How to plot Arome output

- contents of AROME files
- CHAGAL : the traditional Aladin plotter
- DIAPROG: traditional Méso-NH plotter
- Metview icons and macros
- examples of Metview applications

Contents of AROME files

- Arome Model state = 2 files:
  - **atmospheric 'FA' file**: same format as Aladin, model variables on model levels, spectral fields
  - **surfex 'LFI' file**: same format as Méso-NH, model variables on model grid
- **Post-processing will be needed** to convert into usual forecaster's charts (z500, PV, theta...): FullPos job to convert FA file into GRIB lat/lon files (works, but not fully functional yet)
- model state I/O using GRIB could be developed (global GRIBs are used in IFS at ECMWF)

Contents of AROME files

- **atmospheric 'FA' file**:
  - very similar to an ALADIN historical FA file: read/write using the standard FA/LFI ALADIN library
  - extra variables: 2 NH, 5 microphysics, 1 TKE
  - missing variables: most of the surface (i.e. FullPos of surface and screen-level will not work), the clouds
  - U,V,T,ps are biFourier spectral: convert to/from gridpoint using the TAL spectral transforms (~1 sec/field on a good PC)
  - files include the usual ALADIN extension zone
  - spectral truncation must be linear. (Might go from elliptic to rectangular in the future, for coupling file compression)

Contents of AROME atmospheric files

```
///// LFIOUV - Nom='/home/bouttier/tmp/plotgren/aromatmo.fa'

///// LFIOUV - Unite 53 OUVERTE, derniere Modification OK a 20051118_042533,  559 Articles de donnees,  7845888 mots en tout
#DBUG# b
lslfi:number of articles in file =          559
CADRE-DIMENSIONS  len=           5        # header = geometry info (FA 'cadre')
CADRE-FRANKSCHMI  len=           4
CADRE-REDPOINPOL  len=         250
CADRE-SINLATITUD  len=          18
CADRE-FOCOHYBRID  len=          85
ALD               len=           1
DATE-DES-DONNEES  len=          11
SPECSURFGEOPOTEN  len=       44950
SURFPRESSION      len=       13687        # spectral fields
S001WIND.U.PHYS   len=       13687
S002WIND.U.PHYS   len=       13687
[...]
S041WIND.U.PHYS   len=       13687
S001WIND.V.PHYS   len=       13687
S001TEMPERATURE   len=       13687
S001PRESS.DEPART  len=       13687
S001VERTIC.DIVER  len=       13687
PROFTEMPERATURE   len=       14411   #gridpoint fields
PROFRESERV.EAU    len=       14411
[...]
S001HUMI.SPECIFI  len=       14411
S004CLOUD_WATER   len=       14411
S003ICE_CRYSTAL   len=       14411
S002SNOW          len=       14411
S001RAIN          len=       14411
S004GRAUPEL       len=       14411
S003TKE           len=       14411
S002 CLOUD FRACT  len=       14411
```

Spectral transforms for dummies (like me !)

subroutine readaladinfile ! get dimensions and spectral index table

```
[...open file, get truncations truncx,truncy...]
 ! Decode file cadre record 3 named 'CADRE-REDPOINPOL'
call lficas(iret,ilfilun,cname,ilen,ipos,.false.)
allocate(icad3(ilen))
call lfilas(iret,ilfilun,cname,icad3,ilen)

 if(.not.allocated(ispecini)) allocate(ispecini(0:truncy))
  if(.not.allocated(ispecfin)) allocate(ispecfin(0:truncy))
  ispecini(0:truncy)=icad3(11:11+2*truncy:2)
  ispecfin(0:truncy)=icad3(12:12+2*truncy:2)
[...]
```

---

subroutine esperee(p1d,ksp,p2d,kx,ky) **! call Aladin TAL transform**

```
#include "tsmbkind.h"
use metadata, only : dimx,dimy,exty,truncx,truncy,ispecini,ispecfin,ltransok
! general grid metadata read from FA file header
!integer trunc,truncx,truncy    ! truncation, in x- and y-direction
!integer dimx,dimy,nlev         ! grid size
!integer extx,exty        ! x and y width of extension (=biperiodicization) zone
!integer,allocatable :: ispecini(:),ispecfin(:)
!                        ! initial and final indices of spectral coeffs

implicit none
integer, intent(in) :: ksp,kx,ky
real, intent(inout) :: p1d(ksp)
real, intent(out) :: p2d(kx,ky)
integer nspec,ngptot,jj,ii,jn,jm
integer(kind=jprm),allocatable :: kloen(:)
real(kind=jprb),allocatable :: zspbuf(:,:),zgpbuf(:,:,:)
integer(kind=jprm),allocatable :: kesm0(:)  ! optional
!
interface
#include "setup_trans0.h"
#include "esetup_trans.h"
#include "einv_trans.h"
#include "edir_trans.h"
#include "etrans_inq.h"
end interface
```

```
! Setup transforms & allocate work arrays
print*,'esperee: setting up spectral transform...'
call setup_trans0(kprintlev=0)
if (.not.allocated(kloen)) allocate(kloen(2*dimy))
kloen(:)=dimx
call esetup_trans(kmsmax=truncx,ksmax=truncy,
                  kdgux=dimy-exty,kdgl=dimy,kloen=kloen)

! Optional : get index table & print diagnostics
allocate(kesm0(0:truncx)) !optional
call etrans_inq(kgptot=ngptot,kspec=nspec,kesm0=kesm0)  !optional
print*,'esperee:ngptot=',ngptot,' nspec=',nspec,' dimx*dimy=',dimx*dimy
if(allocated(zspbuf)) deallocate(zspbuf) ; allocate(zspbuf(1,nspec))

! Convert one field (coming from an ALADIN file) from spectral to gridpoint

! reorder Aladin :  file ordering = coeffs per blocks of m, 4 reals per coeff
!          Aladin array ordering = coeffs per blocks of n, 4 reals per coeff
zspbuf(1,1:nspec)=p1d(1:nspec)
ii=1
do jm=0,truncx*4+4,4
  do jn=0,truncy
  if (ispecini(jn)+jm+3<=ispecfin(jn)) then
    zspbuf(1,ii:ii+3)=p1d(ispecini(jn)+jm:ispecini(jn)+jm+3)
    ii=ii+4
  endif
 enddo
enddo
if(ii/=nspec+1) call abor1('esperee internal error on spectral reordering')

if(allocated(zgpbuf)) deallocate(zgpbuf) ; allocate(zgpbuf(ngptot,1,1))
print*,'esperee: calling TAL spectral transform'
call einv_trans(pspscalar=zspbuf(:,:),pgp=zgpbuf(:,:,:))
print*,'esperee: internal gp values=',(zgpbuf(jj,1,1),jj=1,10)

! Load gridpoint array, one row per latitude row
do jj=1,dimy
  p2d(:,dimy-jj+1)=zgpbuf(dimx*(jj-1)+1:dimx*(jj-1)+dimx,1,1)
enddo
```

Contents of AROME files

- **surface 'SURFEX' file**:
  - very similar to a Méso-NH historical LFI file: read/write using the standard LFI ALADIN library
  - lots of variables specific to the SURFEX scheme:

    ntiles x nvariables

  - also includes SURFEX's own postprocessing: screen-level variables, fluxes, cumulated precip...
  - WARNING: field values are undefined where the corresponding tile fraction is zero
  - no extension zone ! Otherwise the surface grid is the same as Arome's computational grid

Contents of AROME surface files

```
///// LFIOUV - Nom='/home/bouttier/tmp/plotgren/arosurf.lfi'
///// LFIOUV - Unite 53 OUVERTE, derniere Modification OK a
      20051118_042517,  237 Articles de donnees,  9928704 mots en
      tout
lslfi:number of articles in file =         237
 STORAGE_TYPE       len=          104           #Header,
      geometry info
 MASDEV             len=          103
 DTCUR%TDATE        len=          105
 DTCUR%TIME         len=          103
 CARTESIAN          len=          103
 GRID_TYPE          len=          142
 LAT0               len=          103
 LON0               len=          103
 RPK                len=          103
 BETA               len=          103
 LATORI             len=          103
 LONORI             len=          103
 IMAX               len=          103
 JMAX               len=          103
 XHAT               len=          329
 YHAT               len=          329
 COVER              len=          357
 COVER001           len=        51631           # physiography
      mapping
 COVER002           len=        51631
 COVER003           len=        51631
 COVER004           len=        51631
 COVER151           len=        51631
[...]
 ZS                 len=        51631             # roughness
      info
 AVG_ZS             len=        51631
 SIL_ZS             len=        51631
 SSO_STDEV          len=        51631
 SST                len=        51631
 Z0SEA              len=        51631
 TS_WATER           len=        51631
 Z0WATER            len=        51631
 TG1                len=        51631
 RI_SEA             len=        51631      # per-tile fluxes & variables: on
      Sea
 RN_SEA             len=        51631
 H_SEA              len=        51631
 LE_SEA             len=        51631
 GFLUX_SEA          len=        51631
 T2M_SEA            len=        51631
 Q2M_SEA            len=        51631
 ZON10M_SEA         len=        51631
 MER10M_SEA         len=        51631
[...]
 RI_ISBA            len=        51631      # per-tile fluxes & variables:  on
      ISBA
 RN_ISBA            len=        51631
 H_ISBA             len=        51631
 LE_ISBA            len=        51631
 GFLUX_ISBA         len=        51631
 BUDC               len=          103
 T2M_ISBA           len=        51631
 Q2M_ISBA           len=        51631
 ZON10M_ISBA        len=        51631
 MER10M_ISBA        len=        51631
[...]
 RI_ISBA            len=        51631       # a flux on several tiles
 RI_WAT             len=        51631
 RI_PATCH           len=        51631
 RI_TEB             len=        51631
 BUDC               len=          103                 <-- non-field item
[...]
 RI                 len=        51631         # per-gridpoint fluxes &
      variables
 RN                 len=        51631
 H                  len=        51631
 LE                 len=        51631
 GFLUX              len=        51631
 T2M                len=        51631
 Q2M                len=        51631
 ZON10M             len=        51631
 MER10M             len=        51631
```

The CHAGAL software

Charming Graphics for Aladin

- The traditional ALADIN's own plotter, managed by Jean-Daniel Gril (jean-daniel.gril@meteo.fr) and distributed with the ALADIN export versions ('PALADIN' tools distribution)
- forecaster-oriented plots
- Efficient and free (based on NCAR library)
- Simple, robust, good for batch plots
- Documentation at http://www.cnrm.meteo.fr/aladin/concept/tools.html

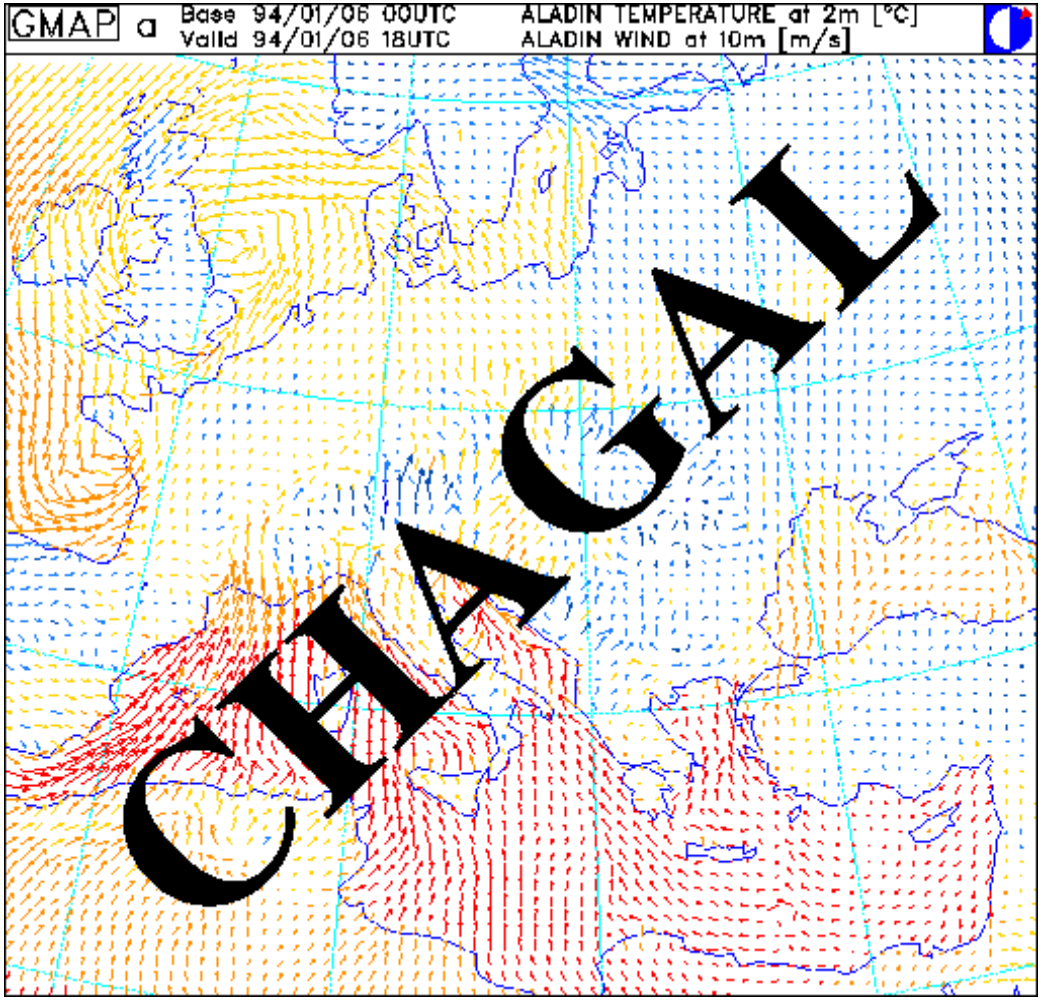
The Aladin/Méso-NH file converters

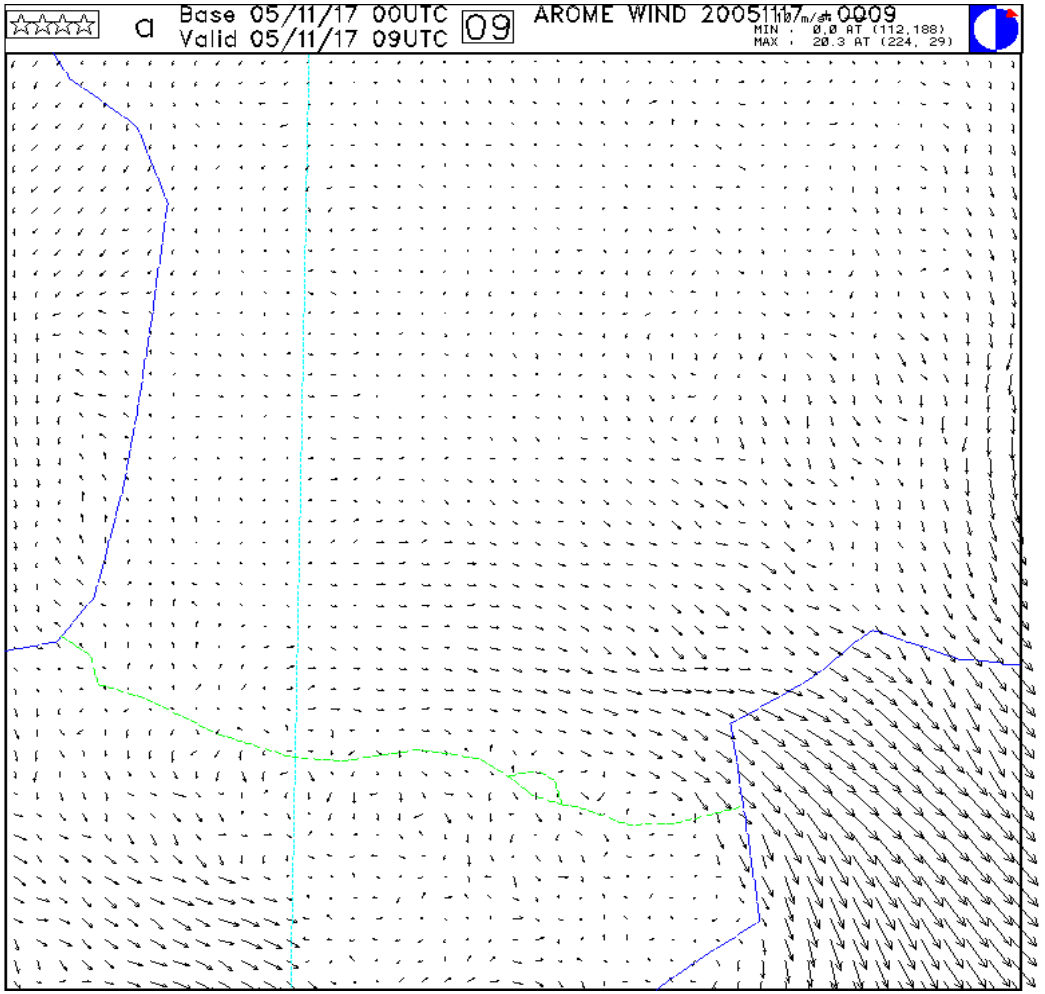
- used for the ‘hybrid data assimilation’= Méso-NH model + Aladin/Arome 3D-Var (convert a MésoNH forecast into a 3DVar background, and a 3DVar increment into a Méso-NH increment)
- used for plotting: plot Aladin/Arome using Méso-NH tools and vice versa
- used for initializing models: dynamic adaptation case studies & model intercomparison (problem of getting comparable plots)
- conversion is handy but usually approximate, not as good as a clean model pre/proprocessing.

Some available software:

- The official Méso-NH prep_experiment tools: (cf. MésoNH doc & support team)
  - to initialize a Méso-NH run from another model (Arpège, Aladin, IFS)
  - does 3D interpolation, accounting for orography change, pressure balance equation
  - technically complex but well maintained
- The ‘convaro’ tools: (cf. GMAP team)
  - quick-and-dirty all-in-one program for the Méso-NH hybrid data assimilation
  - assumes same horizontal grid, does simple vertical interpolation, variable conversion, spectral transform

The CHAGAL software





The DIAPROG software

- Developed for Méso-NH, mostly at Laboratoire d'Aérologie
- Documentation (in French !) at:

  http://www.aero.obs-mip.fr/mesonh/index2.html

- Scientist-oriented plots
- free (based on NCAR library)
- based on a simplified request language
- good for colour plots of uninterpolated model grid values
- for SURFEX file: works
- for AROME FA files: requires conversion to a Méso-NH file (not recommended)

The DIAPROG (and MésoNH !) software documentation

File Edit View Go Bookmarks Tools Help

http://www.aero.obs-mip.fr/mesonh/index2.html

The Mozilla Organiz... Latest Builds

NEW!
Tutorial class
6-9 Feb. 2006

10 Menu

About Us
User Support
User information
Meetings

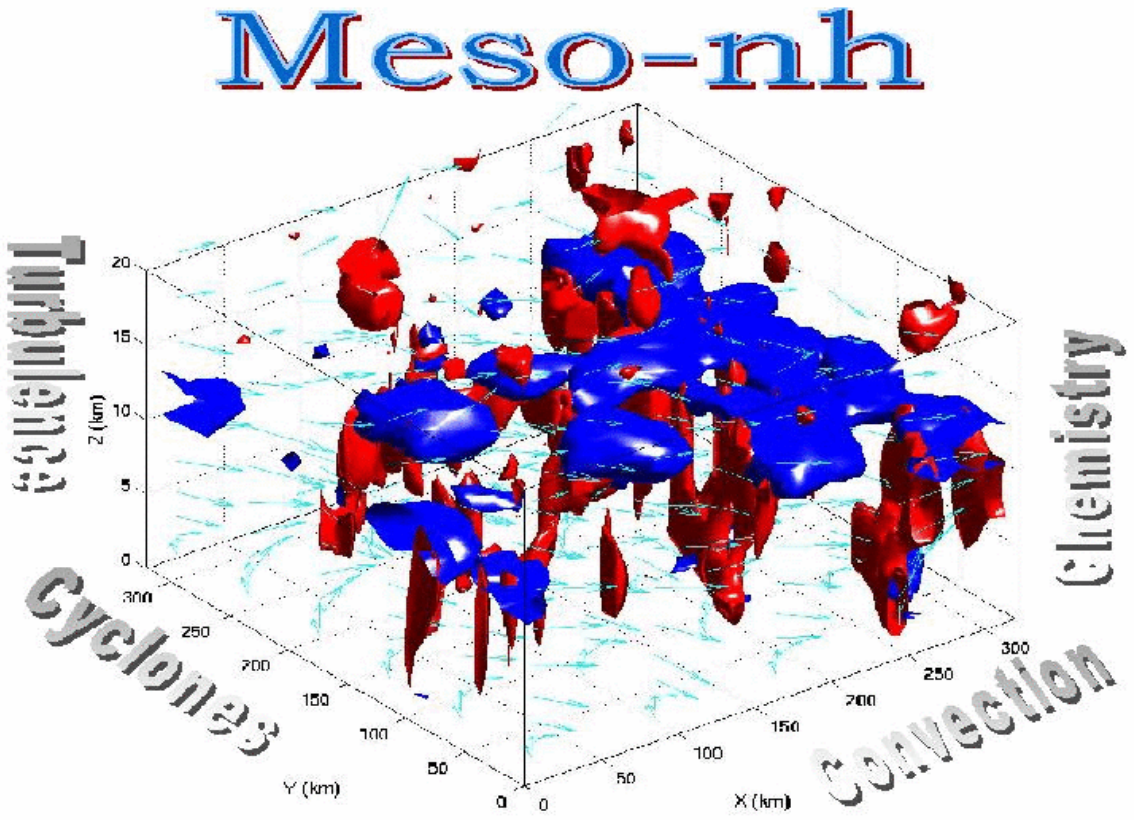
Research Activity
Research Projects
Publications
Examples

Documentation
MesoNH References
Books and Guides
F90 Sources

Tools
Extract ARPEGE/ECMWF
Mail Archive
Meso-NH Statistics

Download
PC LINUX
CVS Libs & Tools
CVS PACKAGE MESONH
Chemistry

Search



Méso-NH is the non-hydrostatic mesoscale atmospheric model of the French research community. It has been jointly developed by the Laboratoire d'Aérologie (UMR 5560 UPS/CNRS) and by CNRM-GAME (URA 1357 CNRS/Météo-France). The model is intended to be applicable to all scales ranging from large (synoptic) scales to small (large eddy) scales and it is coupled with an on-line atmospheric chemistry module
Logo: Méso-nh simulation of a continental thunderstorm with updraft velocity (red), cirrus clouds characterising the upper level outflow (blue), and the wind field ("sky blue" arrows).

Last modified: November 2004. Comments to mesonhweb@aero.obs-mip.fr .

Done

The DIAPROG software

- Example of directives:

```
LINVWB=T
!Ouverture d'un fichier
!******************
_file1_'My_FIC'
!Commandes d'interrogation du contenu du fichier
!*********************************************
print groups
print UT dim proc time
!Ouverture d'une fenetre
!***********************
visu
! Surfaces colorees sans isolignes
!*******************************
lcolarea=t liso=f
! Coupes horizontales altitude 2000,4000,6000 pour les temps de rang 5 et 10
!************************************************************************
THT_z_2000_to_6000_by_2000_T_time5,time10
! Definition pour UT des bornes et increment des isocontours
! RESPECTEZ L'ORTHOGRAPHE de UT (tel qu'enregistre avec MAJ. et min.)
!************************************************************************
XISOMIN_UT=0 xisomax_UT=30 xdiaint_UT=5
nimnmx=1
! Surfaces isobares pour tous les temps contenus dans le fichier
!***************************************************************
UT_pr_980,950
lcoline=t
! Definition d'un profil vertical
!*******************************
nidebcou=10 njdebcou=12 nlmax=2 nlangle=0 profile=1
! Superposition de 3 temps
!************************
WT_PV__t_600_on_WT_PV__t_1200_on_ &
WT_PV__t_1800
quit
```

The DIAPROG software

- List of keywords:

LINVWB=T
_t_ ou _T_ (temps) _P_ ou _p_ (processus) _n_ ou _N_ (masque ou station)

_k_ ou _K_ (CH Niveaux K ou PH ou PV : sel. niveaux) }
_z_ ou _Z_ (Altitudes des CH ou PH) }
_pr_ ou _PR_ (CH Sorties isobares ou PH) } CH
_tk_ ou _TK_ (CH Sorties isentropes ou PH) }
pxt ou _PXT_ (Evolution temporelle d'1 profil horizontal // X
sous forme d'isocontours)
pyt ou _PYT_ (Idem // Y sous forme d'isocontours)

_cv_ ou _CV_ (Coupe verticale) } CV
_pv_ ou _PV_ (Profil vertical) } PV
pvt ou _PVT_ (Profil vertical sous forme d'isocontours f(t))
_rs_ ou _RS_ (RS sous forme d'emagramme (1 profil/1 emagramme) }
_rs1_ ou _RS1_ (RS sous forme d'emagramme (plusieurs profils/1 emagramme)} RS

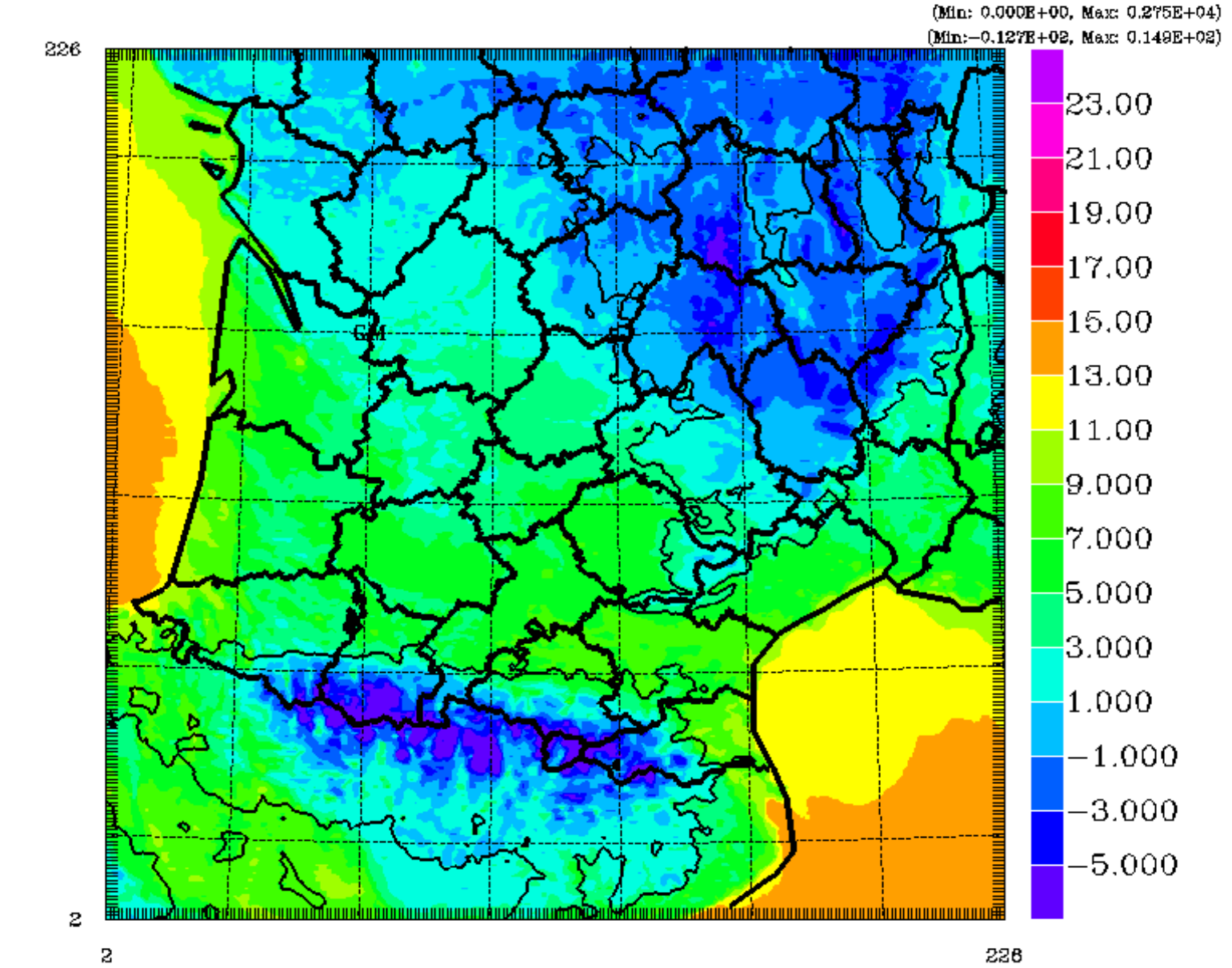
_ft_ ou _FT_ (1 point f(t).)
_ft1_ ou _FT1_ (Id prec. mais 1 seul dessin et calcul des bornes sur l'ensemble des var.)

_on_ ou _ON_ (Superpositions)
+ en representation instantane, ne definir de part et d'autre de _ON_qu'un seul groupe (pas ncessairement le mme),un seul processus (pas ncessairement le mme), un seul temps (pas ncessairement le mme), pour les CH un seul niveau (pas ncessairement le mme), et pour les CV la mme coupe en volution temporelle, mme rgle sauf pour les temps

minus ou _MINUS_ (Diffrence entre 2 champs)
plus ou _PLUS_ (Somme de 2 champs)
+ Diff. et sommes peuvent tre demandes dans une mme directive

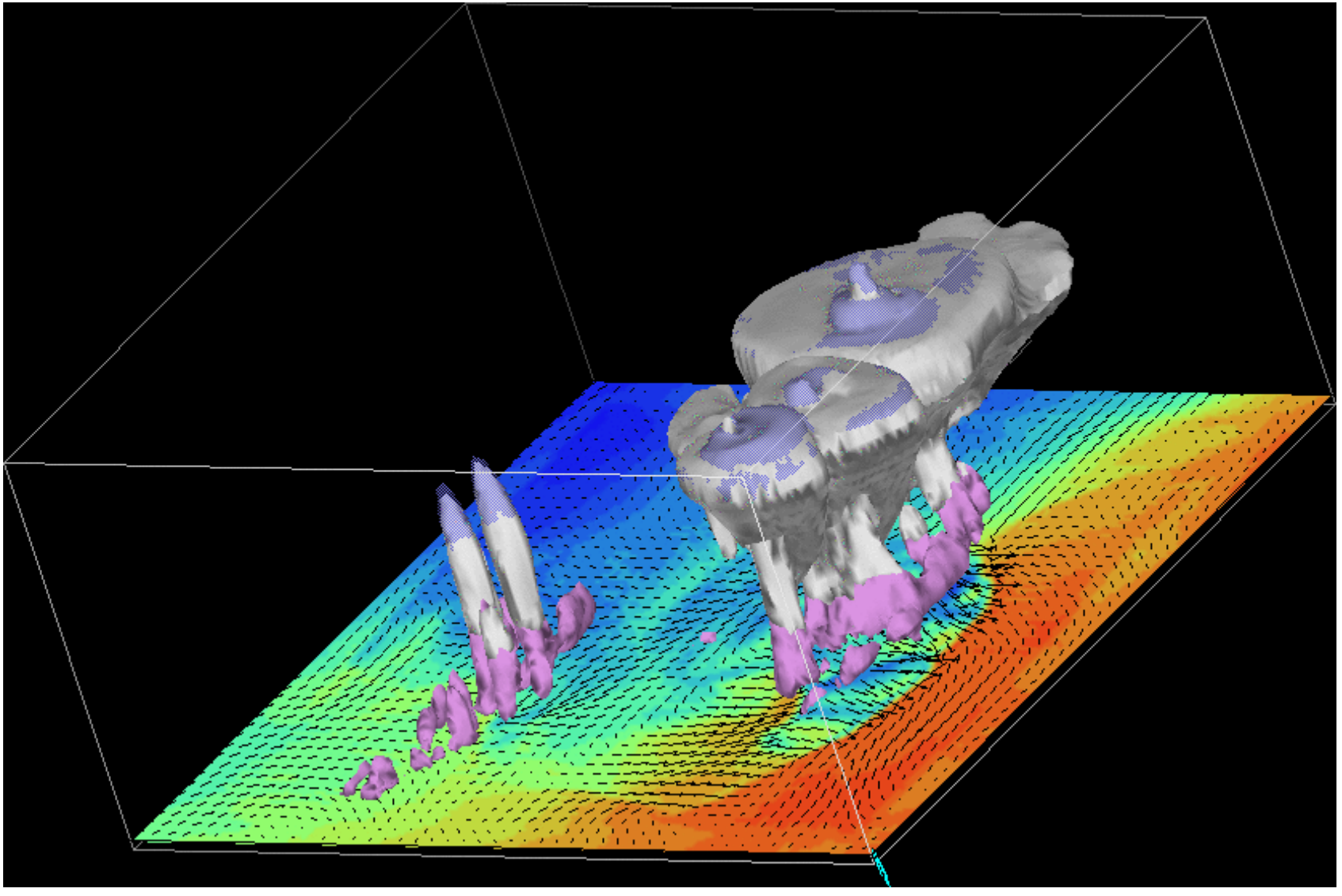
filex'NomFic' (Ouverture d'un fichier avec x=un numro unique)
filex (Dfinition du fichier courant)

The DIAPROG software



Vis-5D (interfaced to MésoNH tools)

usable... with a bit extra work !:



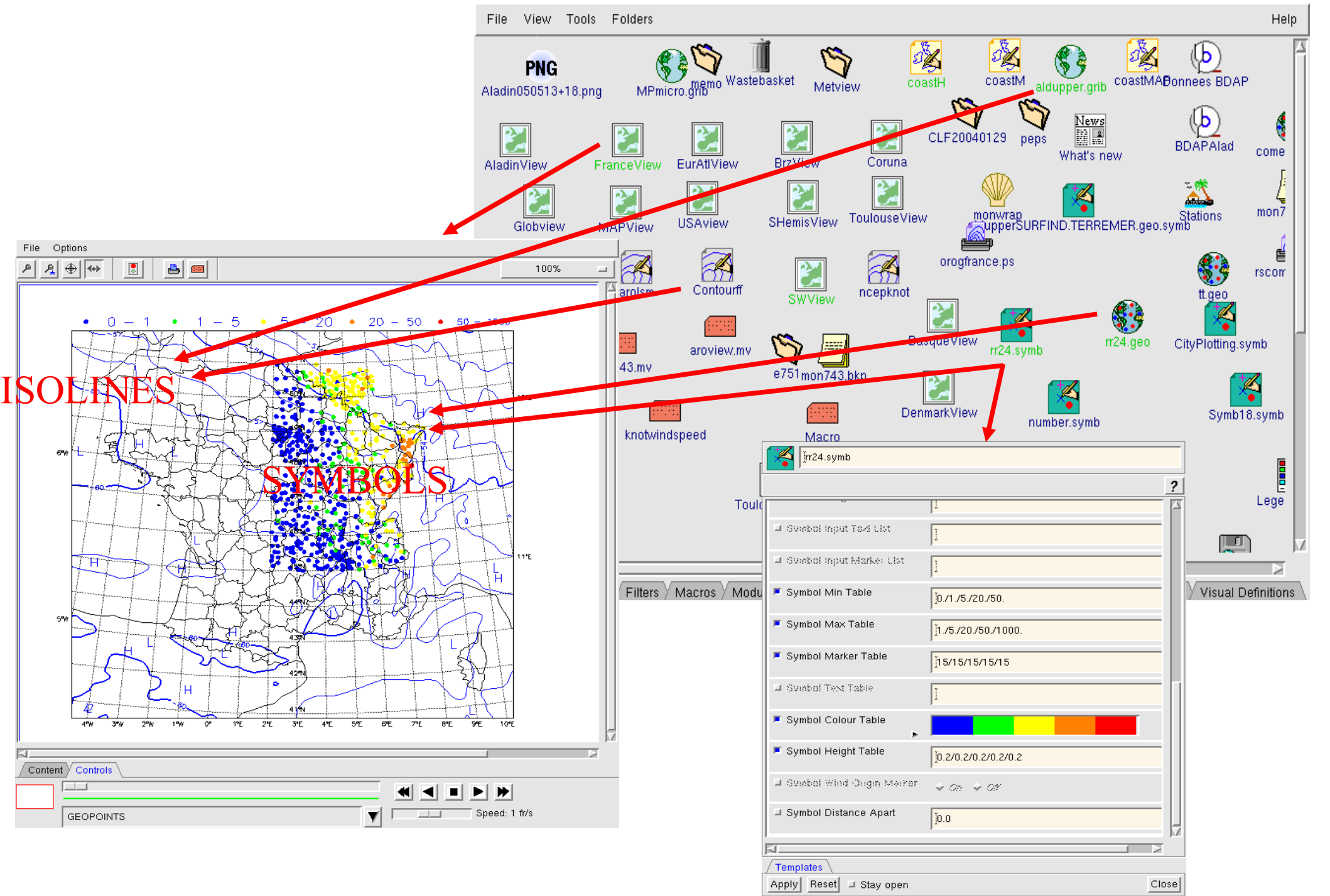
The Metview software

- Developed at ECMWF on top of the MAGICS graphics library, initially for ECMWF own needs
- Oriented for 2D forecaster's plots, with some nice extras for scientists
- Free for ECMWF member & associated states
- To buy (a few €1000, because of internal non-ECMWF library): ask to ECMWF data services
- Documentation at www.ecmwf.int (under 'Manuals')
- Frequent updates (usually interesting)
- Two basic uses:
  - **Metview icons :** Interactive graphical user interface (e.g. PC Linux)
  - **Metview macros:** for batch plotting

The Metview software

- Basic icons used to define plots:
  - data input (GRIB 2D fields, ascii regular or irregular, BUFR obs...)
  - plotting windows (maps, y=f(x), cross-sections, profiles...)
  - plot attributes (isolines, shading, text...) similar to MAGICS
- Drag-and-drop behaviour, good for fine-tuning plots
- a BIG quality: very easy overlaying of multiple fields on same plot

The Metview software using icons



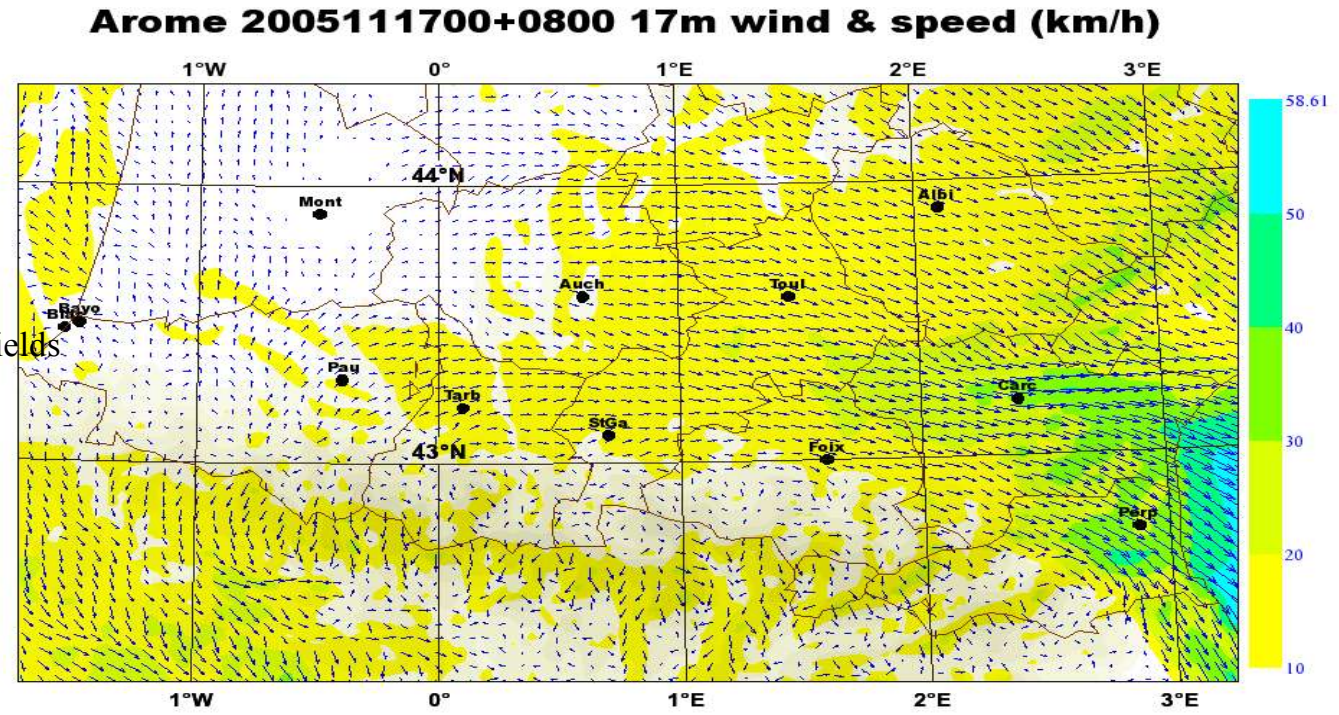
Metview with macros

- automatic conversion of icon plots into a macro that does the same things
- macro language = a complete programming language with math functions, control structures, data I/O, interface to Fortran/C/Perl procedures...
- excellent for doing simple computations on fields and observations: computing averages, scores, parameter conversion...
- write your own macros

Metview with macros

- simple plotting example:

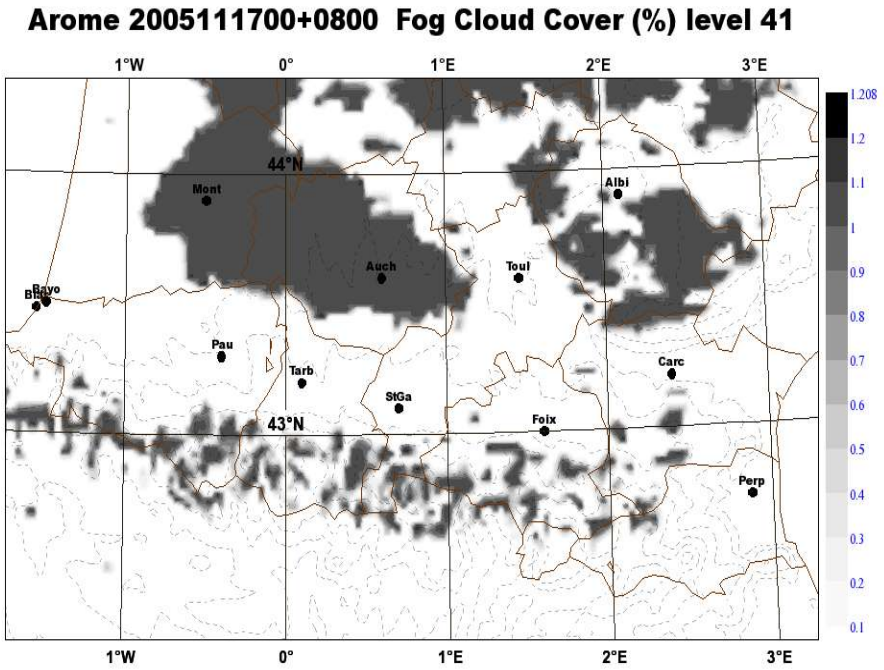
```
# 1. Read GRIB fields from files
path=getenv("METVIEWEX") & "/"
fs=read(path & "ecsample.grib")
print(" this grib contains ",count(fs)," fields.")
wind=read(data:fs,param:["u","v"])
temp=read(data:fs,param:"t")

# 2. Compute wind speed field from wind vector fields
ff=nil
u=read(data:wind,param:"u")
v=read(data:wind,param:"v")
for idx=1 to count(u) do
  ff = ff & sqrt( u[idx]**2 + v[idx]**2 )
end for

# 3. Plot wind arrows and speed isolines
euratl=mapview(map_projection:'polar_stereographic',
            area:[25.,-45.,65.,35. ], map_vertical_longitude:-10.,
            coastlines:pcoast(map_coastline_colour:'black') )
diswin=(plot_superpage(pages:[plot_page(view:euratl)]))

plot(diswin,vent,pwind(wind_arrow_colour:"purple"), temp,pcont(contour_label_height:0.5),
          ff,pcont(contour_line_colour:"red", contour_highlight:"off", contour_min_level:50.) )
```

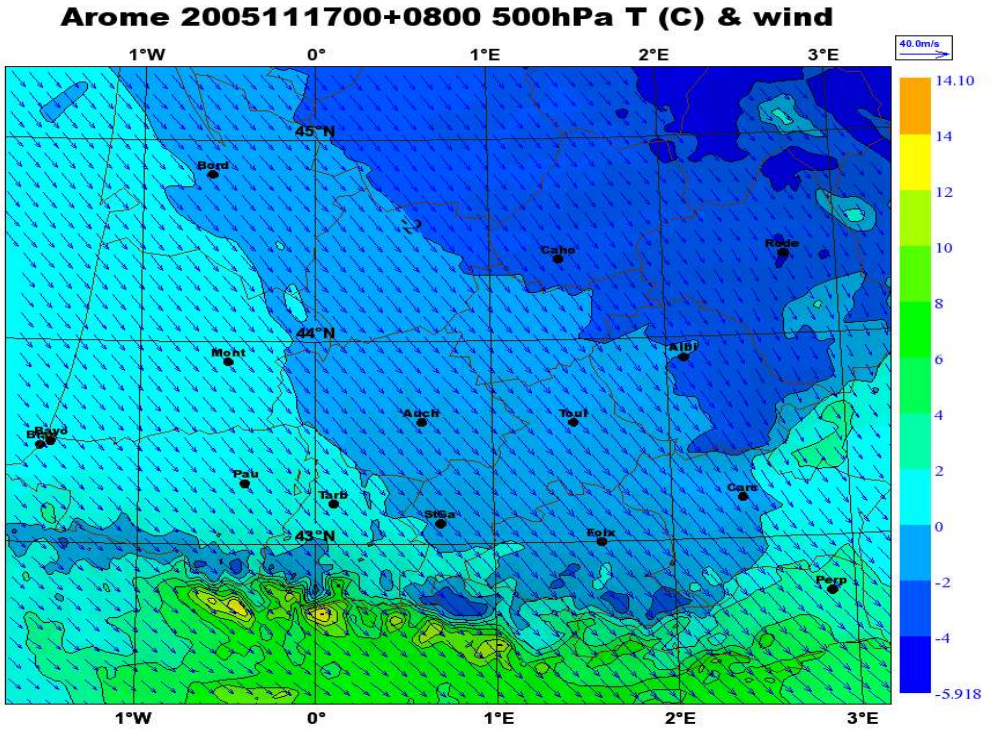


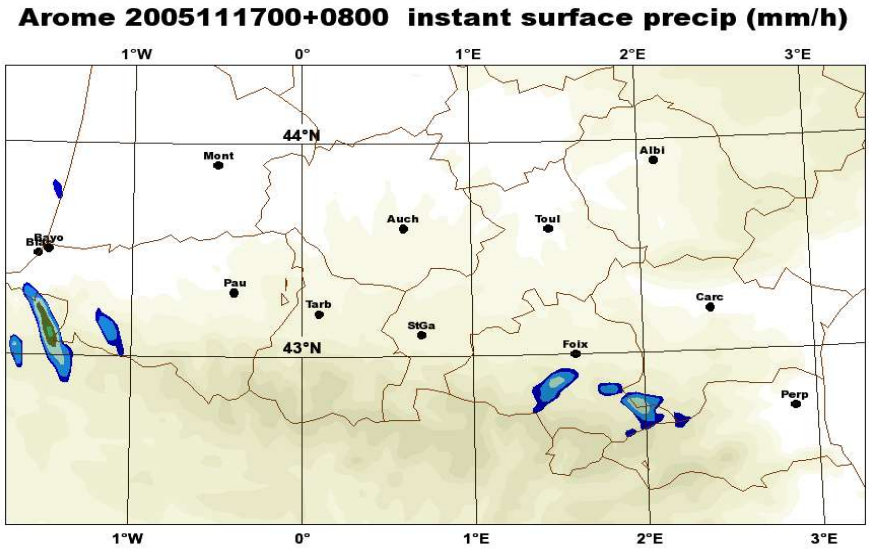
Metview applications

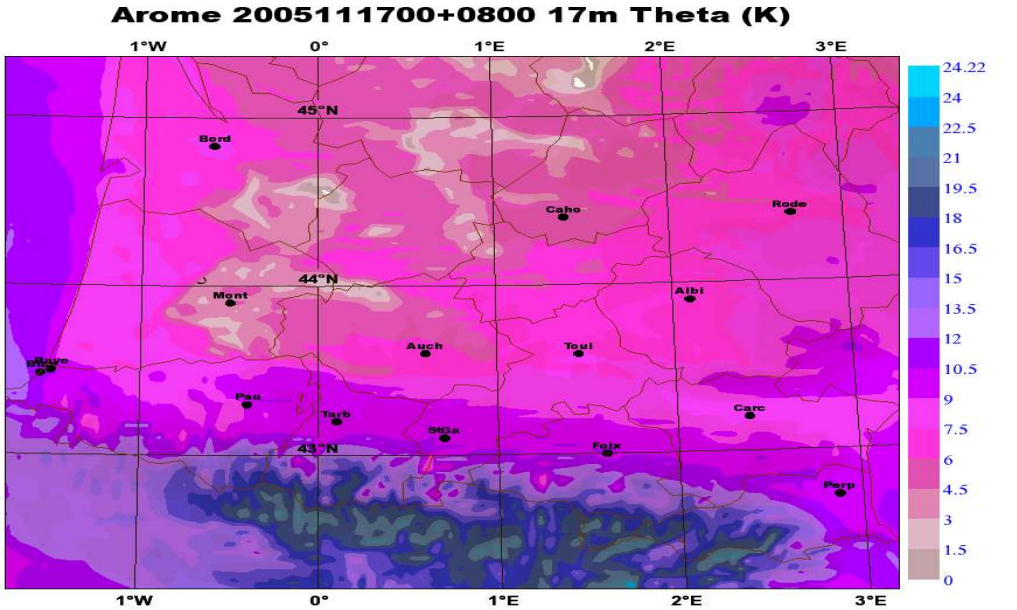
- Metview works best with GRIB horizontal fields on lat/lon grids
- Native Lambert projection is not (yet ?) supported i.e. current Arome fields must be reinterpolated (raw fields can be plotted by cheating - not recommended)
- Vertical p- and hybrid mass coordinates are well supported
- Full-POS produces Metview-compatible GRIBs
- Mandalay software converts ODB into Metview-compatible ascii files ('geopoints') for obs plotting
- Plotting raw model fields requires **field data conversion**, e.g. the lfi2mv program:
  - use FA/LFI libraries to read AROME or SURFEX (or ALADIN) model files
  - call TAL library to convert spectral fields to gridpoint
  - rewrite data as GRIBs or geopoint files
  - plot them using Metview icons or macros

Metview examples



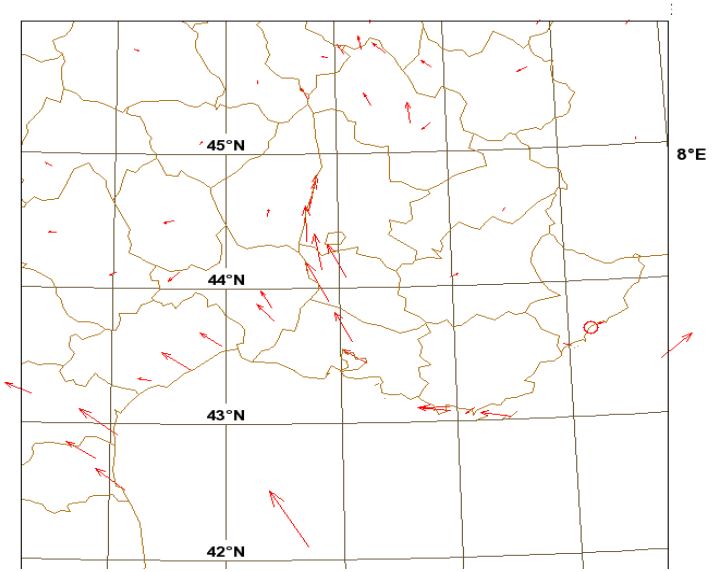




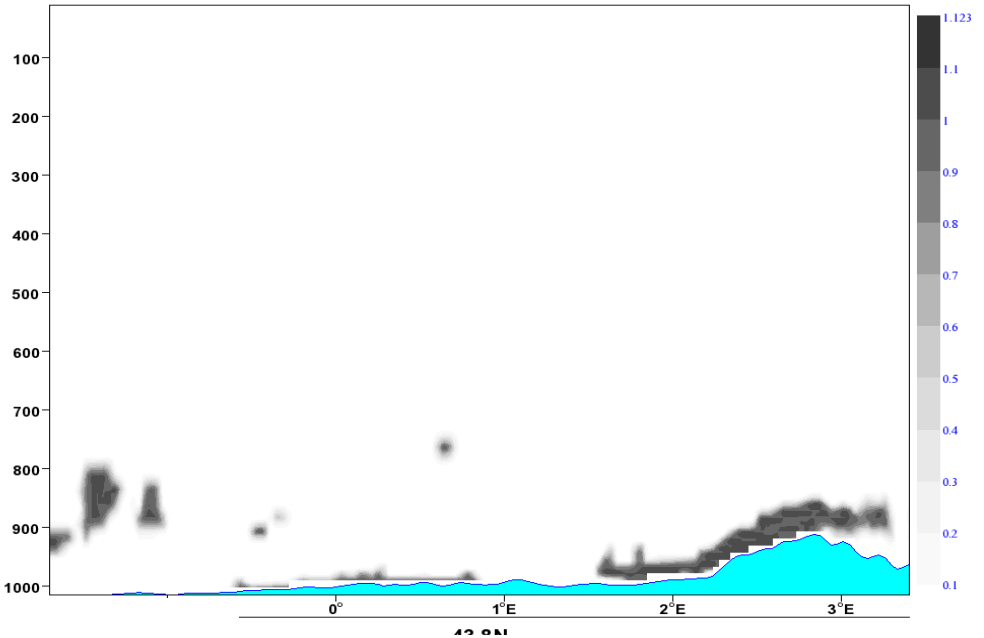


Metview examples

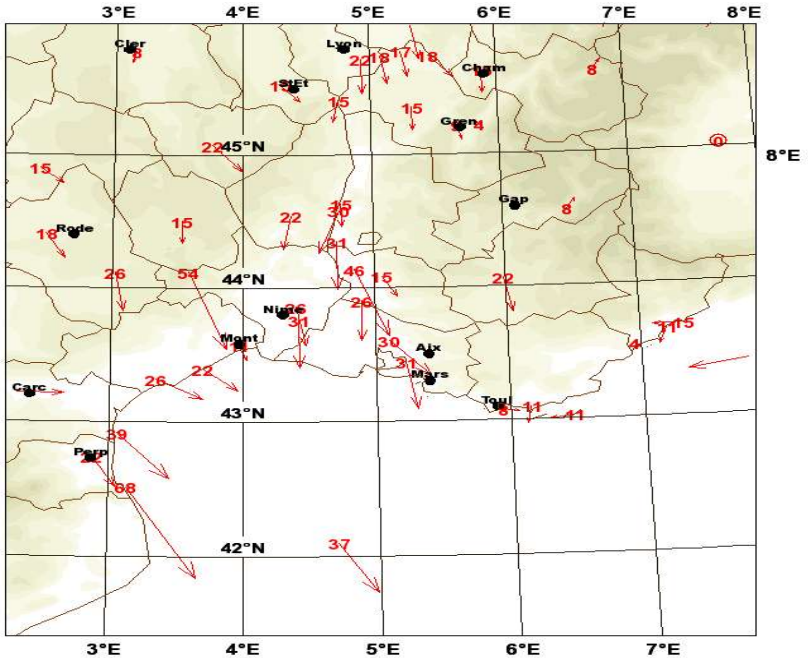
Obs 2005110500+1100 ff obs-aro error



Arome 2005111700+0800 cloud water+/ice (g/kg) EW section



Obs 2005110500+1100 Obs V10m (km/h)



Obs 2005110500+1100 T2m obs-aro error (C)