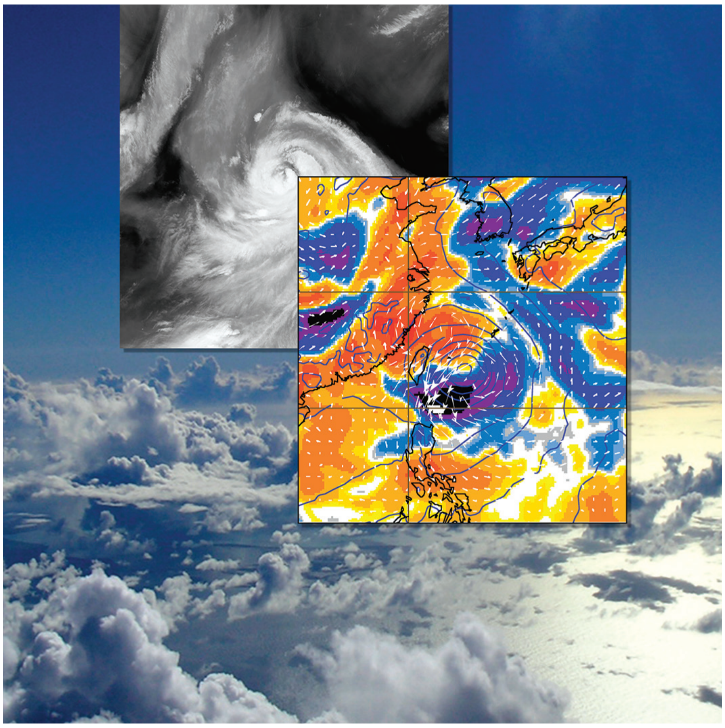
ECMWF Feature article

from Newsletter Number 110 – Winter 2006/07

COMPUTING

The next generation of ECMWF's meteorological graphics library – Magics++



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

doi:10.21957/f0s1fwajmj

This article appeared in the Computing section of ECMWF Newsletter No. 110 – Winter 2006/07, pp. 36-41.

The next generation of ECMWF's meteorological graphics library – Magics++

Stephan Siemen, Sylvie Lamy-Thépaut, Fernando Ii, Iain Russell

Since 1984 the MAGICS (Meteorological Application Graphics Integrated Colour System) graphics library has been at the heart of the plotting of weather maps at ECMWF, its Member States and other weather services around the world. At ECMWF alone, hundreds of maps are produced and printed every day for the MetOps room. Also thousands of plots are produced for the ECMWF web page, and this last number is steadily growing.

MAGICS is not only used directly through its FORTRAN interface, but it is also used to produce graphics inside ECMWF's Metview and Météo-France's Synergie graphical user interfaces.

Why redesign MAGICS?

MAGICS source code has been developed over the past 20 years which brings with it a large legacy. It became hard to extend and incorporate new features. To take advantage of recent developments in software engineering and to clean the code of the legacy the decision was made to rewrite MAGICS. As MAGICS was first designed, techniques such as user interactivity and new media such as the internet were not anticipated. The next generation of MAGICS, called Magics++ to avoid confusion, will make use of these new technologies.

MAGICS was originally designed to provide paper printouts of weather maps, but over the last few years the majority of plots have been produced for the web. An example of how times have changed can be seen in the way text is formatted. Magics++ uses HTML-style formatting that will increase the number of supported fonts (currently three) and improve the setting of special characters (currently non standard).

With Intel's IA-64 and AMD64 appearing as new platforms, MAGICS was in need of a major overhaul because the 64 bit pointers of these platforms were incompatible with the design of MAGICS. The new design is independent of the size of pointers.

Magics++ will also have a new installation procedure which makes it easier to install than the imake based installation of MAGICS. The configure tool has been chosen because it is widely used for many software packages on Unix systems.

Even though the design of MAGICS is changing, the FORTRAN interface used by many programs will be maintained in order to be as backwards compatible as possible with the old MAGICS. However, small changes are necessary (e.g. default coastline colour changed from yellow to black) and newer interactive features may not be used through the FORTRAN interface.

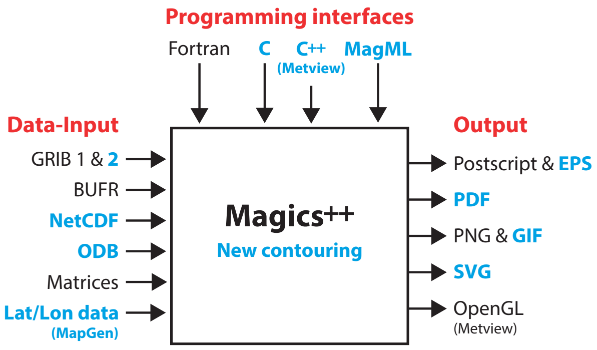


Figure 1 Magics++ as the user will see it. Features new to Magics++ are in blue.

doi:10.21957/f0s1fwajmj

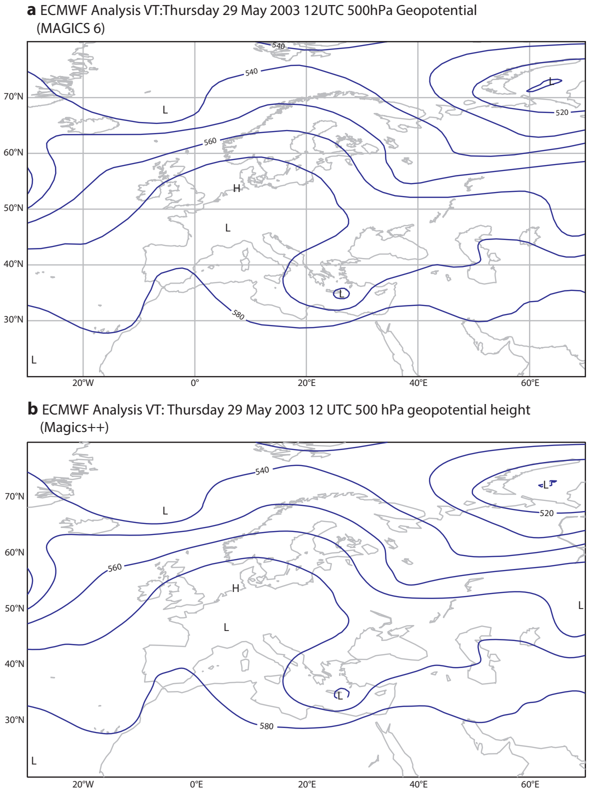
The new architecture

Though Magics++ is programmed in C++ the user should have no problems using the old FORTRAN interface. Figure 1 gives an overview of the old and new features of Magics++.

Run-time information, such as descriptions of observational symbols, is now stored in separate text files. This has the advantage that the information can be changed without a recompilation of the library itself.

Algorithms developed by Hiroshi Akima (ACM Transactions on Mathematical Sciences) will be the basis of the new contouring. The algorithms were chosen because of their flexibility and their support of gridded and scattered data. INPE/CPTEC (Brazil) has successfully incorporated the latest version of the Akima interpolation algorithm in Terralib, their GIS (geographic information system), which is used by Magics++.

Figures 2 shows an example of a field contoured with CONICON of MAGICS 6 and the Akima interpolation in Magics++.



a ECMWF Analysis VT:Thursday 29 May 2003 12UTC 500hPa Geopotential (MAGICS 6)

b ECMWF Analysis VT: Thursday 29 May 2003 12 UTC 500 hPa geopotential height (Magics++)

Figure 2 Examples of a field contoured with (a) CONICON of MAGICS 6 and (b) Akima interpolation in Magics++.

Data formats

Magics++ is one of the first users of ECMWF's new GribAPI library and so not only supports GRIB version 1 but also version 2. NetCDF is now also supported with its own action routine PNETCDF.

Besides complex data formats, such as GRIB and NetCDF, Magics++ is also able to read simple ASCII MapGen data files containing latitude-longitude coordinates. These can be plotted as polylines or polygons. This enables users to easily generate their own overlay of vector information, such as borders or rivers.

Also new is the general support for both retrieving data from databases and plotting it directly in Magics++. In particular ECMWF's ODB (Observational Data Base) is supported and can be directly accessed with Magics++. This is especially challenging because of the large amounts of data involved. These improvements have already led to the development of a Metview icon allowing the access to ODB data, and the plotting and overlaying of the results with other data.

The following is an example of an ODB request in Magics++. The resulting plot is shown in Figure 3.

```
call psetc('odb_database','odb://njord/tmp/odb_data/ECMA.conv/ECMA')
call psetc ('odb_query', 'select lat, lon, obsvalue from hdr, body where varno=$t2m and obsvalue is not null')
call psetc ('odb_latitude', 'lat@hdr')
call psetc ('odb_longitude', 'lon@hdr')
call psetc ('odb_observation', 'obsvalue@body')
call podb
```

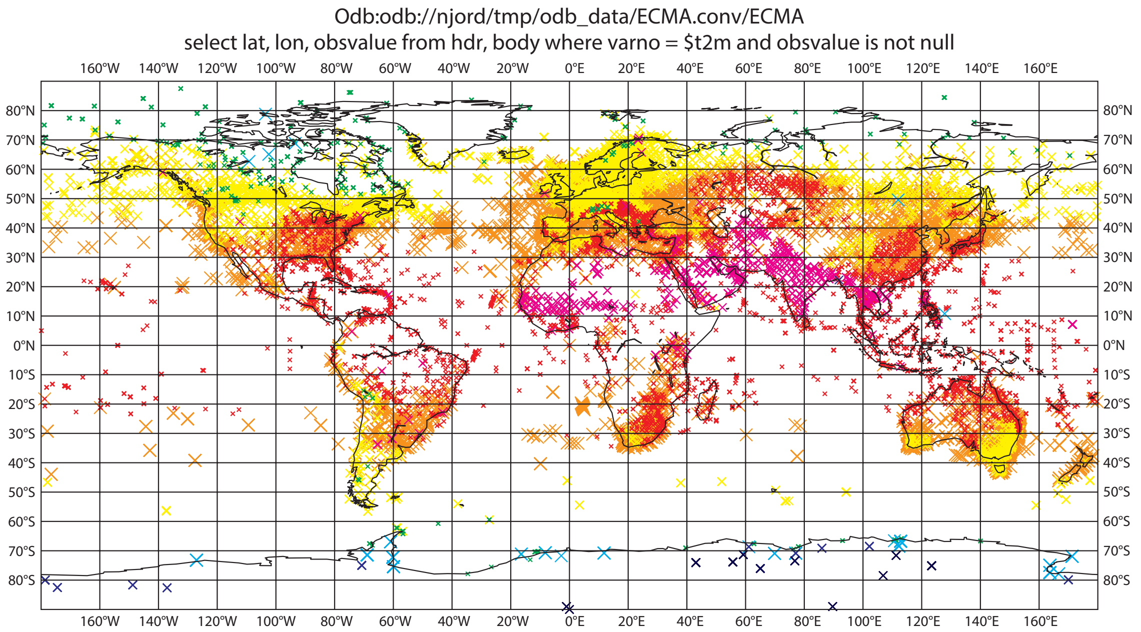


Figure 3 Magics++ and ODB. The title shows details of the request.

Programming interfaces

Until now MAGICS has only been usable through its FORTRAN interface. This will change with Magics++. Beside the familiar FORTRAN interface there is now also a C interface and a descriptive interface called MagML. The latter is based on the Extensible Markup Language (XML) and will be of interest for automatic production of plots, such as for web pages.

The C interface

Magics++ provides a new C programming interface. This is identical to the FORTRAN interface in terms of functionality, but with different action routine names in order to avoid conflicts with some C standard library names.

Example programs and instructions on how to compile C Magics++ programs can be found on the Magics++ webpage. Listing 1 gives an example of a Magics++ C program.

```
#include <magics_api.h>
int main()
{
/* open magics and set the output device */
mag_open ();
mag_setc (“output_format”, “ps”);
mag_setc (“output_name”, “cont_colours”);
/* load the data */
mag_setc (“grib_input_type”, “file”);
mag_setc (“grib_input_file_name”, “data/z500.grb”);
mag_grib ();
/* define the contouring parameters */
mag_setc (“contour”, “on”);
mag_setc (“contour_line_colour”, “sky”);
mag_setc (“CONTOUR_HIGHLIGHT_COLOUR”, “GREEN”);
mag_setc (“contour_label”, “on”);
mag_cont ();
/* plot the title text and the coastlines */
mag_text ();
mag_coast ();
mag_close ();
return 0;
}
```

Listing 1 An example of a Magics++ C program.

The MagML interface

Magics++ offers a new interface, based on XML, to describe Magics plots. This interface is different to other interfaces, which use programming languages to describe a Magics plot. In contrast to programming languages, MagML is a descriptive language with basic support for variables. A plot is described through a hierarchy of objects. Data access is possible by defining an input data file or a database request. Listing 2 shows an example of a MagML description.

MagML offers users a way to describe their plots in a text file. The MagML interpreter program, called magml, reads this text file and generates the plots; this process eliminates the need for compiling a FORTRAN, C or C++ program.

```
<magics version='1.0'>
    <!— Set up our page and subpage specifications —>
    <page format ='a4' orientation = 'landscape' >
    <subpage>
       <!— Describe our view (projection / geographic area)  —>
       <mapview>
          <cylindrical>
            <corners min_longitude = '-20'
                     min_latitude  = '20'
                     max_longitude = '30'
                     max_latitude  = '60' />
          </cylindrical>
       </mapview>
       <!— Describe the coastlines  —>
       <coastlines>
         <coast colour = 'red' />
       </coastlines>
        <!— Describe a new layer (data + visual definitions)  —>
        <layer>
            <grib path = 'data/z500.grb'/>
            <geocontour/>
        </layer>
        <!— Describe our title text  —>
        <text font = 'Times-Roman' colour = 'rgb(0.1,0.6,0.2)'>Z500 Contour Plot</text>
    </subpage>
    </page>
</magics>
```

Listing 2 An example MagML description to read and visualise a GRIB file.

Output formats

In addition to the output formats of MAGICS (PostScript, JPEG, PNG), Magics++ will support EPS (Encapsulated PostScript), PDF, GIF and SVG. The EPS output is geared to be more easily usable for text processors (Word, LaTeX) and submissions for journals. PDF and GIF outputs will reduce post-processing times for web pages because conversions of file formats are not necessary anymore.

Scalable Vector Graphics (SVG) is a vector format defined by the internet standard body W3C and is supported by many browsers directly or indirectly through plug-ins. This vector format not only visualises but also allows interaction with the user. SVG can be used on its own or in cooperation with web services.

As a further support for web pages, in future Magics++ will produce transparent PNGs and JavaScript code describing the dimensions and geographical projections allowing the navigation of raster images. The latter can then be embedded in HTML code to develop web pages which allow users to interact with the display.

New ways of using output drivers

Magics++ introduces new parameters to control the output formats. Although Magics++ supports all output related parameters from MAGICS, the new ones are preferred, and the old parameters are deprecated.

By default, the output format for a MAGICS plot is PostScript. A different output format can be selected by setting parameter OUTPUT_FORMAT to one of those given in Table 1.

The command to set the output to PDF for example is:

```
CALL PSETC ('OUTPUT_FORMAT', 'PDF')
```

Table 1 Magics++ output formats.

Command	Output
PS	single multipage file*
PDF	single file
EPS	multiple files
GIF	multiple files
GIF_ANIMATION	single file
PNG	multiple files
JPEG	multiple files
SVG	multiple files

* This can be changed –
see parameter 'OUTPUT_PS_SPLIT'

Multiple output formats

Magics++ can generate plots in multiple output formats. For instance, a single Magics++ program can create the same plot in both PostScript and GIF formats in a single run. This approach has a performance advantage over running the program separately for each output format as much of the processing is performed only once. The parameter to set is OUTPUT_FORMATS (note the 'S' on the end). The following code shows how to set two output devices.

```
CHARACTER*10     FORMATS
DIMENSION        FORMATS(2)
DATA             FORMATS /'PS', 'GIF'/

CALL POPEN
CALL PSET1C      ('OUTPUT_FORMATS',     FORMATS, 2)
CALL PSETC       ('OUTPUT_NAME',        'multiformat')
```

For a single-page plot, the above code will produce two files: **multiformat.ps** and **multiformat.1.gif**.

New graphical features

Besides supporting new data and output formats, Magics++ also introduces many new graphical features. For example, as shown in Figure 4, Magics++ lets the user now plot maps beyond 360° in longitude with data plotted on all areas.

With more and more plots showing statistical information Magics++ has started to support new techniques of visualising them. For example the box-plots seen in Metgram plots are now also available to users. Through a more flexible design Magics++ allows quick implementations of new displays, such as Taylor diagrams or the new extended Metgrams.

For all these new features there are examples on the Magics++ web page. An article later this year in the ECMWF Newsletter will go into more detail and present more graphical features.

Using Magics++

Magics++ is currently configured with use scripts at ECMWF, but work is in progress to implement modules scripts for the new software configuration set-up. Users who use their own setup should be aware that the Magics++ library has been split up into various smaller libraries and not all parts might be needed. For example, the ODB support is located in a separate library called libMagPlusODB.a and needs only to be linked if the user accesses the ODB.

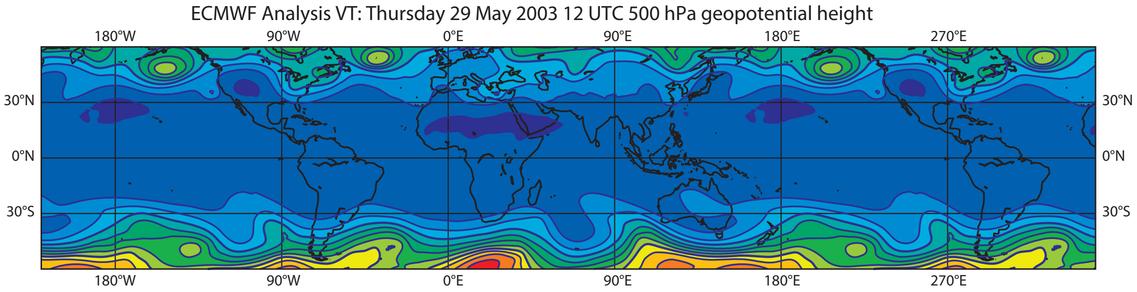

ECMWF Analysis VT: Thursday 29 May 2003 12 UTC 500 hPa geopotential height
180°W
90°W
0°E
90°E
180°E
270°E
30°N
0°N
30°S

Figure 4 Example of a field plot covering more than 360° in longitude.

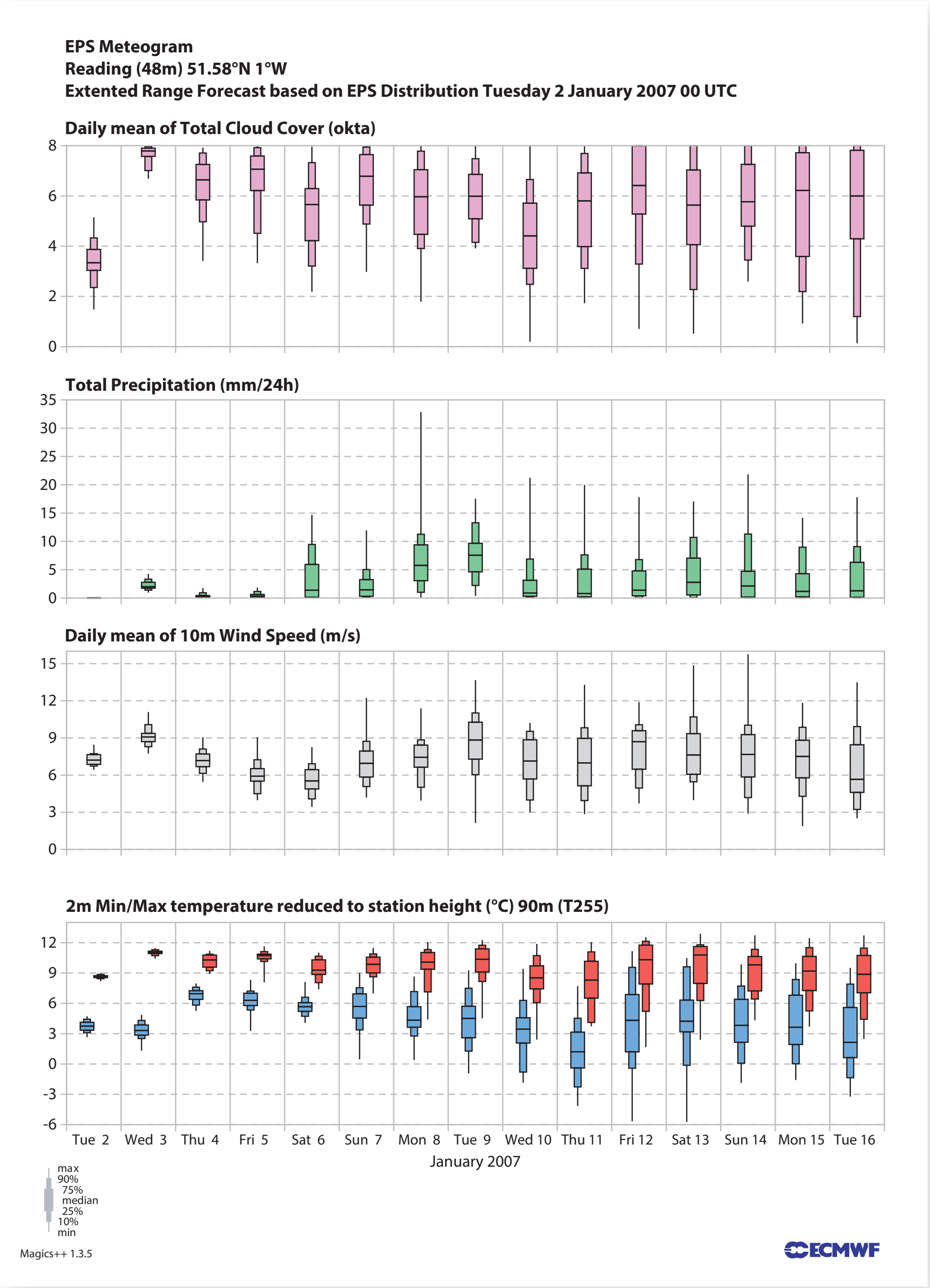


Figure 5 Example of a new 15-day Metgram produced by Magics++.

Future benefits for Metview

Many new features of Magics++ will naturally also benefit Metview which is currently based on MAGICS. The present MAGICS uses the FORTRAN language, wheras Metview and Magics++ are written in C++. Taking advantage of an object oriented language, the internal communication system will be based on objects instead of the current metadata methodology. The initial advantages of this new approach will be straight forward error/warning/text messages handling and simpler program code. This implies higher robustness and better maintainability.

A new visualisation module will be developed to take advantage of all the benefits provided by Magics++. These benefits include:

- Improvements in user interactivity, such as legend and text handling.
- The ability to dynamically toggle certain graphical objects, such as contour labels and maxima/minima.
- Being able to query the properties of graphical objects displayed on screen.

The way ahead

The existing MAGICS manual for the FORTRAN interface has been updated to include the new features and changes since MAGICS 6. Additional information about the C and MagML interface are provided on the Magics++ webpage at: www.ecmwf.int/publications/manuals/magics.

On this page users will find many examples which show how Magics++ can be used.

Magics++ has already been used at ECMWF for the operational Metgrams as shown in Figure 5, the ERA-40 catalogue and is being tested for the service-on- demand project. With the release of version 2.0 Magics++ is now also accessible to users outside ECMWF.

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

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.

doi:10.21957/f0s1fwajmj