

# Metview - Recent evolution of ECMWF's meteorological visualisation and computing software

Rogério Bonifacio, ECMWF

## Abstract

Metview is the ECMWF meteorological visualisation and computation software. It has now matured into a highly adaptable, modular package, under the ultimate aim of providing "meteorological desktop publishing" capacities to the operational and research meteorologist. The computational capacity of Metview rests on an easy to learn, high level macro language particularly adapted to meteorological data. Metview has been recently ported to Linux, therefore widening considerably its potential user base by offering much increased performance from a substantially reduced investment. This paper describes the latest Metview features and the experiences and lessons learned during the porting to Linux.

## Introduction

Metview is the ECMWF's meteorological data visualisation and processing tool. Metview can be operated in two modes:

- as a meteorological desktop plotting package thanks to its WYSIWYG visualisation (it can be used for routine production of meteorological charts in an operational environment)
- as a powerful meteorological data processing software thanks to its macro language

Metview has a modular architecture and was conceived as a fully distributed system, its modules being able to run on different machines. ECMWF software for data access (MARS) and graphics (MAGICS) provide its underlying data handling and plotting capacity.

Metview was designed for the UNIX environment and is highly portable within the UNIX world. At ECMWF, Metview was used mainly on SGI systems and now being rolled out on Linux machines, following the porting of its latest version. Elsewhere it is installed and operational also on SUN, HP, IBM and DEC/Compaq workstations. The choice of a Linux platform widens considerably its potential for deployment by being able to run on comparatively very low cost hardware.

Metview was developed under a cooperative project between ECMWF and INPE/CPTEC, Brazil with assistance from Météo-France. It is now used in most members states, Brazil and Australia.

## Recent Evolution of Metview

Since its first release, Metview went through a number of operational versions, all of which were structurally similar. Over the last few years, Metview underwent its first major structural revision that lead to its version 3.

The basic principles of Metview remain the same - the design principle, Everything in Metview is an icon and the operating principle, Every Metview task is a sequence of actions on icons. Icons represent each and every file and directory inside your Metview work space. A number of data types are recognised (GRIB, BUFR, some NetCDF, ASCII data columns and matrices) and assigned a particular icon. There is no data import facility.

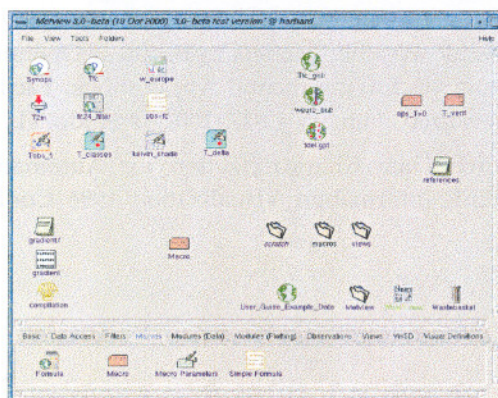


Fig 1 The Metview main desktop (one desktop = one directory). Users can open as many of these as they need. Each and every file in these directories is iconified. Files with recognised structures are assigned specific icons, others either a general ASCII or binary file icon. Icon drawers at the bottom of the desktop window can be filled with user icon templates

Externally, Metview presents a new user interface thoroughly redesigned to give priority to user customisation ability. The greatest changes took place in the visualisation tool, which was completely rewritten from scratch, with a wide range of features being introduced:

- arbitrary layout of different plots in the same window based on a graphical interactive Layout editor
- data plotting is *view* based - a view is a plotting perspective which controls the type of plot (e.g. geographical map, cross-section, ...) and the plot characteristics (geographical projection, cross-section transect coordinates, among many others). Which plot you obtain from a given data unit is determined by the view you have specified - it follows that if there isn't a view for a given type of plot, it cannot be prepared
- drag and drop updating of data, plotting attributes and type of plot (e.g. 2D fields to cross-section, data permitting)
- browsing and animating a stack of fields, including synchronised animation of individual plots
- fully flexible geography manipulation tool including on-the-fly projection changes and windowing

In spite of the revisions and re-structuring, the new version is fully backwards compatible, in that you can use all your older icons and macro programs (see Macro translator below).

Internally, extensive rewriting, streamlining and simplification of the code took place which will make Metview easier to extend and maintain.

Metview was always available for most UNIX platform flavours. A major and now favoured platform was reached when Metview was ported to Linux, details of which are presented in the next section.

### Metview Across UNIX Platforms

Metview is offered on a variety of UNIX flavours - SGI, HP, DEC/Compaq, IBM, Sun, Linux. Over the past couple of years, Metview has been ported to Linux which is the platform currently in use at ECMWF. Most of the issues in porting to Linux had to do with the specifics of GNU software on which the Linux implementation is based:

- The Metview team had already used GNU compilers so the C/C++ compilers presented little problem. However, the GNU FORTRAN compilers have missing features, but problems were solved using the Portland Group F90 compiler (Fujitsu compiler was also tested, but not NAG's)
- Bison's (GNU's YACC) include files had to be re-written
- The echo command (used in Makefiles) had to be adapted as Linux's echo implementation treats the new-line character (\n) differently
- OpenMotif (public domain) worked fine
- OpenGL - Mesa's public domain implementation of OpenGL works fine but Mesa GLU (OpenGL Utilities library) has bugs that crash Metview. SGI OpenGL sample implementation was used to build libGLU. Now this is available as a Linux RPM (RedHat Package Manager) package. Other OpenGL implementations are under testing
- A drawback of OpenGL is lack of printing (PostScript) and font support. We had to develop a custom driver for PostScript and used public domain ones for JPEG and PNG

All in all, the Linux port was quite easy, with the ports to HP and DEC/Compaq Alpha proving the most difficult ones. The Linux port also led to a considerable improvement in Metview's overall performance at a fraction of the cost that would be incurred in other platforms. This widens considerably Metview's potential for deployment.

### New Features in Metview3

#### *Layout Editor*

According to its Meteorological DTP character, users can arrange their plots in any layout they wish - layout design takes place in an interactive graphical layout editor.

This graphical layout editor simulates a sheet of paper (in European standard A-proportions by default, but users can choose any arbitrary size), which you can subdivide at will into layout frames; arbitrary re-sizing and positioning of these frames is available in order to achieve complex layouts with only a few mouse clicks.

Once the layout is designed, users specify which type of plot will be produced in each frame, by supplying appropriate view icons. A finished layout is saved as a display window icon, which when visualised (with a simple mouse click) originates a display window on screen ready for data to be provided and plotted.

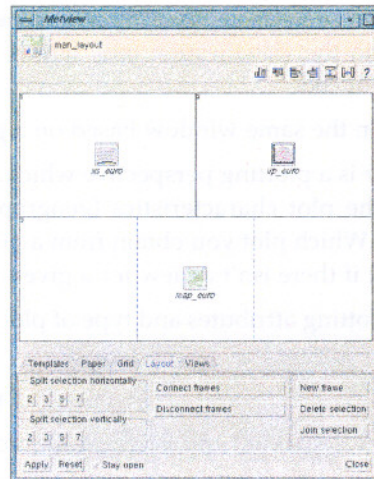


Fig 2 The Metview Layout editor. Users can subdivide and position layout frames at will over a plotting background simulating a sheet of paper. Each plot frame is assigned a view, i.e. a plotting specification detailing the type and characteristics of the plot to be produced

### The Display Window

The overall look and feel of Metview's display tool remains identical to the one presented in the previous workshop. Visualisation is achieved by plain drag and drop of data and visual definition (e.g. a contour specification) icons.

Some data units you may be visualising allow different plots to be prepared - e.g. a multi-level, multi-forecast step set of fields can be visualised as a stack of geographical plots or a number of cross sections or vertical profiles. In this case, you can quickly change from one plot type to another simply by dropping a different view icon in the display window, e.g. drop a cross-section icon to change from whichever plot you have to a cross-section plot (or updating to a different cross-section).

New features include:

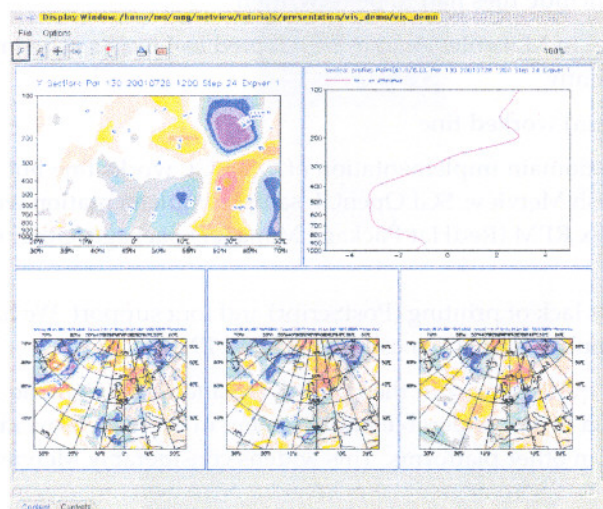


Fig 3 - The Metview Display Window corresponding to the layout of Fig. 2. The same data unit of forecast-analysis differences has been provided to the three layout frames. Because each has a different view, the same data produces different plots (cross-section, vertical profile, map)

### Geography Tool

The display window allows users to change the geographical characteristics of their plots on the fly - through a simple mouse click-right a geography tool is launched which is specific to the plot where you clicked. This tool has different flavours depending on the view in usage for the plot - in the map flavour you can change projection and sub area, in the cross section (vertical profile) flavour you can only change transect (point/area) coordinates; all flavours have a zoom facility to help in selection over small regions.

## Contents handling

Visualisations may involve many different data units and visual definitions. In order to provide users with full control over the visualisation, the Display Window provides access to its component icons (data, views, visual definitions) through the Contents drawer. The Contents drawer is an interactive representation of the visualisation structure which:

- provides the user with a schematic display showing all the icons involved, their relative role and their positioning within the visualisation layout
- allows the user to modify interactively the icons that make up the visualisation, add new icons to the visualisation and replace or remove existing ones

In a word, the Contents drawer offers users a very fine degree of control over the visualisation process. Indeed some modifications to a visualisation may only be possible to do via the contents drawer.

## Applications

Applications in Metview3 work in a more consistent or unified way from those of the previous version. Some applications in Metview2 returned a mix of data and visualisation info, e.g. a cross-section would return the data plus axis details and formatting.

In Metview3 there is now a separation of roles - the new applications return only data and nothing else. All plotting detail is taken care of by the visual definitions and the view.

This also means that you can now save the output of an application. This output may be saved as a GRIB file, when the application returns fields or a geoints file when it returns irregularly spaced data. For other types of output, such as cross section data, the format chosen for output is NetCDF. At the moment there are no plans to make Metview3 fully NetCDF conversant - Metview will simply handle the particular NetCDF structures storing cross section and vertical profile data (among others). Still this enables data generated by Metview to be read/handled by other applications.

## Macro Language

Macro has been extended with the macro equivalents of the new features - there are revised commands for the new plot layout, display window, new output devices. Some functions have been added and a new capacity arose from the fact that now applications return data:

Whereas before you could only visualise the result of a cross-section, now you can take the output of a cross-section application acting on a set of fields and use it as input into suitable functions which take recognised NetCDF structures as input - this allows easy computation of climatological or ensemble averages, variances, etc, of cross-section data.

## Retro-compatibility

Metview 3 is nearly 100% compatible with version 2. Icon structure and usage have not undergone any changes, hence all your previous version icons are still usable. Even the icon of the old visualisation tool when executed will give you an instance of the modern visualisation tool.

Given the changes in the way the visualisation process is specified, there had to be a lot of additions to the macro language - to avoid having to translate all your version 2 macros to the new version, Metview3 includes an automatic translator that seamlessly converts your old macro layout and plotting commands to produce your results in the new visualisation tool. Since the launch of Metview3 this translator has undergone progressive refinements to account for the varied ways in which users had coded their work.

## Enhanced DTP ability

### *Variety of output devices*

So far we focused mainly on the on-screen display of Metview 3. Metview implements what is known as output devices - these are simply the media in which the visualisation is produced. The following output devices are available:

- Screen – visualisation is displayed on screen in the display window tool
- PostScript / printer – visualisation output to a PostScript file or printer
- JPEG/PNG - visualisation output to one of these two graphics formats

## The Import view

For publication purposes, it is sometimes useful to place side by side your own results with those from a non-Metview source. Provided the "external" item is in PostScript/JPEG/PNG formats you can simply display it in your own layout.

This is done using an Import view, instead of the usual data handling/plotting views (such as Map, Cross-Section, etc).

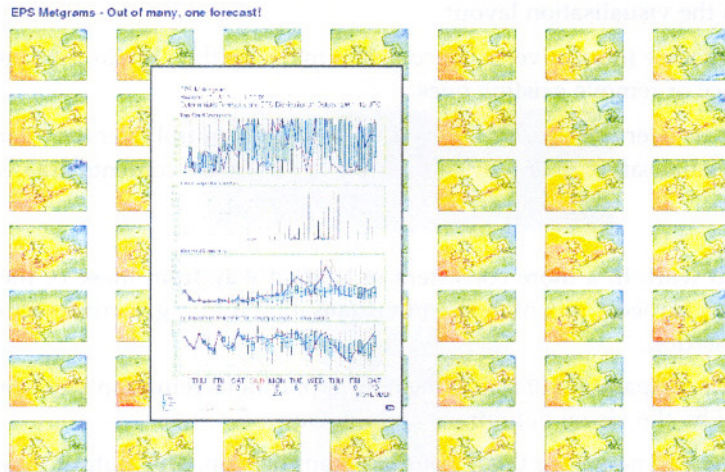


Fig 3 A visualisation composed of an EPS-metgram over a matrix of EPS fields. The EPS-metgram is provided as a PS file, the field matrix generated from retrieved data. The title is generated from a small text file, embedded in a Text view. This facilitates the preparation of plots and pictures for publication without recourse to complex DTP packages.

## The Text view

As a further aid to user plot publication needs, we incorporated a Text view. This view accepts text (e.g. as provided by a plain ASCII file) formatted by means of MAGICs string instructions (for font, style and size specification).

This allows users to incorporate their own text elements into a plot, e.g. a detailed explanatory legend; it is meant to bridge the gap between the plot-only capacities of a visualisation package and the professional publishing standards of conventional DTP packages.

## Help/Documentation

Documentation for Metview3 is an integral part of the installation package. Users can use the Help menu to launch Netscape with the appropriate volume/section of the manual.

For printing purposes the documentation is also available as PDF files; these are prepared with full hyperlink facilities, allowing users to navigate through the document.

## Future Perspectives

At the moment Metview has matured and stabilised and this means it is time to start improving on it again.

There is a continuing routine effort to incorporate new data handling abilities in Metview – new projections and geographical representations, new meteorological data fields. Though these efforts are routine and not visible in terms of new user interface details, they are nonetheless one of the main basis for the continuing usage of Metview - making the processing and visualisation of large and complex data sets as easy and simplified as possible. This brings new data sets closer to operational usage by forecasters and liberates valuable 'brain time' to the researcher.

One such new data type worth mentioning is ODB - ODB is a new format for observation data developed at ECMWF, with an SQL type interface which allows conditional queries to be made. Metview will be fully conversant with observations in ODB format. Currently it is waiting for ODB to be ported to Linux, but the editors and functionality are in place.

A further addition to the desktop publishing dimension of Metview is a straight-to-HTML visualisation output. Users will be able to specify an HTML output device for their visualisation – this means the visualisation will be output to an HTML file which obeys some user (or institution) defined template. This is useful to upload your visualisation products to a web page, a process which can be automated if incorporated into a macro program. This simplifies enormously the work from data archive to dissemination channel.