CURRENT AND PLANNED USE OF RELATIONAL DATABASE SOFTWARE AT ECMWF

A.Hofstadler
ECMWF
Reading, UK

1   INTRODUCTION

Following ECMWF's third Workshop on Meteorological Operational Systems in November 1991 it was decided at ECMWF to gain some experience in the use of a Relational Database Management System (RDBMS). A commercial RDBMS (EMPRESS) was made available for experimentation on one SUN workstation. Following 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, concentrating on the Data Acquisition and Preprocessing system.

2   DATA ACQUISITION AND PREPROCESSING

A new Data Acquisition system has been developed, and currently is run in parallel with the operational Data Acquisition system, which is running on a VAX-cluster under VMS using ISAM (Indexed Sequential Access Method). 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 the 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 GTS (Global Telecommunications System) abbreviated header, with the messages themselves stored as BLOBs (Binary Large Objects), 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 (Preprocessing) 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 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.

Observational data (eg for analysis cycles) can be extracted by using SQL (Structured Query Language) (at the moment the data for the operational analysis is still extracted on the VAX).

An overview of the new Data Acquisition and Preprocessing system can be found in Fig. 1.

WORKSTATION (SUN / SGI)

VAX	Data-acquisition (C, MR-routines)	Decoding and quality-control (FORTRAN, C, MR)		Data-extraction (SQL)

MDB

Meteorological Bulletins

three new tables every day (original, duplicate, station-IDs,) + yearly statistics-table

NFS

Data-files

Bracknell

Offenbach

SYNOP
SHIP
TEMP
PILOT
DRIFTER
AIREP
ACAR
SATEM
SATOB
TOVS250
TOVS120
ERS1
WPROF
PAOB
GRID
GRIB
STORM
FRANCE
WARNING

RDB

Decoded and quality controlled BUFR messages GRIB messages

one new table per BUFR-type + one GRIB table every day
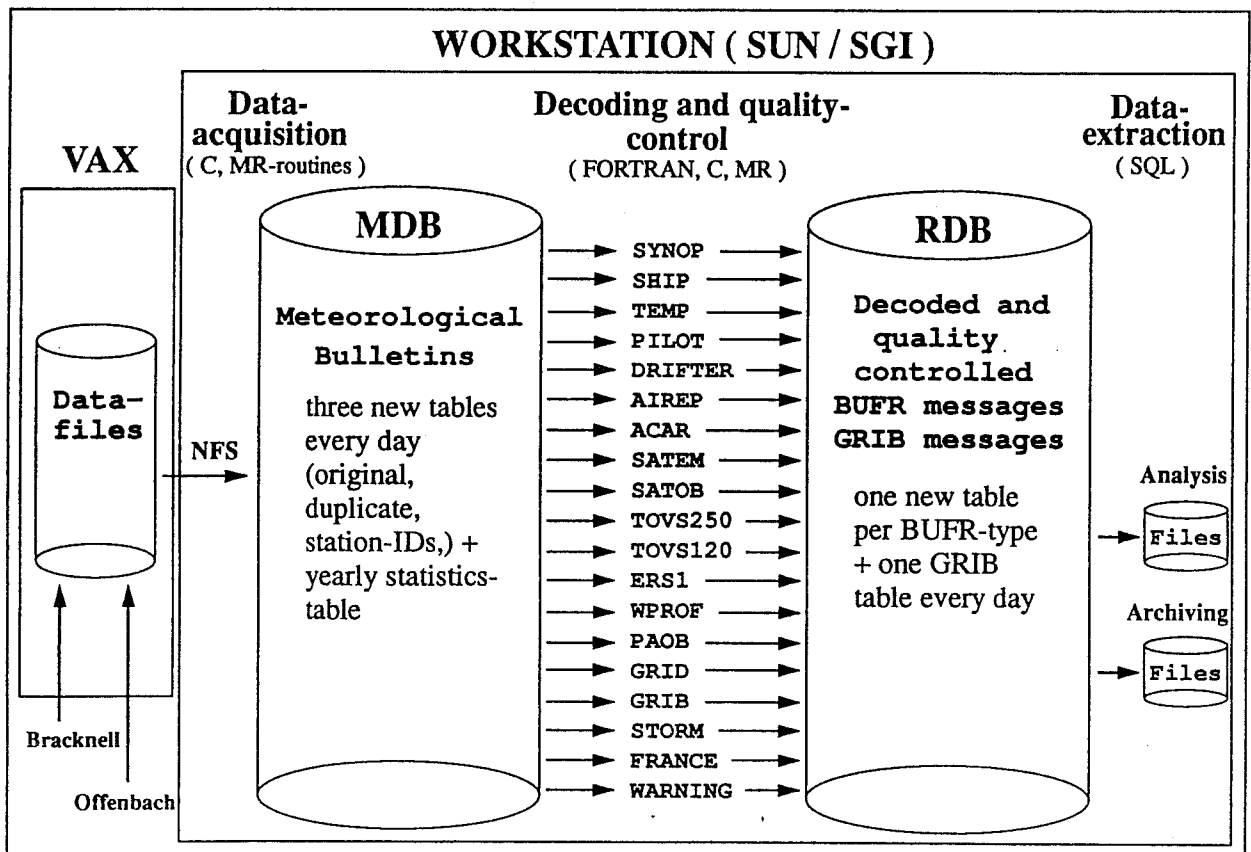
Analysis

Files

Archiving

Files

Fig. 1   The new Data Acquisition and Preprocessing system at ECMWF utilizing EMPRESS.

The new Data Acquisition and Preprocessing system will enable easier access to the data, make the data flow through the system more transparent and improve the monitoring of data. Statistical information about the data flow for a time range of up to one year will be made available.

3   OTHER APPLICATIONS USING EMPRESS

Some smaller read-only databases have been created, which will be used by the new Data
Acquisition and Preprocessing (eg WMO Vol-A station list, WMO Vol-C bulletin list). Most of the
programs to feed data into those databases use ESQL (Embedded SQL). A few tools for
monitoring the dataflow, and allowing manipulation and correction of messages in the MDB are
beeing developed using 4GL (Fourth Generation Language).

Besides those applications, which will form a part of the EMOS (ECMWF Meteorological
Operational System), EMPRESS is used at ECMWF for a rapidly growing number of tasks (eg as
part of MARS on workstations (see talk by B. Raoult in these proceedings), for Data Services to
process observational data and to manage customer information, for the ERA project to handle
BUFR messages and GRIB fields (see talk by J.K. Gibson in these proceedings)).

4   OUTLOOK

The new Data Acquisition and Preprocessing applications and databases will have to be tuned, so
that the performance and stability of the system reaches operational limits. The monitoring
facilities will be improved with the aim of tracing messages which are not received anymore or
which are sent without prior notice. It will have to be tried to utilize advanced EMPRESS features,
such as the database server (to allow a fully networked system) or shared memory (to improve
performance).   After an estimated testing period of six to nine months the new system will serve as
a replacement for the current operational Data Acquisition and Preprocessing.

5   RESUME

After obtaining some basic knowledge about relational databases, and about the capabilities of the
currently supported RDBMS, it has been found to be easy to design a database and write the
necessary database applications. The package used has proved to be a 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 RDBMS, 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.