

ECMWF Feature article

.....
from Newsletter Number 132 – Summer 2012

COMPUTING

.....
BUFR data and Metview
.....



www.ecmwf.int/en/about/news-centre/media-resources

doi:10.21957/rd4r58uu

This article appeared in the Computing section of ECMWF Newsletter No. 132 – Summer 2012, pp. 34–36.

BUFR data and Metview

Vesa Karhila, Sándor Kertész, Iain Russell, Stephan Siemen

BUFR has a reputation of being complex and difficult to use. With this article we want to show how Metview can be used to handle BUFR data easily and efficiently. BUFR, like GRIB, is binary: it requires software to decode it. ECMWF provides a software package BUFRDC to handle BUFR data, but this package requires some expertise to use. Metview has an internal higher-level wrapper around the basic BUFRDC library, making it much easier for Metview developers to write tools to handle BUFR data. As a result, Metview has always provided practical solutions for handling BUFR data and the development of such tools continues in Metview 4. For instance, since version 4.3.0 Metview has been able to decode also PrepBUFR files. With Metview we can easily examine, extract, plot, overlay and process BUFR data. An overview of BUFR is given in Box A.

BUFR data stored in MARS can be accessed with a Metview *Mars Retrieval* icon. Files containing BUFR messages are automatically shown with a *BUFR* icon (Figure 1a) in the Metview Desktop.

Plotting BUFR data

Metview can plot some types of BUFR data directly (mainly some traditional WMO observation types for which WMO has defined a plotting scheme) simply by right-clicking the icon and selecting *Visualise* from the pop-up menu. To customise which values to include (or exclude) from a plot, we use the *Observation Plotting* icon (Figure 1b). An example plot of TEMP data is shown in Figure 2.

To plot any other types of BUFR data we need to extract the required data values and locations into geopoints or CSV (Comma Separated Values) format (see below) and then plot these values. BUFR data can be overlaid with any other data types in a Metview plot.

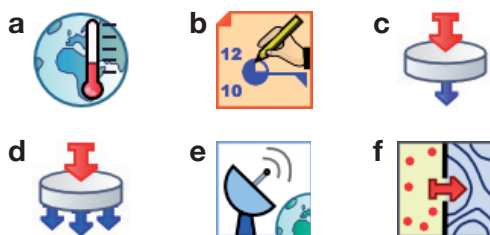


Figure 1 Icons representing (a) BUFR data, (b) Observation Plotting visual definition, (c) Observation Filter tool, (d) BufrPicker tool, (e) OperaRadarFilter tool and (f) Geopoints to Grib tool.

BUFR and PrepBUFR

A

BUFR

BUFR (Binary Universal Form for Representation of meteorological data) is a WMO standard for transmitting and storing observed meteorological data. BUFR is flexible – it can be used to store very different data, ranging from traditional SYNOP and TEMP observations to modern satellite and radar observations.

Data stored in BUFR messages are identified by *data descriptors*. Data descriptors are keys to the BUFR data. To access any data values from BUFR messages, we need to know the data descriptor of the data type.

BUFR is a ‘Table Driven Code Form’. This code form is based on the idea that part of the metadata, including the values used for coding,

is stored in external table files. This is one strong point of BUFR (messages are smaller), but it becomes a weak point if something is wrong with the required external table files.

PrepBUFR

BUFR is so flexible that it is possible to encode BUFR tables in BUFR format. NCEP (National Centers for Environmental Prediction, USA) creates BUFR files, called *PrepBUFR*, where BUFR messages containing meteorological data are preceded by messages that contain the BUFR table entries required to decode the data part. This method avoids all table related errors in decoding, but it sets new requirements for the decoding software.

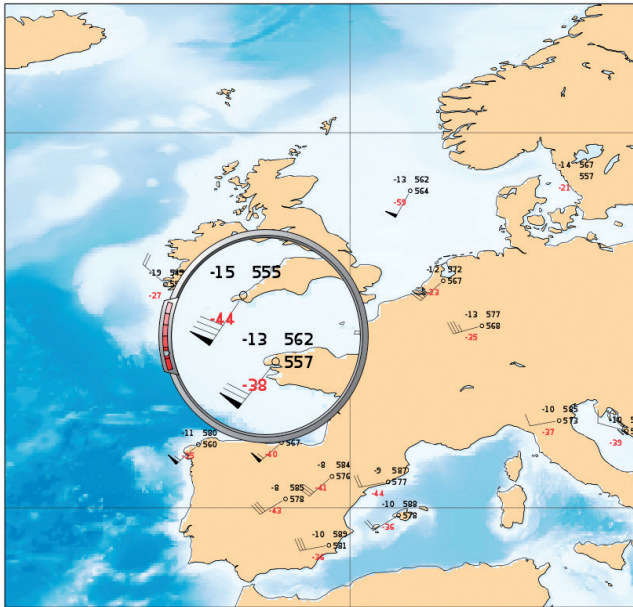


Figure 2 An example plot of TEMP (upper-air sounding) data.

Examining BUFR data

In order to extract values from BUFR data we need to know what data is stored in a BUFR file. *Bufr Examiner* is Metview's tool to study the contents of BUFR files. It is available from the context menu of a BUFR icon. *Bufr Examiner* was completely re-written for Metview 4 and the new version provides options to search and to customise the output. An example output of *Bufr Examiner* is shown in Figure 3.

Extracting data values from BUFR files

Metview provides several modules to extract values from BUFR files, for plotting or for further processing. *Data descriptors* (as shown in the *Descriptor* column in the *Bufr Examiner* data tab output) are used to define which data should be extracted.

Observation Filter, or *ObsFilter* for short, can be used to filter BUFR data in two ways. First, *ObsFilter* can be used to select a subset of BUFR messages from a BUFR file, according to time, type, location, etc. The result is still in BUFR format. Second, *ObsFilter* can be used to extract scalar (or vector) values from BUFR messages (from all messages or from a subset) into geopoints, geovectors or CSV format. Geopoints and geovectors are custom Metview ASCII formats for scattered geospatial data. *ObsFilter* icon is shown in Figure 1c and an example plot in Figure 4.

BUFR messages often contain the same data types repeated. For example TEMP observations repeat the same type of data for different pressure levels, or satellite observations repeat the same type of data for different channels. *ObsFilter* is not suited to working with repeated data. Instead, *BufrPicker* can be used to extract several values or repeated values from BUFR messages into geopoints format. *BufrPicker* was first introduced in Metview 4.1.2. *BufrPicker* icon is shown in Figure 1d and an example plot in Figure 5.

Radar measured precipitation data is available, in BUFR format, from members of the OPERA community (Operational Programme for the Exchange of Weather Radar Information, a EUMETNET programme, <http://www.knmi.nl/opera/>). For certain OPERA BUFR files *OperaRadarFilter*, first introduced in Metview 4.3, can be used to extract the precipitation values into geopoints format, for plotting or for processing. *OperaRadarFilter* icon is shown in Figure 1e and an example plot in Figure 6.

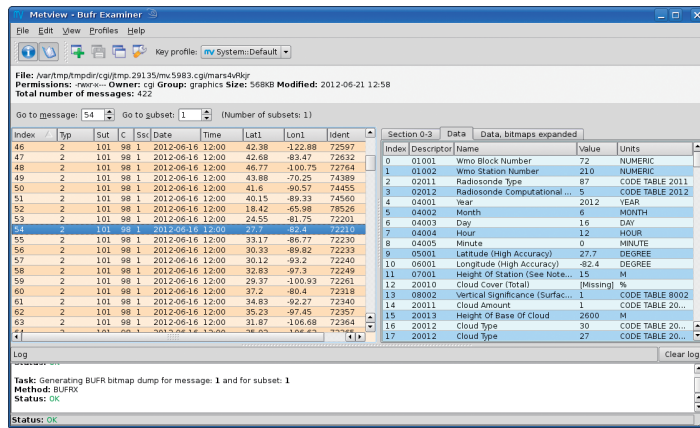


Figure 3 Metview's Bufr Examiner, showing a list of messages (left), the data from the selected message (right), information about the file (top) and a log area which reports any errors encountered when decoding the file (bottom).

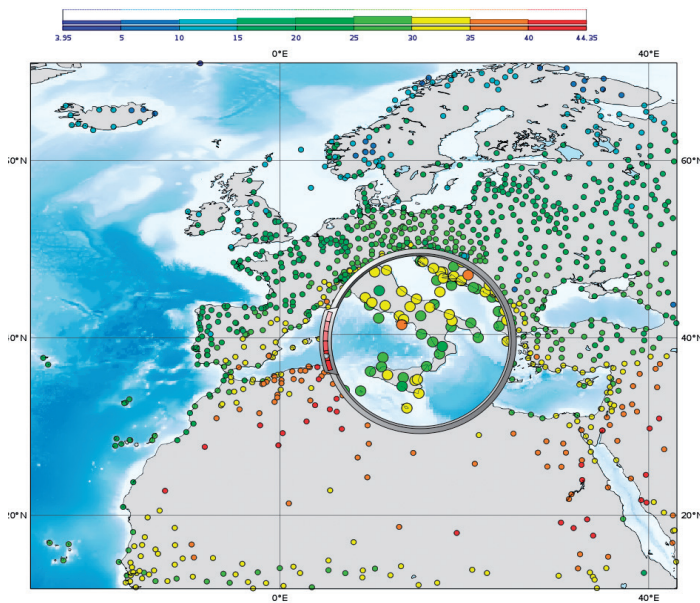


Figure 4 An example plot of temperature values extracted from BUFR data using ObsFilter. The histogram legend shows the distribution of the data.

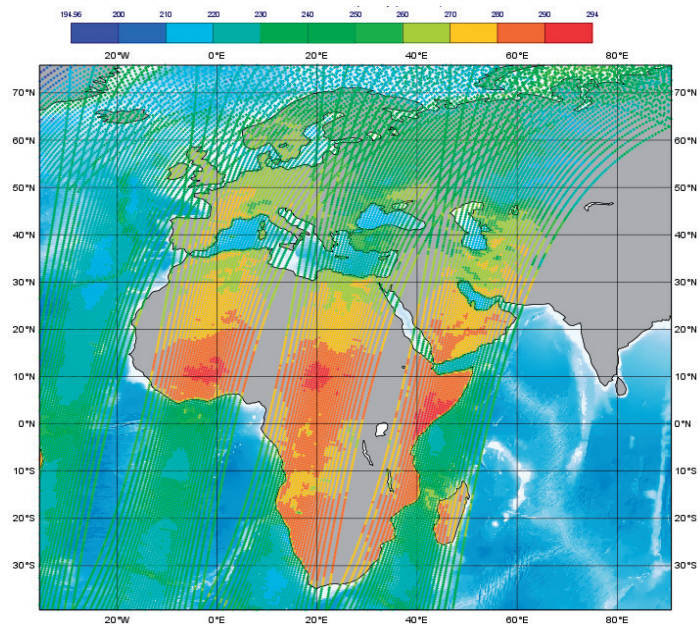


Figure 5 Brightness temperature values from the AMSU-A sensor onboard MetOp-A visualised in Metview 4. The values were extracted from the original BUFR file by Metview's BufrPicker.

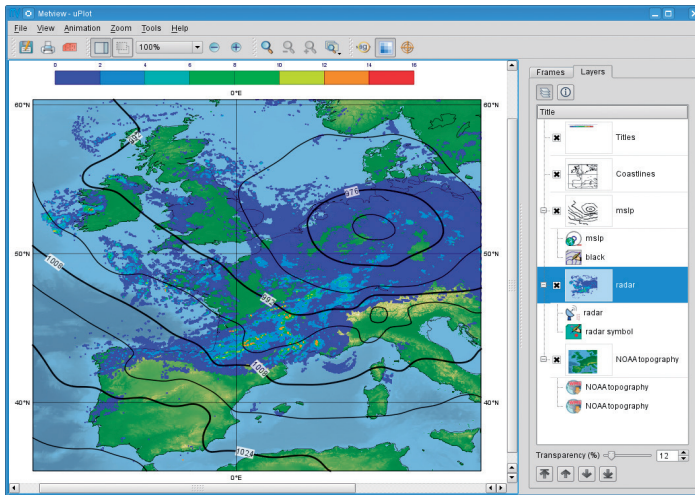


Figure 6 OPERA BUFR image overlaid with the corresponding mean-sea-level pressure analysis from ECMWF's MARS archive.

Processing BUFR data

BUFR messages contain lots of different data, in several possible combinations. Thus it is not possible to compute directly with BUFR messages as such. We have to extract the required individual data items from BUFR messages and do the calculations with the resulting geopoints (or geovector) files. Metview offers lots of functions to operate on these geopoints files.

We can also perform calculations between the resulting geopoints and GRIB data. For such operations, a value for each geopoint location is interpolated from the GRIB field using Metview's own internal routines. The result is a geopoints file with same point locations as in the original geopoints file, but with values replaced by calculated new values. An example product is the departure of a 2-metre temperature field from observed values, shown in Figure 7. The corresponding Metview Macro code is shown in Listing 1.

Data from BUFR messages are normally plotted using numbers or symbols on the geographical location of the data. If the data available in a BUFR file are geographically dense enough, Metview can interpolate the data onto a regular grid and plot the result with isolines, in the same way as GRIB fields. This is done in two steps. First the required data values are extracted into geopoints as before using *ObsFilter*, then the resulting intermediate geopoints data is interpolated using the *Geopoints to Grib* icon, which uses Metview's own internal routines. The result will be a standard GRIB field. Figure 1f shows Geopoints to Grib icon, Figure 8a shows geopoints plotted as symbols and Figure 8b shows the interpolated field plotted with isolines and shading.

```
# retrieve the data
t_analysis = retrieve(
  levtype : "sfc",
  param   : "2t",
  date    : -3
)
obs = retrieve(
  type    : "ob",
  repres  : "bu",
  date    : -3
)
# extract the temperature
# values from the observations
t_obs = obsfilter(
  output : "geopoints",
  data   : obs,
  parameter : 012004
)
# compute the difference
diff = t_analysis - t_obs
```

Listing 1 Macro code to compute the departures of a 2-metre temperature field from observed values

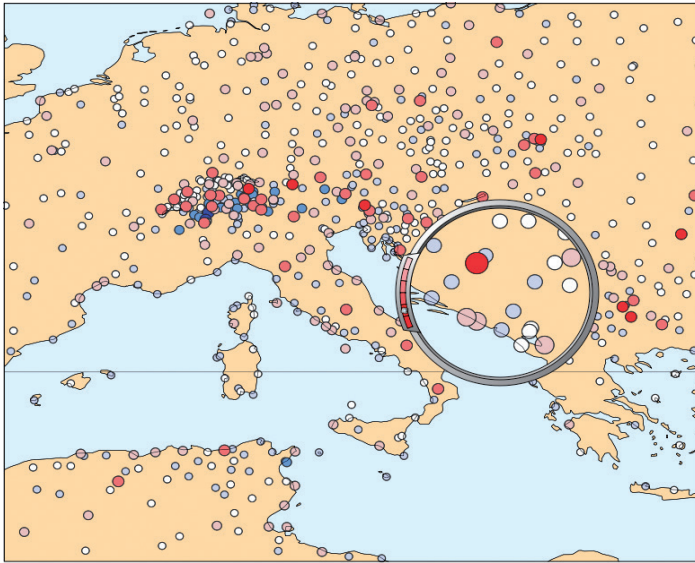


Figure 7 An example plot of departures of a 2-metre temperature field from observed values.

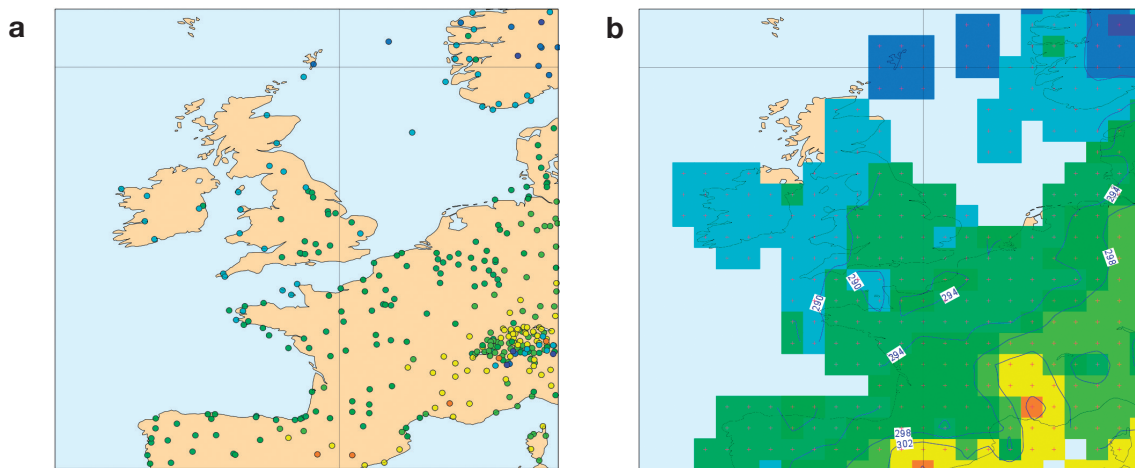


Figure 8 Extracted geopoints (a) plotted with symbols and (b) interpolated and plotted with isolines, shading and grid point markers.

Further information

Users can gain easy access to their BUFR data using the high abstraction level tools available in Metview. More information about Metview can be found at:

- <http://www.ecmwf.int/publications/manuals/metview/index.html>

From the same page, a tutorial is available, describing in step-by-step detail how to handle BUFR data in Metview. It also provides solutions for common problems encountered when processing BUFR data.

© Copyright 2016

European Centre for Medium-Range Weather Forecasts, Shinfield Park, Reading, RG2 9AX, England

The content of this Newsletter article is available for use under a Creative Commons Attribution-Non-Commercial-No-Derivatives-4.0-Unported Licence. See the terms at <https://creativecommons.org/licenses/by-nc-nd/4.0/>.

The information within this publication is given in good faith and considered to be true, but ECMWF accepts no liability for error or omission or for loss or damage arising from its use.