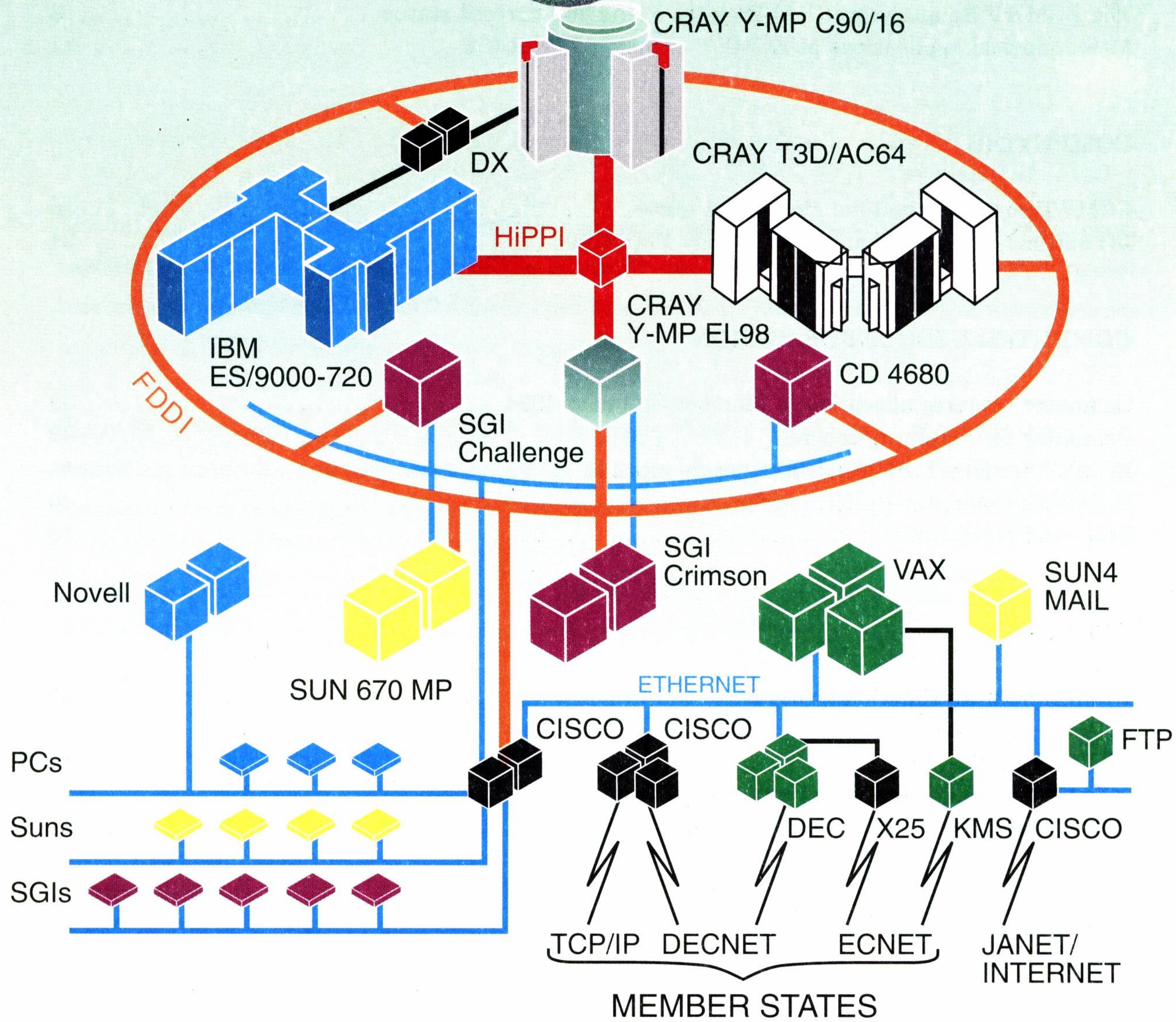
ECMWF Newsletter

Number 64 - December 1993



Shinfield Park, Reading, Berkshire RG2 9AX, England. Telephone: U.K. (0734) 499000
International (+44 734) 499000, Telex: 847908 ECMWF G, Fax: (0734) 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

Editorial 1

METEOROLOGICAL

Changes to the operational forecasting system 2
The ECMWF Re-analysis (ERA) Project - Plans and current status 3
Meteorological applications at ECMWF utilizing EMPRESS 11

COMPUTING

ECMWF computer system: status and plans 16
I/O optimisation techniques on the Cray YMP C90 21

COMPUTER USER INFORMATION

Computer resource allocation to Member States in 1994 39
Computer user training course 42
ECMWF produced computer user documentation 44
ECMWF's Computer Bulletin series 46
Still valid News Sheets 48

GENERAL

The meteorological training course 50
ECMWF calendar 51
ECMWF publications 52
Index of still valid Newsletter articles 53
Useful names and 'phone numbers within ECMWF 56

COVER: ECMWF computer configuration, mid-1994 (see article on page 16)

This Newsletter is edited and produced by User Support.

The next issue will appear in March 1994.

Several articles in this issue give information on plans for 1994: those of the Computer Division, especially concerning developments of the computer configuration; the allocations of computer resources to Member States, and for special projects, in 1994, and the schedule of training courses to be held at ECMWF in the course of the year.

There is a first article on the ECMWF Re-analysis Project (ERA), recently embarked upon, describing its status and planned programme of work. This project will rely on the EMPRESS database, which is also the subject of an article in this Newsletter.

CRAY users will find useful suggestions for the optimisation of use of computing resources in an article detailing I/O optimisation techniques for the Cray Y-MP/90.

There are also articles updating information on computer user documentation and the computer bulletin series produced by ECMWF.

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

CHANGES TO THE OPERATIONAL FORECASTING SYSTEM

Recent changes

On 5 October 1993, a change to the boundary layer parametrisation was introduced, further to the change of 4 August 1993. The profiles of 2 metre temperature in stable conditions were revised to reduce the coupling between the skin temperature and the temperature of the lowest model levels. The cold bias of the predicted night-time temperature was reduced by this change.

On 11 November 1993, the pre-processing was modified to correct radiosonde observations for systematic biases prior to their use in the data assimilation. The correction is applied to the radiosonde stations which present a significant bias in geopotential height due to radiative effects on the sonde (long wave cooling and short wave heating, mainly in the stratosphere).

Planned changes

The IFS code is expected to be implemented in the first quarter of 1994 (no meteorological impact).

The 3-D variational analysis is planned for implementation in the second or third quarter of 1994.

- Bernard Strauss

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

THE ECMWF RE-ANALYSIS (ERA) PROJECT - PLANS AND CURRENT STATUS

INTRODUCTION

The European Centre for Medium-Range Weather Forecasts (ECMWF) has provided data from its archive for Meteorological Research for many years. The ECMWF Re-Analysis (ERA) project was devised in response to the wishes expressed by many users of these data. It is funded or supported by the ECMWF Council, the European Community, the Universities of California and Maryland (USA), the Japanese Meteorological Agency (JMA), and Cray Research Incorporated. In addition contributions in terms of data and assistance have been forthcoming from many quarters, including ECMWF Member States, NMC Washington, NCAR, and the Australian Bureau of Meteorology.

Objectives

The goal is to develop a global atmospheric data assimilation system to analyse in an internally consistent way the record of atmospheric data (from the earth's surface to 10 hPa) for the period 1979-1993, and then to produce a validated and documented 15-year data-set of assimilated data at high horizontal and vertical resolution. It is planned that these new analyses will be ready by the end of 1995. This data-set will provide global three-dimensional descriptions of the velocity, temperature, geopotential and humidity fields for the atmosphere. In addition, analyses of surface parameters (temperature, pressure, soil moisture, etc.), surface fluxes of heat, moisture, radiation and other diagnosed quantities will be included.

ORGANISATION

The Data Assimilation System

The ERA data-sets will be derived from a special version of the ECMWF operational data assimilation system. In this system analyses of atmospheric variables are produced by optimal interpolation of irregularly-spaced observational data to produce increments to be added to global short-range forecasts. Since this system is shortly to be replaced by a three-dimensional variational system, a strategy is being followed to enable the new system to be used if it is sufficiently proven. The short-range forecasts will be produced by a global spectral model with a triangular truncation at wave number 106. The ERA production system, once established, will remain consistent, resulting in data unaffected by the periodic changes introduced over the years within a daily operational system.

The real-time ECMWF data collection (which has a 3-day cut-off) will be the basis for the observational data-set. This will be augmented by the following additional data:

* First Global Geophysical Experiment (FGGE) level II-B data
* Alpine Experiment (ALPEX) level II-B data
* NOAA Climate Analysis Centre and UK Meteorological Office delayed mode sea-surface temperature (SST) analyses, which will serve as the oceanic lower boundary condition for the atmospheric data assimilation
* The COADS ship data-set
* additional radio-sonde data-sets available from National Meteorological Centres
* NOAA satellite radiance data for the 15-year period.

The development of an assimilation system suitable for the fast rate required to generate the assimilated data-sets within the project's time-scale is currently under way. This involves software and data handling development, coupled with experimentation to select the appropriate version of the forecasting model and the spatial analysis procedure employed in the assimilation system.

Data-set production

Data-sets will be produced in three main phases:

Phase A1 - Pilot analyses for June 1985

Phase A2 - Analyses for 1979 - the FGGE year for which there exists a number of comparable data-sets

Phase A3 - Analyses for 1980-1993.

The pilot analyses and the FGGE year analyses will be used by the production team at ECMWF, and by validation teams elsewhere, to critically assess the quality of the analyses, before embarking on the remaining period.

Fig. 1 illustrates the production methodology. Observational data will be fetched from archives, one month at a time, organised in data base form, and checked before use. Immediate results from the 4 times daily first guess forecasts and analyses will be retained on-line to facilitate initial diagnosis and checks. Suitable results from each forecast/analysis cycle will be entered into a data base prior to the generation of monthly statistics and addition to the archive. The analysis phase will require the ECMWF 16-processor Cray C90. Silicon Graphics file servers will be used for the on-line data bases and pre- and post-processing.

Validation

The validation exercise is designed to provide rapid, critical and comprehensive evaluation of the data-sets as they are produced, to detect defects as soon as they arise. This will be done through diagnostic studies of the analyses as they become available. Validation effort will be shared with scientists from the funding agencies, and with ECMWF's partners within the terms of the EC funded project.

The specific diagnostic tasks are:

V1 Validation against global observations used and not used in the analyses; validation through regular forecast experiments. (ECMWF and Spain)

V2 Validation by comparison with earlier analyses from ECMWF and re-analyses from NMC (Washington) - this will be particularly important at the very start of the project; validation of global energetics and the hydrological cycle; validation using ocean models by using the diagnosed surface fluxes to drive an ocean model - the resulting evolution of the SST will be compared with XBT data. (Max-Planck-Institute, Germany)

V3 Comparison of clear sky outgoing radiances diagnosed from the analyses with measurements from the Earth Radiation Budget Experiment (ERBE) using simulations with the SAMSON system - this will help validate in particular the analysed humidity field. (U.K.)

V4 Validation by using surface fluxes to drive an Alpine snow-mantle model - the resulting evolution of the mantle will be compared with observations. (France)

V5 Verification of the analyses over Greenland against observations. (Denmark)

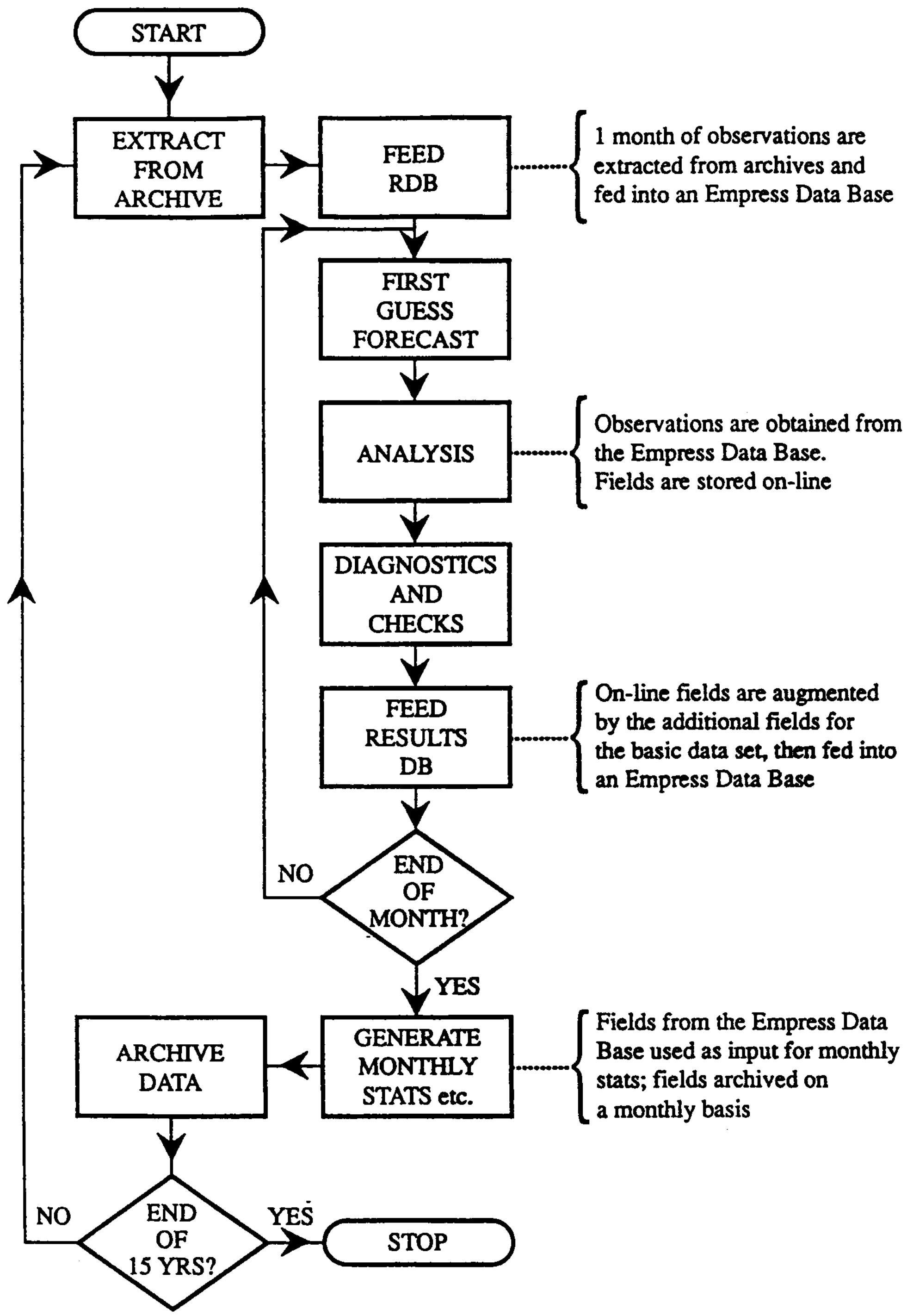


Fig. 1: Production Methodology

V6 Validation of the analysed Planetary Boundary Layer structure using data at the Cabauw Site in The Netherlands. (Netherlands)

V7 Diagnostics of atmospheric low-frequency variability deduced from the analyses will be compared with similar diagnostics from previous analyses produced by ECMWF and re-analyses produced by NMC Washington. (University of Bologna, Italy)

V8 Validation of the analyses over the Mediterranean region. (Instituto per lo Studio delle Metodologie Geofisiche Ambientali, Modena, Italy)

These studies will provide an essential component of the scientific documentation.

Archives and Data Services

The additional observational data, and a comprehensive set of re-analysis results will be added to the ECMWF archive. Appropriate data will be made available through ECMWF's Data Services.

Four main validated and documented data sets will be produced, forming the basis for data services:

ERA Basic Analysis Data Set

ERA Advanced Analysis Data Set

ERA Precipitation, Moisture, and Energy Flux Data Set

ERA Climate and Statistics Data Set.

The Basic Data Set will provide selected variables in compact form at low resolution and will be particularly useful for users with limited data processing resources. It will contain geopotential, temperature, vertical velocity, two components of wind, and relative humidity on a 2.5 degree grid at 15 standard pressure levels, together with a number of surface and diagnostic parameters. One year of such data would require 10 standard 6250 bpi magnetic tapes.

The Advanced Data Set will provide data at higher resolution in time and space, giving access to the results of the re-analysis at the horizontal and temporal resolutions of the generating system. There will be options for users to obtain these data either on the grids used by the generating system (T106 spherical harmonics for upper-air data, N80 Gaussian grid for surface and diagnostic parameters), or interpolated to a regular latitude/longitude grid of the user's choice. The parameters supported will be similar to those described for the basic data set above. In addition to the 15 standard pressure levels, data will be available on the 31-level hybrid vertical co-ordinate system used by the generating models. One year of 15 pressure level data would require 52 standard 6250 bpi magnetic tapes, or approximately 88 tapes if the 31 model levels were specified.

A number of parameters of interest to the international scientific community are not computed within the analysis, but are potentially available from short-range forecasts which provide first guess information for the analyses, provided those forecasts are sufficiently extended in their range to meet such requirements. The Precipitation, Moisture and Energy Flux Data Set will be generated from such forecasts, and will provide parameters such as surface flux of sensible and latent heat, surface and top of the atmosphere radiation fluxes, and rainfall. These will be retained at model resolution; interpolation to other resolutions will be supported. One year of such data at full resolution would require 10 standard 6250 bpi magnetic tapes.

The Climate and Statistics Data Set will contain monthly and seasonal means data together with a number of statistics which will be generated at the end of each month of analyses during production. These data will enable access to many of the climatological aspects of the results without recourse to the full daily data. Considerable discussion has taken place with atmospheric scientists and potential users to determine the contents of this data set. One year of such data would require approximately 5 standard 6250 bpi magnetic tapes.

STATUS

Project Staff and Management

A team of six people, including two EC fellowship students, is currently occupied full time in carrying out the project.

An internal steering group, chaired by the ECMWF Director, reviews the project management. An external advisory group provides advice and facilitates liaison with the external scientific community and the potential users of the results.

Establishment of the Data Assimilation System

A programme of experimentation has been undertaken to determine to what extent the data assimilation system to be used should differ from that of the Centre's daily operations. This is necessary because the system, once established, should remain fixed for the full re-analysis production, and because the basic horizontal resolution to be used will be T106, not the operational resolution of T213. Results from these experiments to date indicate:

a) the use of 31 vertical levels rather than 19 is beneficial in some aspects;

b) the use of the "first guess at appropriate time" scheme provides insufficient benefit to the re-analysis to justify its additional cost in terms of computational resources;

c) mean orography (as opposed to envelope orography) will be used.

Experimentation carried out within the ECMWF's Satellite Group was followed closely to establish how best to use the cloud cleared radiance data. Also considered were the optimal treatment of sea surface temperatures and sea ice coverage data, and the inclusion of the Integrated Forecast System and the Three Dimensional Variational Analysis schemes being developed at ECMWF.

Data acquisition

Work is at an advanced stage to convert FGGE and ALPEX data to the representational forms used within the ECMWF data assimilation system.

The sea surface temperature data produced by NMC Washington, the AMIP sea surface temperature data, and the sea surface temperatures produced by the UK Meteorological Office have been acquired. These have been subjected to comparisons, and examined in detail for various events including El Niño periods. Attention has been given to the representation of sea ice within the SST data, to enable decisions to be made as to whether additional sea ice information should be used.

NMC Washington have agreed to make available the COADS ship data-set as part of the co-operation established between the ECMWF and NMC re-analysis groups.

Various National Meteorological Centres, including a number of ECMWF Member States, have offered access to additional radiosonde and AIREP data. These offers will be taken up as manpower resources permit, beginning with those considered likely to have the most impact for the least effort.

NOAA satellite cloud cleared radiance data have been purchased jointly with NCAR. Delivery of these data has begun, starting with the 1979 data and sufficient 1985 data to cover the experimental period, and will continue during the early part of the production phase. There are, unfortunately, gaps in these data (about 8 months are missing in total). To address the problem of these gaps, level I-B data is being obtained, and will be re-processed to obtain the II-B cloud cleared radiances.

Setting Up the Production System

The various components of the production system are being assembled and checked. To date the EMPRESS data base system to be used for handling observations has been established. The Supervisor-Monitor-Scheduler (SMS) is running on the ERA server; the environment for its use has been established and checked on both the Cray and SGI systems. Integration work is now in progress to generate the more comprehensive suites necessary for full production.

- J.K. Gibson, P. Kållberg, A. Nomura, S. Uppala

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

METEOROLOGICAL APPLICATIONS AT ECMWF UTILIZING EMPRESS

Following ECMWF's third "Workshop on Meteorological Operational Systems" in November 1991 it was decided to gain some experience at ECMWF in the use of a "Relational Database Management System" (RDBMS). A commercial RDBMS (EMPRESS) was made available for experimentation on one SUN workstation. After promising results from initial tests, and in order to meet the requirements of the "ECMWF Re-Analysis" (ERA) project, this licence was extended to a site licence for all SUN and SGI workstations. This article describes how the RDBMS is used, or is planned to be used, for applications in the Meteorological Applications Section and the ERA project.

Data Acquisition and Preprocessing

A new data acquisition system has been developed, and is currently run in parallel with the operational data acquisition system, which is running on a VAX-cluster under VMS using an "Indexed Sequential Access Method" (ISAM). The new system extracts meteorological messages from files received from Bracknell and Offenbach stored on VAX disks and accessed via NFS mounts from the UNIX workstations. The messages are written to a "Message Data Base" (MDB) on a UNIX file server. This is achieved by means of a program written in C, making use of the "MR-routine interface" (EMPRESS C Kernel Level Interface). Three new database tables are created every day - one contains information about the messages, extracted from such information sources as the "Global Telecommunications System" (GTS) abbreviated header, with the messages themselves stored as "Binary Large Objects" (BLOBs), one for duplicate data, and one for station information. After three days, statistical information is extracted from the table containing the messages, written to a statistics table, and the complete table is exported and archived. The oldest table is dropped (deleted) when disk space has reached a certain limit.

A new decoding and quality control system is being developed to be front-ended by the new data acquisition. It runs on UNIX workstations and comprises nineteen programs, one for each data type (eg SYNOP, TEMP, AIREP, ...). Each program reads one message at a time from the MDB, processing one observation, one BUFR report or one GRIB or GRID field at a time, packing it into BUFR or GRIB and writing it and related metadata (information about the data) to the "Reports Data Base" (RDB). The decoding and quality control programs are written in FORTRAN and are essentially the same codes as used within the operational system. They have been adapted to call database routines written in C using the "MR-routine interface". The RDB also resides on a UNIX server and is functionally quite similar to the MDB; one new table per data type is created every day.

Data can be extracted by using "Structured Query Language" (SQL) - a simple but standardized, non-procedural database access language.

Some smaller read-only databases have been created, which are used by the preprocessing (eg WMO Vol-A station list, WMO Vol-C bulletin list). Most of the programs to feed data into those databases use "Embedded SQL" (ESQL; ie SQL commands are inserted into C or FORTRAN code and replaced by a pre-compiler with function calls). A few tools for monitoring the dataflow, and if necessary allowing manipulation and correction of messages in the MDB are written using "Fourth Generation Language" (4GL), a procedural programming language combining database access with a graphical interface.

An overview of the new data acquisition and preprocessing system can be found in Fig. 1.

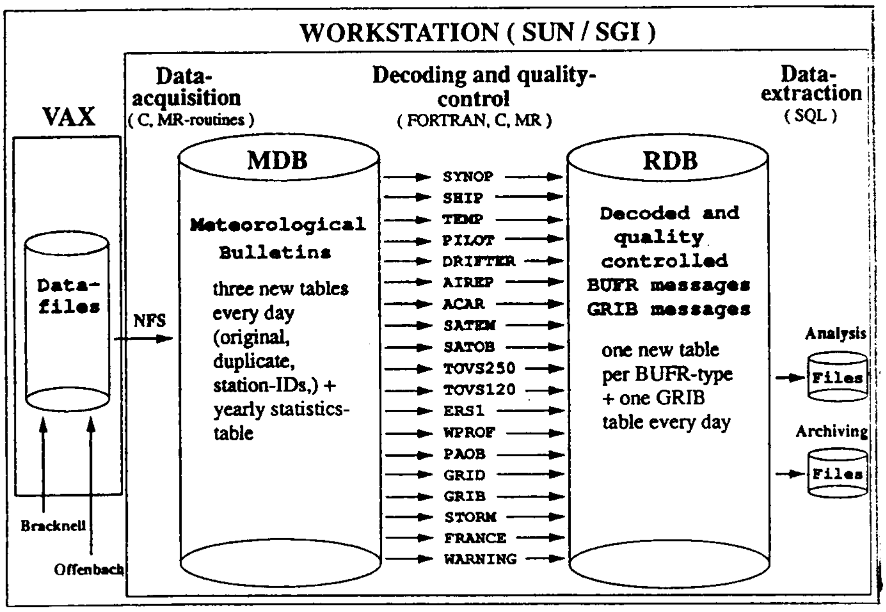


Fig. 1: The new data acquisition and preprocessing system at ECMWF utilizing EMPRESS

The new data acquisition and preprocessing system enables easier access to the data and makes the data flow through the system more transparent. It improves the monitoring of data by facilitating the identification of messages which are no longer received, or which are transmitted without prior notice. Statistical information about the data flow for a time range of up to one year is made available.

MARS Caching System

The "ECMWF Meteorological Archive and Retrieval System" (MARS) has, from its inception, provided more than the basic ability to retrieve data from the archives. An important function of MARS has been to provide all applications with an interface to data such that no distinction is made between those data which reside on-line, and those for which archive retrieval is necessary. When MARS was made available for ECMWF workstation users, a database cache was introduced. Data which has to be retrieved from the archives may not only be supplied to the user, but also ingested into an on-line database. Subsequent requests for data which have recently been retrieved are satisfied from this database. This has the potential to reduce network traffic to the mainframe computers, and to provide more rapid access to those data which are currently "popular", and which by virtue of their "popularity" are cache-resident.

The concept within MARS is simply to regard all data as being database resident, and to develop appropriate interfaces to the various database retrieval methods. Even the MARS archives in CFS are considered as a database, albeit with a unique access method.

For one possible type of cache database, the "MARS Caching System" (MCS), support has been developed in a general way, using EMPRESS. Although MCS could be used for GRIB data (eg analysis and forecast fields) and BUFR data (eg observational reports), only fields are ingested at the moment. This is due to the different way of retrieving BUFR observations and the comparatively small size of each BUFR data entity compared with the much larger sizes of the GRIB entities. When feeding data into this cache database, information is extracted from the self-descriptive data representations. An appropriate database table (using the data type and date) is chosen, attributes are filled and the data are added to the database. If the appropriate table is not available, a new table will be created. A check file describes data which should be cached in one specific database (eg one for satellite images only, ...). Some products, for example analysis and forecast fields, are fed into the database directly at the production stage. A clean-up task, running at fixed times, checks the available disk space and drops (deletes) old tables if necessary according to the last access time.

At the moment there are two EMPRESS databases in use as MCS (both on SGI workstations). One is fed with standard products from the operational suite and with satellite images (for the general user this is a read-only database), the other is used for dynamical caching.

For an overview of the dataflow for a MARS retrieval request see Fig. 2.

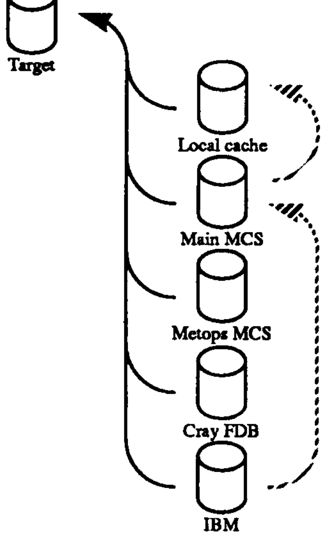


Fig. 2: The data flow for a MARS data retrieval request on workstations

Re-Analysis Project

It is intended, within the ERA project, to use the progress made with respect to the management of observations and fields within the workstation environment as a basis for the principal data management components. These include the provision of support for up to three months of observational data on-line, two months of analysis feedback, and two months of re-analysis results. This will require a dedicated SGI server, the relational database support, and much of the associated software developed by the Meteorological Applications Section, described above.

The database to be used to manage the observational reports will be very similar to the RDB described above, except that several additional attributes will be included. The database will enable data for production analysis to be extracted as required. In addition, it will assist in the identification of duplicate data, the preparation of the definitive observations archive corresponding to the re-analysis, and the supply of observational data to monitoring and verification applications. It will also be exploited to obtain statistical information, and will enable many interactive applications to be supported.

During the data assimilation phase the analysis scheme will run in data checking mode to compare the differences between observations and background fields. The values so produced can provide valuable information both with respect to the assimilation system, and with respect to the observing system. These results will be represented in BUFR, and managed in the same way as the original observational reports.

During the production phase results in field form (GRIB) will be stored using the MCS software (see above). This database will enable the complex post-processing required to obtain all of the planned monthly statistics to be achieved. It will also be necessary to perform monitoring and verification tasks, all requiring access to the results. In addition, the database will act as a pre-archive cache. Since the required production rate is of the order of ten days of re-analysis per day, it should be possible to complete one month of production every three days. By delaying the final archive process until each month is complete, it will be possible to generate an archive which is efficient with respect to the number of archive files, and in turn efficient with respect to future retrieval.

Data Services

A relational database application is being designed to manage information concerning enquiries and data requests dealt with by the "ECMWF Data Services". For entering information about customers and their data requirements, and for calculating charges, an X-interface utilizing "X-Designer" and interfacing to the database with ESQL has been designed. When a customer places a final order, information pertaining to this request will be extracted from the database and used to set up CRAY jobs to process the order and to produce documentation for the customer. Monthly and yearly figures about data types and volumes that were sent will also be made available.

Observational data (TEMP and SYNOP) are being re-archived in the form of monthly time series for distribution by Data Services. Data are extracted from MARS a month at a time and BUFR messages are read one by one into an EMPRESS database (as BLOBs) along with some metadata. Each table within the database contains data for one month and is sorted by station identification and time. The BLOBs for each record are then extracted sequentially into a file which is archived into MARS. This re-archiving system runs on a UNIX workstation and consists of programs written in C using the "MX-routine interface" (EMPRESS C Kernel Level Interface; less functionality than MR-routines, but easier to use).

Conclusion

After obtaining some basic knowledge about relational databases, and about the capabilities of the currently supported relational database system, it has been found to be easy to design a database and write the necessary applications to manage the data. The package used has proved to be a very versatile toolkit, enabling all possible types of data to be managed in a simple and effective way. While the applications developed to date have made use of one particular relational database system, the interfaces to that system have been well defined and are reasonably isolated with respect to their parent applications; in this way it should be possible to adapt in the future to any suitable database system, provided the appropriate levels of functionality and performance are supported.

- Alfred Hofstadler, Rex Gibson, Kathy Rider, Baudouin Raoult

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

ECMWF COMPUTER SYSTEM: STATUS AND PLANS

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

Over the past year a major change has been the replacement of the NOS/VE service by workstation-based services. The NOS/VE service was terminated on 30 November 1993, and the Cyber 962 system subsequently removed. This, in turn, has allowed a move of certain machine hall equipment to begin, the results of which will be an expanded IBM system, plus space for the Cray T3D which arrives in the middle of the year. The computer configuration as it will be in the middle of 1994 is shown in Fig. 1.

Major forecast delays in the first 11 months of the year have been lower than in the past 4 to 5 years.

Cray C90

The service for the first few months of 1993 was marred by higher than expected hardware failures. Cray Research brought in extra staff to investigate the system, checking for possible problems. The result has been a clear improvement (MTBF now at 105 hours, availability > 99%).

The only hardware change in 1993 has been the upgrading of the disk subsystems to the more modern DD62 and DA62 types, the latter being RAID disk technology and hence potentially much more reliable.

Looking at the overall utilisation, both the system time (~ 17%) and idle time (25%) are still too high. It is expected that the introduction of UNICOS 8 in the first half of 1994 will go some way to improve this. Under UNICOS 8, system calls (for single tasked jobs) can be run in parallel. Currently, only one system call at a time can be dealt with.

In mid 1994 the Centre takes delivery of a 64-processor Cray T3D system. This will be the Centre's first experience of an operational MPP (Massively Parallel Processor) system and, hence, creating a trial service on it will be an important task.

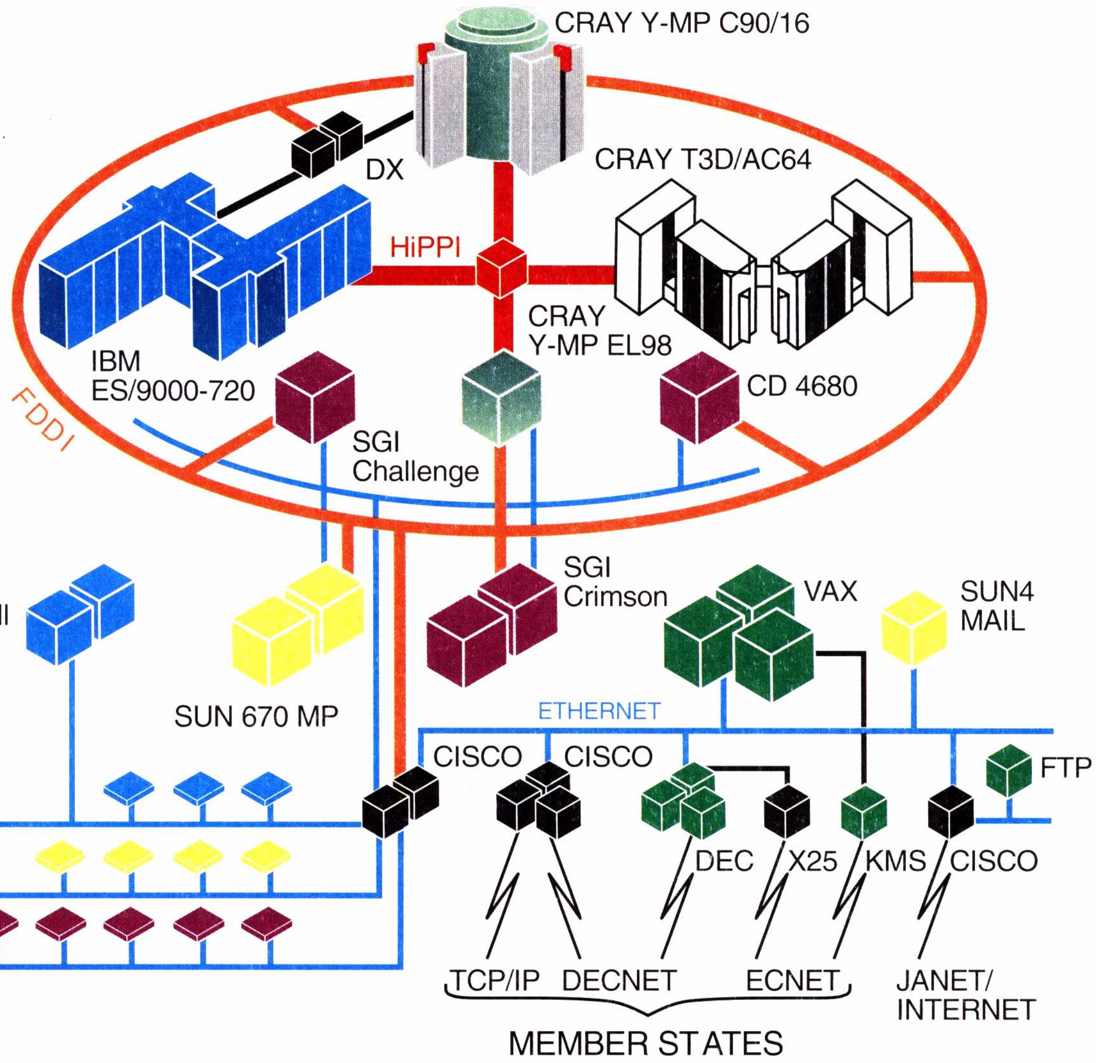


Fig. 1: ECMWF computer configuration, mid-1994

Data handling system

Some of the tape drives on the Storage Technology robots were recently upgraded from 18 to 36 track (doubling the tape capacity). However, initial results with these new drives show higher than expected recovered write errors, and the remaining drives will not be upgraded until the situation is clarified.

The IBM load at the end of 1993 was high; often over 80% utilisation occurs. The load is expected to increase still further now that MARS/Workstation is available. The reliability of the IBM continues to be very good.

In view of the high load seen on the IBM, the system is being upgraded in early 1994 to a 6 CPU system with more memory, channels, and disk capacity. This increase is primarily to provide support for the growth in the workstation load.

The data stored on the IBM is currently some 15 Terabytes in 4 million files.

The amount of data transferred into/out of CFS is about 100 Gbytes a day, of which 30% is MARS related and 70% is ECFILE transfers.

Looking to the future, a (UNIX based) successor to CFS must be found, as CFS cannot be supported indefinitely. Tests with UNITREE (a UNIX based data storage system) have shown that it does not at present offer yet the functionality we need to replace CFS. Extensive investigations in this area will continue. As part of these investigations, a HIPPI-attached disk array subsystem has been installed.

VAXs

A VAX 6210 was replaced in September with a VAX 4100. It is hoped that by mid-1994 all the Member States still using ECNET protocols will have switched to other protocols, and then the VAX 6310, 8250 and 8350 systems will no longer be required. It will then be possible to replace those systems with a second VAX 4100, simplifying the cluster still further.

The Member State traffic through the VAXs continues to show a slow increase and is now up to 100 Mbytes per week received and 300 Mbytes per week transmitted.

Overall, the VAX system continues to operate in a reliable and stable manner.

Workstations

After the decision, early in 1993, to go to Silicon Graphics as the preferred workstation supplier, some 66 SGI systems plus 3 SGI servers have been ordered or purchased. Additional NFS mounted disk space was also made available during the year.

In addition, the Cray YMP-EL server was acquired in a dual CPU configuration. It will shortly be upgraded to a 6 CPU system that will then be used to offload some of the less compute bound work from the Cray C90.

As mentioned earlier, the switch from the NOS/VE environment to a workstation based one has gone well. The workstations have taken over the PC based services.

During 1994 consolidation of the workstation environment will continue, with particular emphasis on improving the reliability of the servers. At the same time, a study will be made of the role of these servers.

Improvements in the user environment, and tools available, will continue. Also, an investigation may be undertaken to see how spare capacity on one system could be used by another.

Networks

Over the past year the Centre's internal network has been upgraded significantly to handle the extra load imposed by the change to using workstations. The network has also been made more resilient.

The Member State network has gradually acquired more 64 kb/s links and, as of today, there are 8 in operation and 2 more on order. Council, at its last session (1-2 December 1993), approved the upgrade of all Member State links to 64 kb/s from 1 April 1994.

Miscellaneous

One workstation is being set up outside the Centre's "security" fence. This machine will then be available for external mail and *ftp* connections.

Another workstation (situated in the User Area) is being equipped with peripherals to handle various tape media (e.g. Exabyte, QUIC, etc.). This will then be available to anyone to import/export data via these new media.

During 1994, the existing ageing cooling system will be refurbished to allow enhancements of its capacity to meet the load from 1995 onwards, if required.

Conclusion

Looking to 1994, the major activities will be the upgrade of the IBM, the installation of the Cray T3D, and the continuing study of alternatives for CFS.

Items with a more direct impact on users include the introduction of UNICOS 8 into production, the continued consolidation of the workstation environment, and the upgrade of Member State telecommunications links to 64 kb/s.

- Andrew Lea

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

I/O OPTIMISATION TECHNIQUES ON THE CRAY YMP C90

In November of this year, the Centre ran a workshop, mainly for Member States users, to help to publicise the various ways of optimising programs which run on the C90. At the end of this workshop it was felt that it would be beneficial to summarise part of it in the Newsletter to reach a wider audience. This article will concentrate on I/O optimisation that may be done to get Fortran programs to perform better, using the Flexible Fortran I/O (FFIO) system, but some mention will also be made of techniques that can be used by C programs. Before starting the article proper, it may be helpful to give a glossary of certain terms which are used, but which may not be known to all readers.

Glossary

CACHE	This is a buffer area, in main memory, designed as an intermediate repository for data blocks, positioned between the user's arrays and the physical disk. There are two reasons for the cache area. The first is to allow the user's logical I/O requests to be "batched up" into physical I/O requests to the disk. The second is to allow frequently accessed blocks of data to be made available, without having to perform disk I/O.
CACHE PAGE	This is the name given to a single cache buffer.
CONTIGUOUS	When applied to disk space, this means that the individual sectors which make up the space on the disk are consecutively allocated. For example a 100 sector contiguous file starting at address "A" on the disk will use up all the sectors through to address "A + 99".
COS	This is an acronym for "Cray Operating System", an earlier operating system for Cray computers. It is a Cray proprietory system, not based upon Unix and although still supported by Cray Research, is used at very few sites.
KERNEL	This is the name given to the lowest level of the Unix operating system. Whenever a program requires an operating system service, such as I/O, it makes a system call to the Kernel. This then handles the I/O control and drives the I/O devices.

LDCACHE	This is a cache area in the SSD which permits I/O to those filesystems which are ldcached to perform at speeds from 8 to 500 MBytes/sec, depending upon whether or not the data is in an LDCACHE page. LDCACHE is allocated on a per filesystem basis and currently cannot be bypassed. If a program does I/O to a filesystem, such as /tmp, which is ldcached, then that data will automatically go via the LDCACHE area for that filesystem.
SSD	This an acronym for "Solid-state Storage Device". The C90's SSD consists of 4 GBytes of memory. This memory is cheaper than main memory, of which we have 1 GBytes. SSD can be used in several ways, of which LDCACHE and SDS are relevant to this article. The SSD is not directly accessible to user programs. They use it by making system calls to the Kernel. These system calls are the result of doing I/O to files which either reside in SDS or which reside on filesystems which are ldcached.
SDS	This is an acronym for "Secondary Data Segments". This is an area in the SSD that can be reserved by a user program, enabling I/O to perform at speeds from 500 to 3000 MBytes/sec. The main difference, from a user's point of view, between LDCACHE and SDS, is that LDCACHE is shared, potentially by all programs running on the C90, while SDS is reserved for a particular file used by a particular program.

File Types and Structures

There are three main UNICOS file types or structures. These are TEXT, COS BLOCKED and UNBLOCKED and are illustrated in Fig. 1.

TEXT files:

These consist of bytes of ASCII data. Records are terminated by an ASCII linefeed character (LF - Hex code 0A). Because there is a record structure it is possible to use the BACKSPACE statement to reposition the file at a previous record. Also it is possible to read partial records. The Fortran standard allows the reading of just one byte from a record, the next read will occur not at the next byte, but at the first byte of the next record. This is denoted in this article as a "partial read".

This data structure is produced by default when using Fortran FORMATTED WRITE statements e.g. WRITE(10, 901) X.

In C programs it is produced using "printf" and "fprintf" statements. It is also the type produced by such Unix utilities as editors, compiler and other utility listings etc.

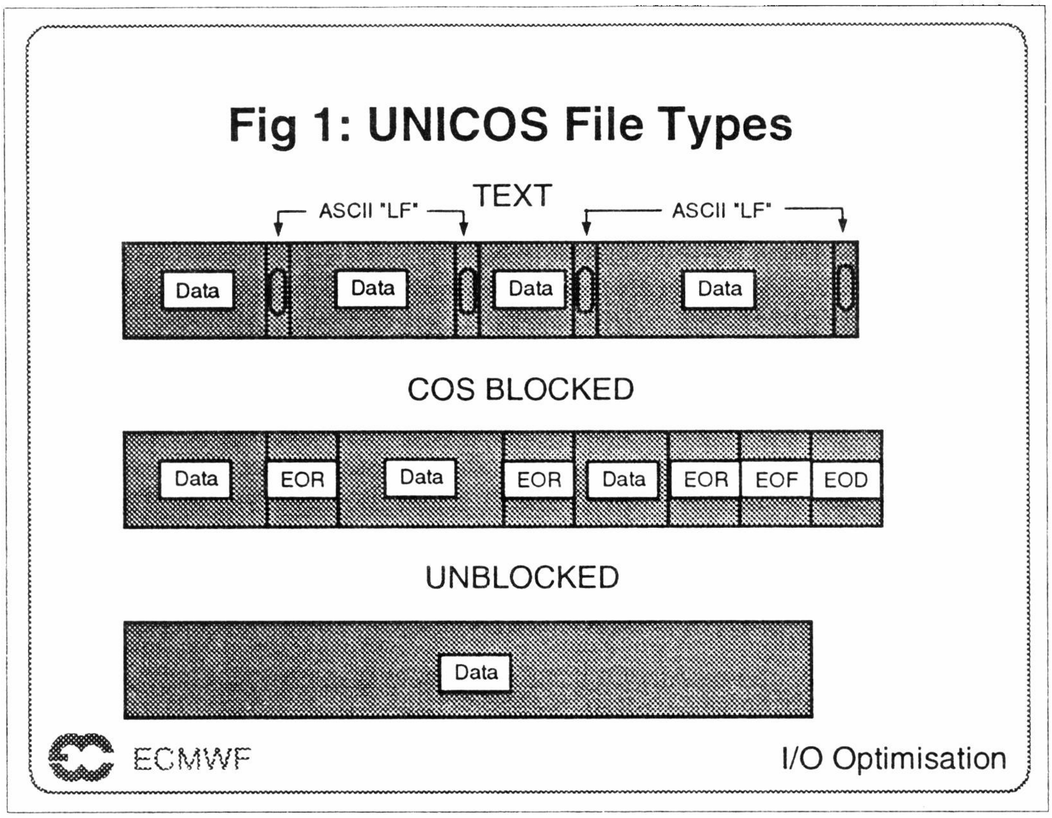


Fig 1: UNICOS File Types
TEXT
ASCII "LF"
ASCII "LF"
Data
Data
Data
Data
COS BLOCKED
Data
EOR
Data
EOR
Data
EOR
EOF
EOD
UNBLOCKED
Data
ECMWF
I/O Optimisation

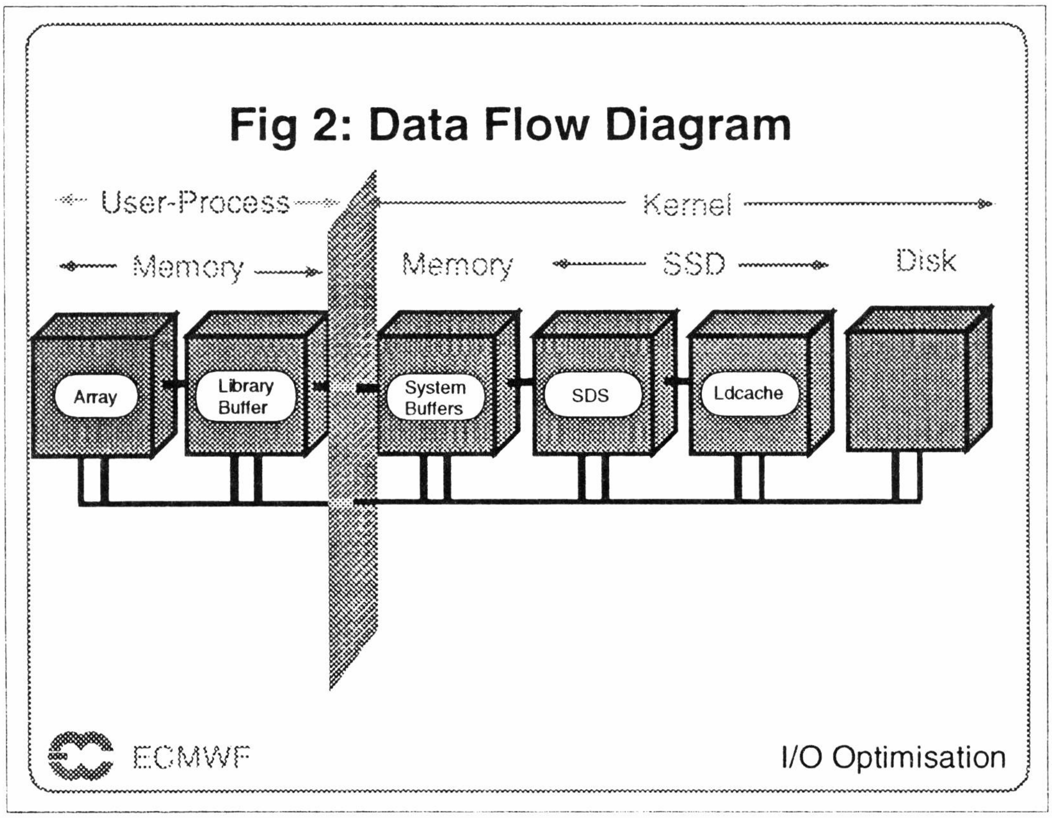


Fig 2: Data Flow Diagram
User-Process
Kernel
Memory
Memory
SSD
Disk
Array
Library Buffer
System Buffers
SDS
Ldcache
ECMWF
I/O Optimisation

COS BLOCKED files:

These consist usually of words (8 bytes) of data. Records are terminated by a special Record Control Word (RCW). The data is blocked up into 512 word blocks, each of which has a Block Control Word (BCW) at the beginning. There are pointers within the BCWs and RCWs to the next control word and there are special RCWs denoting End-of-File and End-of-Data. This type of file also supports BACKSPACE and partial reads.

This data structure is produced by default using Fortran UNFORMATTED WRITE statements e.g.WRITE(10) X, or using the BUFFER OUT statement.

In C programs it can be produced using "ffwrite" statements (FFIO routines) with an appropriate call to "asgcmd" beforehand. It is not produced by any of the normal Unix utilities and is not recognised by other Unix implementations, as it is Cray specific.

UNBLOCKED:

This consists of a sequence of unstructured bytes. Because there is no structure to it, it has no record structure associated with it. Because there is no concept of a record with this file type, it is not possible to use BACKSPACE to get to the previous file position, nor is it possible to read partial records and expect to be positioned at the beginning of the next record.

It can be produced by Fortran UNFORMATTED WRITES or BUFFER OUT statements, provided that an appropriate "assign" control statement has been used beforehand.

In C programs it is produced using "write" and "fwrite" statements.

Of course any file, even TEXT or COS BLOCKED files, may be treated as UNBLOCKED. If, by using an assign statement, the user treats a COS BLOCKED file as UNBLOCKED then the first read executed on that file will cause the BCW for the first block to appear as the first word of data read from the file. This may, but more probably may not, be what the user intended.

Data Flow Throughout the System

Fig. 2 shows the paths over which data flows between the user's program and the physical disk. There is a "brick wall" shown which symbolises the interface at which the data transfers between the user memory and the UNICOS Kernel memory. Since user programs themselves cannot directly write/read the Kernel memory or physical disk, they have to issue system calls to get the Kernel to do this for them. The amount of time that this will take depends upon several factors, including how much data is to be transferred, and how busy the UNICOS Kernel already is processing other system requests. This time is known as "SYSTEM TIME" and is charged to the user and reported at the end of the job.

The current version of the operating system, UNICOS 7.c.2, is single-threaded. By this we mean that, in general, while the Kernel is processing one system call, it cannot simultaneously process another. If, for example, the Kernel were processing a very long request for one process and 15 other processes then made a system request, all 16 CPUs of the C90 would be in use by UNICOS. However, of the 16 CPUs, only 1 would be doing useful work while the other 15 would be waiting for that one to finish. The next major release of the operating system UNICOS 8 is multi-threaded and should solve this problem.

The blocks in Fig. 2 represent points at which the data may "stopover" on its way between the user process and the disk.

The first block represents an ARRAY in the user's Fortran program.

The second block represents a LIBRARY BUFFER or a MEMORY RESIDENT LIBRARY BUFFER. These are blocks of memory within the user's field length, which has been reserved by the I/O library, and as such are "owned" by the user. Each library buffer is specific to a particular file, files do not share library buffers.

The third block represents the SYSTEM BUFFERS. These are blocks of memory within the Kernel's field length (outside the user's field length), which the Kernel uses to match the program's logical I/O requests with physical I/O to the disk. I/O to a physical disk is "quantized", the quantum being a disk sector. It is only possible to write integral multiples of sectors to a disk, whereas the user can, if he so wishes, write any size of data. The system buffers are used to "marry up" this obvious discrepancy in functionality. They are not "owned" by any particular user but are a shared resource, not only between user processes but also between different files in the same process.

The fourth block represents SDS. These are blocks of memory in the SSD. Each SDS buffer is "owned" by a particular user process and is specific to a particular file, but still only accessible via system calls. In this way it acts in a way similar to a MEMORY RESIDENT LIBRARY BUFFER.

The fifth block represents LDCACHE BUFFERS. These are blocks of memory in the SSD. Each buffer is "owned" by the Kernel and used in a way similar to that of the SYSTEM buffers. SDS and LDCACHE are not mutually exclusive; a file with an SDS buffer can, and probably will, still use LDCACHE buffers.

Techniques for Improving I/O Performance

When looking at improving I/O performance several factors need to be taken into consideration, but of all the factors which cause inefficiency, the one that causes the most is the use of the SYSTEM BUFFERS. By cutting out system buffer use it is possible to make programs much more efficient. It has been possible to cut down the elapsed time that it takes a test program to do its I/O from 30 minutes to 20 **seconds**, just by bypassing the system buffers. A real production program now runs in 1/3rd of its original time by doing just the same. The next best way of cutting out inefficiencies is to pre-allocate files contiguously on the disk. "CONTIGUOUS" means that the file is allocated on consecutive sectors on the disk. The third way is to cut down on the number of system calls needed to transfer the data.

Cutting out the use of SYSTEM BUFFERs involves doing "well-formed" I/O. The term "well-formed" means that the I/O must be done from "User memory space" to "Kernel memory space" in integral multiples of the disk sector size, and the file must always be positioned on a sector boundary. From now on the term "block" will have a specific meaning:

1 Block = 512 words (4 KBytes)

There are 2 types of disk connected to the C90:

DD-62 (Sector size = 1 Block) DA-62 (Sector size = 4 Blocks)

At present the DA-62 disks only hold files in the /tmp filesystem, while all other filesystems are on DD-62 disks. From this we can see that all I/O requests which transfer integral multiples of 4 blocks (2 Kwords, 16 KBytes) are well-formed on both types of disk.

PRE-ALLOCATING files contiguously on the disk has several benefits. Normally users do not pre-allocate their files, they let the Kernel allocate chunks of disk as their file grows. A typical example is this:

> The user writes the first sector or two of data to disk. The Kernel allocates one or two sectors to the file. The user writes more data and again the Kernel allocates one or two sectors to the file. Eventually, when the file reaches a certain size, the Kernel allocates not one or two sectors at a time but 28 sectors (28 being the size of a track on the disk). When the file is complete it may consist of hundreds of small chunks which are spread over a wide area of the disk. This can have disastrous consequences if the file is LDCACHEd. It can also take a long time to read this file because the disk heads will have to do a lot of positioning in order to read the data, and head positioning is the slowest of all disk operations.

Pre-allocating the file contiguously will avoid these problems. This can be done at the control statement level or at the C program level. In the following examples "numblks" is the required size of the file in BLOCKS (4 KBytes) and "filesize" is the required size in bytes.

Control statement:

```
setf -c -n numblks  filename  ||  setf -n numblks  filename
```

C program example:

```
if (fstat(filename, &st) == 0)
/* If file already exists ialloc allocates an additional extent */
{
        filesize -= st.st_size;
}
if (ialloc(filename, filesize, IA_CONT, 0) < 0)
/* If contiguous allocation failed */
{
        if (ialloc(filename, filesize, 0, 0) < 0)
        /* If there is insufficient disk space */
        {
                (void) perror("No space left / quota exceeded");
                return (1);
        }
}
```

Note that the "||" construct is used in case there is insufficient contiguous space on the filesystem, at which point we still pre-allocate but without the "-c" (contiguous) parameter. Without the "|| setf .." construct the job would abort if the amount of space requested was not available as one contiguous piece.

Fortran I/O Library Processing

Before rushing to change the Fortran code to do well-formed I/O one needs to look at how the Fortran I/O library handles the COS blocked and UNBLOCKED file types. TEXT files are not mentioned here because formatted Fortran output is usually only a small part of the I/O, being used for human-readable results. Having said that, I have in the past come across a Fortran program which used formatted I/O to pass data to another program, where there was no need for it be formatted. This is very inefficient and should be avoided.

COS BLOCKED FILES

As mentioned earlier COS BLOCKED datasets are the default for files read and written using unformatted READ/WRITE or BUFFER IN/OUT statements. They may also be explicitly requested using an "assign" control statement, the format for which is:

assign -F cos.synctype:bufsize filename or unit-number

Where:

synctype = "sync" for synchronous I/O; "async" for asynchronous I/O which does read-ahead and write-behind; "auto" (default) which is "sync" for bufsize < 64, otherwise "async".

bufsize = size of LIBRARY BUFFER in units of a block, the default being 48.

e.g.

To assign a COS BLOCKED dataset, with a buffer size of 56 blocks, using synchronous I/O, accessed via Fortran unit number 10, the assign statement would be:

```
assign -F cos:56 u:10
```

Fig. 3 shows what happens when the user program does four WRITE statements to create a file with four records, each containing the data belonging to a different array. In this example you can see that the four WRITE statements simply transfer data between the user's arrays and a library buffer. The data does not actually get transferred to disk via a system call until the buffer is flushed, when the file is closed, and then only one system call is involved. If we change the example to write these arrays 56 times to a file with a buffer size of 56, then we can see that these 224 Fortran WRITE statements will generate three system requests each transferring 56 blocks to disk. All these are well-formed requests and so will bypass the system buffers. This is very good, but if the file were continually being created, rewound read, rewound read, rewound ...etc. it would be even better to keep the file totally within in the library buffers until it was closed and flushed to disk. This would require a library buffer of 168 blocks (84 Kwords) and the assign statement would be:

```
assign -F cos.sync:168 u:10
```

UNBLOCKED FILES

Having looked at COS BLOCKED datasets, let us now see how UNBLOCKED datasets are handled by the I/O library. To assign a file as UNBLOCKED the format of the assign control statement is:

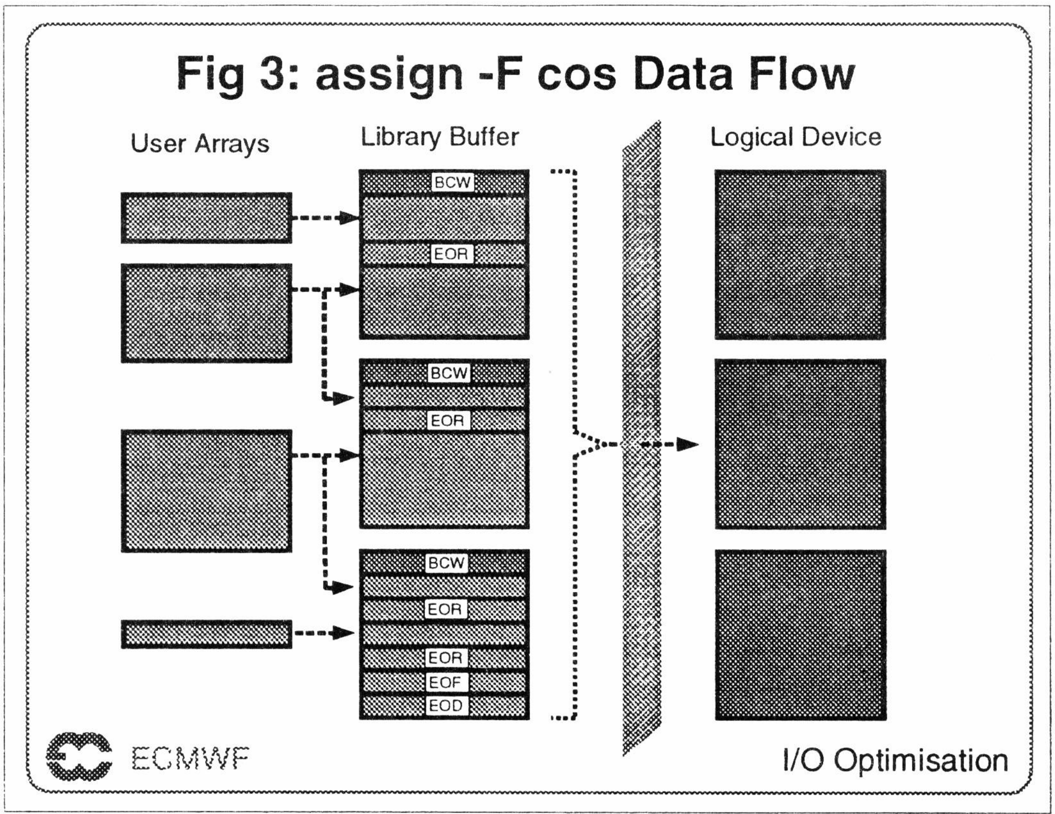


Fig 3: assign -F cos Data Flow
User Arrays
Library Buffer
Logical Device
BCW
EOR
BCW
EOR
BCW
EOR
EOR
EOF
EOD
ECMWF
I/O Optimisation

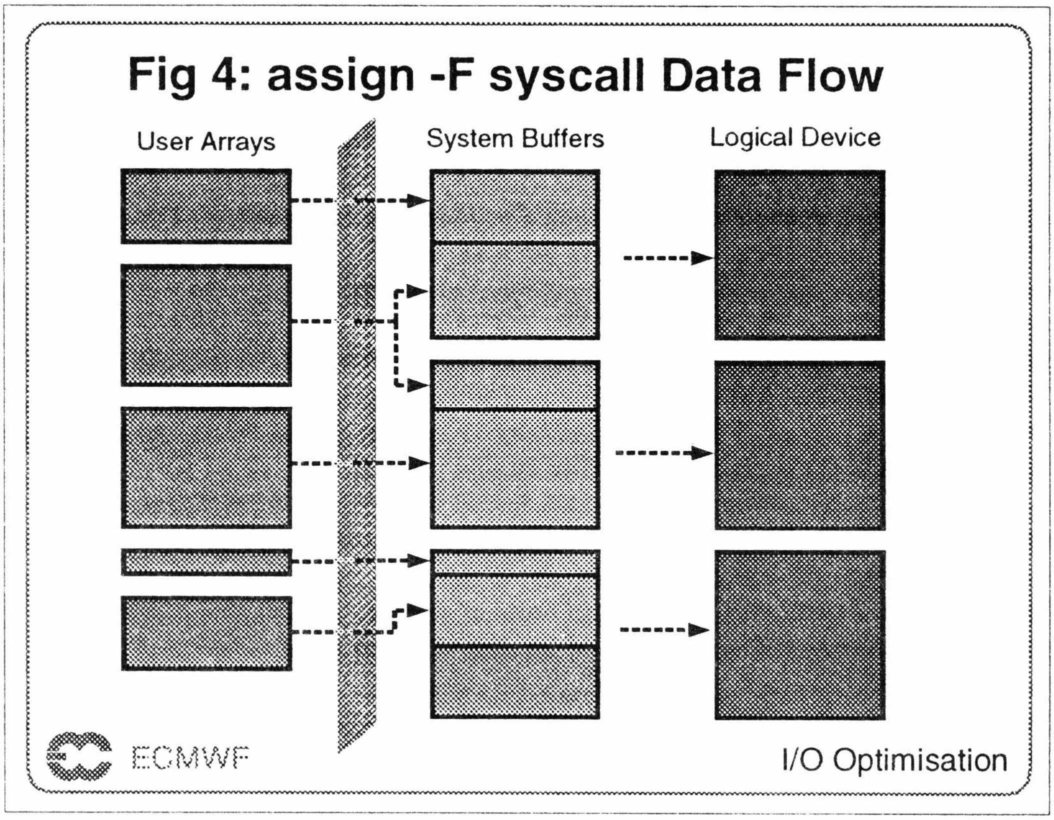


Fig 4: assign -F syscall Data Flow
User Arrays
System Buffers
Logical Device
ECMWF
I/O Optimisation

assign -F syscall filename or unit-number

e.g.

To assign as unblocked the file "fort.11", the assign statement would be:

assign -F syscall f:fort.11

Fig. 4 shows the same Fortran program as before but this time writing to an UNBLOCKED dataset. One can see that this is very inefficient as **every** Fortran WRITE statement generates a system call with an I/O request which is not well-formed. Let us look at this in more detail in "pseudo-code" form.

```
User:   does 1st WRITE --> system call
Kernel:         if no system buffer is free {
                flush the oldest buffer to disk }
        copy data from the user's array to the buffer

User:   does 2nd WRITE --> system call
Kernel:         if the previous buffer is not still around {
                if no system buffer is free {
                        flush the oldest buffer to disk }
                read the previous sector of data from disk into the buffer }
        copy the first part of the data from the user's array to the end of the buffer
        if no other system buffer is free {
                flush the oldest buffer to disk }
        copy the last part of the data from user's array into the buffer

User:   does 3rd WRITE ....etc.
```

As one can see, a large amount of Kernel and disk activity is going on. This is the main cause of "system time" on the C90. How can this be prevented without having to completely rewrite the I/O portion of the Fortran program? The simplest way is to introduce a "well-formed" LIBRARY BUFFER. The format of the assign statement to do this is:

assign -F cache:bufsize:nbufs filename or unit-number

Where:

bufsize = size of LIBRARY BUFFER in units of a block, the default being 8.
nbufs = the number of cache "pages", the default being 4.

e.g.

To assign one cache page of 56 blocks to an unblocked file "fort.11", the assign statement would be:

assign -F cache:56:1 f:fort.11

There is no need to specify "syscall" as that is implied, though one could specify it for clarity e.g.

assign -F cache:56:1,syscall f:fort.11

Now, as can be seen from Fig. 5, all Fortran I/O calls just operate on data in the LIBRARY BUFFER instead of making ill-formed I/O requests.

For UNBLOCKED files which are small enough to fit within the memory of the job it is better to use "mr" (memory resident), rather than "cache", especially if the files are "scratch" files which are created and used solely within this program. The format of the assign control statement to do this is:

assign -F mr.savscr.ovfopt:init:max:incr filename or unit-number

where:

savscr = "save" (default) to pre-load and post-store the data in the file on disk; "scr" to indicate that the data is neither to be read from nor written to the actual disk file.

ovfopt = "ovfl" (default) to allow excess data to overflow onto disk; "novfl" to abort the program if the amount of data exceeds the memory resident buffer.

init = the number of blocks of memory initially allocated to the memory resident buffer.

max = the maximum number of blocks that may be allocated.

incr = the block allocation increment.

e.g.

To assign the same file "fort.11" to be memory resident, one would need to reserve 168 blocks of memory, in order for it to fit completely within the buffer. The assign statement would be:

assign -F mr.save.novfl:168:168:0 f:fort.11

or

assign -F mr.scr.novfl:168:168:0,syscall f:fort.11

The latter statement explicitly specifies "syscall" for clarity and states that the file is a "scratch" file, that will only exist for the duration of the program.

Fig. 6 shows this example. One should not overflow the memory resident buffer, hence use the "novfl" parameter. If this is not coded and the data overflows the buffer then all the overflow portions of the data will (in this example) be ill-formed and system time will again accrue.

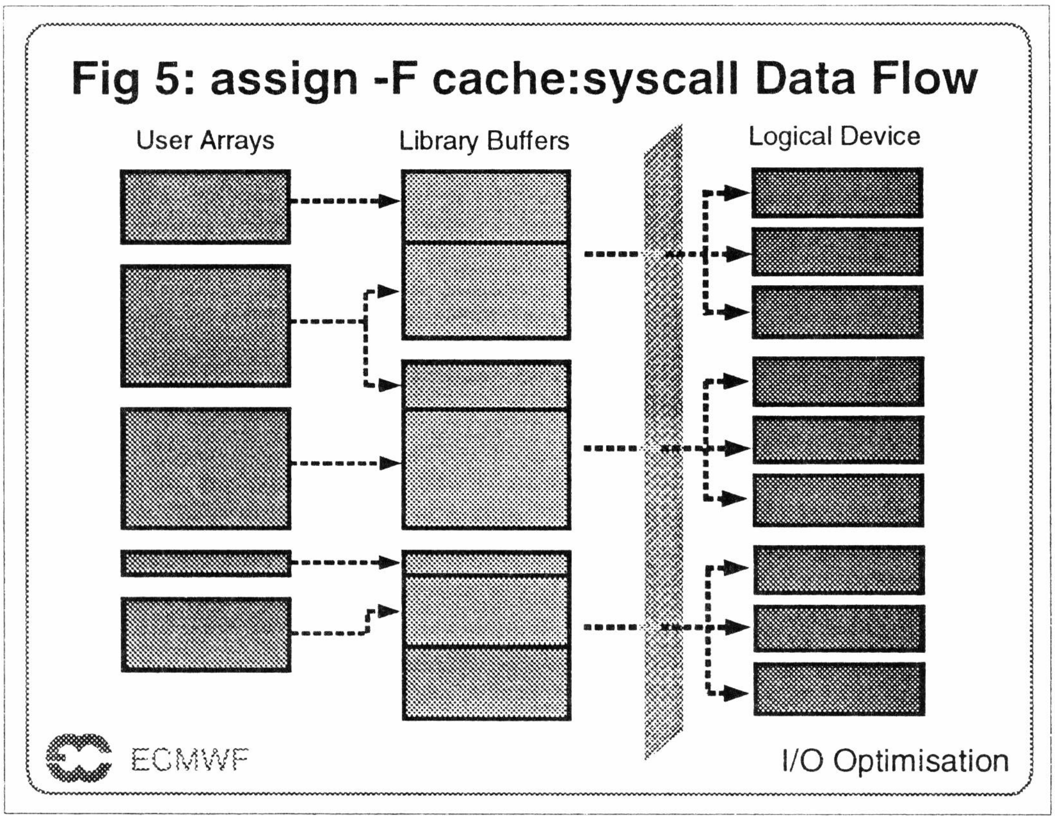


Fig 5: assign -F cache:syscall Data Flow
User Arrays
Library Buffers
Logical Device
ECMWF
I/O Optimisation

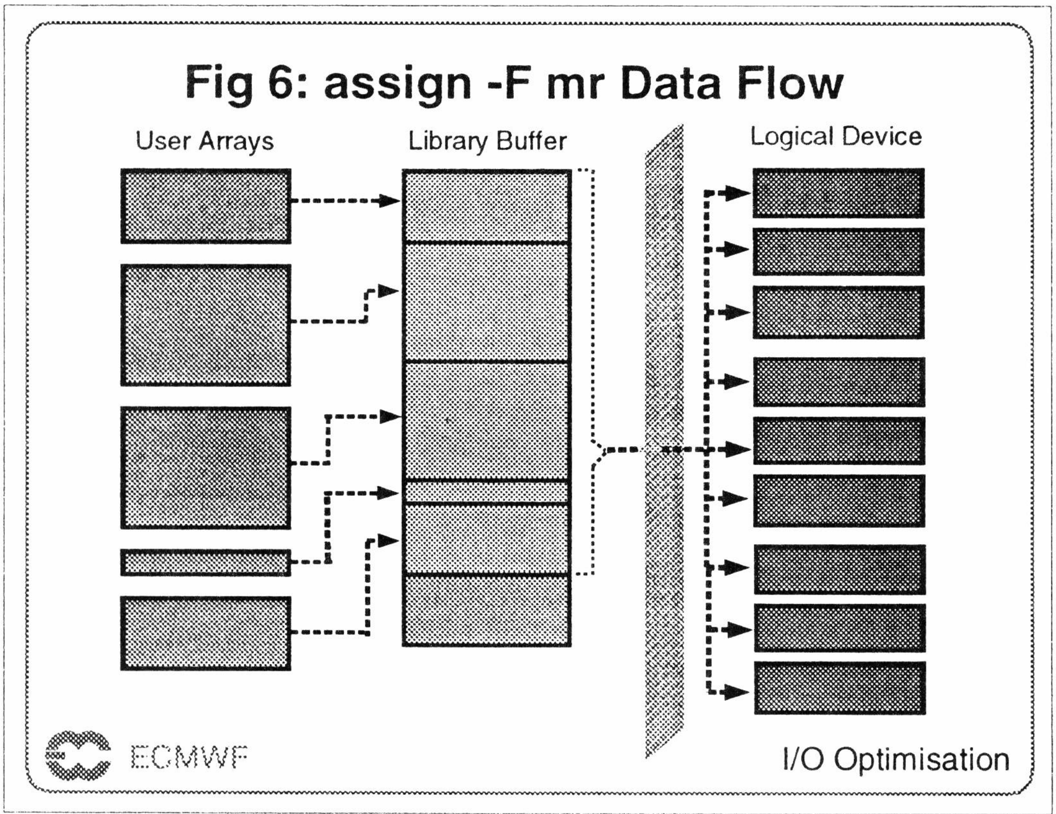


Fig 6: assign -F mr Data Flow
User Arrays
Library Buffer
Logical Device
ECMWF
I/O Optimisation

LDCACHE Data Flow

Since several of the filesystems on the C90, including /tmp, are LDCACHEd it is instructive to see how this works. Fig. 7 shows an example of LDCACHE use. In this example the amount of LDCACHE allocated to the filesystem is four pages each of size two sectors. In reality /tmp has 1280 pages each of seven sectors. The example shows a well-formed I/O request for four sectors of data. It also shows a file which was not allocated contiguously on the disk. The flow is as follows:

User: does the READ --> system call
Kernel:
  if no LDCACHE buffer is free {
  flush the oldest buffer to disk }
1) read two sectors from disk into the buffer (only one of which is from our file)
  transfer our one sector to the user's field length
  if no LDCACHE buffer is free {
  flush the oldest buffer to disk }
2) read two sectors from disk into the buffer (only one of which is from our file)
  transfer our one sector to the user's field length
  if no LDCACHE buffer is free {
  flush the oldest buffer to disk }
3) read two sectors from disk into the buffer (only one of which is from our file)
  transfer our one sector to the user's field length
  if no LDCACHE buffer is free {
  flush the oldest buffer to disk }
4) read two sectors from disk into the buffer (only one of which is from our file)
  transfer our one sector to the user's field length

As can be seen, in this example there could be up to eight physical disk I/O transfers from wildly different areas on the disk, before the data is actually available in the user's program. If the file was contiguous on the disk this would be reduced to four physical I/O requests. In real life, with a buffer size of seven sectors, there would probably be only two physical I/O requests (224 KBytes) for such a contiguous file, instead of eight physical I/O requests (896 KBytes) and the disk positioning, which is the main cause of slow I/O, would be kept to a minimum.

One point that should be made concerning FFIO. When one codes the "-F" parameter on the assign statement, some of the other parameters lose their default values and the "-s" parameter cannot be specified. One important parameter that may need to be explicitly specified is the "-T off" parameter. This is needed if the program uses the SETPOS routine to position within a file and WRITE a record at that point. If a "-F" parameter is not used but a "-s u" used instead, then the default is "-T off", but when using "-F" the default becomes "-T on", the file will become truncated immediately after the record which was written, and any data which was behind this record in the file will no longer exist.

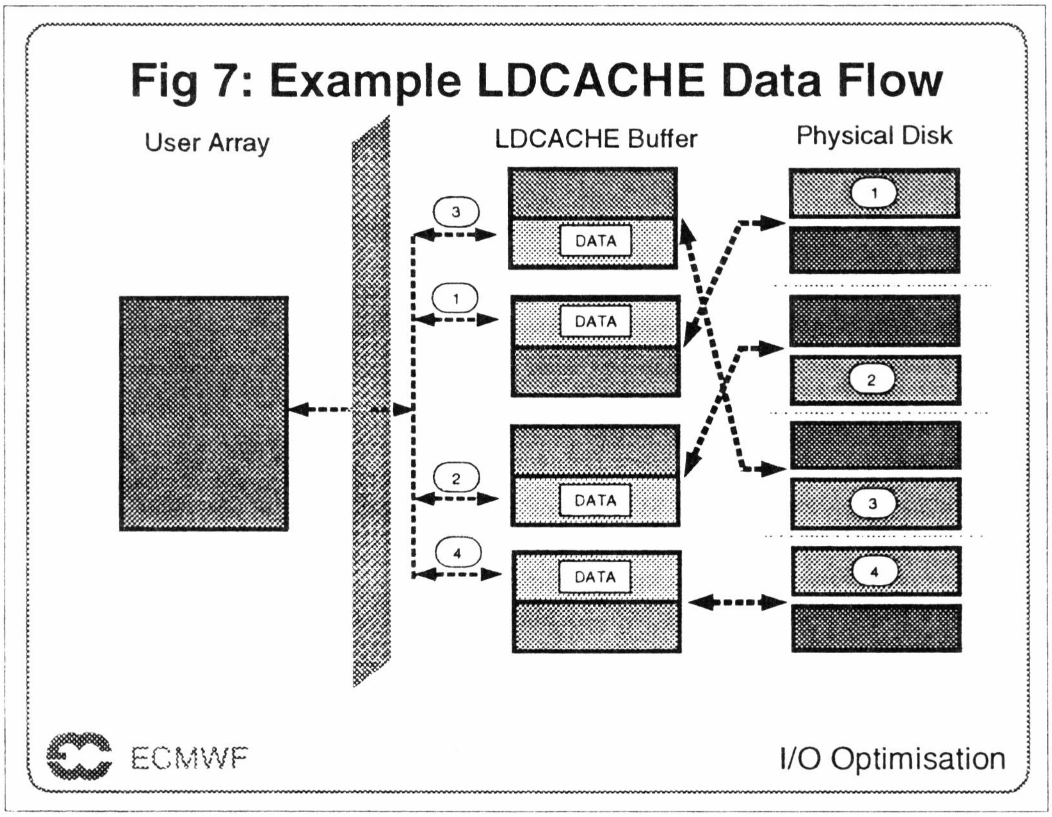


Fig 7: Example LDCACHE Data Flow
User Array
LDCACHE Buffer
Physical Disk
DATA
DATA
DATA
DATA
1
2
3
4
1
2
3
4
ECMWF
I/O Optimisation

I/O Performance Tools

By far the most important performance tool for monitoring I/O is "procstat". Procstat produces a file which is analyzed by the "procrpt" utility (amongst others). Data is put in the procstat "raw file" every time an I/O operation is performed from within the program being monitored. One of the nice features of procstat is its ability to monitor any program without having to recompile it, in fact procstat can also be used on Unix utilities such as "cp", "tar", "cpio" and the like.

Below is a simple procstat example in which the normal "a.out" command is replaced by:

```
procstat    -R $TMPDIR/rawfile    a.out
procrpt  -s -F all $TMPDIR/rawfile
```

it is as simple as that. Now comes the "tricky" part - interpreting the procrpt output.

First, skip to the end of the report to the title:

"PROCSTAT FORTRAN FILE REPORT"

Here you will find a set of reports, one for each of the Fortran files in your program. Such a report (taken from a test program run) looks like this:

```
Fortran Unit Number            11
File Name                      fort.11
Command Executed               a.out
Date/Time of Open              11/02/93 11:52:43
System File Descriptor         4
Type of I/O                    sequential unformatted
File Structure                       unknown

Fortran I/O Count of        Real
Statements  Statements      Time
----------- ----------  ---------------
READ            128000      15:00.1751
WRITE           128000      16:42.9804
REWIND             200          1.0175
CLOSE                1           .0031

32506.0      Bytes transferred per Fortran I/O statement
      0.78% Of Fortran I/O statements did not initiate a system request
===========================================================================
```

From the last value (0.78%) we can see that every READ or WRITE statement caused a system call to be made, which in general is a bad thing. This is because the file was assigned as UNBLOCKED (or UNKNOWN as procrpt reports). The higher this last number is the better, 100% being the target to aim for.

Looking at the REWIND count (200) we can deduce that the file is being continually written, rewound, read and then probably rewound again.

Looking at the time taken to read and write the file, compared to other files in this program which only took 20 seconds in total, it is obvious that this file is a prime candidate for optimisation. One can find out more information about this file by looking earlier in the report. Search backwards for either string:

"File Name = fort.11" or "Fortran Unit Number = 11"

One will see the above again, but there is also other interesting information, such as:

```
Minimum File Size = 4193280 (bytes)
Maximum File Size = 4206592 (bytes)
Final       File Size = 4193280 (Bytes)

System I/O # of   # Bytes    # Bytes     Wait Time      (Clock Periods)
Function   Calls  Processed  Requested   Max         Min          Total
--------   ------ ---------  ----------  ----------  -----  ------------
Read       12800  419328000  4193228000  2523192870  19650  215991603956
Write      12800  419328000  4319328000  1999536112  20170  240666850980
Seek       200          n/a         n/a    31263627   4294      70589203
Truncate   100          n/a         n/a    39963846   6552      26900319

System I/O  Avg Bytes   Percent of  Average I/O Rate
Function    Per Call    File Moved  (MegaBytes/Second)
----------  ----------  ----------  ------------------
Read           32760.0      9968.4        0.466
Write          32760.0      9968.4        0.418
==========================================================================
```

From this one can see that the file was 4193280 Bytes long (0.5 MWords) and that it was continually being written then read. It also confirms that each of the 12800 Fortran READs and WRITEs issued a system call. This file is a prime candidate for MEMORY RESIDENT or CACHE. Just putting a 56 block CACHE on this file with the assign statement:

```
assign -F cache:56:1 u:11
```

changes the total I/O wait time on it from 31 minutes 41 seconds to 19 seconds; while converting to MEMORY RESIDENT with the assign statement:

```
assign -F mr.scr.novfl:130:130:0 u:11
```

changes the total I/O wait time on it to 0.85 seconds.

C Program I/O Optimisation

This article has concentrated on Fortran programs, since these make up the majority of programs run on the C90. With C programs things are not quite so easy.

It is still possible to optimise the I/O of C programs. The use of "ialloc" has been shown in an earlier example. In order to use FFIO, the source code has to be modified to call the FFIO routines (ffopen, ffread, ffwrite, ffclose etc.). Unless special constructs, such as #define macros, are used, this could cause portability problems.The use of "read" and "write" with well-formed I/O is an obvious way of optimising. Library buffering can be used by coding "fopen" calls instead of "open" and "fread/fwrite" instead of "read/write". The size of library buffers (by default 8 blocks) can be specified using the "setvbuf" routine e.g.

```
if (setvbuf(file_out, (char *) NULL, _IOFBF, 56*4096) != 0)
{
        perror("Cannot set buffer");
        return (1);
}
```

Summary

In this article I have attempted to give some ideas of ways in which I/O can be optimised. In most cases this does not necessarily involve changing any code, merely using procstat and procrpt and then adding one or two assign control statements to the jobs. In the past few weeks we have tried to identify jobs on the C90 which are performing inefficiently as regards the amount of system time they use, since these jobs affect all other jobs on the machine. The user is then approached and encouraged to use the information given in this article, with the end result that I/O wait time, and hence turnaround time, is decreased dramatically (from e.g. one hour to 10 minutes).

Everyone's I/O requirements are different and it would be impossible to design a default I/O system to satisfy everyone's needs. I believe that Cray have done a good job in designing procstat and FFIO, and users are more than willing to use these features when they see how easy they are to use and how much benefit they can gain from using them. Please make use of these, especially if you are developing code that is going to be used not only by you but by many different users. Obviously it is better to design such code from the very beginning with "well-formed" I/O, contiguous data and reasonable buffer sizes, rather than have to rely upon users coding assign statements at a later stage. For "dusty deck" codes FFIO etc. is the next best thing.

If you need help in implementing or even understanding any of the features outlined in this article, please do not hesitate to contact the User Support section at ECMWF. Two Cray manuals describing I/O techniques in detail are:

(SG-3075 7.0): I/O User's Guide
(SG-3076 7.0): Advanced I/O User's Guide.

- Neil Storer

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

COMPUTER RESOURCE ALLOCATION TO MEMBER STATES IN 1994

Table 1: Allocation of CRAY resources and data storage to Member States in 1994 (including a 10% reserved allocation for Special Projects)

MEMBER STATE	Cray (kunits)	Data (Gbytes)
BELGIUM	108	40
DENMARK	90	34
GERMANY	479	178
SPAIN	188	70
FRANCE	373	139
GREECE	75	28
IRELAND	67	25
ITALY	348	130
YUGOSLAVIA*	82	30
NETHERLANDS	133	50
NORWAY	85	32
AUSTRIA	99	37
PORTUGAL	73	27
SWITZERLAND	120	45
FINLAND	91	34
SWEDEN	118	44
TURKEY	85	32
UNITED KINGDOM	322	120
SPECIAL PROJECTS	324	105
TOTAL	3260	1200

At its 37th session on 2-3 December 1992 the Council decided that the telecommunications link between ECMWF and Belgrade would be terminated with immediate effect. As a consequence, access to the Centre's computer system is not available to Belgrade.

Table 2: Special Project allocations for 1994

Member State	Institution	Project title	1994 Resources requested		Future Cray resources	
			Cray Kunits	CFS Gbytes	1995 Kunits	1996 Kunits
<u>Continuation Projects</u>						
Austria	Institut für Meteorologie und Geophysik, Vienna (Hantel)	Diagnostic tests of the ECMWF physics package	0.8	1.5	1.3	-
Finland	FMI, Helsinki (Lönnberg)	The HIRLAM 3 project	15	5	15	15
	Fin.Inst.Marine Research (Kahma)	Implementation of WAM wave model to the Baltic Sea	1.3	3	-	-
France	CNET/CRPE (Eymard)	Determination of ocean surface heat fluxes using satellite data and the ECMWF model	0.8	1.4	-	-
	METEO France, Toulouse (Roquet)	AVISO. Study of surface winds and surface fluxes at the interface ocean/atmosphere	5	1	5	5
	LMD, Palaiseau (Duvel)	Validation of spatial and temporal variabilities of the ECMWF model	3	2	3	3
	L.A.M.P., Aubière (Chaumerliac)	Chemistry, cloud and radiation interactions in a meteorological model	2	2	-	-
Germany	Institute for Geophysics and Meteorology (Speth)	Interpretation and calculation of energy budgets	2	6	2	2
	GKSS, Geesthacht (Raschke/ Rockel)	Parametrization of radiation and clouds for use in general circulation models	1	0.2	1	1
	MPI Hamburg (Roeckner)	Modelling the earth's radiation budget and evaluation against ERBE data	40	5	40	40
	MPI, Hamburg (Bengtsson)	Numerical experimentation with a coupled ocean/atmosphere model	90	25	110	130
	MPI, Hamburg (Bengtsson)	Simulation and validation of the hydrological cycle	60	10	90	100
	MPI Hamburg (Hasselmann)	Extraction of two-dimensional wave spectra from ERS-1 SAR wave mode fast delivery products	2	1.5	2	2
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	3
Netherlands	KNMI, De Bilt (Siegmund)	Analysis of a CO_2-experiment performed with a GCM	2	0.5	2	2

Table 2: Special Project allocations for 1994 (continued)

Member State	Institution	Project title	1994 Resources requested		Future Cray resources	
			Cray Kunits	CFS Gbytes	1995 Kunits	1996 Kunits
	KNMI, De Bilt (Cuijpers/ Duijnkerke)	Large eddy simulation of (strato)cumulus clouds	1	1	1	1
	KNMI, De Bilt (Können)	Climatological scenarios	0.5	5	0.5	0.5
<u>New projects</u>						
Austria	Institut für Meteorologie und Geophysik, Vienna (Hantel)	Diabatic heating of the global atmosphere	0.2	0.5	0.3	0.3
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	24	24
Germany	University of Munich (Wirth/Egger)	The behavior of cut-off cyclones in ECMWF analysis: impact of diabatic processes on their development and decay	1	1	-	-
Netherlands	KNMI (Velders)	Chemistry and transport studies with a 3D off-line tracer model	25	15	25	25
Norway	Geophysical Institute, University of Bergen (Grønås/Kvamstø)	Cloud parametrization in general circulation models	1	0.1	1	1
Sweden	Stockholm University (Sundqvist)	Parametrization of cirrus cloudiness in numerical models	10	2	10	10
		Total requested	290.6	92.2	336.1	364.8
		Reserve (to be allocated by ECMWF)	32.6	12.0		
		Overall total requested	323.2	104.2		
		Amount available	326.0	120.0		

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

COMPUTER USER TRAINING COURSE

The Centre is offering a computer user training course for Member States' personnel and ECMWF staff from 21 February - 18 March 1994. Full information and a request for nominations has been sent to all Member State meteorological services.

The course is divided into four one-week modules. Attendees may register for separate modules.

Week 1: An introduction to UNIX, for those who have no knowledge of this operating system.

Week 2: UNICOS extensions to UNIX, ECMWF utilities, the ECFILE file system, MARS.

Week 3: MAGICS, Advanced UNICOS.

Week 4: MPP usage.

Each week will consist partly of lectures and partly of practicals. In more detail, the four modules are:

MODULE 1 (21 - 25 February 1994): UNIX

Introduction to UNIX history and basic structure
Introduction for the file system
Basic commands
File manipulation and attributes
I/O commands
Basic shell scripts

MODULE 2 (28 February - 4 March 1994): ECMWF'S UNICOS SERVICE

System and hardware overview
UNICOS batch jobs
FORTRAN
ECFILE file storage system
MARS data retrieval
File transmission services, including **sendtm**

Those attending module 2 are expected to know basic UNIX commands and be able to use the *vi* editor.

MODULE 3 (7 - 11 March 1994): MAGICS & Advanced UNICOS

MAGICS

Introduction and overview

Concepts

Parameters

Subroutines
Action and pseudo-action routines
Data input

Plotting features.

ADVANCED UNICOS

Program maintenance tools

Introduction to vectorising

Specialist I/O techniques.

Those attending module 3 are expected to know basic UNIX commands, be able to use the *vi* editor and submit jobs to UNICOS.

MODULE 4 (14 - 18 March 1994): MPP USAGE

Overview of parallel processing

Architecture of the Cray T3D

PARMACS

Running a program on the Cray T3D

Optimisation & debugging tools.

Those attending module 4 are expected to know basic UNIX commands, be able to use the *vi* editor and submit jobs to UNICOS.

- Andrew Lea

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

ECMWF PRODUCED COMPUTER USER DOCUMENTATION

Ever since the beginning of its computer service in 1978, ECMWF has produced its own user documentation to supplement that supplied by the manufacturers. This documentation has been published in two series, namely the ECMWF Computer Bulletins and ECMWF News Sheets. The former (Bulletins) are for items of a lengthy nature that do not change very often, e.g. MAGICS Reference Manual. The latter (News Sheets) contain items of an urgent, often transitory, nature.

Both series, up to now, have been published in paper form and distributed widely to in-house and to Member State users.

In the past couple of years, another distribution medium has become possible with the adoption of UNIX-based servers, namely in an electronic form. More specifically electronic copies of user documentation have been collected together in one location, and a simple user interface provide to browse and search through those documents. This on-line document facility is known as "*echelp*".

Since 1 September 1993, Member State users have had interactive access to a UNIX-based server at ECMWF. Thus the majority of users now have direct access to "*echelp*" and its on-line user documentation. Hence, in future, ECMWF will move away from providing paper copies of documents and instead concentrate on enhancing the on-line document system. From the user point of view this is advantageous in that documents can be browsed and searched much more readily. From Computer Division's point of view documents can be kept up-to-date much more readily, and the (sometimes) lengthy printing process can be eliminated.

To access the ECMWF on-line document system log on to any UNIX-based workstation or server (e.g. *munin*) and type "*echelp*". Alternatively for those with an X Windows interface, select "Local Help Tool" from the User Menu (cursor on background, press right hand mouse button). A public domain software system (gopher/xgopher) is started which displays a top level menu of documents and/or directories.

The documents themselves are stored in a set of directories under ***/usr/local/information***. They consist of:

text files	-	each file holds an individual document which can be immediately viewed;
binary files	-	these files hold documents in a format which is not immediately viewable, e.g. in PostScript or WordPerfect formats;

directories	-	each contains a set of similar files, e.g. all the Computer Bulletins;
index	-	a word index which can be searched to locate all documents containing a specific topic;
user list	-	an index of user names, identifiers, telephone numbers, etc.

The gopher/xgopher interface initially displays the contents of the directory */usr/local/information*. For X Windows users a full windows-based service is available, allowing quick access to all documents by simple mouse point and click procedures. For other users a line mode interface based on "curses" gives access to the same information, via the cursor keys.

The facilities provided by gopher/xgopher allow one to:

* select a given text file and view it immediately;
* search within a given document for a word or phrase;
* search all documents for a given topic;
* print documents, or save them to a file of one's own;
* remember past searches, and so be able to return to them quickly (bookmark facility);
* view on-line help in how to use gopher/xgopher.

Hopefully, the use of "*echelp*" should be self evident, however, new users are recommended to read through once the entry "*All about ECMWF's help and information service*".

As was stated earlier on, it is now ECMWF's intention to concentrate in the future on providing user documentation in this electronic, on-line, format. Hence, please get to know "*echelp*" and its facilities, as this should be your first point of search whenever you have a query about ECMWF's computer services.

\- Andrew Lea

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

ECMWF'S COMPUTER BULLETIN SERIES

As is stated in the previous article, ECMWF will in future concentrate on providing computer user documentation in an electronic on-line format (via "*echelp*"). Information will primarily be held in files, often fairly short, that deal with rather specific issues.

Conversely each Computer Bulletin was designed to cover a reasonably wide topic in some detail, and to be held by users as a printed document. The growth therefore of on-line documents means that the need for Computer Bulletins will now diminish, although not disappear altogether.

It is intended in future to restrict Computer Bulletins to only those documents that need to cover a large field, e.g. introductory guides and reference manuals. Such documents will still be printed where relevant, but an electronic copy will also always be available in the on-line system.

With the closure of the NOS/VE service, several ECMWF Computer Bulletins now become redundant. In addition, it is not proposed to update the *Index to Computer Bulletins* any further, as it is replaced by the on-line word search index facility. Finally, some other bulletins have been replaced by alternative documentation. Hence the following Computer Bulletins should now be discarded:

Index, B1.1/1, B2.5/1, B2.5/2, B3.5/1, B4.1./1, B4.1/3, B6.1/1,
B6.1/2, B6.1/3, B6.7/5, B6.7/6, B6.7/7, B7.6/1, B7.6/2, B8.3/3.

The current contents of valid bulletins is therefore:

0.1/1	ECMWF Computer Division management and personnel list
0.2/3	Security
1.0/2	An introduction to the VAX interactive system
1.0/3	Introduction to the UNICOS system at ECMWF
1.2/1	UNICOS accounting
1.5/1	Advisory and visitor services
2.8/1+	Terminal emulators for PCs - a basic introduction
2.8/2+	An (outgoing) fax service for PC users
3.3/1*	The ECMWF data link protocol
3.3/2*	ECMWF end to end protocol definition
3.3/3*	The ECMWF file transfer protocol
3.3/4*	Remote job facility for ECMWF's computer system
3.3/5*	The interactive facility for medium speed line connections

3.4/2	Integrated electronic mail services
3.4/3	INTERNET
5.2/5	The ECMWF meteogram system
5.2/8	Reference manual for MAGICS
5.2/9	User's guide for MAGICS
5.2.10	Pocket guide for MAGICS
5.2/14+	The diagnostic and plotting package BPP
6.0/1	Software libraries available at ECMWF
6.7/2	MARS user guide
8.2/1	Supporting incoming/outgoing magnetic tapes at ECMWF
8.3/1	ECFILE concepts
8.5/1	Alphanumeric Microfiche

NOTES:

+ Bulletins issued within ECMWF only
* Bulletins distributed to Member States only.

- Andrew Lea

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

STILL VALID NEWS SHEETS

Below is a list of News Sheets that still contain some valid information which has not been incorporated into the Bulletin set (up to News Sheet 305). All other News Sheets are redundant and can be thrown away.

No.	Still Valid Article
204	VAX disk space control
224	Job information cards
235	VAX public directory - how to create
253	Copying complete UNICOS directories to ECFILE
254	UNICOS carriage control
260	Changes to PUBLIC directories for VAX users
261	Meteogram system on UNICOS
265	Lost UNICOS outputs submitted via RJE or VAX Microfiche changes
266	Reminders on how to import/export magnetic tapes
268	Changes to WMO FM 92 GRIB
270	Changes to the Meteogram system
271	New ECFILE features on UNICOS
280	UNICOS on-line documentation: docview
281	File transfer via FTP (possible problems)
283	New features for Member State batch users (RQS 1.1)

<u>**No.**</u>	<u>**Still Valid Article**</u>
284	UNICOS 7 features & differences
286	Improving the performance of "model" jobs on the Cray Y/MP8 at ECMWF (pre-allocating disk space)
294	Changes to the Meteogram system
296	Introduction of the new (TCP/IP) ECFILE on the Y/MP16-C90
297	Automount on the C90
298	New Member State job queues
300	Techniques to reduce memory requirements and improve I/O performance on the C90
301	Use of *sendtm* (hints on how to use)
303	Change to the default retention period for files in ECFILE
305	TCP/IP network connections to external hosts

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

METEOROLOGICAL TRAINING COURSE

18 April - 17 June 1994

The objective of the meteorological training course is to assist Member States in advanced training in the field of numerical weather forecasting. The course is divided into four modules:

Numerical Weather Prediction

Met 1 (18 April - 6 May):	Numerical methods, adiabatic formulation of models, data assimilation and use of satellite data
Met 2 (9-20 May):	Parametrization of diabatic processes
Met 3 (23 - 26 May):	General circulation, systematic model errors and predictability

ECMWF Products

Met 4 (6 - 17 June):	Use and interpretation of ECMWF products.

Students attending the course should have a good meteorological background and are expected to be familiar with the contents of standard meteorological textbooks. Some practical experience in numerical weather prediction is also an advantage.

Students can attend any combination of the modules.

In each module there will be lectures, exercises and problem or laboratory sessions. Participants are encouraged to take an interest in the work of ECMWF and to discuss their own work and interests with the staff of the Centre. All the lectures will be given in English and a comprehensive set of Lecture Notes will be provided for modules 1, 2 and 3.

A booklet describing the course in more detail has been mailed to the Member States early January. The booklet also includes information on the application procedures and application forms.

Further inquiries to:

E. Kooij-Connally, ECMWF, Shinfield Park, Reading, Berks., RG2 9AX, U.K.

- Els Kooij-Connally

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

ECMWF CALENDAR 1994

21 February - 18 March	Computer user training course
28 February - 4 March	Workshop on seasonal forecasting
23 - 24 March	Finance Committee, 52nd session
31 March (pm) - 4 April	*ECMWF HOLIDAY*
12 - 13 April	Technical Advisory Committee Subgroub meeting
21 - 22 April	Scientific Advisory Committee, 22nd session
25 - 26 April	Technical Advisory Committee, 19th session
27 - 28 April	Policy Advisory Committee, 2nd session
2 May	*ECMWF HOLIDAY*
9 - 10 May	Security Representatives meeting
30 May	*ECMWF HOLIDAY*
18 April - 17 June	Meteorological training course:
18 April - 6 May	**Met 1:** Numerical methods, adiabatic formulation of models, data assimilation and use of satellite data
9 - 20 May	**Met 2:** Parametrization of diabatic processes
23 - 26 May	**Met 3:** General circulation, systematic model errors and predictability
6 - 17 June	**Met 4:** Use and interpretation of ECMWF products
1 - 2 June	Council, 40th session
20 - 22 June	Workshop - Semi-Lagrangian methods
29 August	*ECMWF HOLIDAY*
5 - 9 September	Seminar - Parametrization of the physical processes in models

26 - 28 September	Scientific Advisory Committee, 23rd session
28 - 30 September	Technical Advisory Committee, 20th session
4 - 5 October	Finance Committee, 53rd session
6 - 7 October	Policy Advisory Committee
14 - 16 November	Workshop - Modelling and assimilation of clouds
21 - 25 November	Workshop - Parallel processing in meteorology
1 - 2 December	Council, 41st session
23 - 27 December	*ECMWF HOLIDAY*

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

ECMWF PUBLICATIONS

TECHNICAL MEMORANDA:

No. 193	Synoptic cloudiness validation in the ECMWF model
No. 194	Variational assimilation at ECMWF
No. 195	Evaluation of forecast error covariance matrix
No. 196	Computation of optimal perturbations using a local projection operator: A winter case study
No. 197	Sensitivity of forecast error to initial conditions using the adjoint method

Forecast and Verification Charts to 31 December 1993

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

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.

		No	Newsletter Date	Page
GENERAL				
ECMWF publications - range of		26	June 84	16
Obsolete manuals, bulletins, and news sheets		53	Mar 91	30
	&	58	June 92	33
Technical Advisory Committee and Computing Representatives, Meteorological Contact Points		61	Mar 93	46
COMPUTING				
CFT77		36	Dec 86	12
Data handling system		57	Mar 92	16
Fortran 90 features in CF77		60	Dec 92	23
	&	61	Mar 93	29
	&	62	June 93	26
	&	63	Sept 93	33
MAGICS - the ECMWF graphics package		62	June 93	15
Massively parallel computing - ECMWF's current investigations		61	Mar 93	15
Migration of ECMWF's meteorological operational system to UNICOS on the CRAY Y-MP		52	Dec 90	10
Multitasking ECMWF spectral model		60	Dec 92	3
Networks				
- ECMWF's internal network		54	June 91	20
- New LANs at ECMWF		59	Sept 92	20
Supervisor Monitor Scheduler (SMS)		59	Sept 92	13
Telecommunications				
- The ECNET system		54	June 91	16
- Digital links to Member States		57	Mar 92	19
UNICOS 7		58	June 92	23
UNICOS scripts for ECMWF's meteorological operational system		52	Dec 90	11
Workstations at ECMWF		55	Sept 91	7
	&	61	Mar 93	21
Y-MP/C90: An introduction		57	Mar 92	10

METEOROLOGY	No	Newsletter Date	Page
Comparison between SSM/I and ECMWF total precipitable water	57	Mar 92	3
Data acquisition - ECMWF's new system	46	June 89	21
Development of the operational 31-level T213 version of the ECMWF forecast model	56	Dec 91	3
Envelope orography - discussion of its effects	33	June 86	2
ECMWF Analysis			
- New version of analysis system	35	Sept 86	16
- Divergent structure functions	42	June 88	2
- Revised use of satellite data	39	Sept 87	4
- The variational analysis scheme - main features and some early results	62	June 93	5
- Use of TOVS satellite data at ECMWF: a new approach	61	Mar 93	3
ECMWF Preprocessing - new scheme	43	Sept 88	3
Ensemble prediction	58	June 92	5
Ensemble prediction system: expert meeting	63	Sept 93	18
ERS-1 mission	54	June 91	8
Evaporation from tropical oceans	51	Sept 90	3
Forecast model			
- Cloud cover scheme	29	Mar 85	14
- Convection - parametrisation of	43	Sept 88	6
- Increased resolution - studies of	38	June 87	10
- Initial conditions - the spin-up problem	39	Sept 87	7
- New surface/boundary layer formulation	63	Sept 93	3
- Parametrisation of gravity wave drag	35	Sept 86	10
- Revisions to physics	46	June 89	3
- Revision of the clear-sky and cloud radiative properties	61	Mar 93	3
Global forecast experiment at T213 resolution	41	Mar 88	3
Good prediction of a severe storm over southern Sweden	50	June 90	10
GORBUSH - a storm in the Mediterranean Sea	53	Mar 91	4
MARS - the ECMWF meteorological archival	32	Dec 85	15
and retrieval system &	33	Mar 86	12
Minimum temperature forecasts at the Regional Meteorological Service of the Emilia Romagna region (N. Italy) by the application of the Kalman filter technique	60	Dec 92	9
Monte Carlo forecast	49	Mar 90	2
Performance of the ECMWF model in tropical cyclone track forecasting over the western north Pacific during 1990-1991	58	June 92	16
Potential vorticity maps at ECMWF	50	June 90	3
Recent verification of 2m temperature and cloud cover over Europe	54	June 91	3

	No	Newsletter Date	Page
METEOROLOGY (continued)			
Skill forecasting - experimental system	40	Dec 87	7
Systematic errors - investigation of, by relaxation experiments	31	Sept 85	9
Use of reduced Gaussian grids in spectral models	52	Dec 90	3

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

USEFUL NAMES AND 'PHONE NUMBERS WITHIN ECMWF

		Room*	Ext.**
DIRECTOR	- David Burridge	OB 202	2001
DEPUTY DIRECTOR and HEAD OF OPERATIONS DEPARTMENT	- Michel Jarraud	OB 010A	2003
ADVISORY: Available 9-12, 14-17 Monday to Friday			2801
Other methods of quick contact:	- Telefax (+44 734 869450) - VMS MAIL addressed to ADVISORY - Internet mail addressed to Advisory@ecmwf.co.uk		
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	- +44 734 499 840		
Reception Counter	- Tape Librarian	CB Hall	2315
Tape Requests	- Tape Librarian	CB Hall	2315
Terminal Queries	- Norman Wiggins	CB 026	2308
Telecoms Fault Reporting	- Michael O'Brien	CB 028	2306
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				
Head of Research Department	- Anthony Hollingsworth	OB 119A	2005	
Computer Co-ordinator	- David Dent	OB 123	2702	

---

* CB - Computer Block
OB - Office Block

** The ECMWF telephone number is READING (0734) 499000, international +44 734 499000, or direct dial to (0734) 499 + last three digits of individual extension number, e.g. the Director's direct number is (0734) 499001.

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

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