ObsConvert

from cy48 (under development)

ObsConvert is a port of BATOR to the new EcCodes API, which replaces BUFRDC. As Bator using BUFRDC API is the well known preprocessing tool used in our operational suites since many years, you will only find in this paper the Bator changes which are needed by the use of EcCodes.

Contents

I. Required items and knowledges

- EcCodes > v. 2.14.0,

- all needed BUFR tables are not included in the ECMWF package (locals, Eumetsat…), so you have to get or create missing ones for your needs.

- new input files for Obsconvert : param.cfg and NAMELIST.

- knowledge of EcCodes is essential. Please refer to the documentation on the ECMWF site at : https://confluence.ecmwf.int/display/ECC/ecCodes+Home

II. Matching source file names

```
ObsConvertFunctions.h              <=>  bator_fonction.func.h
AbstractBufrReader.F90             NEW
CompressedBufrReader.F90           NEW
EcFactory.F90                      NEW
ObsConvertBufrToOdb.F90            <=>  bator_decodbufr_mod.F90
ObsConvertDateAndTime.F90          <=>  bator_datetime_mod.F90
ObsConvertGlobalDefinition.F90     <=>  bator_module.F90
ObsConvertHdfToOdb.F90             <=>  bator_decodhdf5_mod.F90
ObsConvertInits.F90                <=>  bator_init_mod.F90
ObsConvertNetcdfToOdb.F90          <=>  bator_decodnetcdf_mod.F90
ObsConvertPoolBalance.F90          <=>  bator_pool_balance_mod.F90
ObsConvertPrintings.F90            <=>  bator_impr_mod.F90
ObsConvertRadiancePostproc.F90     <=>  bator_rad_postproc_mod.F90
ObsConvertReadings.F90             <=>  bator_lectures_mod.F90
ObsConvertTools.F90                <=>  bator_util_mod.F90
ObsConvertWritingsToOdb.F90        <=>  bator_ecritures_mod.F90
ObsConvertSaisies.F90              <=>  bator_saisies_mod.F90
SingleUncompressedBufrReader.F90 NEW
UncompressedBufrReader.F90         NEW
ObsConvertNamelist.h               <=>  bator_namelist.nam.h
ObsConvert.F90                     <=>  Bator.F90
```

III. Aim of the new modules

EcCodes does not allow to recover data in the same way whether you work on a single-subset, an uncompressed multi-subsets or a compressed multi-subsets BUFR file.

To deal with these differences, I chose to create a specific 'reader' based on the currently decoded BUFR file type. Thus, the user will be able to use the same function/subroutine calls regardless of the BUFR file. The 5 new modules allow this.

IV. Initializing a reader

```
Here is a simple example, you can also have a full example in ObsConvertBufrToOdb.F90 :

program example
 use ECCODES
 use parkind1,        only : jpim
 use BufrReaderFactory,  only : ReaderFactory
 use AbstractBufrReader, only : GenericReader

 type(ReaderFactory)        :: Factory
 class(Genericreader), pointer :: Reader
 integer(kind=jpim)         :: FileId, BufrId, Ret, Compressed, Subsets

 call codes_open_file(FileId,'bufr.temp','r',Ret)
 if (Ret == CODES_SUCCESS) then
 call codes_bufr_new_from_file(FileId,BufrId,Ret)
 call codes_get(BufrId,'compressedData',Compressed)
 call codes_get(BufrId,'numberOfSubsets',Subsets)
 Reader => Factory%CreateReader(BufrId,Compressed,Subsets)
 ...
 ...
 ...
 call Factory%Release
 call codes_release(BufrId)
 call codes_close_file(FileId)
 endif
end program
```

V. New subroutines and functions

Once the reader is created, you can use any subroutine or function described below. Thereafter we will assume your reader is called Reader.

1) Subroutine Reader%GetSingle

```
Reader%GetSingle( character(len=*),                   intent(in)    key,
 integer(kind=jpim),                                  intent(in)    subscript,
 integer(kind=jpim),                                  intent(in)    numsubset,
 integer(kind=jpim),                                  intent(out)   res )
```

Get a single integer value for a key from a message. In case of error returns nabsi.

key	key name
subscript	desired occurrence of the key in the chosen subset
numsubset	subset number from which res is to be read
res	single integer value.

```
Example : get the first occurrence of SID in the first subset from a bufr.amsub file.

integer(kind=jpim) :: SatId
call Reader%GetSingle('satelliteIdentifier',1,1,SatId)
```

```
Reader%GetSingle( character(len=*),                    intent(in)     key,
  integer(kind=jpim),                                  intent(in)     subscript,
  integer(kind=jpim),                                  intent(in)     numsubset,
  real(kind=jprd),                                     intent(out)    res )
```

Get a single real value for a key from a message. In case of error returns `rabsi`.

key	Key name
subscript	desired occurrence of the Key in the chosen subset
numsubset	subset number from which res is to be read
res	single real value.

```
Example : get the first occurrence of latitude displacement in the first subset from a
bufr.temp file.

real(kind=jprd) :: DeltaLat
call Reader%GetSingle('latitudeDisplacement',1,1,DeltaLat)
```

```
Reader%GetSingle( character(len=*),                    intent(in)     cond,
  character(len=*),                                    intent(in)     key,
  integer(kind=jpim),                                  intent(in)     numsubset,
  integer(kind=jpim),                                  intent(out)    res )
```

Get a single integer value for a key with a condition from a message. In case of error returns `nabsi`.

cond	the condition
key	key name
numsubset	subset number from which res is to be read
res	single integer value.

```
Example : get the first occurrence of vertical sounding reference in the first subset
from a bufr.temp file.

integer(kind=jpim) :: Vsr
call Reader%GetSingle('extendedVerticalSoundingSignificance',1,1,Vsr)
```

```
Reader%GetSingle( character(len=*),                    intent(in)     cond,
  character(len=*),                                    intent(in)     key,
  integer(kind=jpim),                                  intent(in)     numsubset,
  real(kind=jprd),                                     intent(out)    res )
```

Get a single real value for a key with a condition from a message. In case of error returns `rabsi`.

cond	the condition
key	key name
numsubset	subset number from which res is to be read
res	single real value.

```
Example : get Td following the condition in the first subset from a bufr.temp file.

real(kind=jprd) :: Td
call Reader%GetSingle('verticalSoundingSignificance=4','dewpointTemperature',1,Td)
```

2) Subroutine Reader%GetReal

```
Reader%GetReal( character(len=*),                          intent(in)      key,
  integer(kind=jpim),                                      intent(in)      subscript,
  real(kind=jprd),dimension(:),allocatable,                intent(out)     res )
```

Get real array values for a key from a message. On error returns `rabsi`.

key	Key name
subscript	desired occurrence of the Key
res	real array values.

```
Example : get all first occurrences of Td from a bufr.temp file.

real(kind=jprd),dimension(:),allocatable :: Td
call Reader%GetReal('dewpointTemperature',1,Td)
```

```
Reader%GetReal( character(len=*),                          intent(in)      cond,
  character(len=*),                                        intent(in)      key,
  real(kind=jprd),dimension(:),allocatable,                intent(out)     res )
```

Get real array values for a key with a condition from a message. On error returns `rabsi`.

cond	the condition
key	key name
res	real array values.

```
Example : get all occurences of Td following the condition from a bufr.temp file.

real(kind=jprd),dimension(:),allocatable :: Td
call Reader%GetReal('verticalSoundingSignificance=4','dewpointTemperature',Td)
```

```
Reader%GetReal( character(len=*),                          intent(in)      key,
  integer(kind=jpim),                                      intent(in)      nbiter,
  integer(kind=jpim),                                      intent(in)      numsubset,
  real(kind=jprd),dimension(:),allocatable,                intent(out)     res )
```

Get real array values for a key in a particular subset of a message. On error returns `rabsi`.

key	Key name
nbiter	number of iteration to do. For a <u>compressed BUFR file</u>, If <=0, 10000 iterations will be done.
numsubset	subset number from which res is to be read
res	real array values.

```
Example : get the first 10 occurrences of Td in the first subset from a bufr.temp file.

real(kind=jprd),dimension(:),allocatable :: Td
call Reader%GetReal('dewpointTemperature',10,1,Td)
```

3) Subroutine Reader%GetInteger

```
Reader%GetInteger( character(len=*),                 intent(in)     key,
  integer(kind=jpim),                                intent(in)     subscript,
  integer(kind=jpim),dimension(:),allocatable,       intent(out)    res )
```

Get integer array values for a key from a message. On error returns `rabsi`.

key	Key name
subscript	desired occurrence of the Key
res	integer array values.

```
Example : get all first occurrences of vertical sounding reference from a bufr.temp file.

integer(kind=jpim),dimension(:),allocatable :: Vsr
call Reader%GetInteger('verticalSoundingSignificance',1,Vsr)
```

```
Reader%GetInteger( character(len=*),                 intent(in)     cond,
  character(len=*),                                  intent(in)     key,
  integer(kind=jpim),dimension(:),allocatable,       intent(out)    res )
```

Get integer array values for a key with a condition from a message. On error returns `rabsi`.

cond	the condition
key	key name
res	integer array values.

```
Example : get all occurences of time difference following the condition from a bufr.temp file.

integer(kind=jpim),dimension(:),allocatable :: DeltaTime
call Reader%GetInteger('verticalSoundingSignificance=4','timePeriod',DeltaTime)
```

```
Reader%GetInteger( character(len=*),                 intent(in)     key,
  integer(kind=jpim),                                intent(in)     nbiter,
  integer(kind=jpim),                                intent(in)     numsubset,
  integer(kind=jpim),dimension(:),allocatable,       intent(out)    res )
```

Get integer array values for a key in a particular subset of a message. On error returns `rabsi`.

key	Key name
nbiter	number of iteration to do. For a <u>compressed BUFR file</u>, If <=0, 10000 iterations will be done (can be modified in ObsconvertGlobalDefinition. F90).

numsubset   subset number from which res is to be read

res   integer array values.

```
Example : get the first 10 occurrences of vertical sounding significance in the first
subset from a bufr.temp file.

integer(kind=jpim),dimension(:),allocatable :: Vsr
call Reader%GetInteger('verticalSoundingSignificance',10,1,Vsr)
```

4)   Subroutine Reader%GetString

```
Reader%GetInteger( character(len=*),               intent(in)     key,
  character(len=*),dimension(:),allocatable,       intent(out)    res )
```

Get a string array values for a key of a message. On error returns empty string.

key   Key name

res   string array values.

```
Example : get the fight number from a bufr.amdar file.

character(len=8),dimension(:),allocatable :: Flight
call Reader%GetString('aircraftFlightNumber',Flight)
```

5)   Subroutine Reader%GetDate

```
Reader%GetInteger( integer(kind=jpim),             intent(in)     subscript,
  integer(kind=jpim),dimension(:,:),allocatable,   intent(out)    res )
```

Get 'date+time' values of a message and store them in a 2 dimensions array.

Res(:,1)=year, res(:,2)=month, res(:,3)=day, res(:,4)=hour, res(:,5)=minutes, res(:,6)=seconds.

For each element, returns `nabsi` when error occurs, except for seconds :
- returns `0` when the descriptor for 'seconds' is missing or its value is nabsi,
- returns `59` when the descriptor value for 'seconds' exceeds 59.

subscript   desired occurrence of the Key

res   integer 2D array values.

```
Example : get the second date and time from a bufr.buoy file.

integer(kind=jpim),dimension(:,:),allocatable :: DateAndTime
call Reader%GetDate(2,DateAndTime)
```

6) Function Reader %IsDefined

```
Reader%IsDefined( character(len=*),                    intent(in)     key,
  integer(kind=jpim),                                  intent(in)     subscript )
```

Returns `.TRUE.` if the key exists in message, `.FALSE.` if not.

key   Key name
subscript   desired occurrence of the Key

```
Example: check if the first occurrence of WMO block number exists from a bufr.synop
file.

  if (Reader%IsDefined('blockNumber',1)) then
    ...
    ...
  endif
```

```
Reader%IsDefined( character(len=*),                    intent(in)     cond,
  character(len=*),                                    intent(in)     key )
```

Returns `.TRUE.` if the key exists in message, `.FALSE.` if not.

cond   the condition
key   key name

```
Example: check if  Td exists following the condition from a bufr.temp file.

  if (Reader%IsDefined('verticalSoundingSignificance=4','dewpointTemperature')) then
    ...
    ...
  endif
```

VI. other documentation

Full EcCodes documentation can be found on the ECMWF website.

Additional ObsConvert documentation can be found on the GMAPDOC website.

BDMO documentation can be found on the DSI website.