Magics++

The next generation of ECMWF’s meteorological graphics library

Stephan Siemen
Sylvie Lamy-Thépaut
Contents

- MAGICS 6.x
- Why redesign MAGICS?
- The new Architecture
- Contouring – Akima
- Data formats
- Programming interfaces
- Output formats
- Lessons learned
- The way ahead
MAGICS 6.x

- Meteorological Applications Graphics Integrated Colour System
- In operational use since 1984
- Used by Member States and many other weather services
- Used in Metview and Synergie
- Hundreds of plots are produced daily for MetOps
- Thousands of plots for the Web with growing demand
Why redesign MAGICS?

- 20 years of code history/legacy
- Take advantage of recent developments in software engineering
- Work practice changing: from paper to web usage (e.g. formatting text)
- New architecture allows more interactivity for users with Metview and formats such as SVG
- GNU configure simplifies installation and allows shared libraries to be built
- MAGICS 6.x way of handling pointers is not working on IA64/AMD64 bit platforms

But ensure backwards compatibility!
CALL POPEN
CALL PSETR ('SUBPAGE_LOWER_LEFT_LATITUDE', 30.0)
CALL PSETR ('SUBPAGE_LOWER_LEFT_LONGITUDE', -30.0)
CALL PSETR ('SUBPAGE_UPPER_RIGHT_LATITUDE', 65.0)
CALL PSETR ('SUBPAGE_UPPER_RIGHT_LONGITUDE', 70.0)
CALL PSETC ('MAP_COASTLINE_COLOUR', 'BLACK')
CALL PSETC ('MAP_GRID_COLOUR', 'GREY')
CALL PSETC ('GRIB_INPUT_TYPE', 'FILE')
CALL PSETC ('GRIB_INPUT_FILE_NAME', 'z500.grb')
CALL PGRIB
CALL PCONT
CALL PTEXT
CALL PCOAST
CALL PCLOSE
END
Magics++ - how the user sees it

Programming Interfaces
- Fortran
- C++
- MagML

Data-Input
- GRIB 1 & 2
- BUFR
- NetCDF
- ODB
- Matrices

Features new in Magics++

Output
- PostScript & EPS
- PDF
- PNG & GIF
- SVG
- OpenGL → Metview
The new architecture (1)

- Object-oriented architecture, where objects have a meteorological meaning (e.g. EPSgram)
- Magics++ objects are defined in XML so that code and documentation are produced automatically
- Standard C++ features such as the STL and well-known design patterns make Magics++ more portable and easier to optimise
- More user control over the resolution of contouring and coastlines
- Allows clearer warning and error messages
The new architecture (2)

- Easier to implement a new data format or add a driver
- XML based formats for descriptions of titles and symbols is convenient and allows users to change these without recompiling the library

**MAGICS 6.x:**
3 & pot temp & potential temperature & 1.0 & 0.0 & K & deg k & 5.0 & 0 & 0 & & PT &

**Magics++:**
<centre code='98'>
<table code='128'>
...
<param code='3'
  short_title='pot temp'
  long_title='potential temperature'
  scaling='1.0' offset='-0.0'
  original_unit='K' derived_unit='deg k'
  vector1='0' vector2='0'
  contour_interval='5.0' />
...

Contouring - Akima

- Successful implementation of Akima algorithms, as presented at last Workshop
- Algorithms developed by Hiroshi Akima - documented in the ACM Transactions on Mathematical Software
- INPE/CPTEC (Brazil) has been implementing a C++ version
- Algorithms handle gridded and scattered data
- Speed and accuracy are configurable by the user, although Magics++ will always choose sensible automatic values by default
Akima Contouring; Relative Humidity, 850hPa; 1st March 2005.
Data formats

- Grib versions 1 & 2 are supported through new ECMWF GribAPI library
- NetCDF support added
- Observations no longer only read from BUFR
- Magics++ supports the retrieval of data through ECMWF’s Observational DataBase (ODB)
- ODB request can be sent from Magics++ and instantly plotted
- The high data volume of the ODB is challenging, especially for interactive usage
- Magics++ can easily be extended to access other databases
Magics++ and ODB example

Odb Access (400000 points)

Odb Database: odb://igraine/bigtmp/odb_data/ECMA.amsub/ECMA

select lat, lon, obsvalue from hdr, body where obsvalue is not null
Programming Interfaces

- Definitions of Magics++ plots can be written in Fortran, C/C++ and MagML (XML)
- Procedural (Fortran/C) and object-oriented (C++, MagML)
- The simple API for Fortran with its parameter concept stays – with a few default values changed
- Aim is to plot meteorological data as simply as possible with meaningful automatic scaling and title
MagML

- XML based format to describe Magics++ plots
- Descriptive, not procedural
- No need for (re-)compilation
- MagMLInterpreter program processes a MagML file
- Interpreter can be easily called in user code
- Description is very different from Fortran API – more closely reflecting the internal structure of Magics++
- Can be integrated into more complex XML request descriptions → see “plot on demand” demo
MagML example (1) – the code

<magics>
  <page format='a4' orientation = 'landscape' >
    <subpage>
      <mapview>
        <cylindrical>
          <corners min_longitude='-20' min_latitude='20' max_longitude='30' max_latitude='60' />
        </cylindrical>
      </mapview>
      <coastlines>
        <coast>
          <colour> grey </colour>
        </coast>
      </coastlines>
      <layer>
        <grib path = '/path_to_data/z500.grb' />
        <contour/>
      </layer>
      <text font = 'Times-Roman'>This is an MAGML Demo Plot
        <colour> avocado </colour>
      </text>
    </subpage>
  </page>
</magics>
MagML example (2) – the output
Output formats

- In a single run Magics++ can produce multiple output formats to save computing time (calculations are only performed once)
- New object-oriented architecture allows easy implementing of new output drivers
- PostScript driver extended to output PDF and EPS
- Magics++ uses GD to produce GIF, PNG and JPEG (much faster than with ImageMagick’s convert)
- GIFs can be animated
### Multiple output formats - example

<table>
<thead>
<tr>
<th>Format</th>
<th>MAGICS 6.10</th>
<th>Magics++</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>+convert</td>
<td></td>
</tr>
<tr>
<td>PostScript</td>
<td>0.51</td>
<td>0.46</td>
</tr>
<tr>
<td>GIF</td>
<td>1.58</td>
<td>0.50</td>
</tr>
<tr>
<td>PDF</td>
<td>1.39</td>
<td>0.88</td>
</tr>
<tr>
<td>All three</td>
<td>2.46</td>
<td>1.12</td>
</tr>
</tbody>
</table>
Output formats - SVG

- SVG is an XML based vector graphics format for the web
- Supports interaction with user to enable navigation through a plot
- Problem is the support of SVG in web browsers (differences in scripting and font sizes)
- The hope of the last workshop that SVG would be much better supported by now has not materialised
- Driver can easily be adapted to support any other future XML based vector graphics format (e.g. MS WVG/Avalon)
Future benefits for Metview

- Magics++ and Metview will both use C++
- More input and output formats
- More interactivity for users
  - toggle contour labels, shading, HiLo
  - change quality of contours/coastlines
- Better display through the new OpenGL driver
  - higher quality text
  - improved import of graphics
- Generate MagML for plot-on-demand (web)
Magics++ in Metview

Point selection

Area selection
Lessons learned

- Using XML for the description of objects and configuration has made Magics++ very flexible and stable
- Choice of STL container can affect the performance a lot
- C++ exceptions are not working with Fortran inside static libraries
- Converting from ClearCase to Perforce for version control was a big improvement
- Limit the number of third-party dependencies
- Benefit of automated test suite with HTML output (various platforms / ECMWF SMS)
The way ahead (1)

- Magics++ was already used for the ERA-40 catalogue (‘PNETCDF’) and is being tested at ECMWF for web requests through MagML
- Nov. 2005: Internal release of Magics++ 1.0 in ECMWF
- Version on ecgate, to test for Member States, follows shortly
- Followed by external webpage

http://www.ecmwf.int/publications/manuals/magics
The way ahead

- Implementing further interactive features in SVG output
- Release of external version second half 2006
- Integrate Magics++ into Metview and extend Metview to take advantage of new features (i.e. interactivity)
- Consider using GIS data in Magics++ (see Working Group)
Overview

Programming interfaces

Fortran  C++  MagML

Data-Input

GRIB 1&2  BUFR  NetCDF  ODB  Matrices

Features new in Magics++

Output

PostScript & EPS  PDF  PNG & GIF  SVG  OpenGL → Metview

10th Workshop on Meteorological Operational Systems