

DB-All.e: Database for punctual meteorological data

ARPA-SIM, Bologna, Italy

DB-All.e is a fast on-disk database where meteorological observed and forecast data can be stored, searched, retrieved and updated. Many modern meteorological applications, like data assimilation, quality control and verification, need complex procedures for input of meteorological data, such as decoding and standardisation, and their organisation in memory. The need to manage a high number of measurement points and the need of long spans of space and time dimensions lead to a disproportionate use of RAM and increase the complexity of programming. This framework allows to manage large amounts of data using its simple Application Program Interface (API), and provides tools to visualise, import and export in the standard formats BUFR, AOF and CREX.

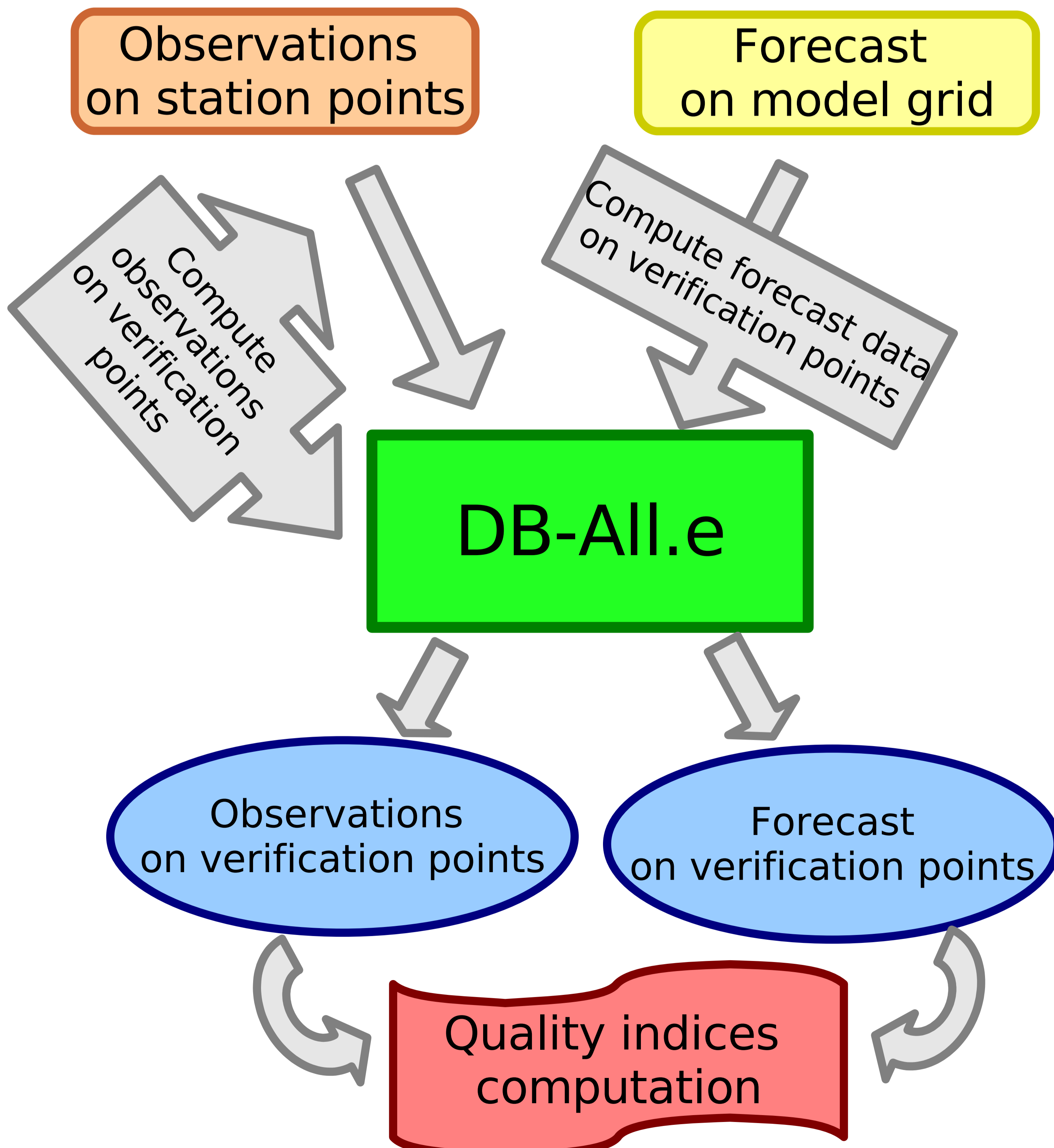
Main characteristics of DB-ALL.e

- ✓ **Free Software** GNU/GPL licensed
- ✓ **Fortran, C, C++** and **Python** API are provided.
- ✓ Developed using **ODBC** programming layer based on **MySQL** relational database.
- ✓ Includes an interactive graphical navigation tool to visualize, search, modify and delete the database contents.
- ✓ To make computation easier, data is stored as **physical** quantities, that is, as values of a variable in a specific point of space and time, rather than as a sequence of encoded reports.
- ✓ Internal representation is similar to **BUFR** and **CREX** WMO standard (table code driven) and utility for **import** and **export** are included (generic and ECMWF template).
- ✓ Representation is in 7 dimensions: **observation network, x, y, z, datetime, timerange, variable**, where x,y are geographic coordinate, z table driven vertical coordinate, datetime the reference time, timerange table driven observation and forecast specification, variable table driven unique definition.
- ✓ Any data may have **attribute**, containing more information linked to the data.
- ✓ Unlimited station information data are possible.
- ✓ **Real, integer** and **character** data type are supported.
- ✓ It is **fast** for both read and write access.
- ✓ It is based on physical principles, that is, the data it contains are defined in terms of **homogeneous** and **consistent** physical data. For example, it is impossible for two incompatible values to exist in the same point in space and time.
- ✓ It can manage **fixed stations** and **moving stations** such as **airplanes** or **ships**.
- ✓ It can manage both **observational** and **forecast** data.
- ✓ It can manage data along **all three dimensions in space**, such as data from soundings and airplanes.
- ✓ Report information is preserved. It can work based on physical parameters or on report types.

It is a powerful and flexible alternative to Netcdf for punctual data.

Example work flow: verification package

Package for the objective verification of "forecast" data (direct model output, post-processed model output, product derived by radar data) against reference ("observed") data (observations on station points, analysis on a regular grid). Forecast and observed data are reported on the same verification points, e.g. by interpolating forecast on station points, or by computing average values on a regular grid, and compared by computing objective quality indices.



Example use of Fortran API

```

c Set the default error handling behaviour
call idba_error_set_callback(0,idba_default_error_handler,1,handle_err)

c Connect to the database and open a working session
call idba_presentati(idbhandle,database,user,password)
call idba_preparati(idbhandle,handle,"read","read","read")

c Set the extremes for the query
call idba_setd(handle,"latmin",latmin)
call idba_setd(handle,"latmax",latmax)
call idba_setd(handle,"lonmin",lonmin)
call idba_setd(handle,"lonmax",lonmax)

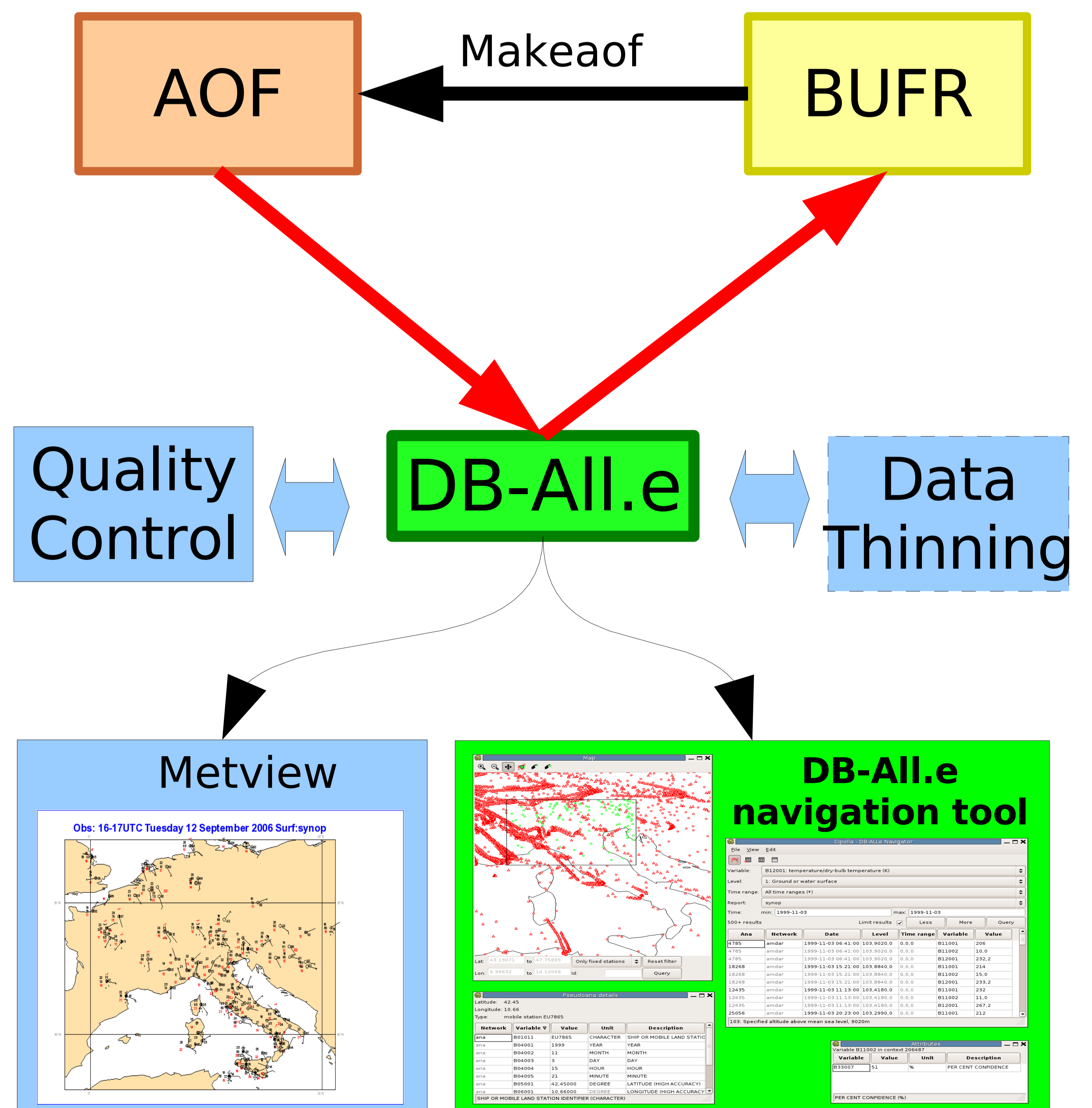
c Query data about the stations in the given area
call idba_quantesimo(handle,N)
do i=1,N
  call idba_elencamele(handle)
c Query name, coordinates and height and print them
call idba_enqc(handle,"name",name)
call idba_enqr(handle,"lat",lat)
call idba_enqr(handle,"lon",lon)
call idba_enqi(handle,"height",height)
print *,name,lat,lon,height
end do

c Get all the data from the stations in the given area
call idba_voglioquesto(handle,N)
do i=1,N
  call idba_dammelo(handle,btable)
c Query report type, level, time range, date and value and print them
call idba_enqc(handle,"rep_memo",rep_memo)
call idba_enqlevel(handle,leveltype,11,12)
call idba_enqtimerange(handle,pindicator,p1,p2)
call idba_enqdate(handle,year,month,day,hour,min,sec)
call idba_enqc(handle,btable,dato)
print *,btable,dato
enddo

c End the work
call idba_fatto(handle)
call idba_arrivederci(idbhandle)
  
```

Example work flow: LM data assimilation

DB-All.e adds a management infrastructure for the AOF (Analysis Observation File) processing that can be applied to data assimilation with LM. Data can be searched, visualised, corrected, deleted, annotated. It is an optimal framework for the development of custom procedures for quality control and data thinning.



Reference:

Milan Dragosavac: *BUFR Reference Manual*, ECMWF
 Format for the Analysis Observation File AOF -- 32 bit version for LM data assimilation
 Guide to WMO Table Driver Code Forms: FM 94 BUFR and FM 95 CREX -- Layer 3: Detailed Description of the Code Forms, WMO

Download site:

<http://www.smr.arpa.emr.it/software/DBalle.html>

Corresponding author:

<mailto:ppatruno@arpa.emr.it>