2012 GSI Summer Tutorial, 22 Aug. 2012, Boulder, CO

Community Tools (1) - PrepBUFR/BUFR:
Basic tools, NCEP data, and BUFR table

Ruifang Li

NCAR/MMM

Developmental Testbed Center

Topics covered

- NCEP observation data
  - Observation data processing
  - BUFR/PrepBUFR file structure
  - Operation BUFR/PrepBUFR types and data servers
- Community BUFR/PrepBUFR basic tools
  - Encode, decode and append a simple BUFR file
- NCEP DX BUFR table
  - DX BUFR table structure and examples
  - DX BUFR table application examples

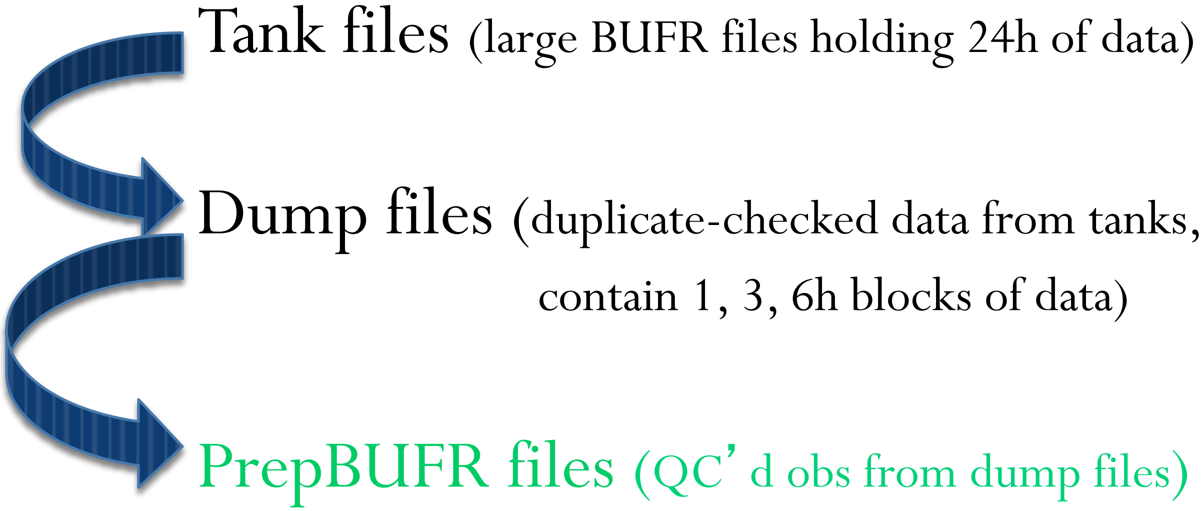
This talk is based on DTC BUFR/PrepBUFR User's Guide:
http://www.dtcenter.org/com-GSI/BUFR/docs/index.php

NCEP observation data

- Observation data processing
- BUFR/PrepBUFR file structure
- Operation BUFR/PrepBUFR types and data servers

Observation processing at NCEP

- Managed jointly by NCEP Central Operations (NCO) and EMC
- Relies on NCEP BUFRLIB software
- Three stages:

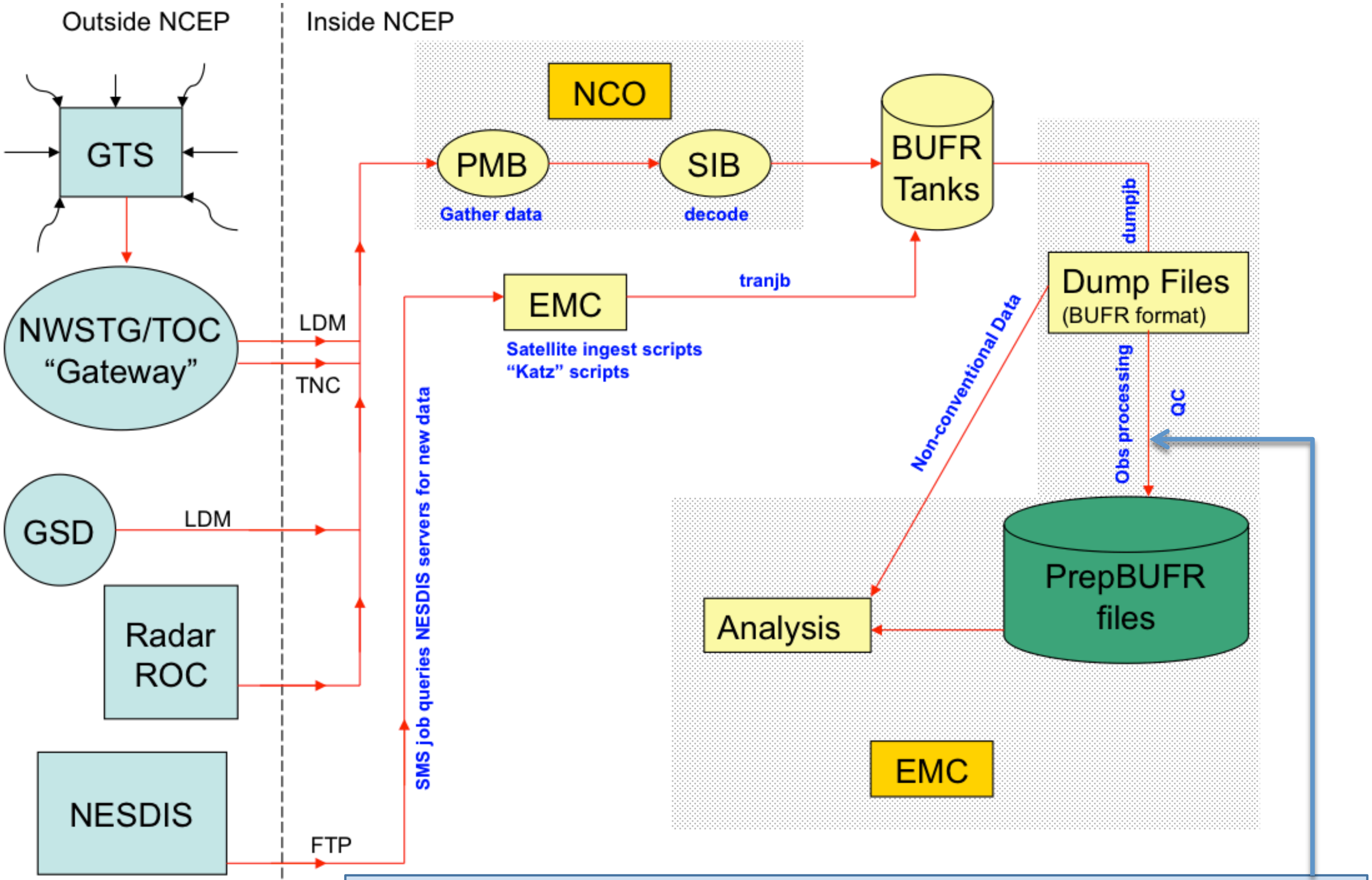


Tank files (large BUFR files holding 24h of data)

Dump files (duplicate-checked data from tanks, contain 1, 3, 6h blocks of data)

PrepBUFR files (QC' d obs from dump files)

Data processing system at NCEP



Dennis Keyser’s website on PREPBUFR PROCESSING AT NCEP:
http://www.emc.ncep.noaa.gov/mmb/data_processing/prepbufr.doc/document.ht

** Diagram originally developed by Shelley Melchior **

Developmental Testbed Center

What is BUFR/PrepBUFR

- **B**inary **U**niversal **F**orm for the **R**epresentation of meteorological data (BUFR)
- BUFR is a “self-descriptive” table driven code form
  - The form and content of the data contained within a BUFR message are described within the BUFR message itself
- BUFR is one of the code forms WMO recommends for the representation and exchange of observational data

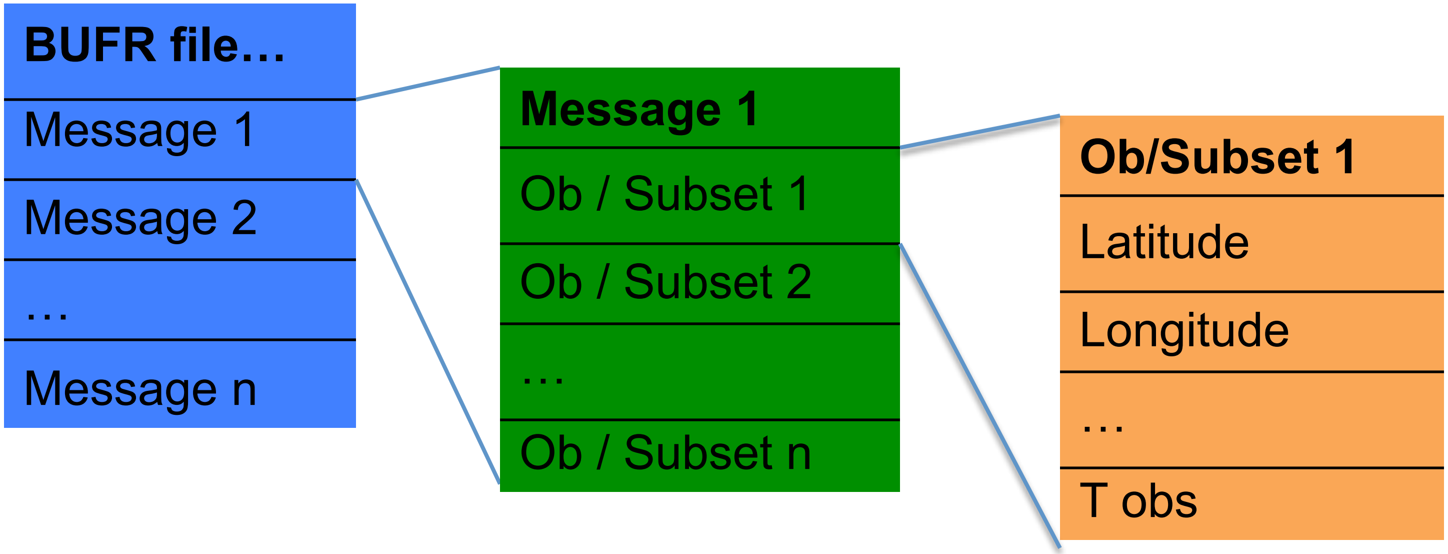
- **PrepBUFR is the NCEP term for “prepared” or QC’d data in BUFR format (NCEP convention/standard)**

- **PrepBUFR file is still a BUFR file**

BUFR/PrepBUFR file structure

A bit of terminology:

BUFR files (including "PrepBUFR" files) contain "messages". Each message contains "subsets." Each subset contains meteorological observation.



BUFR file…

Message 1

Message 2

…

Message n

Message 1

Ob / Subset 1

Ob / Subset 2

…

Ob / Subset n

Ob/Subset 1

Latitude

Longitude

…

T obs

BUFR/PrepBUFR file structure

BUFR file example: *gdas1.t12z.prepbufr.nr*

Message 1: ADPSFC: Surface land (synoptic, metar) reports

Message 2: ADPUPA: Upper air (raob, pibal, recco, drops) reports

.

.

Message n:

Message 1	Lat	Lon	P	T	Q	U	V	Type	
Ob / Subset 1	52.1	12.5	984.4	10e10	10e10	1.7	4.7	281	**wind report**
Ob / Subset 2	52.1	12.5	984.4	23.1	12979.0	10e10	10e10	181	**mass report**
.									
.									
Ob / Subset n									

Operation BUFR/PrepBUFR types

- 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
- 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

1st section: operation system
2st section: analysis time
3st section: data type
4st , 5st sections:
nr: non-restricted data
bufr_d: bufr format
tm**: indicate the catch up cycle analysis time,

=0 analysis time = 2st section
>0 analysis time = 2st section - 4st section
For example:
ndas.t18z.lbamub.tm03.bufr_d
has analysis time=18z-03z=15z

See BUFR User's Guide Chapter 5.2

Operation BUFR/PrepBUFR data servers

- Resources 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 tools

- All tools are in GSI util/bufr_tools
- Encode, decode and append a simple BUFR file
- Detailed information is in BUFR User's Guide.

All tools based on NCEP BUFRLIB

- BUFRLIB contains close to 250 Fortran and C subprograms and functions, no more than 10-20 of them are directly called by a user, the rest are used to accomplish various underlying tasks.

  The detailed BUFRLIB documentation:

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

- The BUFRLIB requires BUFR/PrepBUFR files to be FORTRAN-blocked before they are used by BUFRLIB.
  - Almost always, any BUFR file is already blocked and you can use the BUFRLIB routines directly to read/write BUFR file.
  - If your BUFR file is not blocked, Use NCEP cwordsh utility to block it http://www.nco.ncep.noaa.gov/sib/decoders/BUFRLIB/toc/cwordsh/

BUFR Processing Actions

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

Encode BUFR file

- Write the observation into a BUFR file

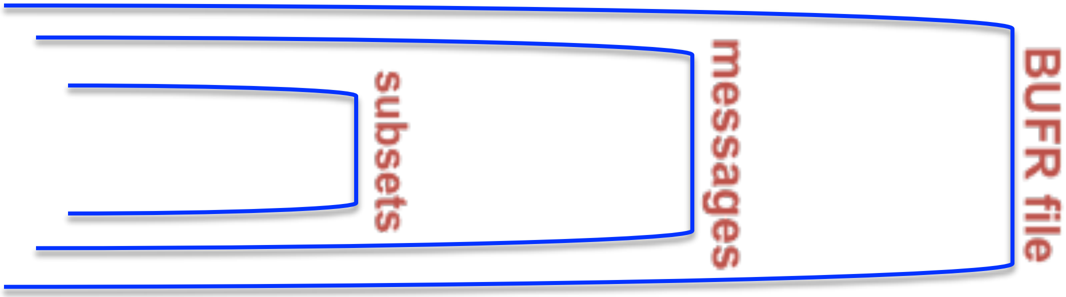
```
program bufr_encode_sample   (34 Lines)
!
!  example of writing one value into a bufr file
!
 implicit none

 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
```

"A BUFR file contains one or more BUFR messages, each containing one or more BUFR data subsets, each containing one or more BUFR data values

Define variables and assign obs values

Open BUFR tables file

Open BUFR file



subsets

messages

BUFR file

BUFRLIB subroutines

```
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
```

openbf, closbf, openmb, closmg, ufbint, writsb, datelen are the BUFRLIB subroutines and functions. They are very often used to read/write BUFR file. Understanding usage of them will be very helpful in users own application.

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='unforma
 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 and form for each of message types. It is embedded within the first few BUFR messages of the file itself. 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

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

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

Setup data

Data written to subset.

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: severs the connection between logical unit *LUBFR* and the BUFRLIB.

```
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
```

Set cycle date

***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
```

***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.

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

Data subsets

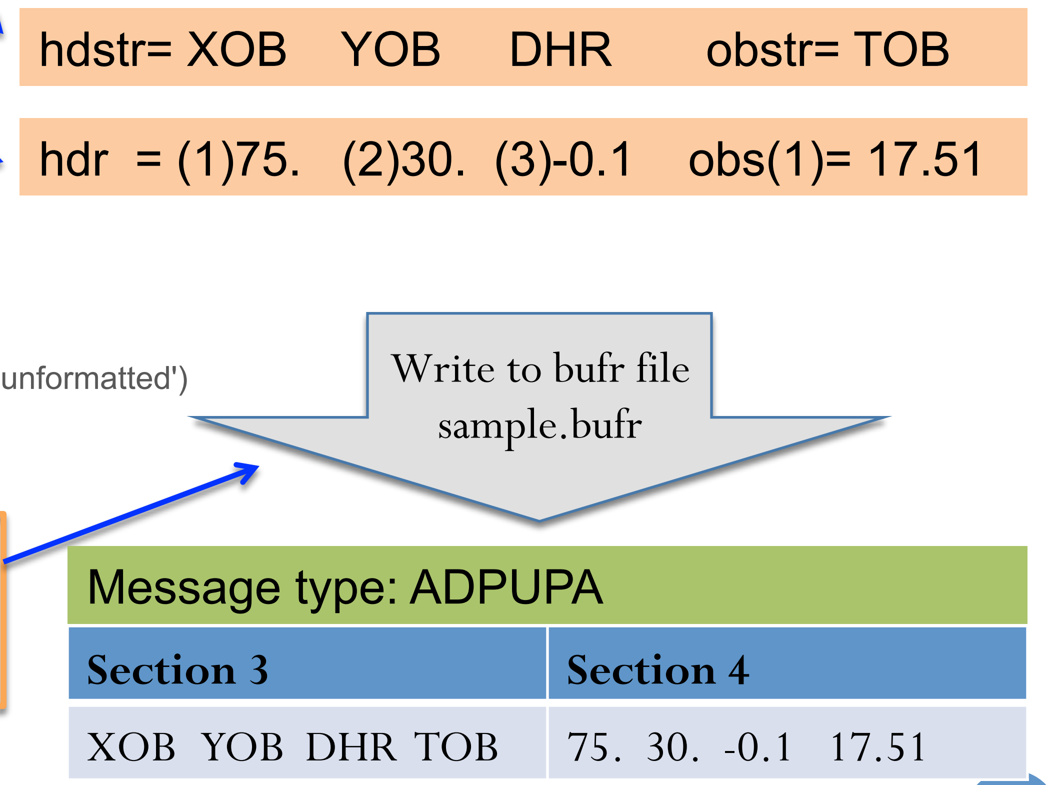
```
program bufr_encode_sample   (34 Lines)
!  example of writing one value into a bufr file
 implicit none

 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.

```
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
```

Mnemonics and data array



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 from BUFR file

```
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
```

"A BUFR file contains one or more BUFR messages, each containing one or more BUFR data subsets, each containing one or more BUFR data values

subsets

messages

BUFR file

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
```

Message and data subset

```
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
```

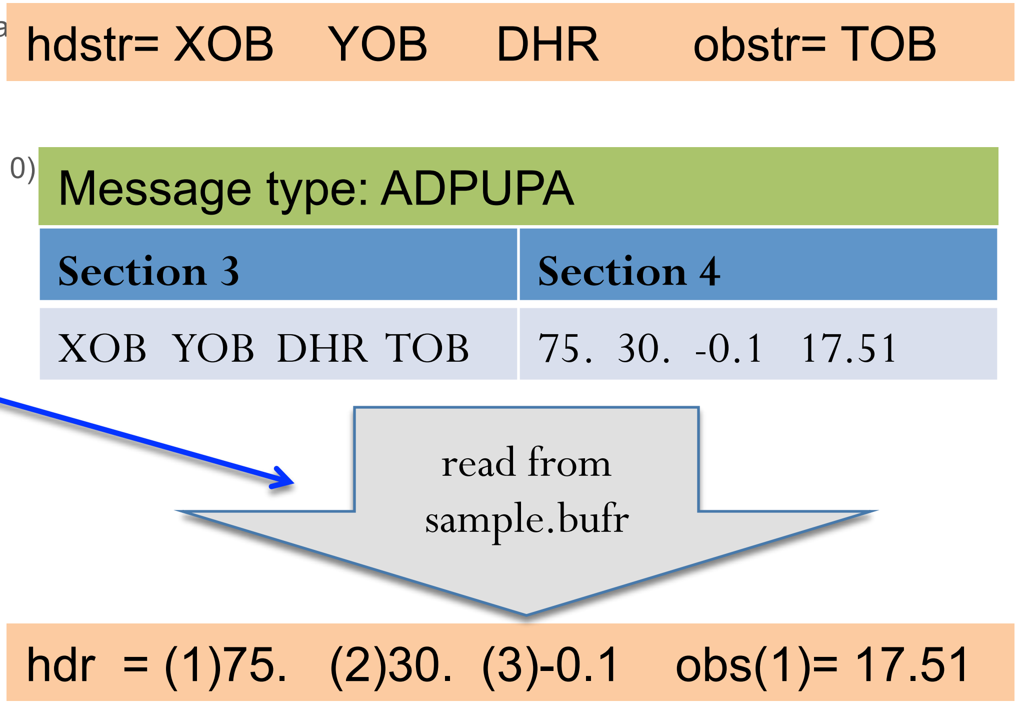
IRET = *IREADMG* (LUBFR, CSUBSET, IDATE)
Purpose: reads the next BUFR message from the given BUFR file pointed to by *LUBFR*.

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='unforma
 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
```

BUFR table

```
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.

```
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.

Test results (Basic Practice case 0):

```
./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   17.1
```

```
./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   17.1
    2008120101   2  ADPSFC
        1    1    85.0    50.0     0.2   15.0
```

Examples for GSI BUFR/PrepBUFR files

Code name	Illustrated process function
prepbufr_decode_all.f90	• read BUFR table from an existing prepbufr file • read all observation information used by GSI analysis from an existing prepbufr file.
prepbufr_encode_surface.f90	• write a surface observation into a new prepbufr file
prepbufr_encode_upperair.f90	• write a upper air observation into a new prepbufr file
prepbufr_append_upperair.f90	• read BUFR table from an existing prepbufr file • append a upper air observation into an existing prepbufr file
prepbufr_append_surface.f90	• read BUFR table from an existing prepbufr file • append a surface observation into an existing prepbufr file.
prepbufr_append_retrieve.f90	• read BUFR table from an existing prepbufr file • append retrieved data into an existing prepbufr file.
bufr_decode_radiance.f90	• read BUFR table from an existing radiance bufr file • real radiance data from an existing radiance bufr file.

- 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)
```

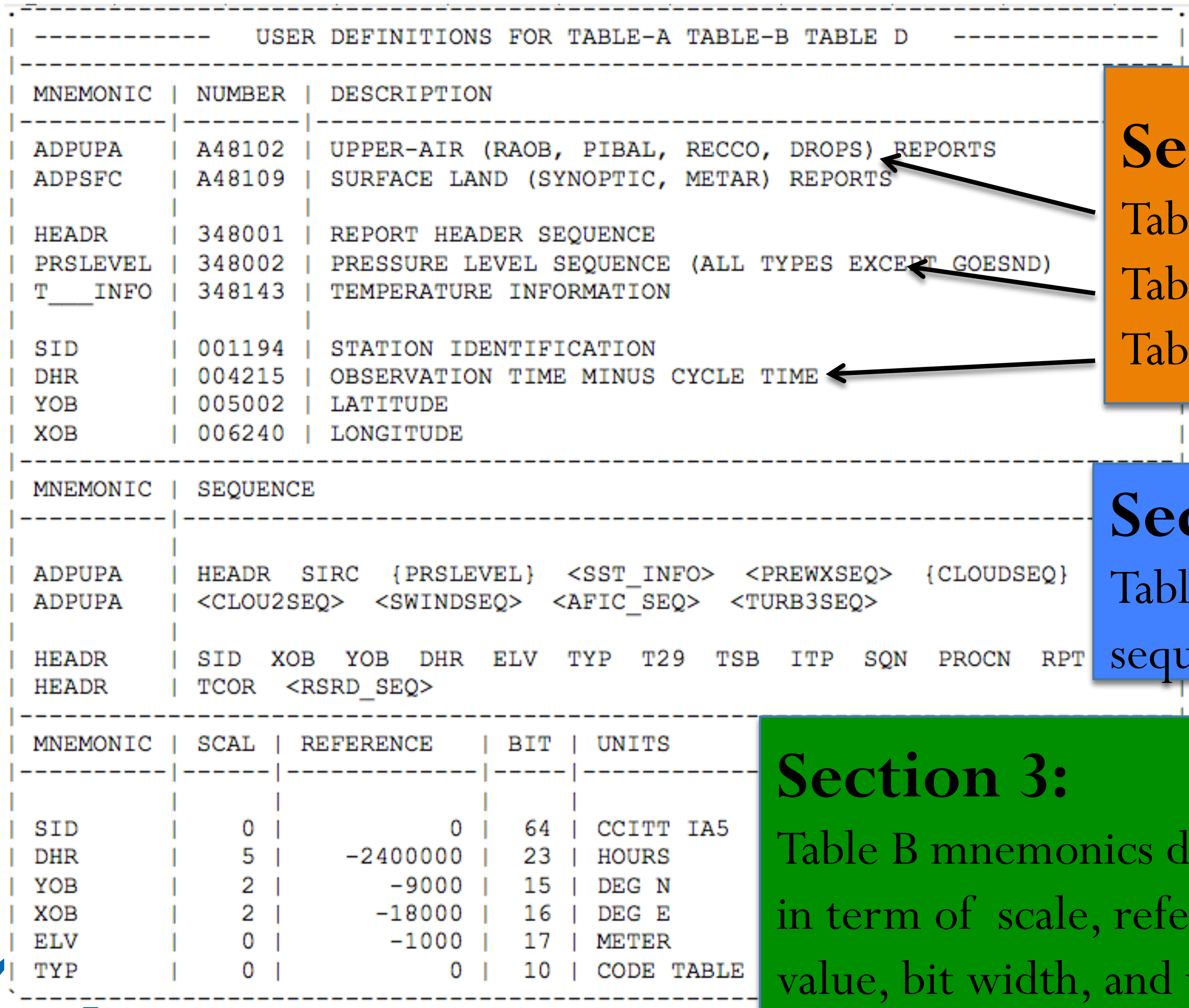
NCEP DX BUFR table

- DX BUFR table structure and examples
- DX BUFR table application examples
- Detailed information is in BUFR user guide.

DX BUFR table

- Define report structures in any kind of BUFR/ PrepBUFR files.
- Report structures for various types of observations are defined by "NCEP BUFR Tables" when using the NCEP BUFRLIB software.
- 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   -------------- |
|------------------------------------------------------------------------------|
| 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
`-------------------------------------------------
```

Section 1:

Table A mnemonic

Table D mnemonic

Table B mnemonic

Section 2:

Table A and Table D sequences

Section 3:

Table B mnemonics defined in term of scale, reference value, bit width, and unit.

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

- **Section 1**: all Table A, B and D mnemonics are declared, assigned a unique FXY number, and given a short description.

Table A mnemonic:
Refer to report types

```
|------------------------------------------------------------------------|
| MNEMONIC | NUMBER | DESCRIPTION                                        |
|----------|--------|----------------------------------------------------|
| ADPUPA   | A48102 | UPPER-AIR (RAOB, PIBAL, RECCO, DROPS) REPORTS      |
```

Table B mnemonic:
Refer to basic data values

```
|------------------------------------------------------------------------|
| MNEMONIC | NUMBER | DESCRIPTION                                        |
|----------|--------|----------------------------------------------------|
| SID   | 001194 | STATION IDENTIFICATI                                  |
| XOB   | 006240 | LONGITUDE                                             |
| YOB   | 005002 | LATITUDE                                              |
| DHR   | 004215 | OBSERVATION TIME MINUS CYCLE TI                       |
```

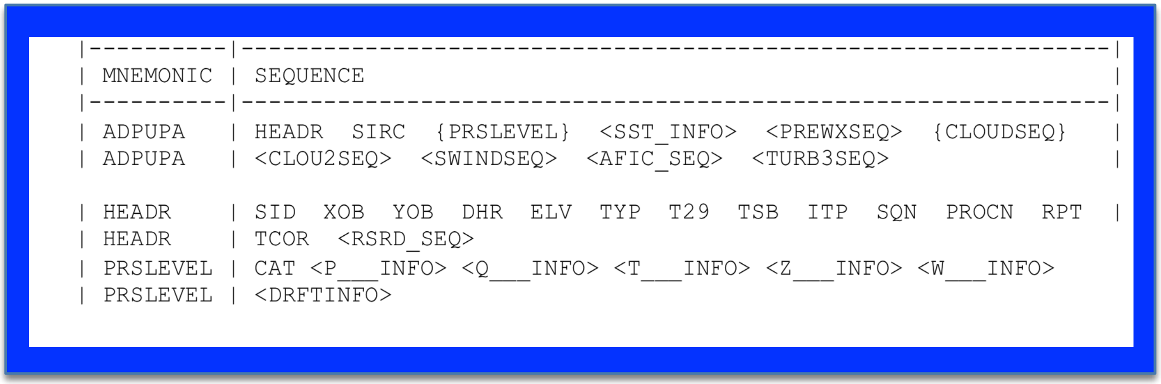
Table D mnemonic:
Constituents of a particular Table A mnemonic.

```
|------------------------------------------------------------------------|
| MNEMONIC | NUMBER | DESCRIPTION                                        |
|----------|--------|----------------------------------------------------|
| HEADR    | 348001 | REPORT HEADER SEQUENCE |
| PRSLEVEL | 348002 | PRESSURE LEVEL SEQUENCE (ALL TYPES EXCEPT "GOESND",
                        "AIRCFT" and "AIRCAR")
```

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

- Section 2

Table A, D mnemonic making up sequence



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

Replication:

a way to efficiently store data in BUFR format

< >	Indicates that the enclosed mnemonic is replicated using 1-bit delayed replication (either 0 or 1 replications). e.g. **<SST_INFO>**
{}/[ ]	Indicates that the enclosed mnemonic is replicated using 8-bit delayed replication (between 0 and 255 replications) e.g. **{PRSLEVEL}**
( )	Indicates that the enclosed mnemonic is replicated using 16-bit delayed replication (between 0 and 65535 replications)
" "n	Indicates that the enclosed mnemonic is replicated using regular (non-delayed) replication, with a fixed replication factor of *n*. e.g. **"GQCPRMS"3**

Developmental Testbed Center

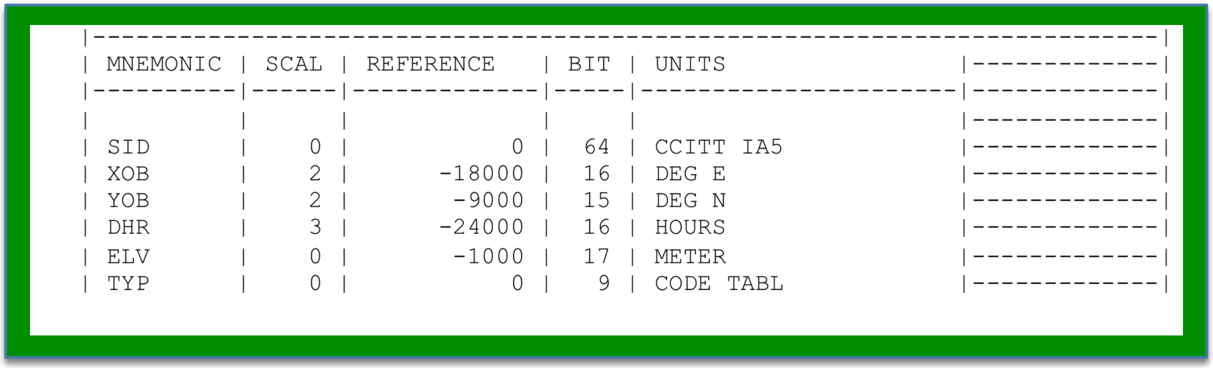
DX BUFR table example (GSI util/bufr_tools/prepobs_prep.bufrtable)

- **Section 3**

Table B mnemonic

scale, reference,

bit, unit



```
|---------------------------------------------------------------------------|
| 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.

Developmental Testbed Center

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                |-------------|
`------------------------------------------------------------------------------'
```

Developmental Testbed Center

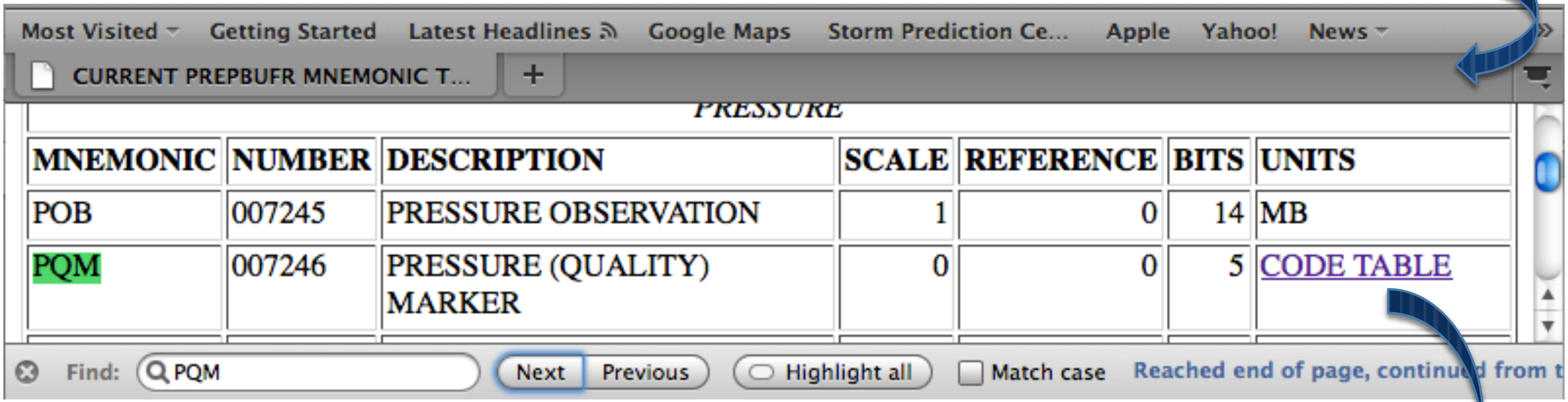
DX BUFR table application examples: understand PQM

```
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



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

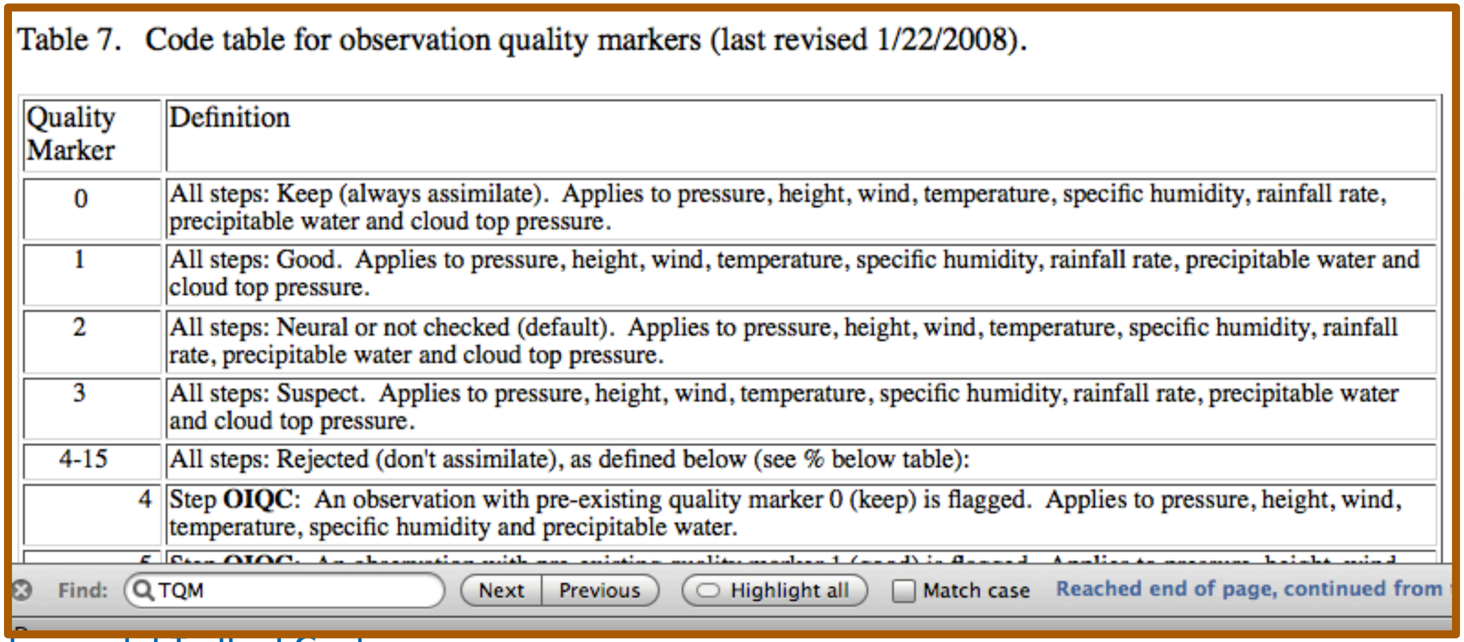


Table 7. Code table for observation quality markers (last revised 1/22/2008).

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.
5	Step **OIQC**: An observation with pre-existing quality marker 1 (good) is flagged. Applies to pressure, height, wind

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];

Questions?

gsi_help@ucar.edu