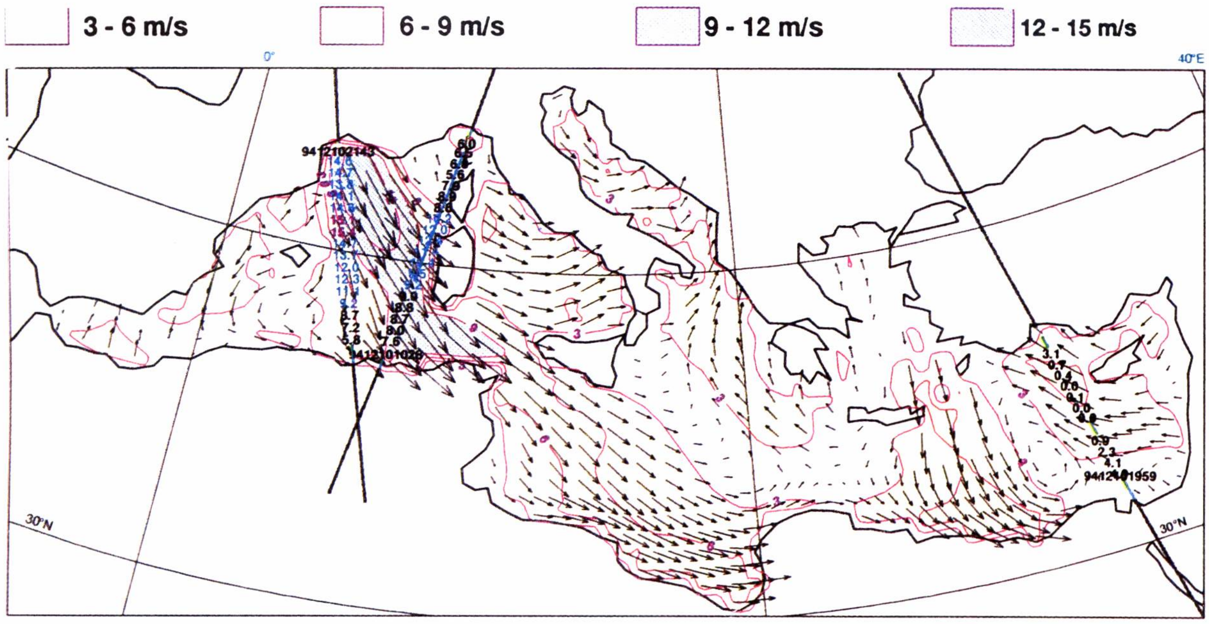
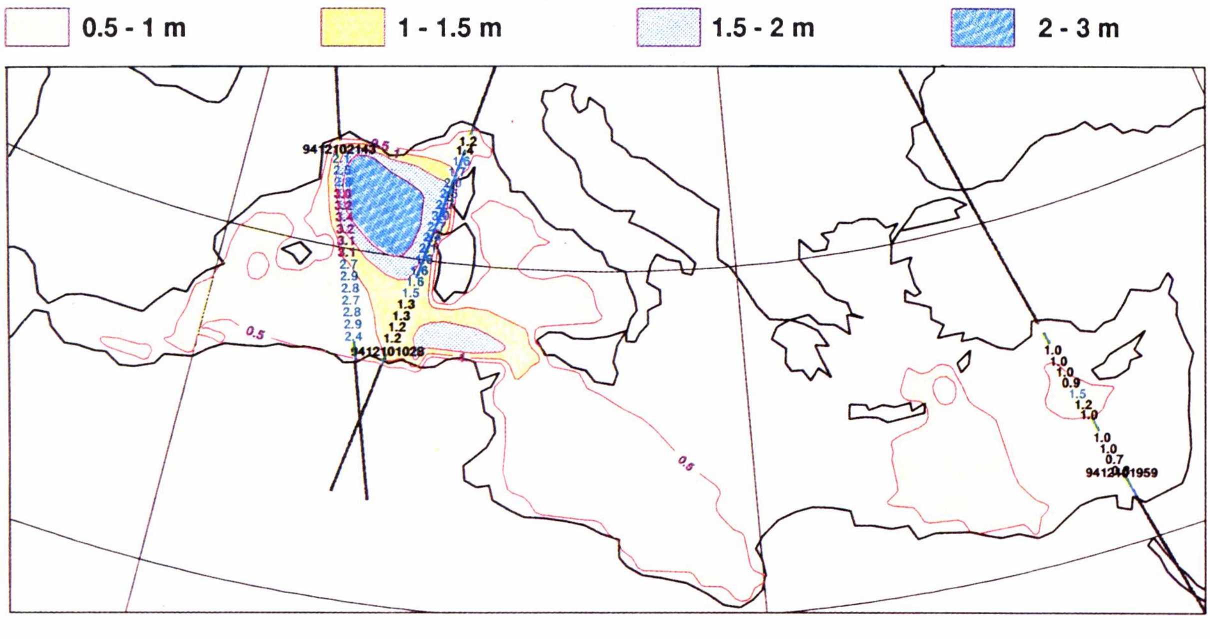
ECMWF Newsletter

Number 68 - Winter 1994/95





Shinfield Park, Reading, Berkshire RG2 9AX, England. Telephone: U.K.(01734) 499000,
International +44 1734 499000, Telex: 847908 ECMWF G, Fax (01734) 869450

European Centre for Medium-Range Weather Forecasts
Europäisches Zentrum für mittelfristige Wettervorhersage
Centre européen pour les prévisions météorologiques à moyen terme

IN THIS ISSUE

COVER: Mistral outbreak.
(top) comparison of ERS-1 altimeter-derived 10m windspeed against ECMWF analyses -the background field is for 10 December at 12:00 - the along. track measurements by the altimeter are plotted using the same colour coding as for the background field. The ascending track towards France is at 21:43 UTC and the descending track from Italy at 10:28 UTC.

(bottom) Comparison of ERS-1 altimeter-derived significant wave height against the WAM model analyses obtained with the fine-mesh Mediterranean Sea wave model regularly fed with the ECMWF 10m wind analyses.

This Newsletter is edited and produced by User Support.

The next issue will appear in Spring 1995.

Telephone No.: CHANGE OF AREA CODE

Please note that the telephone number for ECMWF is now

National	01734 499000
International	+44 1734 499000

This edition contains an extensive description of METVIEW, which should be of interest to most readers. This is the first article publicising this new application, which is the result of a long and intensive programme of development.

Also in this newsletter is the promised follow-up to the article on the CRAY T3D at ECMWF in Newsletter no. 67, in which programming on the CRAY T3D is discussed.

There is a second article on ocean wave forecasting, this time covering work done on areas of the Mediterranean Sea.

Also in this edition are some routine items at this time of year, namely the allocation of computer time for use by Member States, and for special projects, during 1995, and a report on the current status and plans for the computer system.

* * * * * * * * * * * *

CHANGES TO THE OPERATIONAL FORECASTING SYSTEM

Recent changes

Since 6 December 1994, the 1D-VAR system for the processing of cloud cleared radiance data received from the NOAA satellites has been used in the Southern Hemisphere and the Tropics in addition to the Northern Hemisphere, to derive temperature and humidity data for the analysis using a variational inversion technique.

Planned changes

Pre-operational trials are in progress on a 3-d variational analysis system.

Changes to the model physics are under test. They include a new cloud scheme with prognostic equations for liquid water content, use of mean orography and new subgrid orography parametrisation.

- Bernard Strauss

* * * * * * * * * * * *

OCEAN WAVE FORECASTING IN THE WESTERN MEDITERRANEAN

Since July 1992, ECMWF has been producing operational wave forecasts up to 10 days for the globe, with a 3° resolution until August 1994, and with a 1.5° resolution thereafter. Wave forecasts have also been produced for the Mediterranean Sea, up to 5 days, with a finer resolution of 0.5°. In the global model the altimeter-derived significant wave heights from the ERS-1 satellite have, since August 1993, been assimilated using the method by Lionello et al (1992). For the Mediterranean Sea, there is no assimilation of wave data in the wave model.

The Mediterranean Sea is known for a high spatial and temporal variability of the wind due to the complex orography of the periphery and of the islands. Strong winds such as the Mistral from the Rhone Valley are generated by channelling in valleys and many local effects occur. Until recently, this variability was not obtained with operational atmospheric models, mainly because their resolution was not fine enough to approach the orography of the many mountains surrounding the Mediterranean Sea. With the development of limited area models and with the increase in resolution of global atmospheric models, the accuracy of model surface winds has improved and has led to the operational production of wave forecasts in the Mediterranean Sea.

The main facts governing the accuracy of ocean wave forecasts are directly related to the physics and dynamics of ocean waves and to their modelling. Waves acquire energy from the local wind. this depends on the background sea state and on the local surface wind intensity. The winds for the wave model are provided by the atmospheric model. The complex physical mechanisms involved in the modification of the local wave pattern are approximated by the wave model. Both the accuracy of the atmospheric model surface winds and of the wave model physics have an impact on the final accuracy of the wave forecasts. Waves also propagate. This implies a propagation scheme and requires a proper description of the propagating area, to obtain correctly the masking effects by coasts and islands and, in shallow water, a proper description of the bathymetry. In the WAM model version implemented at ECMWF for the Mediterranean Sea (WAMDI, 1988), a deep water assumption is made for the whole area. This assumption is valid when the wave length is short compared to the depth. With a 0.5° resolution, this is nearly always verified in the Mediterranean Sea, because long waves do not often generate due to fetch limitation and because there is a steep increase in depth along most of the Mediterranean coast.

To summarise, the accuracy of ocean wave forecasts depends on the accuracy of the model winds, of the wave model, and of the wave model grid. The relative importance of these three factors depends on the spatial and temporal scale of the meteorological phenomena to be taken into account. As both the temporal and spatial scales of meteorological phenomena in the Mediterranean Sea are small, each factor plays an important role. This makes the analysis of verification results in the Mediterranean Sea more difficult because many interwoven possible causes of errors need to be separated. This task is especially challenging considering the scarcity of the observations.

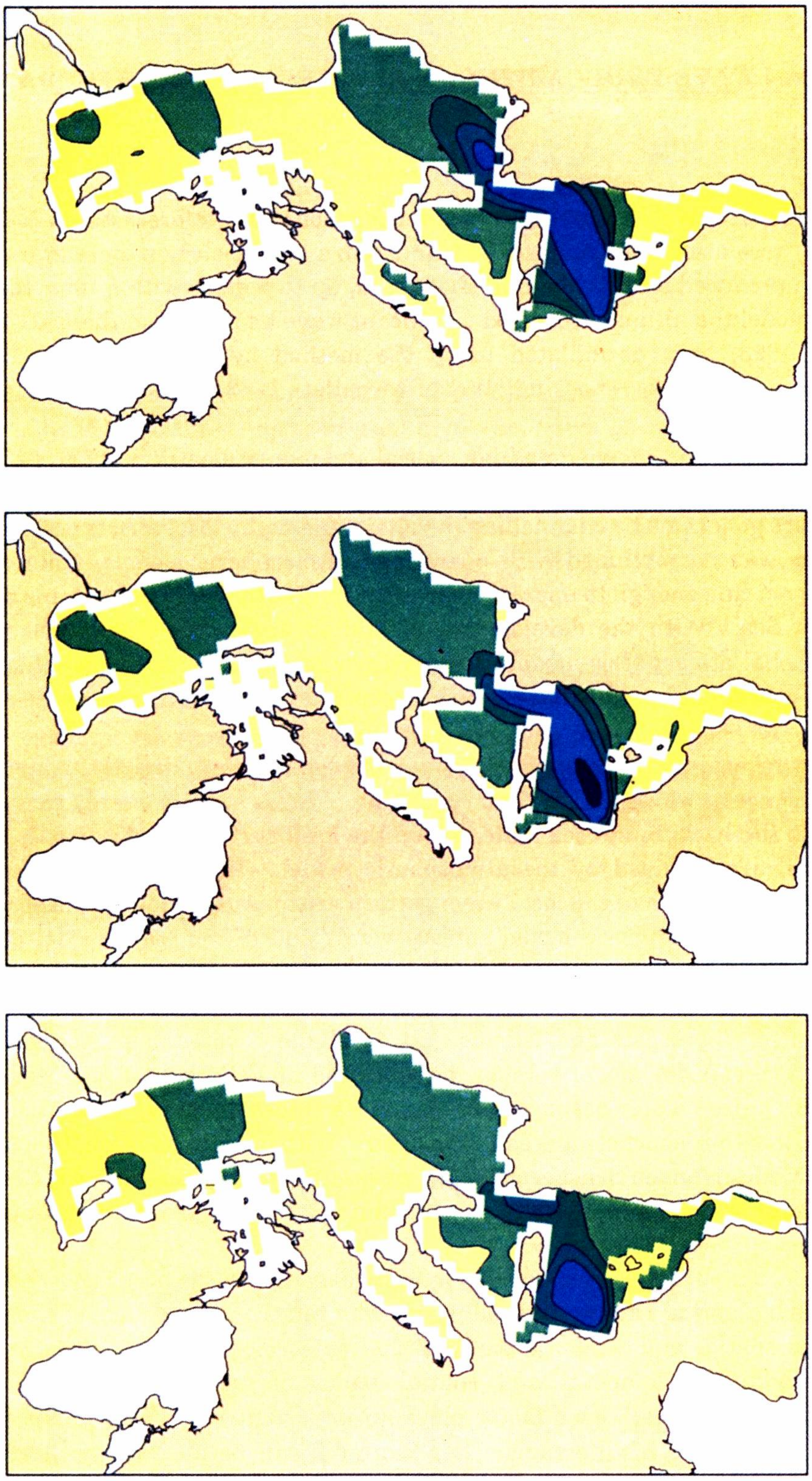


Fig. 1 Sequence of significant wave height fields
Top: 10 December 1994, 12:00 UTC
Middle: 10 December 1994, 18:00 UTC
Bottom: 11 December 1994, 00:00 UTC

During the last decade, many efforts have been made to improve the observational network over the Mediterranean Sea with the extension of a buoy network, mainly by Italy and Spain, and the development of satellite observations, as for example by the altimeters from the Geosat, the ERS-1, and the Topex/Poseidon satellites. Altimeters, which principally observe the geoide, also observe the 10m wind speed and the significant wave height. Because measurements are only at the nadir of the satellite, one satellite only provides a poor daily coverage over the Mediterranean Sea, and most of the time the weather phenomena of interest are not observed. However, altimeter measurements are valuable and enabled a study, for the first time, of the spatial variability of wind and waves in the Mediterranean Sea (Guillaume et al, 1992). At ECMWF, the analysed 10m wind speed from the atmospheric model and the significant wave height (swh) from the wave model using these analysed winds are compared daily to the altimeter observations from the ERS-1 satellite, which is the only satellite providing information in real-time.

Maps showing the 10m wind speed and swh observed by the ERS-1 altimeter together with the respective model fields at 12:00 UTC are produced daily. On these plots, the time delay between the model results and the satellite observations, which is easily identified from the passing time plotted at the end of each track, can be as much as 12 hours. This time delay could be reduced by producing more frequent plots but, because of the poor coverage of the Mediterranean Sea by ERS-1, this was not found to be useful in a daily routine. An example of such plots, for a typical Mistral situation, is shown on the front page of this newsletter. The background fields are for 10 December 1994 at 12:00 UTC. The descending track along Corsica and Sardinia occurs at 10:28 (1½ hour earlier than the background field) and shows a greater expanse of wind stronger than 9 m/s in the northern part and higher waves compared to the models. The ascending track towards the Rhone Valley is at 21:43 (nearly 10 hours later than the background fields) and gives significantly higher wind speed and significant wave heights compared to the model results. Part of this discrepancy is explained by the time lag difference, as can be seen from the sequence of swh fields for this day (see Fig. 1).

The other means of verification of the wave model results is by comparison with observations from buoys. The main advantage of buoys compared to satellites are regular time coverage, a greater number of available sea surface parameters, and greater accuracy of these parameters. With the present technology, the significant wave height and the full frequency spectrum are easily measured. More sophisticated buoys provide some information on the directions of the waves. However, compared to satellite, buoys have some serious shortcomings. The number of buoy is small and does not assure a good coverage of the Mediterranean Sea. Because the buoy network was set up with the primary aim of shore and harbour protection, and, also for technical (and cost) reasons, the buoys are moored in the coastal zone. In the case of the Mediterranean Sea, the coastal zone is influenced by the large diurnal cycle over land and is affected by local breezes. This adds a few more sources of discrepancy and further complicates the verification studies. As, moreover, the Mediterranean Sea buoys do not carry any anemometer, a concomitant verification of wind and wave cannot be made. It is then extremely difficult to distinguish between model wind and model wave errors. Finally, these buoys do not report in real-time on the Global Transmission System and cannot be used on a routine basis.

In a recent study by Guillaume et al (1994), measurements from three buoys moored in the Spanish coastal zone, Cabopalos, Capdepera and Palamos (see Fig. 2), have been used to verify the significant wave height from the operational ECMWF wave analyses and from the 48-hour forecasts during a period of five months, Oct. 1992 - Feb. 1993. These three buoys were selected from the network of 22 buoys monitored by Clima Maritimo in Spain, because they were in sufficient depth and the farthest from the shore (from 3 to 6 km).

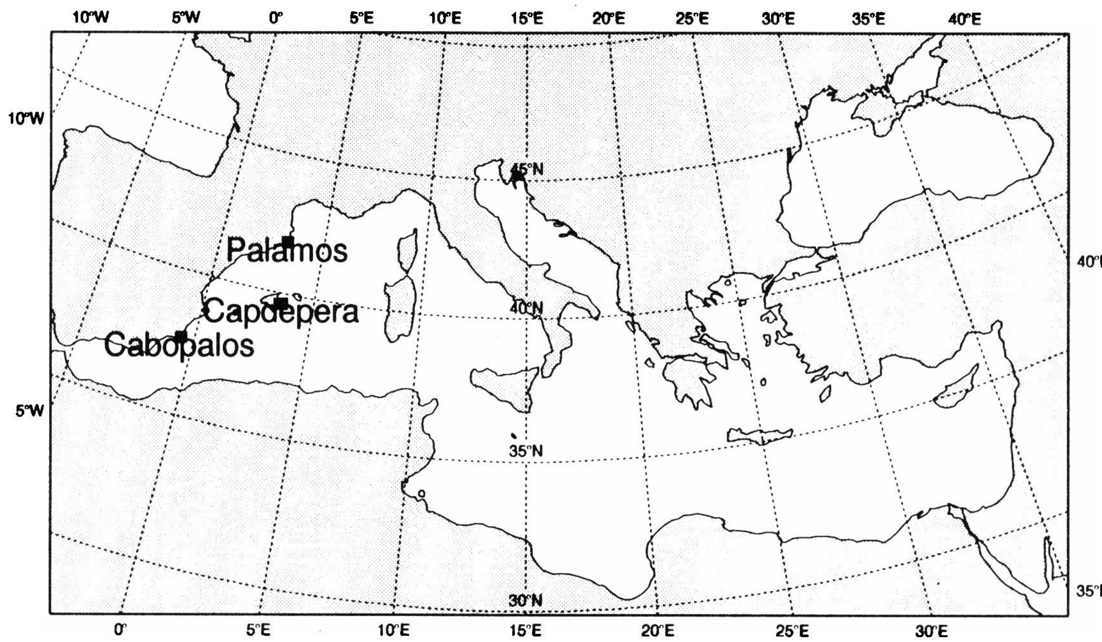


Fig. 2 The Spanish buoys used for the WAM model verification in the Mediterranean Sea.

At the three Spanish buoy locations, the wave were on average 1m with maximum swh values of between 3 and 5m, and with short periods, 4-5 sec on average. The model swh analysis and 48-hour forecast were found to have a bias to below the buoy observations, and the scatter of the differences, model minus buoy, were from 30% to 40% high. The model mean wave period analysis and 48-hour forecast were biased short compared to the buoy observations. Both the swh and mean wave period comparisons indicated a less developed sea-state in the WAM results than in the buoy observations. In order to distinguish between wind model errors and wave model errors and overcome the lack of joint measurement of wind speed and significant wave height, the directional distribution of model significant wave heights (Fig. 3) was compared to the distribution of model wind speeds (Fig. 4). This permitted a distinction between windsea and swell situations. For windsea situations, which are identified when the wave distribution is similar to the wind distribution, model wind errors are expected to be the primary source of discrepancy, as the WAM

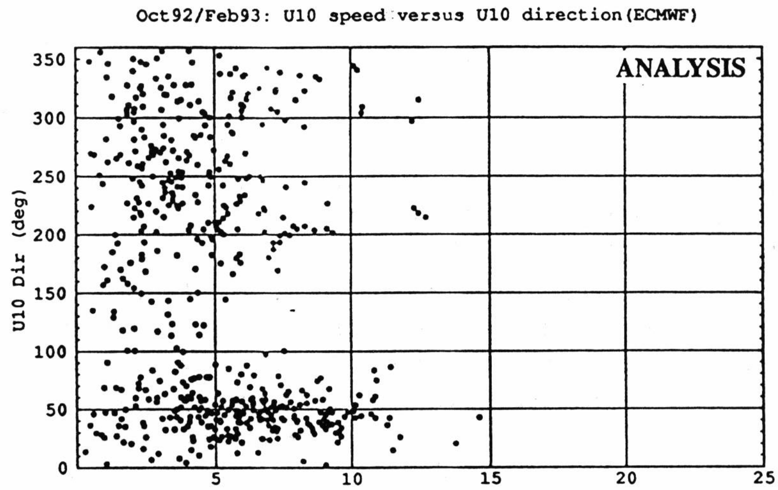


Fig. 3 Angular distribution of wind speeds at Cabopalos. Scatter plots of 10m wind direction versus 10m wind speed from the ECMWF T213 analyses for the period Oct. 1992 / Feb. 1993.

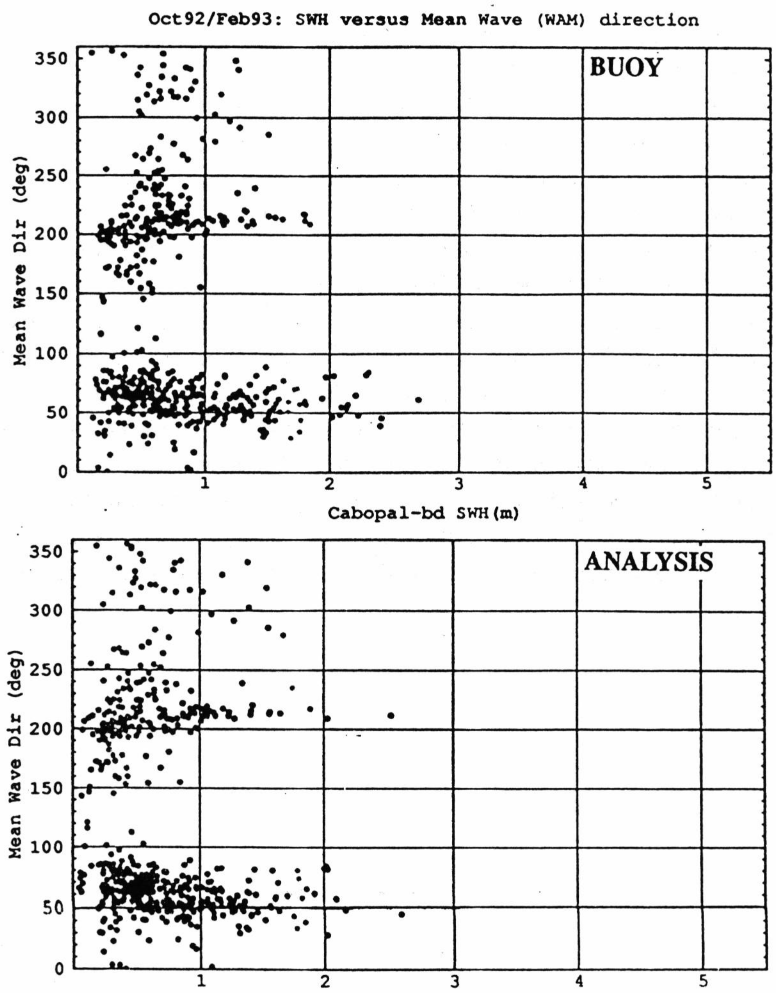


Fig. 4 Angular distribution of significant wave heights at Cabopalas. Scatter plots of mean wave direction versus significant wave height for the buoy (top diagram), and the WAM model analyses (bottom diagram). The mean wave directions shown for the buoy (top diagram) are the WAM analysed mean wave direction which are assumed to be the best estimate of the buoy mean wave directions which are not measured.

model physics have been extensively tested (WAMDI, 1988). For swell situations, propagation which depends also on the grid and spectral resolution of the wave model also need to be taken into account. At Cabopalos, where windsea situations dominate, the wave model compares well with the observations. Discrepancies between the 48-hour swh forecast and swh analyses were explained by wind model discrepancies. At Capdepera, a strong coastal influence was identified. The discrepancies between the model and buoy swh were explained by too coarse a resolution of the atmospheric model, too coarse a resolution of the wave model grid and too coarse a directional discretisation of the wave spectra in the wave model. At Palamos, an underestimation of storm waves was found, which cannot be attributed to the resolution of the wave model grid, but could be explained by an underestimation of storm peaks by the WAM model (which had already been reported in "the Mediterranean Sea" by Guillaume et al (1992)) and/or by an underestimation of storm wind speed by the ECMWF atmospheric model (as reported by Dell'Osso et al (1991)).

In conclusion, the higher resolution of the Mediterranean Sea wave model which is planned for early 1995 is expected to improve the wave analyses and forecasts. However, substantial improvements will require some in-depth investigations of the many highlighted qualitative aspects of the 10m winds fields. These include the accuracy of storm winds, the impact of atmospheric model resolution on coastal winds accuracy, the accuracy of forecast winds, and the systematic overestimation of wind speed in the ECMWF 48-hour forecasts compared to the analyses, which led to higher waves with longer period in the 48-hour forecasts compared to the analyses. To this end, wind and wave observation should be improved in the Mediterranean Sea. Wind speed measurements should be encouraged when and where waves are measured. The transmission on the GTS of wind and wave data recorded from buoys should be encouraged to assure better usage for marine forecasting activities. In the Mediterranean Sea, sea-state observation from space will have to rely mainly on altimeters. This is because long waves that could be measured by other instruments such as the SAR are rare. It is thus important that altimeters be validated for the specific climatological properties of the Mediterranean Sea, i.e. over the full range of significant wave height, including low waves, and for rapidly varying situations. Again, to assure the best coverage, the transmission in real-time of altimeter-derived parameters should be encouraged.

- Anne Guillaume

* * * * * * * * * * * * *

METVIEW - INTERACTIVE ACCESS, MANIPULATION AND VISUALISATION OF METEOROLOGICAL DATA ON UNIX WORKSTATIONS

Introduction

Metview is an interactive meteorological workstation application. It enables operational and research meteorologists to access, manipulate and visualise meteorological data on UNIX workstations.

Metview was designed as a flexible, modular and extendible system which will be able to accommodate the evolving user needs. Since Metview is, in its entirety, a very large system, it was not desirable nor feasible to aim at the implementation of the complete system at the initial stage.

The development of Metview started in 1991 [Câmara and Daabeck, 1991] and the first release became available to internal ECMWF users in December 1993. The third enhanced version has now been released.

The system is based on ECMWF standards for data access, MARS [ECMWF, 1993], and graphics, MAGICS [O'Sullivan, 1993]. The user interface is based on MOTIF and the X Window System. Metview is a fully distributed system where modules can run on different UNIX workstations and servers.

This paper introduces the basic concepts used in Metview followed by an example of a Metview user session. The features of Metview are outlined and a technical overview is given. Finally, the Metview Project and future plans are briefly mentioned.

The Metview concept

The general philosophy is that the user creates Metview definitions and performs operations on them. The definitions are represented in the graphical user interface as named objects with small pictures (icons). The objects or definitions are referred to as **Metview icons**. The icons can be manipulated by mouse operations and are stored in the work space (the large central area in Fig. 1).

Metview icons

Metview supports a range of icon classes and operations. The valid icon operations depend on the class of the icon.

Some basic Metview icons are: a Mars retrieval definition (MARS request), a visual definition (MAGICS attributes for plotting) and a plot window definition (specification of geographical area).

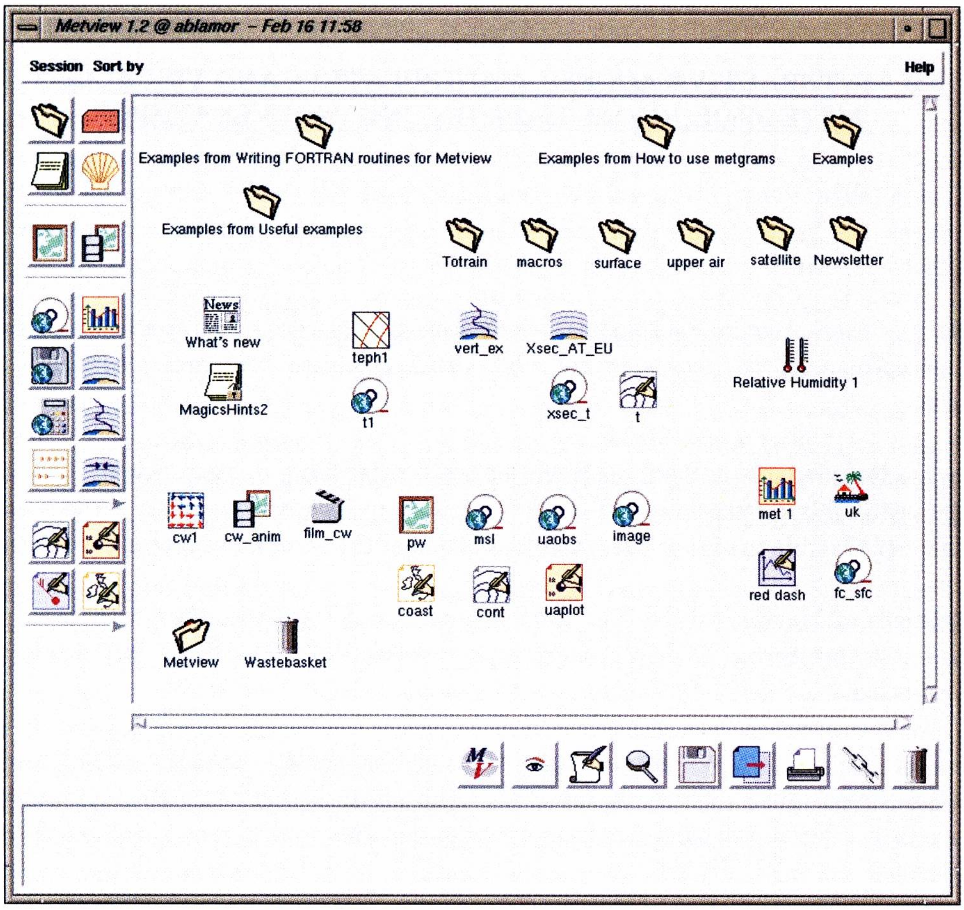


Fig. 1 Metview main graphics user interface window

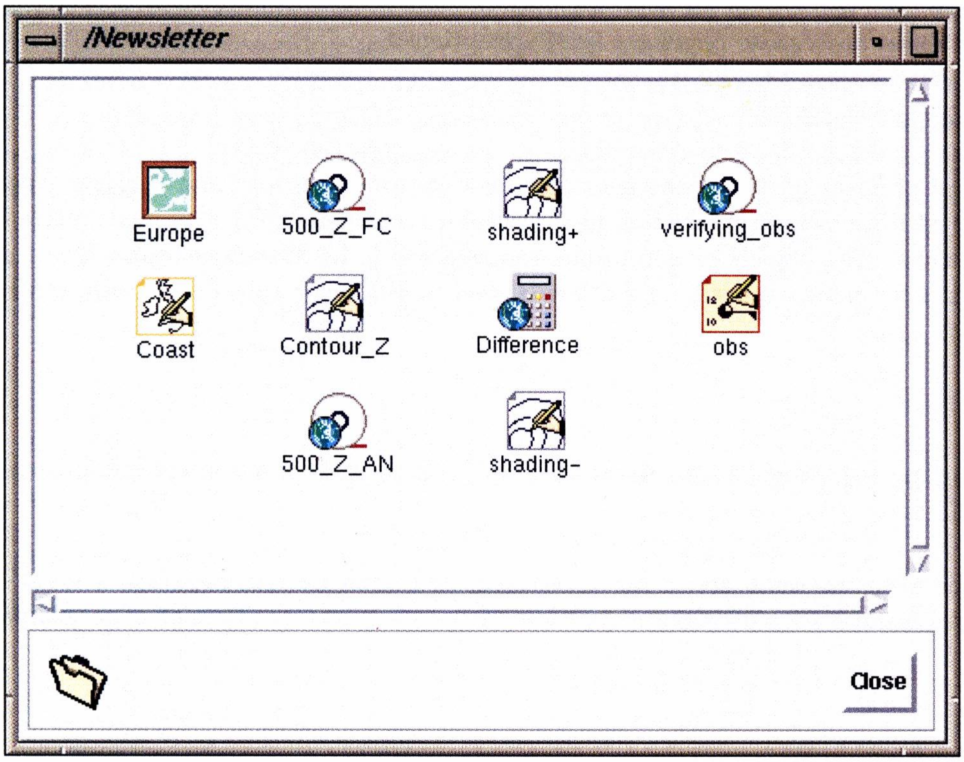


Fig. 2 Metview folder window

Operations, that can be performed on these icons, include:

- visualising a Mars retrieval definition or using it as input to a formula definition;
- applying a visual definition when visualising data to overwrite default plotting attributes;
- realizing a plot window definition, i.e. opening a window with a geographical map

There are several ways to visualise a data request, e.g.

- perform the visualize action on the Mars retrieval icon. A plot window will be created and default attributes will be used for visualising the data;
- open a plot window and drag the Mars retrieval icon into the plot window. A visual definition can be dragged into the plot window and its attributes will be used for re-displaying the data.

Example of a Metview session

The main user interface window contains the work space, the create buttons (left of the work space) and the action buttons (below the work space). The area at the bottom of the window is reserved for Metview status messages.

Metview icons are stored in the work space and are kept between Metview sessions. Folders (equivalent to directories) can be used to structure the storage of icons. The icons used to generate the following example plots have been grouped into a folder (Fig. 2).

By opening the editor for the particular icon, modifications can be made. Fig. 3 shows a portion of the editor for the Mars retrieval icon '500_Z_FC' and Fig. 4 for the visual definition icon 'Contour_Z'. Some values can be set using menus (e.g. *Type* and *Contour Line Style*), some can be set using an assistance button to the left of the value to aid selection (e.g. *Param* and *Contour Line Colour*), and for the others the value is simply typed in. The boxes on the extreme left show where a selected value is different from the default.

Actual steps

Double-click on the 'Europe' plot window icon. This opens the plot window which has been defined to have polar stereographic projection over the European area and has the coastline icon 'Coast' associated with it to define the coastline colour.

Contouring with default plotting attributes

Drag the Mars retrieval icon '500_Z_FC' (editor shown in Fig. 3) into the plot window. This action causes a 500 hPa height forecast from 24 December valid at T+120 to be retrieved and plotted in the plot window with default plotting attributes (Fig. 5).

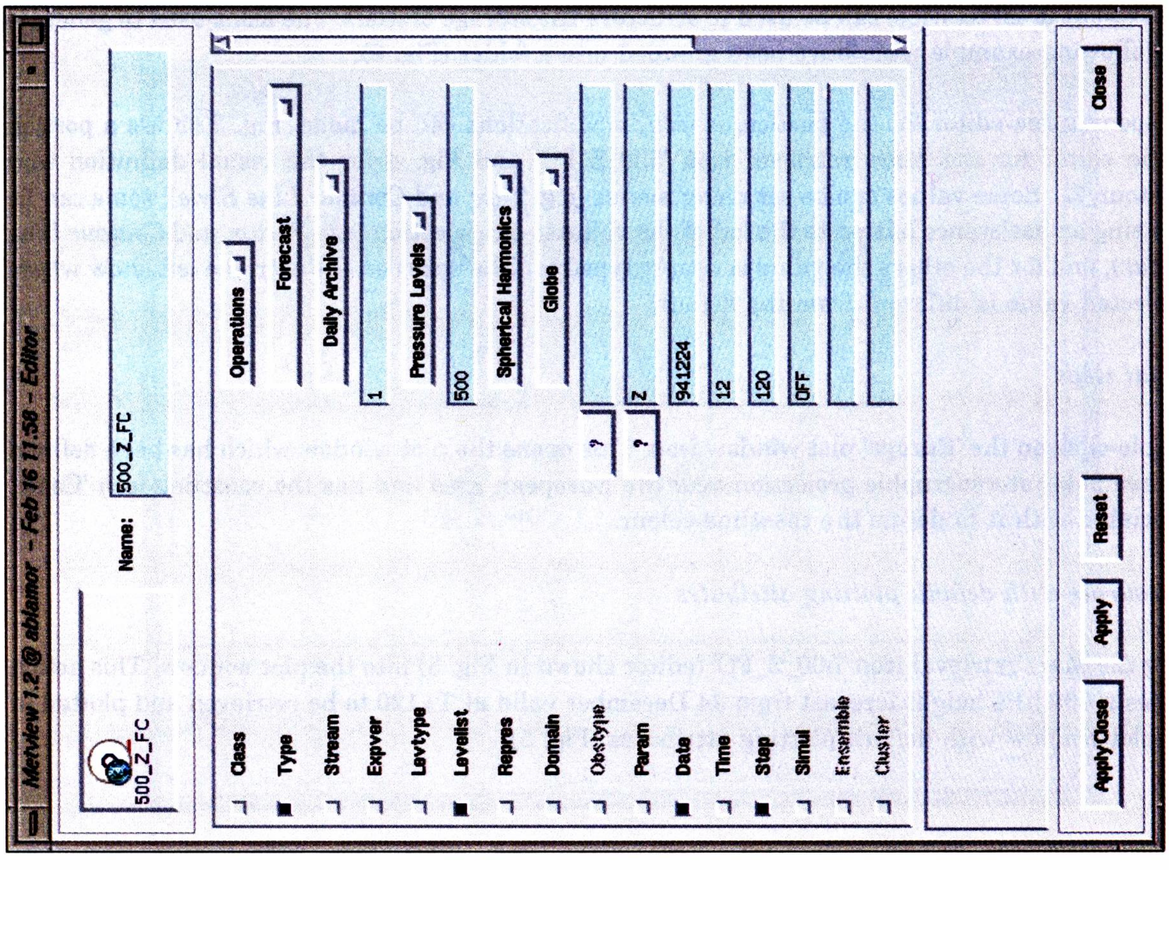


Fig. 3 Mars retrieval editor window

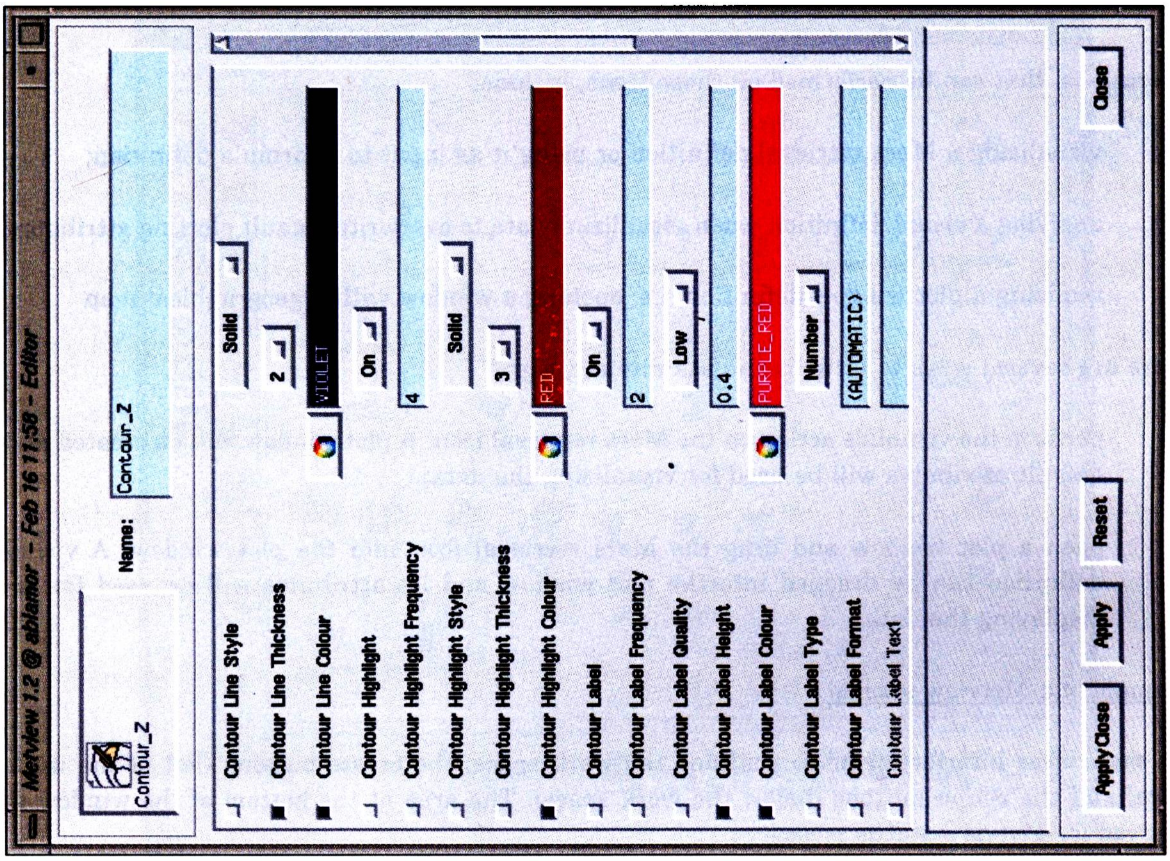


Fig. 4 Metview visual definition (MAGICS) editor window

Enhanced contouring

Drag the visual definition icon 'Contour_Z' (editor shown in Fig. 4) into the plot window. This icon defines particular colours for contour lines, labels etc, a specific contour interval, and maxima plotted as numbers. The contouring is changed accordingly (Fig. 6).

Difference field

To plot the difference between the forecast and its verifying analysis, we also need a Mars retrieval icon for the 500 hPa height analysis for 29 December ('500_Z_AN'). A formula icon 'Difference' can then be defined which simply specifies '500_Z_FC'-'500_Z_AN'. The visual definition icon 'shading+' specifies that shading is on, only values of 2 and above are plotted and that the colours range from yellow for low values to red for high values. A similar icon 'shading-' exists for plotting values of -2 and below. The icons 'Difference', 'shading+' and 'shading-' are dragged together into the plot window by drawing a box, with the mouse, around the three icons and dragging the box (Fig. 7).

Observations and zooming

We now wish to see the forecast together with the verifying observations. The icons '500_Z_FC' and 'Contour_Z' are dragged together into the plot window. The 'verifying_obs' icon is a Mars retrieval icon defining upper air observations at 500 hPa for 29 December. The 'obs' icon defines how the observations will appear; in this case, all values are default except for the observation size and dewpoint colour. Both icons are dragged into the plot window. A smaller area can be inspected by using a zoom tool which is selected by clicking on the 'magnifying glass' button and drawing a box with the mouse to define a geographical sub-area (Fig. 8).

Metview features

Online help

A context-sensitive online help system with hypertext support is available in Metview.

Creating icons

Currently, there are 31 create buttons available in a scrollable area. Pressing one of these buttons will create a new icon in the work space. New icons can also be created directly in a folder by using a pop-up menu. Icons will initially have system or user defined default values.

Some icons in the work space represent the output from other actions, e.g. the film icon represents a stored animation loop.

Actions on icons

The following actions for icons are defined: *execute, visualise, edit, examine, save, duplicate, print, delete, link*. Certain actions are only valid for a subset of the icon classes. Pressing an action

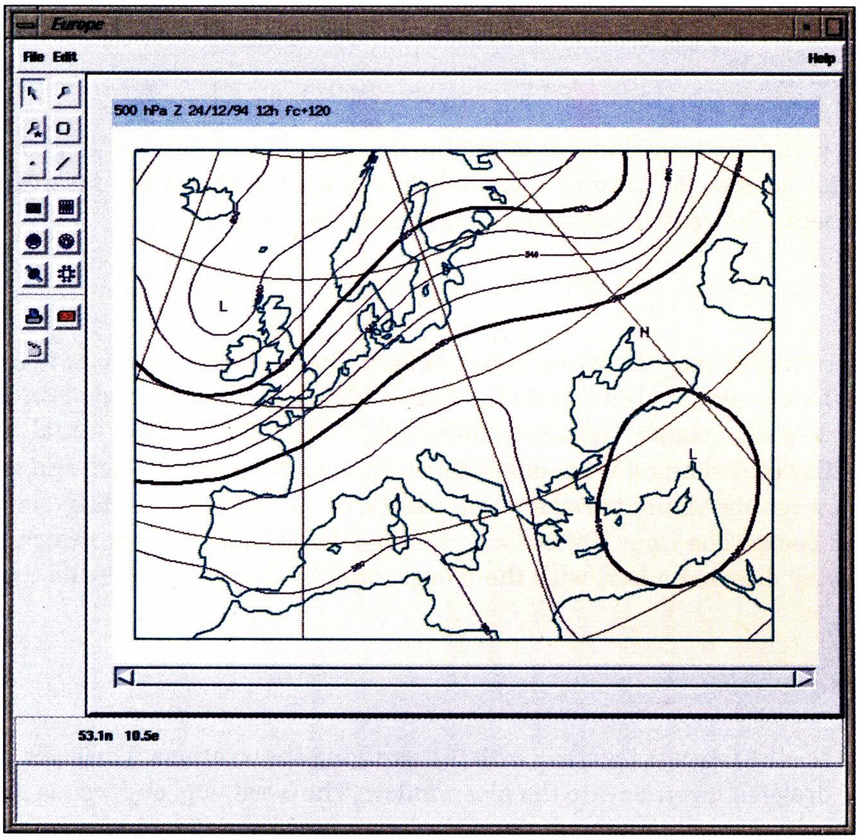


Fig. 5 Metview plot window with default plotting attributes

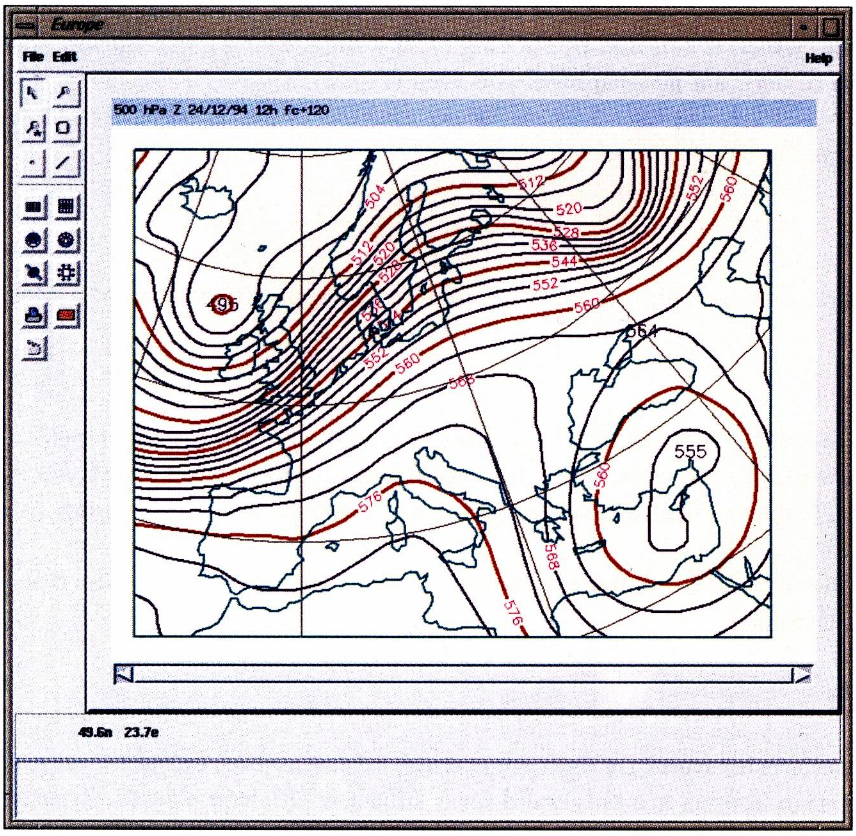


Fig. 6 Metview plot window with enhanced contouring

button will perform the action on the selected icons. For a single icon, actions can also be performed by using a pop-up menu.

Modifying the content of definitions

The *edit* action on an icon will bring up a context-sensitive editor window and the parameter values of the definition can be modified. Several editor windows can be open simultaneously. An icon of the same type can be dragged to the editor window to copy the parameter values.

For some editor parameters, assistance buttons exist. Pressing one of them, a window will appear which supports the selection process, e.g. for colours parameters, the assistance window will display the available colours.

Data formats

Meteorological data can be retrieved by Metview from the MARS archive, which holds 15 years of meteorological data (10+ Tera bytes), or read from a file.

The supported data formats are:

- WMO GRIB format for fields.
- WMO BUFR format for observations.
- The experimental extended WMO GRIB format for satellite images.
- A simple matrix format (only read from file).

The content of GRIB data (GRIB headers etc.) can be investigated by the *examine* action. The hierarchical structure will be shown by folders.

Retrieved data (Mars retrieval definition) or derived data (e.g. output from a Formula definition) can be saved into a file with optional generation of an icon in the work space to represent the file.

Managing icons in the work space

Automatic icon ordering and positioning is supported in the work space as well as shortcuts to locate an icon. Colour is used to indicate the status of an icon (red = error, yellow = busy, green = ready). To delete an icon, it should be moved to the *Wastebasket* folder. Icons are removed when deleted from this folder. Dragging icons from the *Wastebasket* folder will 'undelete' them.

A *link* icon is used to store references to multiple definitions. Without the *link* definition, visualisation of multiple icons would each time require a multiple selection of icons before dragging them into a plot window.

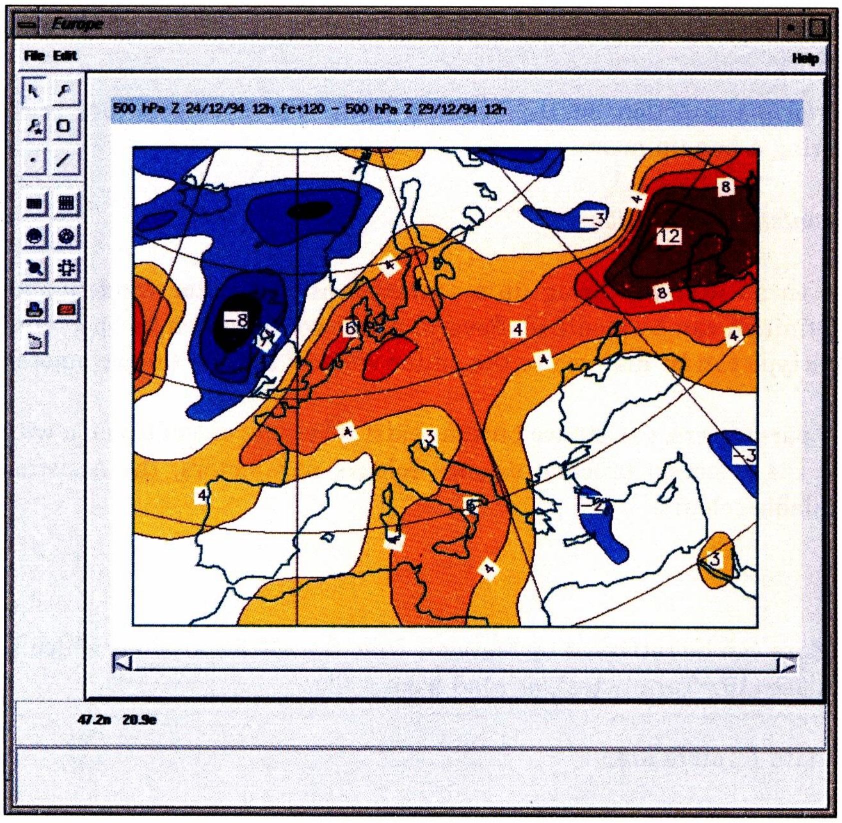


Fig. 7 Difference field with shading in Metview plot window

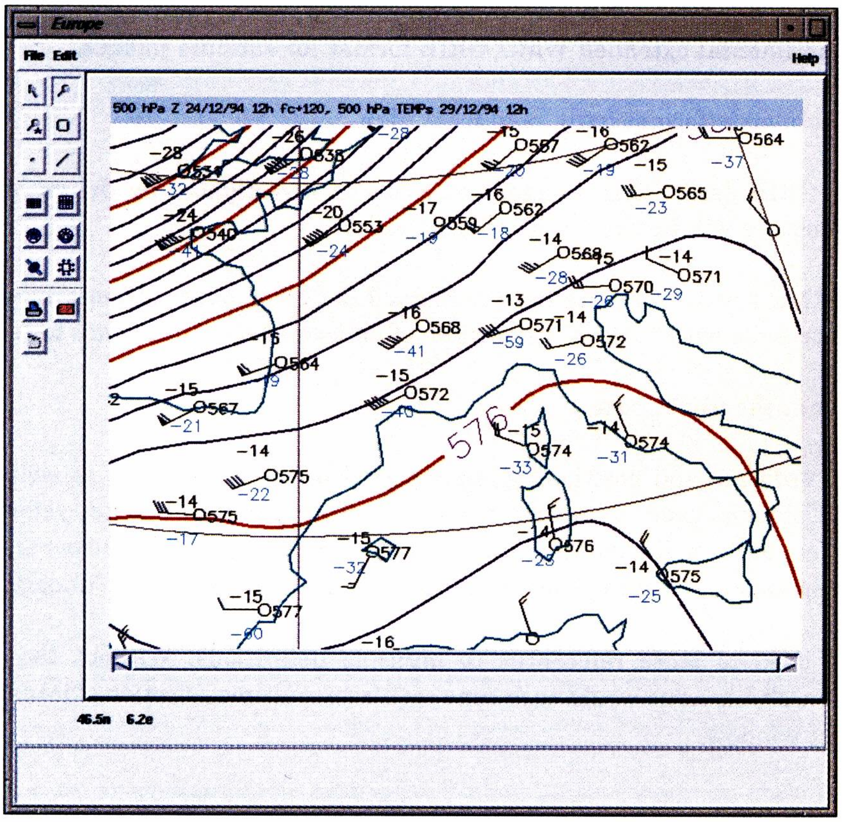


Fig. 8 Metview plot window with observations and zooming

Metview applications

Some of the Metview definitions are called applications and are represented by some of the create buttons.

Currently, the Metview applications include:

- Mars retrieval	accesses the MARS archive.
- Data in file	extracts selected fields from a GRIB file and processes them, including sub-area extraction and change of representation/ resolution.
- Observation filtering	extracts a subset (time, space, parameter), values range or extremes of observations from a BUFR file.
- Formula	performs mathematical computations on GRIB fields (Fig. 7). To facilitate typing, the reference to a data icon can be entered into the formula by dragging the data request icon to the formula editor window. More complex data manipulation can be performed by using the Metview macro language (see below).
- Cross section	vertical cross-sectional plot of upper air fields (Fig. 9 bottom left).
- Metgram	displays the evolution over forecast at selected points time of meteorological fields represented as curves or bar charts (Fig. 9 bottom right).
- Tephigram	plots tephigrams (soundings) from observations or from fields at selected points (Fig. 9 top right).
- Average plot	vertical cross-section of averaged upper air parameters (zonal or meridional).
- Vertical profile	graph for the vertical profile of upper air fields.
- Colour wind	displays wind fields coloured by a temperature field (Fig. 9 top left).
- Relative humidity	computes relative humidity from temperature and specific humidity.
- Total rain	computes total precipitation (convective precipitation added to large scale precipitation).

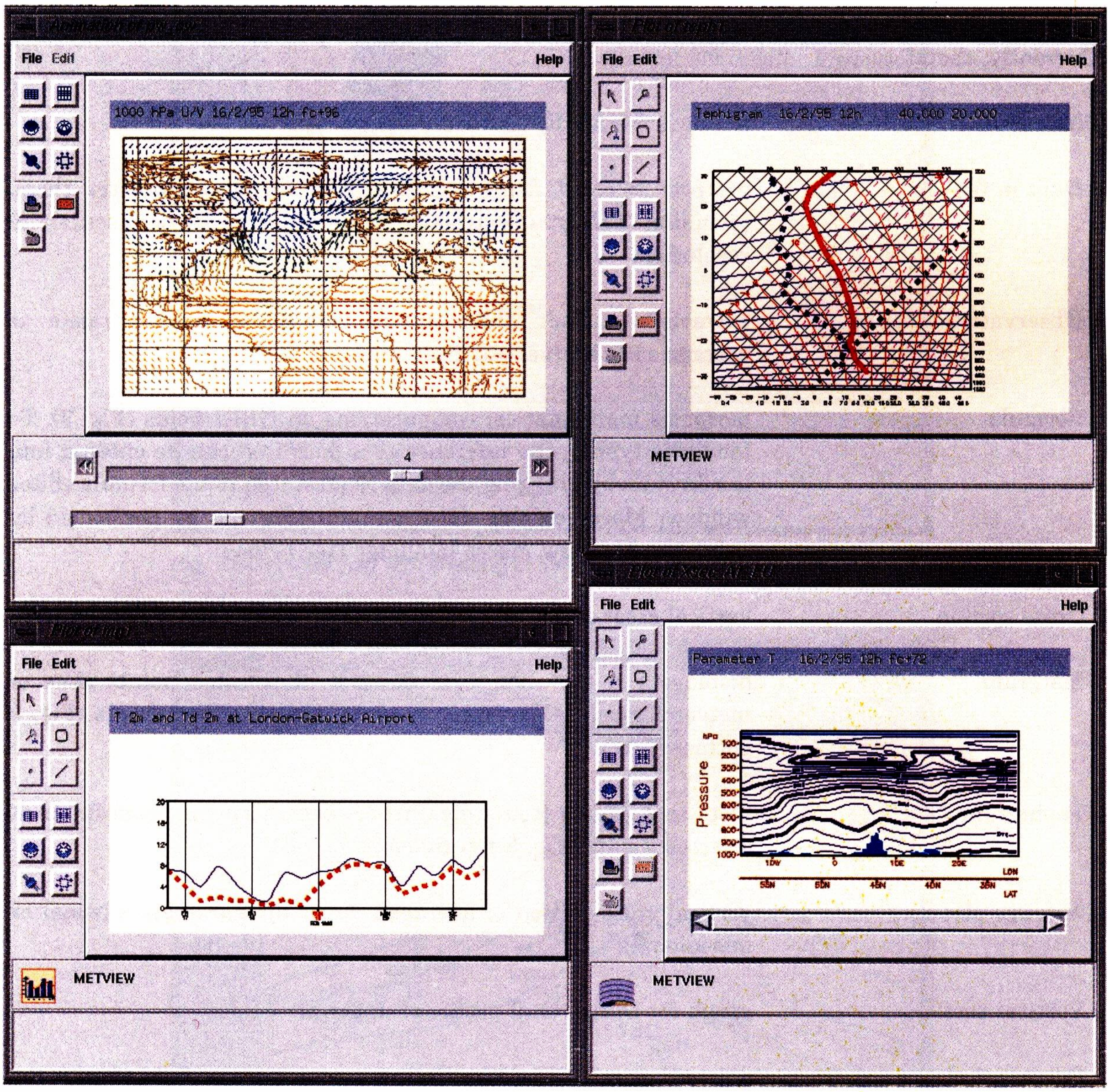


Fig. 9 Example of output from the Metview applications: Colour wind in animation window, Tephigram, Metgram and Cross section

Plot windows and animation windows

Visualisation of data is shown in either a plot window or an animation window. The user can define and open a plot or animation window and then drag a data request into it for visualisation. When matching is enabled, Metview will ensure that overlaid maps are consistent e.g. same time step. Annotation such as title and legend is also supported.

Geographical projections

The selection of the *Cylindrical, Mercator, Polar Stereographic* and *Spaceview* projections is made by pressing one of the projection buttons. An assistance window will appear and the area can be selected.

Sequence of maps

When the *visualise* action results in a sequence of maps, only the first map will be shown in the plot window and a scroll bar can be moved to visualise the following maps. The icons visualised in the plot window can be inspected and definitions can be removed and added. A plot window can contain rows and columns of panes (sub-windows). Each pane can contain an independent sequence of maps. Alternatively, a group of panes can display a range of steps from the same map sequence. All the panes of a plot window have the same projection and area.

Zooming

A plot window can be resized and two zoom tools are available. The *quick zoom* makes a graphical enlargement of a sub-area of the plot window whereas the *smart zoom* button will regenerate the map from the data and show more details.

Tools to select input for applications

Three tools (area, line, point) are available to facilitate the specification of input parameters for an application, e.g. select the line tool in a convenient plot window and use it to specify the line for the Cross section application. The line definition will now be applied to the line parameter in the editor window for the Cross section application.

Printing

When printing, the maps are re-rendered in a high resolution PostScript format. By default, the layout on the printer pages will be as in the plot window. All maps in a sequence, possibly not concurrently visible in the plot window, can be included for printing. The printer and page layout can also be specified.

The maps can also be written to a file in the PostScript format. Before printing or generating a PostScript file, previewing can be performed. If required, the source of a MAGICS FORTRAN program to generate the same plot, can be written to a file. This feature is mainly used for debugging purposes.

Animation window

The display of animation loops can be controlled by direction and speed (Fig. 9 top left). Other control features are similar to the control features for plot windows, i.e. for projection and area, coastlines and matching. However, resizing and zooming are not supported and an animation window only contains one pane.

With an opened animation window, data definitions and visual definitions can be dragged into it as for plot windows. Alternatively, from a plot window, the animate option can be selected. A temporary animation window for the data in the plot window will be created. Projection, size and layout from the plot window are preserved.

An animation sequence can be saved on disk, represented by a film icon in the work space, and be replayed at a later stage.

Metview macro language

The Metview macro language [Raoult et al., 1995] is fully integrated into the Metview user interface and can be used to automate repetitive procedures (Fig. 10).

The macro language is a high level language aimed at the research user. The language has all the capability of a computer language, such as variables, loops and procedures. In addition to the handling of real numbers and character strings, the macro language defines entities such as dates, sets of meteorological fields, satellite images and observations. Computations can be performed on these entities using a set of operators and functions.

Tools to filter observations and manipulate fields and images are available. The user can provide functions written in the macro language or FORTRAN. The macro language supports retrieval of data and page layout for plotting the resulting fields can be specified.

From within a macro, dialogue windows can dynamically be created to prompt the user for parameters.

Creating macros

Macros can be entered by typing in the macro editor window. Icons can be dragged into the macro editor window to facilitate the writing of macros. The contents of the definitions will be expanded in the macro editor window. A macro can automatically be created from a plot window, i.e. the plot window and its content can be visualised by executing the macro.

Macros can be stored in user or system-wide accessible libraries.

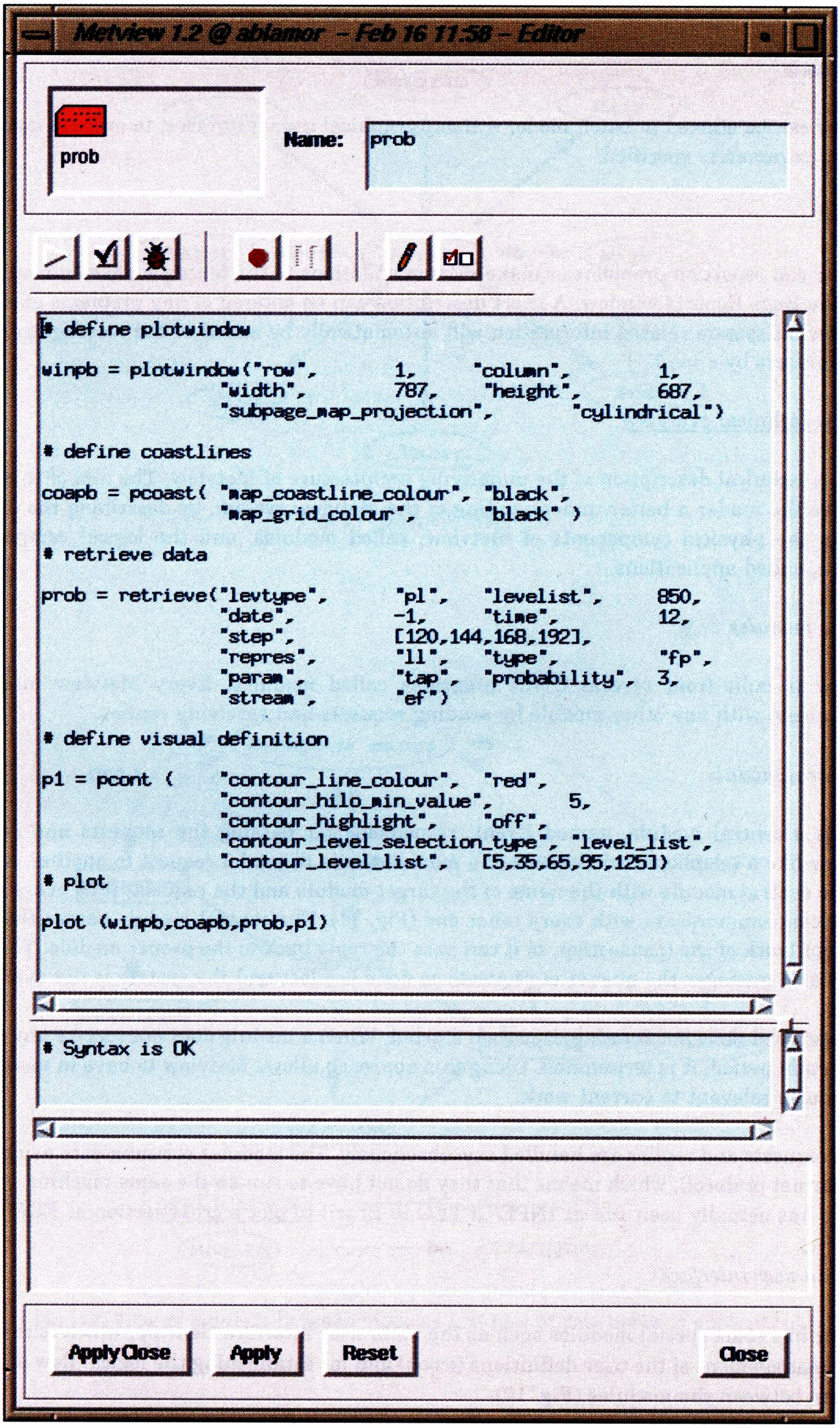


Fig. 10 Metview macro editor window

Batch mode

Metview can be started in batch mode, without graphical user interfaces, to execute macros with optional parameters specified.

Support

The user can report on problems or make recommendations to the Metview development team by using the Bugs Reports window. A short description can be entered of any problems encountered and relevant system related information will automatically be added to the message and sent to the developers by e-mail.

Metview technical overview

This is a technical description of the underlying architecture of Metview. The aim of this chapter is to give the reader a better understanding of the Metview system, by describing the difference between the physical components of Metview, called modules, and the logical components of Metview, called applications.

Metview modules

Metview is built from several UNIX programs called modules. Every Metview module can communicate with any other module by sending requests and receiving replies.

The central module

There is a central module, named Event, responsible for passing the requests and replies. It functions like a telephone switchboard: if a module wants to send a request to another module, it calls the central module with the name of the target module and the request, thus every Metview module can communicate with every other one (Fig. 11). The central module passes the request and keeps track of the transaction, so it can pass the reply back to the proper module. The central module also manages the queues of requests so none are lost and the system is not overloaded.

If a target module is not running, it is then started. When a module does not receive any requests for a certain period, it is terminated. Using this approach allows Metview to have in memory only the modules relevant to current work.

All the requests and replies are handled asynchronously. The modules communicate using TCP/IP (the Internet protocol), which means that they do not have to run on the same machine. As a test, a macro has actually been run at INPE/CPTEC in Brazil to plot a cross-section at ECMWF.

The main user interface

Metview has some special modules such as the main user interface GenApp, which is responsible for the management of the user definitions (icons) and for establishing the logical flow of requests and data between the modules (Fig. 12).

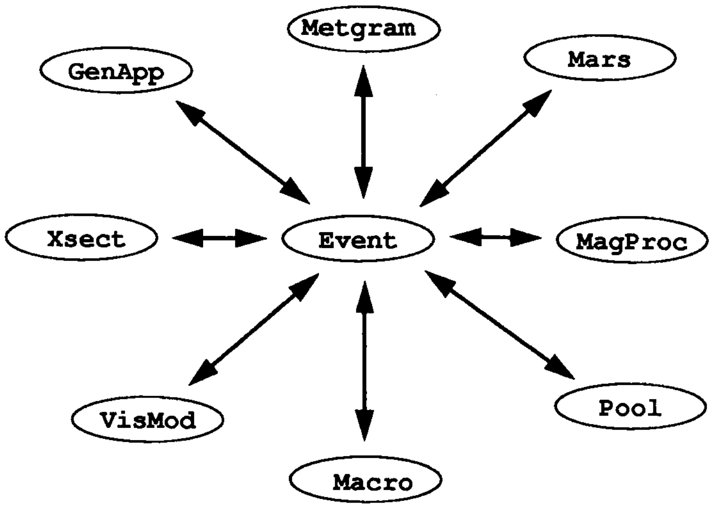


Fig. 11 Metview modules

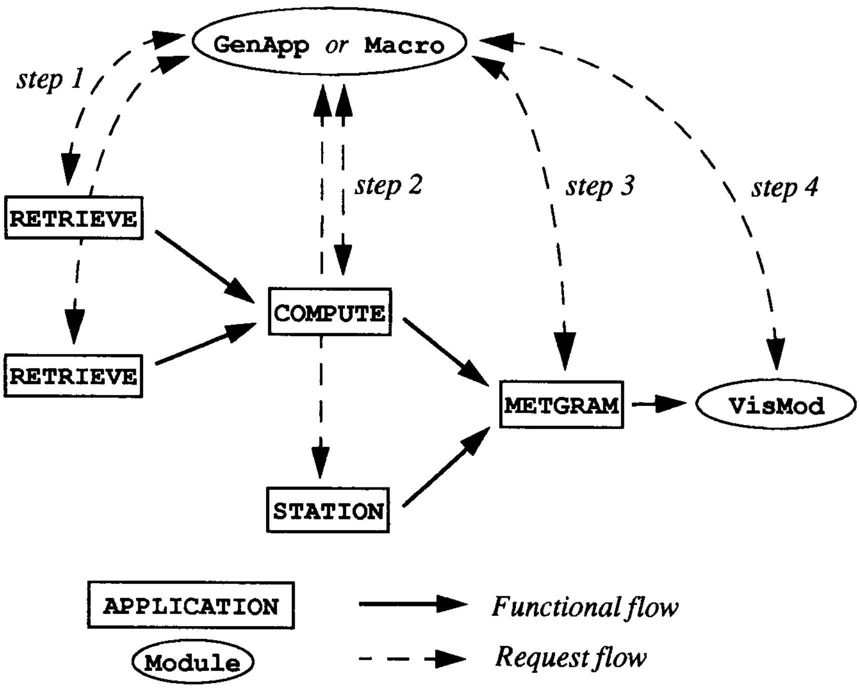


Fig. 12 Flow of requests between modules and flow of data between applications

Visualisation

The visualisation module VisMod manages the plot windows. It uses the actual rendering module MagProc, which performs the calls to MAGICS. Splitting the visualisation into different modules allows the user to perform certain actions on the plot window (resizing, quick zoom, . . .) while another map is being drawn.

However, the drawings are performed by a single module, so Metview can only plot one map at a time. In order to print a plot window, Metview re-renders the plot for the printer, so the drawing is of the best possible quality.

The applications

Some of the modules handle one or several Metview applications. Metview applications are represented by some of the create buttons.

An application can be seen as a simple function: it receives an input (the content of a given icon) and returns a set of values. The applications can be chained to perform more complex actions. Mars, Metgram, XSect are modules running the basic Metview applications: RETRIEVE and READ for Mars, METGRAM for Metgram, etc.

Most of the modules running an application will *fork* before processing, which means that a module can handle several requests at once.

Fig. 12 shows how the metgram of the difference between two fields can be plotted. The dashed arrows show the module to module communication via requests and replies. The solid arrows show the logical flow of information. In step 1, GenApp sends two requests to the RETRIEVE application. The resulting data is passed back to GenApp, which in turn will pass it as input to the COMPUTE application, while calling the STATION application to find the geographical position of a given station (Step 2). Once the two replies are ready, they are passed to the METGRAM application (Step 3). The output of METGRAM is then passed to VisMod (Step 4).

The macro player

The macro player will interpret a macro and execute its statements, possibly with a set of input arguments, and return a value to its caller. In this respect it behaves like an application. But it can also invoke another application, as the main user interface is doing. Every application (or icon class) has a corresponding macro function. The metgram example could be written

```
f = retrieve (. . .)
g = retrieve (. . .)
s = station (. . .)
plot ( metgram ( data : f - g, station : s ) )
```

In this case, `retrieve()` calls the RETRIEVE application (handled by the Mars module), `station()` calls the STATION application, `metgram()` the METGRAM application and `plot()` passes the result to VisMod. The only difference with the user interface is that the computation is performed directly by the macro and not by the COMPUTE application, for the sake of efficiency.

File space management

Metview creates a temporary directory to hold its working files. When an icon is executed, the relevant application is called and the result, which can be a file, is placed in this directory.

As long as the icon is not modified, the file is kept. If the icon is visualised, the application is not called again, the cached result is used instead. If the icon is modified, the file is removed as it does not match the definition in the icon any more. Quitting Metview will remove all temporary files.

The file space management is handled by a module called Pool. This module also checks the dependencies between the icons, i.e. a Cross section icon can be dependent on a Mars retrieval icon. If one icon is modified, any dependent icon is also marked as modified and its cached data purged. If the data retrieval is modified, the Cross section output is no longer valid.

Two other modules, Exampool and Savepool, help the user to examine and save the cached data.

The Mars module also provides a data caching mechanism. The data retrieved from the main archive (on the IBM), are copied into an Empress database on one of the ECMWF servers. As a result, a frequently accessed field will be cached, and the retrieval time can drop from several minutes to a few seconds.

The Metview environment

All of the user definitions are written in the Metview user directory. Each icon has a corresponding file in this directory. Each folder is a UNIX sub-directory.

For each of the icons, there is also a hidden file with the same name as the icon but prefixed with a dot. The hidden file contains the position of the icon in the user interface, and its class.

If a foreign file is found, the main user interface will try to guess its type. Although they cannot be created with one of the create buttons, some files can be displayed as an icon in the work space. This is the case for GRIB and BUFR files, as well as shell scripts, PostScript files and plain text files.

This feature is useful for users having GRIB files in a directory structure. Creating a symbolic link to this directory in the Metview user directory allows the user to see these files as Metview icons and use them in computation or plot them like any other Metview object.

Creating a symbolic link to someone else's Metview user directory or to a directory shared by a group of people can prove very useful.

Developing Metview applications

Metview is built from several software layers (Fig. 13). The main idea was to design an open system so new applications can be added very easily and transparently. This was achieved by:

- implementing the Metview applications as various programs (modules) that can be developed and tested by different people sharing a common environment but working independently;

- providing a set of very high level C++ classes, so the technical aspects of Metview (communication, data caching, input parameter validation, error reporting, etc.) are totally hidden and the application developer can focus on his/her problem.

To write a Metview application, the following steps are required:

- design a meaningful icon which will be used to represent requests to this application in the main user interface;

- create a file that contains the definition of the parameters your application needs, what their valid values are, their default values, and a set of rules to perform simple consistency checks. This file is used by the main user interface to create the various items in the editors, like the menus or the input fields. It is also used by the macro player to check the validity of the parameters;

- write a program using the C++ classes.

Alternatively, a macro could be made into an application and, because macros can call FORTRAN routines, any FORTRAN programmer can easily implement a Metview application.

<u>The Metview Project</u>

Metview is a cooperative project between ECMWF and INPE/CPTEC, Brazil. ECMWF has also been assisted by a staff member from Météo-France.

Metview 1.2 was developed at ECMWF with contributions from Ricardo Cartaxo[1], Bruno David, David Dent, Fernando Ii[1], Arne Jørgensen, Vesa Karhila, Elisa Nishimura, Patrick O'Sullivan, Baudouin Raoult and Sylvie Lamy-Thépaut[2]. The project leaders for Metview are Jens Daabeck and Brian Norris at ECMWF and Gilberto Câmara[1] at INPE/CPTEC.

Further references to Metview can be found in Daabeck et al., 1994; DMI, 1994 and Cartaxo et al., 1993.

---

[1] Centro de Previsão de Tempo e Estudos Climáticos (CPTEC), Instituto Nacional de Pesquisas Espaciais (INPE), São José dos Campos, Brazil

[2] MétéoFrance, Toulouse, France

<table>
<tr>
<td><code>MagProc</code>
5 600 lines
Interface module to MAGICS. Printing Postscript files.</td>
<td><code>VisMod</code>
22 000 lines
Visualisation module. Plot windows and animation.</td>
<td><code>GenApp</code>
22 000 lines
Main user interface module. Editing and management of the user's definitions. Triggers and chains requests.</td>
<td><code>Applications</code>
3 500 lines
Various applications such as Cross section or Metgram.</td>
<td><code>Modules</code>
4 500 lines
Utility modules such as What's new, fieldset browser, shared data pool.</td>
<td rowspan="3"><code>Macro</code>
10 000 lines
Macro language compiler and runtime module. Data manipulation on numbers, strings, dates, vectors, matrices, observations, images and fields.
Batch. Access to any Metview module.</td>
<td rowspan="3"><code>Event</code>
1 000 lines
Inter module message router.
Process management.</td>
</tr>
<tr>
<td rowspan="3">libMagics
Plotting routines.</td>
<td colspan="4"><code>libMetview</code>
6 000 lines
High level C++ routines library.
C++ wrapping of lower level routines. Meta-classes for Module definitions, high level inter-modules communication classes, high level fields and observations manipulation, and various graphical objects.</td>
</tr>
<tr>
<td colspan="4"><code>libUtils</code>
30 000 lines
Low level C++ routines library.
Widgets. A few low level classes, a C++ wrapping of Motif, low level X11 and Xt routines. Drag and drop widget, HyperText widgets, Plotting widgets and some layout widgets.</td>
</tr>
<tr>
<td colspan="6"><code>libMars</code>
20 000 lines
Low level C routines library.
Networking, Low level inter-modules communication routine, string hashing, request management. Languages processing, check and expansion. Free-form formulae, field manipulation, database access, file access. General purpose routines.</td>
</tr>
<tr>
<td colspan="5">libEmos
GRIB and BUFR routines.</td>
<td colspan="2">libSpec
Interpolation routines.</td>
</tr>
</table>

In addition

- 1 000 lines of examples
- 2 500 lines help files
- 3 500 lines of icons
- 11 000 lines of configurations and language files
- 1 000 lines of shell scripts

Fig. 13 Metview software layers

Future plans

Future plans for the development of Metview include adding more applications, e.g. the migration of applications from the Metview/batch system [Norris, 1994] which is implemented on the Centre's Cray C90 and YMP-EL computers. Support for Member States use of Metview with transparent access to MARS data via a proxy server running at ECMWF is planned. At the Centre, Metview has been implemented on Silicon Graphics workstations. Elsewhere, implementations on different platforms and interfaces to SQL based databases are under development.

- Jens Daabeck, Brian Norris, Baudouin Raoult

References

Raoult, B., J.Daabeck, B.Norris and G.Câmara, 1995: Data manipulation and visualisation at ECMWF using the Metview/ws macro language, 11th International Conference on Interactive Information and Processing Systems for Meteorology, Oceanography, and Hydrology (IIPS), AMS.

Daabeck, J., B.Norris, K.Petersen, E.Nishimura, S.Lamy-Thépaut, G.Câmara and R.Cartaxo Modesto de Souza, 1994: Metview/ws - an application for data processing and visualization at ECMWF and INPE/CPTEC, 10th International Conference on IIPS, AMS.

DMI, 1994: 5th European Group on Operational Meteorological Workstations (EGOWS), Danish Meteorological Institute.

Norris, B. (editor), 1994: Metview/batch user's guide, ECMWF Meteorological Bulletin, M1.10/1(3).

Cartaxo, R., J.Daabeck, F.Ii, E.Nishimura, B.Norris, B.Raoult, S.Lamy-Thépaut, 1993: Metview/ws - a distributed system for meteorological data processing and visualisation, Fourth Workshop on Meteorological Operational Systems, ECMWF.

ECMWF, 1993: MARS User Guide, ECMWF Computer Bulletin, B6.7/2.

O'Sullivan, P., 1993: MAGICS - The ECMWF graphics package, ECMWF Newsletter, Number 62, June 1993.

Câmara, G., J.Daabeck, 1991: The conception of a new system for visualization of meteorological and climatological data at ECMWF, 7th International Conference on IIPS, AMS.

* * * * * * * * * * * *

PROGRAMMING THE CRAY T3D

As described in the Autumn 1994 Newsletter (p. 17), the Cray T3D consists of 128 processors (PEs) each of which consists of a microprocessor connected to 64 MB of local memory. It is distinguished from a collection of workstations by the fast interconnect and global address capabilities which allow a PE to address any part of memory directly. The distinction between this type of 'virtual shared memory' and the shared memory of the CRAY-C90 lies in the relative performance. In the C90, any part of memory can be considered to be the same 'distance' away from a processor in that the time taken to retrieve a word is essentially the same regardless of where that word resides. In the T3D, a PE can access its local memory an order of magnitude faster than other memories. This distinction has a large bearing on the way that the T3D is programmed.

When individual processors are very powerful as in the C90, it is perfectly reasonable for many applications to run serially, i.e. utilising only one processor. This becomes insufficient when:

(a) the application is time critical
(for example, a production run of IFS);

(b) it occupies substantial amounts of main memory.

In these cases, the serial C90 code has to be parallelised using manual methods such as MACROTASKING or more automatic methods such as AUTOTASKING. Both techniques rely on the global shared memory hardware in order to achieve satisfactory performances.

On the T3D, serial codes occupy only one PE and execute at speeds similar to a workstation. This will not be sufficient for many applications. Hence, parallel versions have to be developed using either

(a) manual methods (MESSAGE PASSING) or

(b) more automatic techniques via a data parallel variation of Fortran such as that defined by HPF (High Performance Fortran) or the Cray variation of HPF called CRAFT (CRay Adaptive ForTran).

"Message Passing" is a mechanism which allows data to be transferred between separate memories, possibly belonging to completely independent computers (e.g. workstations).

In order to overcome the significant manual work needed to implement message passing, extensions to the Fortran language standard allow the programme some control over the layout of data in the distributed memories.

Message passing

This view of the machine considers all data to be local to the processor. Any data required which resides in the memory of other PEs must be copied between memories using a 'message'. The time taken to transfer the message will depend on the underlying hardware and the message passing software interface and is therefore crucial to the ultimate performance of the parallel application. On the T3D, the network is very fast compared to most other parallel distributed memory computers.

Usually, messages are sent and received by 'co-operating' processes (see Fig. 1), i.e. the sender addresses the message to a specific logical PE number which expects to receive from that sender. Additional security can be obtained by adding a unique 'tag' to the messages. This is useful for distinguishing between several arriving messages. There is usually an intermediate 'mailbox' provided by the message passing system to hold the message in case the receiver is not ready.

Different message passing interfaces are available e.g. PVM (Parallel Virtual Machine) and PARMACS. An emerging de-facto standard known as MPI (Message Passing Interface) is likely to be the most portable method for the near future. Up to today, we have chosen PARMACS because of its wide availability and good performance. MPI will be available quite soon on the T3D and will provide not only portability but also enhanced functionality.

Message passing can be likened to MACROTASKING in that the programmer must have a wide and detailed knowledge of the application. This means that it is likely to be expensive to develop but if done well should lead to a portable code that runs efficiently. There are additional advantages from using message passing. Greater user control over the computations carried out on each PE allows for improved optimisation, re-use of serial code, and reproducibility of results. We believe that message passing will remain a viable programming model for many years to come.

Today, we have a parallel implementation of IFS using PARMACS. It runs on the T3D and can utilise any number of PEs. However, large resolution runs require a certain minimum number to provide enough memory. For example, T63 needs 8 nodes and T213 needs 128 nodes.

PARMACS imposes a host/node programming model on to the application. We choose to make the host program as small as possible and to run it on the YMP-2E front-end (so called heterogeneous mode). In this case, any messages passed between host and nodes must be converted to take care of the differing machine representation of floating point numbers. This can be done rather easily by invoking a PARMACS 'MSG_FORMAT' macro. The IFS host contains initialising code such as namelist and initial data reading, PARMACS initiation calls and execution of the node program. Data needed by the nodes has to be sent there using message passing (or intermediate files). Finally, output generated by the nodes (such as model diagnostics) is received and printed by the host. By keeping the host program small and simple, it should be possible to convert easily to a hostless programming model which is allowed by MPI.

High Performance Fortran

A view of the machine which is more like standard Fortran is obviously an attractive idea. HPF attempts to achieve this by defining compiler directives which describe the layout of data arrays across multiple PE memories. Directives are also used to indicate e.g that iterations of a DO loop may be distributed across PEs, thus 'parallelising' the code. It is crucial to good performance to ensure that the iterations computed by each PE are exactly those for which it owns the data elements as defined by the data distribution directives.

A significant limitation of the first definition of HPF lies in its inability to handle other than simple rectangular arrays. Much effort is currently being devoted attempting to create a standard way to describe the layout of more general data structures, for example the triangular shapes used by spectral models and the irregular requirements of reduced Gaussian grids. An HPF2 definition may become available within the next few years.

Use of directives in this fashion is reminiscent of micro- or auto-tasking on C90. Normally, this activity is at a low level in the code (SUBROUTINE or DO loop level) and as such generates fine grained parallelism which is likely to be less efficient than the relatively coarse grained parallelisation that can be achieved with message passing. Its attraction lies in the relatively small amount of effort needed to port a serial code to run in parallel. However, to achieve good performance one must have a detailed understanding of the data flow and data structures of the application.

HPF compilers are beginning to become available, though many merely provide a Fortran interface to message passing, that is the distributed nature of the data is handled by compiler generated message passing. Today, there is no HPF compiler for the T3D. There is, however, a CRAY compiler which supports CRAFT. This provides the functionality of HPF plus many additional features but is non portable since the directives are unique to T3D platforms. This means that since the compiler directives are in the form of comments, the code will run serially on non-T3D platforms. We cannot recommend the general use of CRAFT due to the lack of portability, but CRAFT can be useful as a debugging aid. Using a shared array, one can get a global view of distributed data without having to write additional message passing code. Portions of data computed by individual PEs may be stored into memory to allow easy printing for trace purposes. Fig. 2 shows an example of how shared arrays are allocated across the memories of the available PEs.

Codes written using CRAFT generally achieve rather poor performances from today's early compiler, as is generally true for any presently available HPF-like compilers. We believe that this will be the case for at least the next few years.

Since the T3D hardware is able to address any part of memory from any node, it is possible for an application to take advantage of this feature by means of 'shared memory' library calls. A PE can transfer data between any nodes using 'shmem get and put' calls. These are much faster than message passing since the software message passing protocol is bypassed. However, the user must explicitly take care of synchronisation (e.g. letting a remote PE know that some data has been sent to it), and of course the interface is non-portable.

MESSAGE PASSING

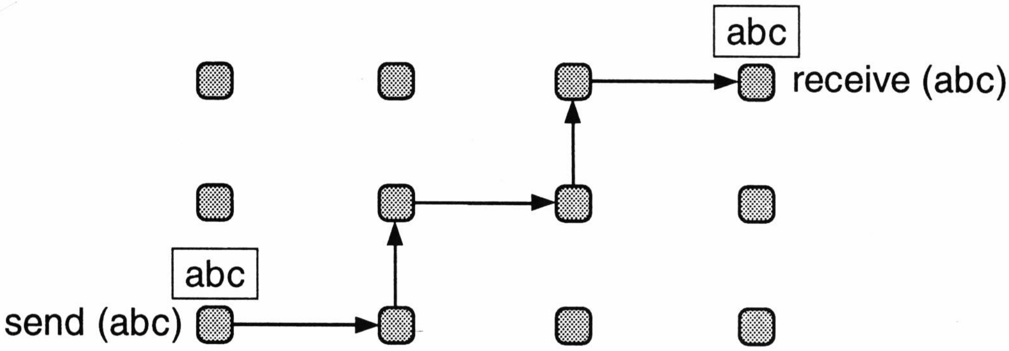


Fig. 1 Transferring data (abc) from one PE to another

CRAFT SAMPLE

```
      COMMON /PRVT/ A(4)
      COMMON/SHRD/ X(512)
CDIR$ SHARED X(:BLOCK)
```

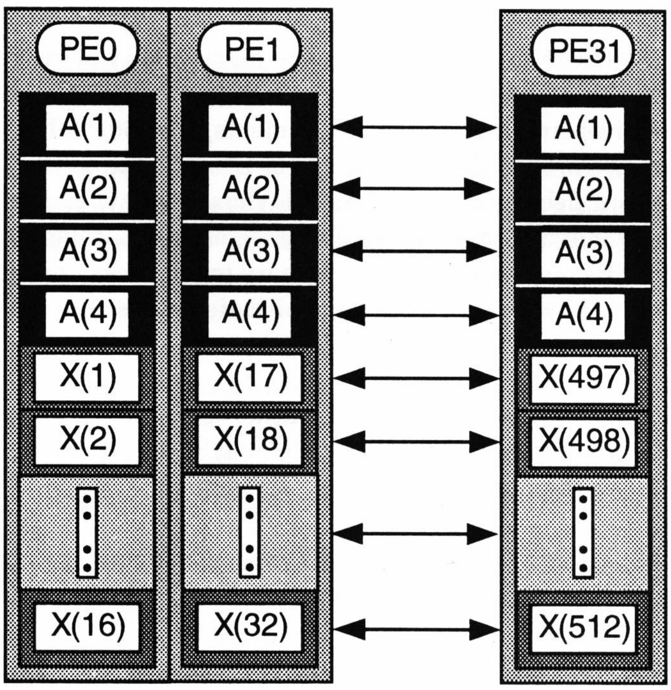


Fig. 2 Declaration of private and shared arrays using CRAFT

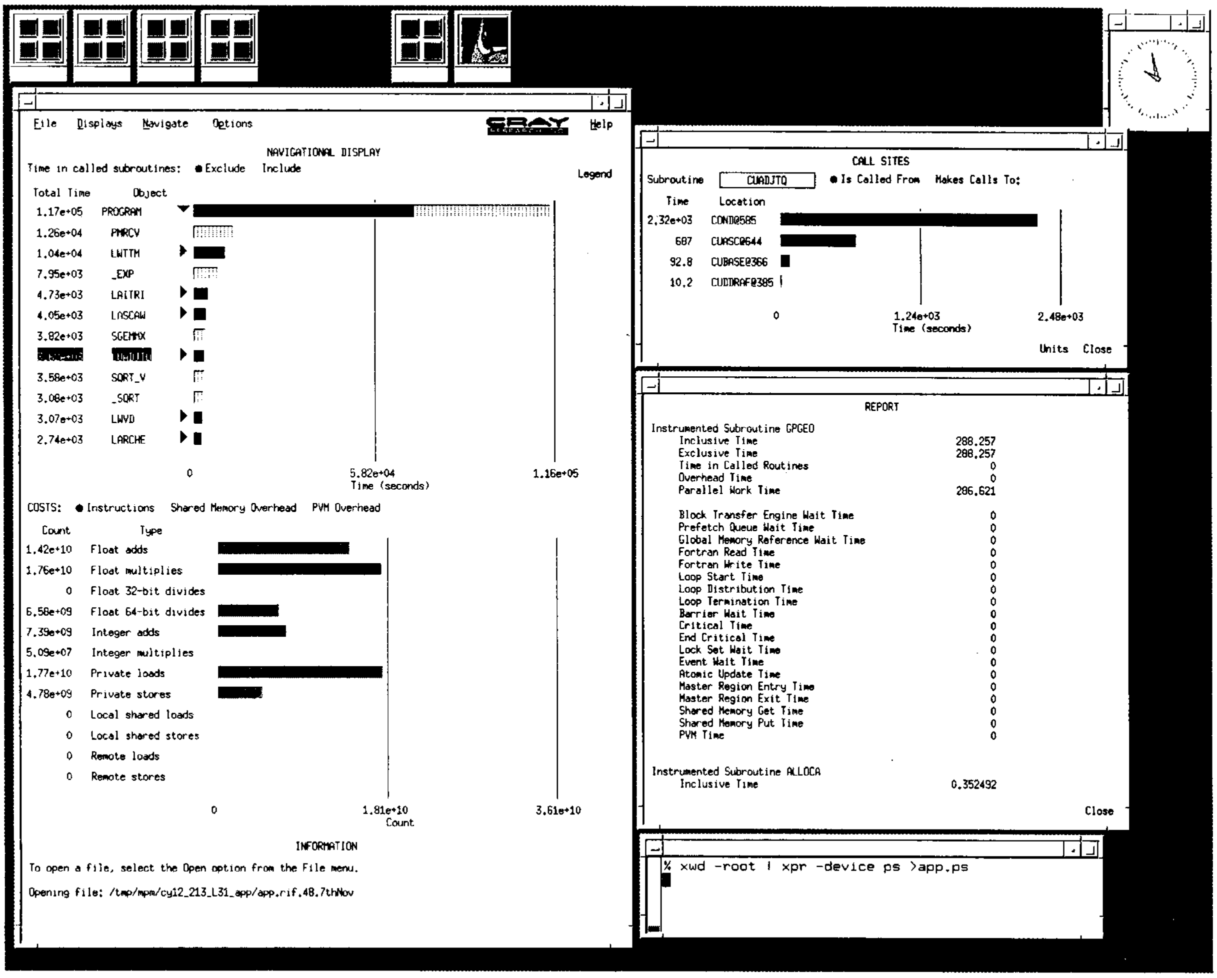


Fig. 3 APPRENTICE showing profile displays from IFS

I/O

It is important to remember that the operating system available on the T3D nodes is a 'kernel' system, meaning that very few OS functions are provided. All other Unix functions are provided by the front end YMP-2E UNICOS system. This can lead to substantial overheads. An advantage of the kernel approach is that it occupies only about 4 MBytes of local memory (out of 64 MBytes).

Fortunately, Fortran I/O statements such as READ and WRITE translate into system I/O requests only after passing through a library buffering stage. Small record transfers are blocked/deblocked by the library software which executes on the T3D. Therefore, for good performance, it is important not to disable this buffering by, for example, utilising an unblocked file format (assign -u). Unblocked I/O avoids the cost of blocking/deblocking but generates a system call for every user record. **assign** statement options may be used to provide automatic data conversion between T3D and YMP. The article in Newsletter No. 64 gives detailed information on useful I/O optimising features many of which remain relevant on T3D.

It is worth pointing out that inter-node communication on T3D is very much faster that I/O. However, message passing from the YMP-2E to T3D nodes is slower than doing I/O directly on the nodes.

Tools

Cray provides a number of tools to aid development and optimisation of T3D programs. For debugging, TotalView is supported on the T3D (as well as on other Cray systems), and for profiling, Apprentice is provided. An example of Apprentice output from an IFS run is shown in Fig. 3, giving quite detailed information on the computations within any parts of the code. This is clearly useful during the optimisation phase of program development.

For applications with complex communications, a public domain tool called ParaGraph is available to visualise the communication and compute portions. Fig. 4 shows time on the X axis. There is a horizontal line showing processor status for each participating PE. The interconnecting lines show messages passing between the PEs.

All of these tools have been used with some success as part of the IFS parallel implementation and optimization on the T3D.

Performance

No matter how much effort is put into parallelising a serial code, the single node performance is going to be a crucial factor in the overall parallel performance. So far, very few real applications have achieved more than about 10% of peak speed (i.e. 15 Mflops per node) on a T3D. This is partly due to the micro-processor design (small data cache), the immaturity of the compiler, and our limited experience in being able to write optimal code for cache based microprocessors. The IFS single node performance has improved over a period of some months from less than 10 Mflops to

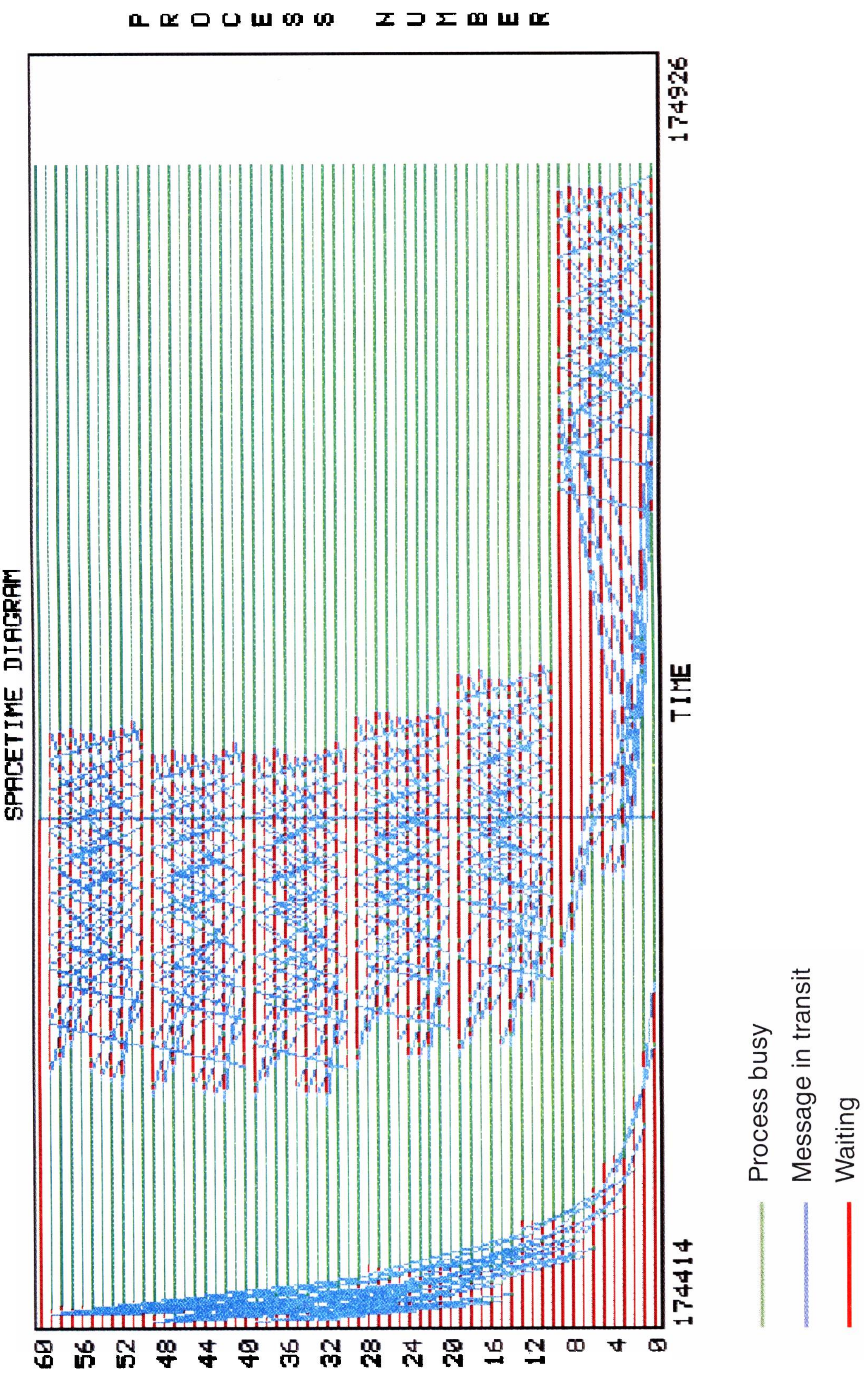


Fig. 4 PARAGRAPH visualisation of message passing

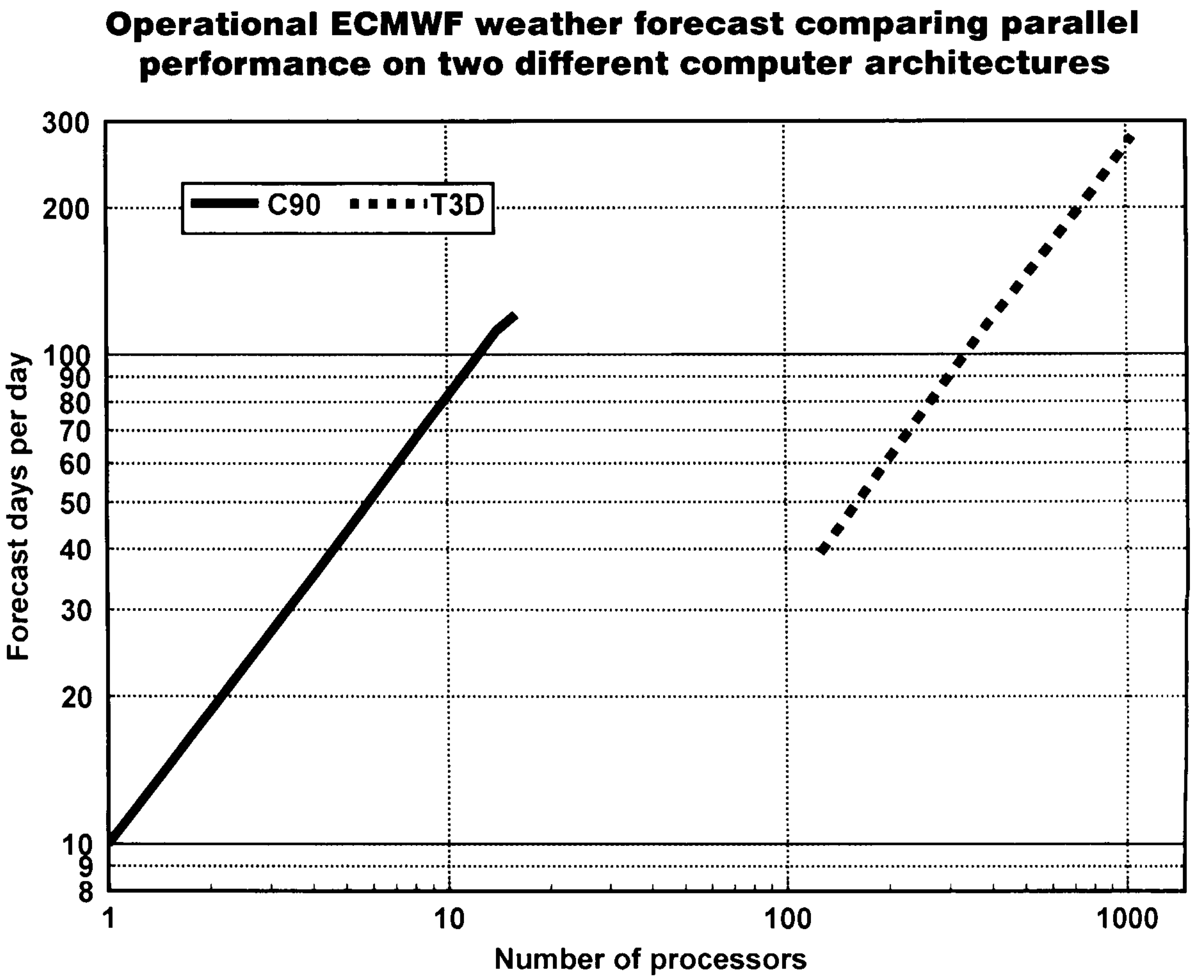


Fig. 5 IFS performance on T3D and C90

almost 15 Mflops. This improvement is due entirely to minor code changes and incorporation of optimised library routines. Further gains due to compiler improvements are expected soon when we invoke new optimising features in the latest compiler release.

Fig. 5 shows IFS performance plotted as forecast days per compute day against the number of PEs utilised. The good news is that the problem scales well, i.e. adding more PEs provides the expected improvement in run time. However, due to poor single node performance, about 500 T3D PEs are needed to perform as well as the C90.

- David Dent, Lars Isaksen, George Mozdzynski

* * * * * * * * * * * *

ECMWF COMPUTER SYSTEM: STATUS AND PLANS

(This article is based on a recent talk given by G.-R. Hoffmann to ECMWF staff.)

This past year has seen an upgrade to the IBM Data Handling system in January, upgrades to the Cray C90 in the summer, and the bringing into service of the Cray T3D system. Towards the end of the year Council approved the financing of a replacement for the data handling system (over the period 1995-98), and of a replacement of the Cray C90 and T3D in 1996.

On the operational forecast side this year has been the best ever, only three times was the forecast delayed more than one hour.

In terms of Cray account units, Member State usage of the Cray C90 has risen to 12%, largely as a result of the optional projects and of the ECSN study.

Cray C90

As mentioned above, the system was upgraded in the summer to full memory with full banking (256 Mwords). In addition, extra disks were added, some in the form of RAID technology. These new RAID disks (DA 301, 5 GB per unit) have proved very reliable, hopefully solving problems we have had in the past with unreliable disks.

Over the past year the availability and MTTI (mean time to interruption) have been erratic. Spells of several weeks of good performance have been marred by periods where several parts have failed in quick succession. Despite extensive studies no overall cause for these bad periods has been found, a high level joint task force with Cray is still studying the issues. Recently, the MTTI has improved to about 200 hours, but this is still not as good as some sites which report MTTIs in the region of 1000-2000 hours.

On a much more positive side the CPU utilisation is now excellent, in fact it is the best we have recorded on any Cray system at the Centre, past or present. From a level of 60% in 1992 the user CPU utilisation has risen to over 90% today (e.g. 93.8% for the month of November). This has been due to three main factors:

- optimisation of key jobs, especially their I/O
- introduction of UNICOS 8
- extra main memory.

As a result present job turnaround seems to be acceptable to all concerned.

Cray T3D

Initially approved by Council in June 1993, the Centre gained access to a T3D in Minneapolis in November 1993 to begin experiments in transferring the IFS model to an MPP system. Then in August 1994 ECMWF took delivery of its own Cray T3D, details of which have been given in a previous Newsletter article (No. 67, p. 17-26).

So far, its availability has been good, over 99% since installation. The first major task for this machine will be to run the Ensemble Prediction System, both in operational and research modes.

Experiments with the IFS code show that this code scales very well. For example, when run on a 1024 processor system, the speed-up obtained was just over 900. The initial single node performance was low (8.6 MFlops/sec), but with a modest amount of work this has already risen to 14.7 MFlops/sec, and further work promises to push it higher.

Data handling system

In January, the IBM ES-9000 was upgraded to a model 720 containing 6 CPUs and 384 MB of main memory. More recently, some of the disks have been upgraded so there is now 300 GB of on-line disk.

Its availability continues to be excellent (over 99% user availability). The daily amount of data being transferred to other systems is around 100 GB/day, much as predicted. This is a comfortable level, leaving some spare capacity for when the load will increase after the Cray C90 is replaced.

The total volume stored on the system continues to rise steadily, now being almost 24 TB. More worrying is the number of files, currently at 5.6 million. Neither CFS nor any other system of this nature can handle a very large number of files. Hence ways are being studied of storing more data per file, reducing the total number of files overall.

Workstations and servers

Some further SGI systems are on order and will be used to replace the old SUN SPARC 1 and 1+ machines. Two SGI Challenge servers will be bought shortly, so that we will then have two (2-processor) machines for Member States and Operations Department, plus one (4-processor) machine for Research Department. This will release an existing SGI Crimson server which will then go to the re-analysis project.

In addition, some of the existing IBM workstations will be enhanced so that they can be used to test the concept of a "high availability" server for future operational support.

The existing SGI servers have continued to operate reliability, with the exception of a single machine which has suffered a series of annoying but short interruptions.

The Cray YMP-EL server suffered a bad spell of reliability earlier this year when attempts were made to add extra disk capacity. However, it now appears to be settling down again, recent overall availability being just below 99% (13 week moving average).

Networks

Despite its apparent complexity the internal network continues to operate smoothly. A new HIPPI based router (GigaRouter from NetStar Corp.) is being purchased following a recent trial. It proved easy to instal and has run very reliably over its trial period.

The major change on the Member State network has been the upgrading of all lines but one to 64 Kb/s. This has led to increased traffic both incoming and outgoing. In addition, the following has been or will be done shortly:

- a 64 Kb/s link to Iceland has been installed
- a 9.6 Kb/s link to Hungary is operational
- the line to Météo-France (Toulouse) has been upgraded to 128 Kb/s.

Miscellaneous

The VAX cluster consists of two 4100s, an 8250 and a 3100. The VAX 8250 will go at the end of 1994 when the last remaining Member State stops using ECNET level 3 protocols.

Of the various MPP projects:

- RAPS (Real applications on parallel systems) is ongoing with the next meeting planned for May 1995 in Austria

- GP-MIMD2 has ceased as far as ECMWF is concerned

- PPPE (Portable Parallel Programming Environment) has been extended by one year to June 1996.

Plans

The year 1995 will be dominated by two events:

(i) the selection of a replacement data handling system, with the installation of phase 1 towards the end of the year

(ii) the ITT and selection of the C90 replacement.

Then in 1996 we will see the installation of the Cray C90 replacement, and the termination of the present Cray C90 and T3D services. Phase 1 of the new data handling system will also come into operation. Thus, from the user point of view, 1996 could be a year of major change.

Looking ahead further it is planned to bring phase 2 of the new data handling system into operation in 1997, with phase 3 in 1998. At that point (mid 1998) the present IBM based data handling system will be switched off. Also in 1998 the C90 replacement system will double its capacity.

Finally, some other projects to be investigated over the next four years are:

- smart card access for Member State users
- use of managed networks for the Member States
- possible remote use of MARS and METVIEW
- operational use of servers, including a possible batch service
- introduction of DCE/DFS for file sharing.

Conclusions

From a user point of view, 1994 has seen improvements in the C90 service, the addition of the T3D and an upgrading of the IBM. For 1995 we expect a year of consolidation, before (possibly) large changes take place in 1996.

- Andrew Lea

* * * * * * * * * * * *

COMPUTER RESOURCE ALLOCATION TO MEMBER STATES IN 1995

Table 1: Allocation of CRAY resources and data storage to Member States in 1995 (including a 10% reserved allocation for Special Projects)

MEMBER STATE	CRAY C90 (kunits)	CRAY T3D (kunits)	Data (Gbytes)
Belgium	129	33	46
Denmark	108	28	39
Germany	572	147	205
Spain	225	58	81
France	446	115	160
Greece	90	23	32
Ireland	80	21	29
Italy	416	107	149
Yugoslavia*	98	25	35
Netherlands	159	41	57
Norway	101	26	36
Austria	118	31	43
Portugal	88	23	32
Switzerland	143	37	51
Finland	109	28	39
Sweden	141	36	51
Turkey	102	26	37
United Kingdom	385	99	138
Special Projects	390	9	140
TOTAL	3900	913	1400

* In accordance with UN Security Council Resolution 757 (1992) of 30 May 1992, the Council instructed the Director to suspend the telecommunications connection to Belgrade with immediate effect. This took place on 5 June 1995. As a consequence, no operational products are disseminated to Belgrade and access to the Centre's computer system is not available to Belgrade.

* * * * * * * * * * * *

Table 2: Special Project Allocations 1995

Member State	Institution	Project title	1995 Resources requested		1995 Resources proposed	
			Cray kunits	CFS Gbytes	Cray kunits	CFS Gbytes
Continuation Projects						
Austria	Institut für Meteorologie und Geophysik, Vienna (Hantel)	Diagnostic tests of the ECMWF physics package	1.3	1.5	1.3	1.5
	Institut für Meteorologie und Geophysik, Vienna (Hantel)	Diabatic heating of the global atmosphere	0.5	1	0.5	1
Belgium	Belgian Institute for Space Aeronomy (Fonteyn)	The development of an interactive chemical transport model based on a low resolution version of the ECMWF weather forecast model	24	2	17.6	2
France	CNET/CETP (Eymard)	Use of spaceborne measurements to infer surface fluxes	0.8	1.4	0.8	1.4
	METEO France, Toulouse (Tarrieu)	AVISO. Study of surface winds and surface fluxes at the interface ocean/atmosphere	5	1	5	1
	LMD, Palaiseau (Duvel)	Validation of spatial and temporal variabilities of the ECMWF model	3	2	3	2
	L.A.M.P., Aubière (Chaumerliac)	Chemistry, cloud and radiation interactions in a meteorological model	2	2	2	2
Germany	Institute for Geophysics and Meteorology (Speth)	Interpretation and calculation of energy budgets	2	6	2	6
	GKSS, Geesthacht (Raschke/Rockel)	Parametrization of radiation and clouds for use in general circulation models	1	0.2	1	0.2
	MPI Hamburg (Roeckner)	Modelling the earth's radiation budget and evaluation against ERBE data	60	15	43	13.8
	MPI, Hamburg (Bengtsson)	Numerical experimentation with a coupled ocean/atmosphere model	130	30	93	27.6
	MPI, Hamburg (Bengtsson)	Simulation and validation of the hydrological cycle	110	15	79	13.8
	MPI, Hamburg (Hasselmann)	Extraction of two-dimensional wave spectra from ERS-1 SAR wave mode image spectra	2	1.5	2	1.5

Member State	Institution	Project title	1995 Resources requested		1995 Resources proposed	
			Cray kunits	CFS Gbytes	Cray kunits	CFS Gbytes
	University of Munich (Wirth/Egger)	The behaviour of cut-off cyclones in ECMWF analysis: impact of diabatic processes on their development and decay	1	1	1	1
	FU, Berlin (Fischer/Thoss)	Comparison of the ECMWF cloud scheme with multi-spectral satellite data in the Baltic Sea	1.5	5	1.5	5
	Joh.Gutenberg Univ, Mainz (Zimmermann)	Computation of correlation dimensions from observed data	1	0.5	1	0.5
Italy	Istituto per lo Studio della Dinamica delle Grandi Masse, Venezia (Cavaleri)	Testing and applications of a third generation wave model in the Mediterranean Sea	3	1.5	3	1.5
Netherlands	KNMI, De Bilt (Siegmund)	Tropopause folding analysis	1	1	1	1
	KNMI, De Bilt (Können)	Climate scenarios	0.3	6	0.3	6
	KNMI (van Velthoven)	Chemistry and transport studies with a 3D off-line tracer model	25	15	11	13.8
Norway	Geophysical Institute, University of Bergen (Grønås/Kvamstø)	Cloud parametrization in general circulation models	1	0.1	1	0.1
Sweden	SMHI, Norrköping (Källén)	The HIRLAM 3 project	20	5	20	5
New projects						
Netherlands	KNMI (Komen)	North Sea wave climate	15	4	15	4
Sweden	Uppsala University (Andrén)	Evaluation and development of PBL parametrization schemes suitable for GCM's and long-range forecasts	50	10	36	9.3
	SMHI (Funkquist)	HIROMB (High resolution operational model for the Baltic Sea)	10	5	10	11
		Total	470.4	132.7	351.0	126.0
		Reserve (to be allocated by ECMWF)	39.0	14.0	39.0	14.0
		Overall total	509.4	146.7	390.0	140.0
		Amount available	390.0	140.0		

SIXTH WORKSHOP ON USE OF PARALLEL PROCESSORS IN METEOROLOGY 21 - 25 NOVEMBER 1994

The computing requirements in meteorology are continuing to increase at a very high rate. The development of 4-dimensional variational data assimilation and ensemble prediction systems, and the ongoing refinements of models will demand computing capacity many times higher than is available today.

Since the only feasible and affordable way to achieve such high computing performance is through the use of massively parallel systems, the interest in the workshop series on use of parallel processors in meteorology has been increasing steadily. More than 100 researchers in meteorology, climate and computing science attended the sixth workshop on this theme. Their contributions show that the use of parallel systems has now matured to such an extent that operational use of MPPs is eminent. It can be shown that programming methods exist to parallelise both grid point and spectral models efficiently. Agreed strategies for the different model types seem to be emerging and the whole field has grown up from its infancy ten years ago. The motto of the workshop "*Coming of Age - Use of Parallel Processors in Meteorology*" therefore reflected the current situation correctly.

Over 40 presentations were given by invited lecturers, computer vendors, managers of meteorological computing centres and, especially, researchers in the field. Written versions of the lectures will shortly be published by World Scientific Publishing Ltd. thus continuing the series of proceedings of these workshops.

The workshop ended with a well attended discussion where not only the current status was looked at but also views on future trends were exchanged. Among the questions addressed were the following:

- What future role will traditional vector computers play?
- What will the structure of a possibly heterogeneous computing environment look like?
- Are future MPP systems with a bigger number of processors or are fewer, faster processors more desirable?
- Will it be possible to achieve sufficiently good load balancing, if more processors become available?
- How important is reproducibility on MPP systems for different classes of meteorological usage (climate modelling, operational forecasting, research experiments, etc.)?
- Which degree of portability of the software is desirable and affordable?

- Are the current programming environment and the available tools suitable?

- Will future MPP hardware be sufficiently reliable and will the operating systems be resilient and flexible enough to sustain daily operational and research use?

All the above questions were discussed in a distinctively positive atmosphere - areas of concern, yes, but unsurmountable problems, no.

A summary of the discussion will be included in the forthcoming publication of the proceedings.

Given the undiminished enthusiasm for the field all over the world, yet another workshop is anticipated in 1996.

- Norbert Kreitz

* * * * * * * * * * * *

ECMWF CALENDAR 1995

6 - 24 March	Computer user training course
13 (pm) - 17 April	*ECMWF holiday*
26 April - 23 June	Meteorological training course:
	Met 1 (26 April - 5 May):
	Numerical methods, adiabatic formulation of models
	Met 2 (9 - 19 May):
	Parametrization of diabatic processes
	Met 3 (22 May - 2 June):
	Data assimilation and use of satellite data
	Met 4 (5 - 9 June):
	General circulation, systematic model errors and predictability
	Met 5 (12 - 23 June):
	Use & interpretation of ECMWF products
2 - 3 May	Technical Advisory Committee, 21st session
4 - 5 May	Finance Committee, 54th session
8 May	*ECMWF holiday*
26 - 29 May	*ECMWF holiday*
4 - 5 July	Council, 42nd session
28 August	*ECMWF holiday*
4 - 8 September	Seminar: Predictability
11 - 13 September	Workshop: Non-linear aspects of data assimilation
25 - 27 September	Scientific Advisory Committee, 24th session
11 - 13 October	Technical Advisory Committee, 22nd session
17 - 18 October	Finance Committee, 55th session
24 - 27 October	Workshop: Surface fluxes (WGNE)
30 Oct - 3 November	WGNE meeting
6 - 8 November	Workshop: Semi-Lagrangian methods
13 - 17 November	Workshop: Meteorological Operational Systems
4 - 5 December	Council, 43rd session
25 - 27 December	*ECMWF holiday*

* * * * * * * * * * * *

ECMWF PUBLICATIONS

Technical Memoranda:

No. 205	Linearity of optimal perturbation time evolution and sensitivity of ensemble forecasts to perturbation amplitude
No. 206	Extended versions of the convective parametrization scheme at ECMWF and their impact on the mean and transient activity of the model in the Tropics
No. 207	Use of the conventional surface observations in three-dimensional variational data assimilation
No. 208	The singular-vector structure of the Atmospheric General Circulation
No. 209	Report of the eighth meeting of Member State Computing Representatives, 27-28 September 1994
No. 210	Quasi-continuous variational data assimilation

Technical Reports:

No. 73	Raw HIRS/2 radiances and model simulations in the presence of clouds
No. 74	Ocean wave forecasting in the Mediterranean Sea. A verification study in the Spanish Coastal Zone

Forecast and Verification Charts to December 1994

Meteorological Bulletin No. M1.4/4: BUFR User Guide and Reference Manual

* * * * * * * * * * * * *

INDEX OF STILL VALID NEWSLETTER ARTICLES

This is an index of the major articles published in the ECMWF Newsletter series. As one goes back in time, some points in these articles may have been superseded. When in doubt, contact the author or User Support.

* * * * * * * * * * * *

USEFUL NAMES AND 'PHONE NUMBERS WITHIN ECMWF

		Room*	Ext.**
DIRECTOR	- David Burridge	OB 202	2001
DEPUTY DIRECTOR and HEAD OF RESEARCH DEPARTMENT	- Anthony Hollingsworth	OB 119A	2005
HEAD OF OPERATIONS DEPARTMENT	- Massimo Capaldo	OB 010A	2003
ADVISORY: Available 9-12, 14-17 Monday to Friday			2801
Other methods of quick contact:	- Telefax (+44 1734 869450)		
	- VMS MAIL addressed to ADVISORY		
	- Internet mail addressed to Advisory@ecmwf.int		
REGISTRATION			
Project Identifiers	- Pam Prior	OB 225	2384
User Identifiers	- Tape Librarian	CB Hall	2315
COMPUTER OPERATIONS			
Console	- Shift Leaders	CB Hall	2803
Console fax number	- +44 1734 499 840		
Fault reporting	- Call Desk	CB Hall	2303
Service queries	- Call Desk	CB Hall	2303
Tape Requests	- Tape Librarian	CB Hall	2315
ECMWF LIBRARY & DOCUMENTATION - DISTRIBUTION	- Els Kooij-Connally	Library	2751
LIBRARIES (ECLIB, NAG, etc.)	- John Greenaway	OB 226	2385
METEOROLOGICAL DIVISION			
Division Head	- Horst Böttger	OB 007	2060
Applications Section Head	- John Hennessy (acting)	OB 014	2400
Operations Section Head	- Bernard Strauss	OB 328	2420
Meteorological Analysts	- Andreas Lanzinger	OB 314	2425
	- Ray McGrath	OB 329	2424
	- Anders Persson	OB 315	2421
Meteorological Operations Room	-	CB Hall	2426

		Room*	Ext.**	Bleeper
COMPUTER DIVISION				
Division Head	- Geerd-R. Hoffmann	OB 009A	2050	150
Systems Software Sect.Head	- Claus Hilberg	OB 104A	2350	115
User Support Section Head	- Andrew Lea	OB 227	2380	138
User Support Staff	- Antoinette Alias	OB 224	2382	154
	- John Greenaway	OB 226	2385	155
	- Norbert Kreitz	OB 207	2381	156
	- Dominique Lucas	OB 206	2386	139
	- Pam Prior	OB 225	2384	158
Computer Operations Section Head	- Peter Gray	CB 023	2300	114
Security, Internal Networks and Workstation Section Head	- Walter Zwieflhofer	OB 140	2352	145
GRAPHICS GROUP				
Group Leader	- Jens Daabeck	CB 133	2375	159
RESEARCH DEPARTMENT				
Computer Co-ordinator	- David Dent	OB 123	2702	

* CB - Computer Block
OB - Office Block

** The ECMWF telephone number is READING (01734) 499000, international +44 1734 499000, or direct dial to (01734) 499 + last three digits of individual extension number, e.g. the Director's direct number is (01734) 499001.

DEC MAIL: Contact scientific and technical staff via VMS MAIL, addressed to surname.

Internet: The ECMWF address on Internet is **ecmwf.int**
Individual staff addresses are firstname.lastname, e.g. the Director's address is David.Burridge@ecmwf.int