI, *read_prepbufr.f90* reads PrepBUFR file. The following mnemonic lists come from *read_prepbufr.f90* and are used in PrepBUFR sample code.

```
integer, parameter :: mxmn=35, mxlv=250
character(80):: hdstr='SID XOB YOB DHR TYP ELV SAID T29'
character(80):: obstr='POB QOB TOB ZOB UOB VOB PWO CAT PRSS'
character(80):: qcstr='PQM QQM TQM ZQM WQM NUL PWQ     '
character(80):: oestr='POE QOE TOE NUL WOE NUL PWE     '
real(8) :: hdr(mxmn),obs(mxmn,mxlv),qcf(mxmn,mxlv),oer(mxmn,mxlv)
```

```
call ufbint(unit_in,hdr,mxmn,1   ,iret,hdstr)
call ufbint(unit_in,obs,mxmn,mxlv,iret,obstr)
call ufbint(unit_in,oer,mxmn,mxlv,iret,oestr)
call ufbint(unit_in,qcf,mxmn,mxlv,iret,qcstr)
```

NCEP DX BUFR table

- DX BUFR table structure and examples
- Understand a mnemonic by referring to the DX BUFR table

Detailed information is in BUFR user guide.

http://www.dtcenter.org/com-GSI/BUFR/docs/users_guide/BUFR_PrepBUFR_User_Guide_v1.pdf

DX BUFR table

- BUFR is a **table-driven** code form

  → the meaning of data elements is determined by referring to a set of tables

BUFR Table A - Data category

BUFR Table B - Classification of elements and tables

BUFR Table C - Data description operators

BUFR Table D - List of common sequences

http://www.wmo.int/pages/prog/www/WMOCodes/WMO306_vI2/LatestVERSION/LatestVERSION.html

DX BUFR table

- In NCEP BUFR files, the BUFR tables are embedded at the top of the files.

- Excellent reference for NCEP BUFR Tables: http://www.nco.ncep.noaa.gov/sib/decoders/BUFRLIB/toc/dfbftab/

DX BUFR table: structure example

USER DEFINITIONS FOR TABLE-A TABLE-B TABLE D

Section 1

MNEMONIC	NUMBER	DESCRIPTION
ADPUPA	A48102	UPPER-AIR (RAOB, PIBAL, RECCO, DROPS) REPORTS
ADPSFC	A48109	SURFACE LAND (SYNOPTIC, METAR) REPORTS
HEADR	348001	REPORT HEADER SEQUENCE
PRSLEVEL	348002	PRESSURE LEVEL SEQUENCE (ALL TYPES EXCEPT GOESND)
T___INFO	348143	TEMPERATURE INFORMATION
SID	001194	STATION IDENTIFICATION
DHR	004215	OBSERVATION TIME MINUS CYCLE TIME
YOB	005002	LATITUDE
XOB	006240	LONGITUDE

Section 2

MNEMONIC	SEQUENCE
ADPUPA	HEADR SIRC {PRSLEVEL} <SST_INFO> <PREWXSEQ> {CLOUDSEQ}
ADPUPA	<CLOU2SEQ> <SWINDSEQ> <AFIC_SEQ> <TURB3SEQ>
HEADR	SID XOB YOB DHR ELV TYP T29 TSB ITP SQN PROCN RPT
HEADR	TCOR <RSRD_SEQ>

Section 3

MNEMONIC	SCAL	REFERENCE	BIT	UNITS
SID	0	0	64	CCITT IA5
DHR	5	-2400000	23	HOURS
YOB	2	-9000	15	DEG N
XOB	2	-18000	16	DEG E
ELV	0	-1000	17	METER
TYP	0	0	10	CODE TABLE

DX BUFR table: structure example

USER DEFINITIONS FOR TABLE-A TABLE-B TABLE D

MNEMONIC	NUMBER	DESCRIPTION
ADPUPA	A48102	UPPER-AIR (RAOB, PIBAL, RECCO, DROPS) REPORTS
ADPSFC	A48109	SURFACE LAND (SYNOPTIC, METAR) REPORTS
HEADR	348001	REPORT HEADER SEQUENCE
PRSLEVEL	348002	PRESSURE LEVEL SEQUENCE (ALL TYPES EXCEPT GOESND)
T___INFO	348143	TEMPERATURE INFORMATION
SID	001194	STATION IDENTIFICATION
DHR	004215	OBSERVATION TIME MINUS CYCLE TIME
YOB	005002	LATITUDE
XOB	006240	LONGITUDE

Table A mnemonic

Table D mnemonic

Table B mnemonic

MNEMONIC	SEQUENCE
ADPUPA	HEADR SIRC {PRSLEVEL} <SST_INFO> <PREWXSEQ> {CLOUDSEQ}
ADPUPA	<CLOU2SEQ> <SWINDSEQ> <AFIC_SEQ> <TURB3SEQ>
HEADR	SID XOB YOB DHR ELV TYP T29 TSB ITP SQN PROCN RPT
HEADR	TCOR <RSRD_SEQ>

MNEMONIC	SCAL	REFERENCE	BIT	UNITS
SID	0	0	64	CCITT IA5
DHR	5	-2400000	23	HOURS
YOB	2	-9000	15	DEG N
XOB	2	-18000	16	DEG E
ELV	0	-1000	17	METER
TYP	0	0	10	CODE TABLE

DX BUFR table: structure example

USER DEFINITIONS FOR TABLE-A TABLE-B TABLE D

MNEMONIC	NUMBER	DESCRIPTION
ADPUPA	A48102	UPPER-AIR (RAOB, PIBAL, RECCO, DROPS) REPORTS
ADPSFC	A48109	SURFACE LAND (SYNOPTIC, METAR) REPORTS
HEADR	348001	REPORT HEADER SEQUENCE
PRSLEVEL	348002	PRESSURE LEVEL SEQUENCE (ALL TYPES EXCEPT GOESND)
T___INFO	348143	TEMPERATURE INFORMATION
SID	001194	STATION IDENTIFICATION
DHR	004215	OBSERVATION TIME MINUS CYCLE TIME
YOB	005002	LATITUDE
XOB	006240	LONGITUDE

MNEMONIC	SEQUENCE
ADPUPA	HEADR SIRC {PRSLEVEL} <SST_INFO> <PREWXSEQ> {CLOUDSEQ}
ADPUPA	<CLOU2SEQ> <SWINDSEQ> <AFIC_SEQ> <TURB3SEQ>
HEADR	SID XOB YOB DHR ELV TYP T29 TSB ITP SQN PROCN RPT
HEADR	TCOR <RSRD_SEQ>

Table A and Table D sequences

MNEMONIC	SCAL	REFERENCE	BIT	UNITS
SID	0	0	64	CCITT IA5
DHR	5	-2400000	23	HOURS
YOB	2	-9000	15	DEG N
XOB	2	-18000	16	DEG E
ELV	0	-1000	17	METER
TYP	0	0	10	CODE TABLE

DX BUFR table: structure example

USER DEFINITIONS FOR TABLE-A TABLE-B TABLE D

MNEMONIC	NUMBER	DESCRIPTION
ADPUPA	A48102	UPPER-AIR (RAOB, PIBAL, RECCO, DROPS) REPORTS
ADPSFC	A48109	SURFACE LAND (SYNOPTIC, METAR) REPORTS
HEADR	348001	REPORT HEADER SEQUENCE
PRSLEVEL	348002	PRESSURE LEVEL SEQUENCE (ALL TYPES EXCEPT GOESND)
T___INFO	348143	TEMPERATURE INFORMATION
SID	001194	STATION IDENTIFICATION
DHR	004215	OBSERVATION TIME MINUS CYCLE TIME
YOB	005002	LATITUDE
XOB	006240	LONGITUDE

MNEMONIC	SEQUENCE
ADPUPA	HEADR SIRC {PRSLEVEL} <SST_INFO> <PREWXSEQ> {CLOUDSEQ}
ADPUPA	<CLOU2SEQ> <SWINDSEQ> <AFIC_SEQ> <TURB3SEQ>
HEADR	SID XOB YOB DHR ELV TYP T29 TSB ITP SQN PROCN RPT
HEADR	TCOR <RSRD_SEQ>

MNEMONIC	SCAL	REFERENCE	BIT	UNITS
SID	0	0	64	CCITT IA5
DHR	5	-2400000	23	HOURS
YOB	2	-9000	15	DEG N
XOB	2	-18000	16	DEG E
ELV	0	-1000	17	METER
TYP	0	0	10	CODE TABLE

Define data elements

DX BUFR table example (GSI util/bufr_tools/prepobs_prep.bufrtable)

MNEMONIC	SCAL	REFERENCE	BIT	UNITS
SID	0	0	64	CCITT IA5
XOB	2	-18000	16	DEG E
YOB	2	-9000	15	DEG N
DHR	3	-24000	16	HOURS
ELV	0	-1000	17	METER
TYP	0	0	9	CODE TABL

Units:

CCITT IA5: character

CODE TABL: go to http://www.emc.ncep.noaa.gov/mmb/data_processing/prepbufr.doc/table_1.htm, search that Table B mnemonic, click CODE TABL link and see the code table.

DX BUFR table application examples: understand SID

```
character(80):: hdstr='SID XOB YOB DHR TYP ELV SAID T29'
```

```
.-------------------------------------------------------------------------------.
| ------------   USER DEFINITIONS FOR TABLE-A TABLE-B TABLE D   -------------- |
|-------------------------------------------------------------------------------|
| MNEMONIC | NUMBER | DESCRIPTION                                              |
|----------|--------|----------------------------------------------------------|
| ADPUPA   | A48102 | UPPER-AIR (RAOB, PIBAL, RECCO, DROPS) REPORTS            |
|          |        |                                                          |
| HEADR    | 348001 | REPORT HEADER SEQUENCE                                   |
| PRSLEVEL | 348002 | PRESSURE LEVEL SEQUENCE (ALL TYPES EXCEPT GOESND)        |
| T__EVENT | 348173 | TEMPERATURE EVENT SEQUENCE                               |
|          |        |                                                          |
| SID      | 001194 | STATION IDENTIFICATION                                   |
| TOB      | 012245 | TEMPERATURE OBSERVATION                                  |
| TQM      | 012246 | TEMPERATURE (QUALITY) MARKER                             |
|-------------------------------------------------------------------------------|
| MNEMONIC | SEQUENCE                                                                |
|----------|-------------------------------------------------------------------------|
| ADPUPA   | HEADR  SIRC  {PRSLEVEL}  <SST_INFO>  <PREWXSEQ>  {CLOUDSEQ}             |
|          |                                                                         |
| HEADR    | SID  XOB  YOB  DHR  ELV  TYP  T29  TSB  ITP  SQN  PROCN  RPT            |
| HEADR    | TCOR  <RSRD_SEQ>                                                        |
|          |                                                                         |
| PRSLEVEL | CAT  <P___INFO>  <Q___INFO>  <T___INFO>  <Z___INFO>  <W___INFO>         |
| PRSLEVEL | <DRFTINFO>                                                              |
|          |                                                                         |
| T__EVENT | TOB  TQM  TPC  TRC                                                      |
|-------------------------------------------------------------------------------|
| MNEMONIC | SCAL | REFERENCE    | BIT | UNITS                    |-------------|
|----------|------|--------------|-----|--------------------------|-------------|
| SID      |    0 |            0 |  64 | CCITT IA5                |-------------|
| TOB      |    1 |        -2732 |  14 | DEG C                    |-------------|
| TQM      |    0 |            0 |   5 | CODE TABLE               |-------------|
`-------------------------------------------------------------------------------'
```

DX BUFR table application examples: understand PQM

```
character(80):: qcstr='PQM QQM TQM ZQM WQM NUL PWQ      '
```

```
.-------------------------------------------------------------------------------.
| ------------   USER DEFINITIONS FOR TABLE-A TABLE-B TABLE D   -------------- |
|-------------------------------------------------------------------------------|
| MNEMONIC | NUMBER | DESCRIPTION                                              |
|----------|--------|----------------------------------------------------------|
| ADPUPA   | A48102 | UPPER-AIR (RAOB, PIBAL, RECCO, DROPS) REPORTS            |
|          |        |                                                          |
| HEADR    | 348001 | REPORT HEADER SEQUENCE                                   |
| PRSLEVEL | 348002 | PRESSURE LEVEL SEQUENCE (ALL TYPES EXCEPT GOESND)        |
| T__EVENT | 348173 | TEMPERATURE EVENT SEQUENCE                               |
|          |        |                                                          |
| SID      | 001194 | STATION IDENTIFICATION                                   |
| POB      | 007245 | PRESSURE OBSERVATION                                     |
| PQM      | 007246 | PRESSURE (QUALITY) MARKER                                |
|-------------------------------------------------------------------------------|
| MNEMONIC | SEQUENCE                                                              |
|----------|-----------------------------------------------------------------------|
| ADPUPA   | HEADR  SIRC  {PRSLEVEL}  <SST_INFO>  <PREWXSEQ>  {CLOUDSEQ}           |
|          |                                                                       |
| HEADR    | SID  XOB  YOB  DHR  ELV  TYP  T29  TSB  ITP  SQN  PROCN  RPT          |
| HEADR    | TCOR  <RSRD_SEQ>                                                      |
|          |                                                                       |
| PRSLEVEL | CAT  <P___INFO>  <Q___INFO>  <T___INFO>  <Z___INFO>  <W___INFO>       |
| PRSLEVEL | <DRFTINFO>                                                            |
|          |                                                                       |
| P__EVENT | POB  PQM  PPC  PRC
||-------------------------------------------------------------------------------|
| MNEMONIC | SCAL | REFERENCE   | BIT | UNITS                    |-------------|
|----------|------|-------------|-----|--------------------------|-------------|
| SID      |    0 |           0 |  64 | CCITT IA5                |-------------|
| POB      |    1 |           0 |  14 | MB                       |-------------|
| PQM      |    0 |           0 |   5 | CODE TABLE               |-------------|
`-------------------------------------------------------------------------------'
```

DX BUFR table application examples: flag and code table

http://www.emc.ncep.noaa.gov/mmb/data_processing/prepbufr.doc/table_1.htm

Most Visited | Getting Started | Latest Headlines | Google Maps | Storm Prediction Ce... | Apple | Yahoo! | News

CURRENT PREPBUFR MNEMONIC T...

PRESSURE

MNEMONIC	NUMBER	DESCRIPTION	SCALE	REFERENCE	BITS	UNITS
POB	007245	PRESSURE OBSERVATION	1	0	14	MB
PQM	007246	PRESSURE (QUALITY) MARKER	0	0	5	CODE TABLE

Find: PQM | Next | Previous | Highlight all | Match case | Reached end of page, continued from t

Table 7. Code table for observation quality markers (last revised 1/22/2008).

Quality Marker	Definition
0	All steps: Keep (always assimilate). Applies to pressure, height, wind, temperature, specific humidity, rainfall rate, precipitable water and cloud top pressure.
1	All steps: Good. Applies to pressure, height, wind, temperature, specific humidity, rainfall rate, precipitable water and cloud top pressure.
2	All steps: Neural or not checked (default). Applies to pressure, height, wind, temperature, specific humidity, rainfall rate, precipitable water and cloud top pressure.
3	All steps: Suspect. Applies to pressure, height, wind, temperature, specific humidity, rainfall rate, precipitable water and cloud top pressure.
4-15	All steps: Rejected (don't assimilate), as defined below (see % below table):
4	Step **OIQC**: An observation with pre-existing quality marker 0 (keep) is flagged. Applies to pressure, height, wind, temperature, specific humidity and precipitable water.

Find: TQM | Next | Previous | Highlight all | Match case | Reached end of page, continued from

DX BUFR table application examples: message content

```
msg_report: do while (ireadmg(unit_in,subset,idate) == 0)
subset = ADPUPA, ADPSFC, …
```

```
.-------------------------------------------------------------------------------.
| ------------   USER DEFINITIONS FOR TABLE-A TABLE-B TABLE D   -------------- |
|-------------------------------------------------------------------------------|
| MNEMONIC | NUMBER | DESCRIPTION                                              |
|----------|--------|----------------------------------------------------------|
| ADPUPA   | A48102 | UPPER-AIR (RAOB, PIBAL, RECCO, DROPS) REPORTS            |
|          |        |                                                          |
| HEADR    | 348001 | REPORT HEADER SEQUENCE                                   |
| PRSLEVEL | 348002 | PRESSURE LEVEL SEQUENCE (ALL TYPES EXCEPT GOESND)        |
| T__EVENT | 348173 | TEMPERATURE EVENT SEQUENCE                               |
|          |        |                                                          |
| SID      | 001194 | STATION IDENTIFICATION                                   |
| TOB      | 012245 | TEMPERATURE OBSERVATION                                  |
| TQM      | 012246 | TEMPERATURE (QUALITY) MARKER                             |
|-------------------------------------------------------------------------------|
| MNEMONIC | SEQUENCE                                                                |
|----------|-------------------------------------------------------------------------|
| ADPUPA   | HEADR  SIRC  {PRSLEVEL}  <SST_INFO>  <PREWXSEQ>  {CLOUDSEQ}             |
|          |                                                                         |
| HEADR    | SID  XOB  YOB  DHR  ELV  TYP  T29  TSB  ITP  SQN  PROCN  RPT            |
| HEADR    | TCOR  <RSRD_SEQ>                                                        |
|          |                                                                         |
| PRSLEVEL | CAT  <P___INFO>  <Q___INFO>  <T___INFO>  <Z___INFO>  <W___INFO>         |
| PRSLEVEL | <DRFTINFO>                                                              |
|          |                                                                         |
|
|-------------------------------------------------------------------------------|
| MNEMONIC | SCAL | REFERENCE    | BIT | UNITS                    |-------------|
|----------|------|--------------|-----|--------------------------|-------------|
| SID      |    0 |            0 |  64 | CCITT IA5                |-------------|
| TOB      |    1 |        -2732 |  14 | DEG C                    |-------------|
| TQM      |    0 |            0 |   5 | CODE TABLE               |-------------|
`-------------------------------------------------------------------------------'
```

DX BUFR table example: expand ADPUPA

```
| MNEMONIC | SEQUENCE                                                                |
|----------|-------------------------------------------------------------------------|
| ADPUPA   | HEADR  SIRC  {PRSLEVEL}  <SST_INFO>  <PREWXSEQ>  {CLOUDSEQ}             |
| ADPUPA   | <CLOU2SEQ>  <SWINDSEQ>  <AFIC_SEQ>  <TURB3SEQ>                          |

| HEADR    | SID     XOB   YOB   DHR   ELV   TYP   T29   TSB   ITP   SQN       |
| HEADR    | PROCN   RPT   TCOR <RSRD_SEQ>                                     |

| PRSLEVEL | CAT    <P___INFO>  <Q___INFO>  <T___INFO>  <Z___INFO>  <W___INFO> |
| PRSLEVEL |        <DRFTINFO>

      | P___INFO | [P__EVENT]                <P__BACKG>  <P__POSTP>
      | Q___INFO | [Q__EVENT]    TDO         <Q__BACKG>  <Q__POSTP>
      | T___INFO | [T__EVENT]    TVO         <T__BACKG>  <T__POSTP>
      | Z___INFO | [Z__EVENT]                <Z__BACKG>  <Z__POSTP>

            | P__EVENT |              POB                PQM    PPC    PRC
            | Q__EVENT |              QOB                QQM    QPC    QRC
            | T__EVENT |              TOB                TQM    TPC    TRC
            | Z__EVENT |              ZOB                ZQM    ZPC    ZRC

            | P__BACKG | POE       PFC       <PFC__MSQ>
            | Q__BACKG | QOE       QFC       <QFC__MSQ>
            | T__BACKG | TOE       TFC       <TFC__MSQ>
            | Z__BACKG |           ZFC       <ZFC__MSQ>

            | P__POSTP | PAN       <PCLIMATO>   POETU   PVWTG   PVWTA
            | Q__POSTP | QAN       <QCLIMATO>   QOETU   QVWTG   QVWTA   ESBAK
            | T__POSTP | TAN       <TCLIMATO>   TOETU   TVWTG   TVWTA
            | Z__POSTP | ZAN       <ZCLIMATO>
```

It is always a good idea to fully expand all the sequences for Table A mnemonic, which help you easily understand exactly what is contained within the report.

(GSI util/bufr_tools/prepobs_prep.bufrtable)

Reference

- http://www.dtcenter.org/com-GSI/BUFR/docs/index.php
  - BUFR User's Guide version 1.0:

    [pdf (1M), last update: Jan. 13, 2012];

- GSI summer tutorial 2013
  - Observation processing

    [Dennis Keyser & Jeff Whiting (NCEP/EMC)] [pdf]

- WMO BUFR User's Guide
  - https://www.wmo.int/pages/prog/gcos/documents/gruanmanuals/ECMWF/bufr_user_guide.pdf

Questions?

gsi-help@ucar.edu