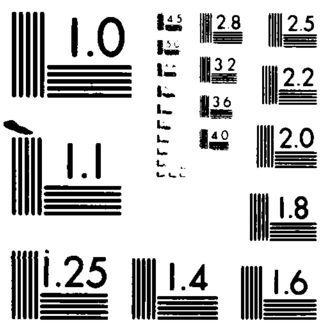
AD-A099 429 ROYAL AIRCRAFT ESTABLISHMENT FARNBOROUGH (ENGLAND) F/G 1/1
A SCIENTIFIC DATA BASE FOR THE ROYAL AIRCRAFT ESTABLISHMENT, FA--ETC(U)
APR 80 T R SIZER, D S WOODWARD, C GRAFF
UNCLASSIFIED RAE-TR-80048 DRIC-BR-74911 NL

END
DATE
FILMED
6 81
DTIC



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS 1963-A

UNLIMITED

BR74911

TR 80048

AD A099429

LEVEL II

ROYAL AIRCRAFT ESTABLISHMENT

*

Technical Report 80048

April 1980

DTIC ELECTE MAY 27 1981 S D E

A SCIENTIFIC DATA BASE FOR THE ROYAL AIRCRAFT ESTABLISHMENT, FARNBOROUGH

by

T.R.H. Sizer, CEng, FIEE, FBCS
D.S. Woodward, PhD, BScEng, MRAeS, AMBIM, MBCS
C. Graff, BSc, MSc, MBCS
I.M. Cummings, BSc, MBCS
H.E. Taylor, BSc, MBCS

*

DTIC FILE COPY

Procurement Executive, Ministry of Defence
Farnborough, Hants

80 9 8 052

UDC 681.3.065 : 5.001

ROYAL AIRCRAFT ESTABLISHMENT

Technical Report 80048

Received for printing 11 Apr[redacted]80

A SCIENTIFIC DATA BASE FOR THE ROYAL AIRCRAFT ESTABLISHMENT, FARNBOROUGH

by

T. R. H. Sizer, CEng. FIEE, FBCS
D. S. Woodward, PhD, BScEng, MRAeS, AMBIM, MBCS
C. Graff, BSc, MSc, MBCS
I. M. Cummings, BSc, MBCS
*H. E. Taylor, BSc, MBCS

SUMMARY

After specifying the data handling problems within the Establishment, details are given of a scientific data base which has been developed for use by scientists, engineers and administrators. A specific problem from the aerodynamics field is described and used as an example illustrating the application of the data base.

The conclusions are drawn that the use of a data base can confer benefits; that penalties have to be paid; that more experience is needed before the extent of the costs and benefits can be determined accurately.

Departmental Reference: IT 167

* H.E. Taylor is with Triad Computing Systems Ltd.

LIST OF CONTENTS

1 INTRODUCTION

The Royal Aircraft Establishment has its origins in military aviation and is today the Air Systems Establishment of the Procurement Executive, Ministry of Defence, with overall responsibility for the conduct and coordination of most research and development in aerospace activities.

This role requires that the RAE maintains and develops expertise over a wide range of disciplines and this expertise is deployed in support of the Services, Government agencies and industry, extending from research in the conceptual stages of aerospace projects to the evolution of new operational techniques and the solutions of problems as they arise in service.

Particular emphasis is placed on the rapid and effective transfer to industry of the knowledge and expertise stemming from research at the Establishment. The areas of research at RAE are complementary to those of other research and development establishments within MOD(PE). Relationships between the Establishment and the academic world are also close.

A characteristic feature of aerospace technology is a reliance on facilities for testing and calculations. RAE maintains a range of experimental and computation facilities both for research and as a national resource on which the UK industry can call during the development of aircraft and weapons projects. Research in flight is equally necessary, and RAE maintains a fleet of aircraft nearly all of them specifically adapted for trials projects.

An intermediate stage in much of this testing and calculation is often a 'data base' relevant to the activity. Sets of numbers require to be distinguished from and related to other similar sets and maintained for easy access by a variety of people together with details of the circumstances in which they were generated. This Report describes work done to harness computer data base techniques to the needs of the RAE.

2 DATA PROCESSING ASPECTS IN GENERAL AT RAE

The data processing activities at RAE cover a wide spectrum. For present purposes they can be divided into a number of categories: small to medium scientific and engineering computations; processing of experimental data; processing of administrative data; workshop scheduling; modelling and simulation; machine tool control programs; and large scale computations (sometimes referred to as 'number crunching'). Much of the computing work associated with these categories is processed on the central computing facilities, which at present

048

consist of an ICL 1906S, an ICL 1904A and a dual DEC 1099 system. Plans exist for a considerable increase in these facilities in 1982. In addition to these three mainframe computers there are over 100 mini-computers throughout the Establishment.

Data handling by the Central Computing Facilities is extensive. On average 480 million characters are on-line for immediate access by the many users with remote access terminals; 2.8×10^9 characters reside in the off-line file store on magnetic tape and a further 3.68×10^{10} characters are archived outside the file store.

The rate of growth in file store is considerable. Currently files are being created at a daily rate exceeding the total on-line capacity and there is a marked trend towards larger individual file stores. The growth rate indicates that about 1% of the on-line file store is retained for a long period.

3 VOLUME OF DATA AND HANDLING PROBLEMS

Generally it is a requirement that scientific and experimental data be retained for several years and it became clear that so far as computer-based data was concerned the volume of accumulated data was making its management a problem. Standardisation only applied insofar as users had to adapt to the requirements of the ICL George 3/4 operating system. No attempt was made to consider standardised data formats or anything resembling a data base approach. There was a lack of portability of data between research groups though it is of interest that the problem of interchangeability and portability was not present to the same degree between groups within RAE and associated external collaborators and contractors.

Problems had become particularly acute in Aerodynamics Department where, because of the extensive use of test facilities and on-line data logging systems, the volume of computer-based data was growing at a very high rate and the data is required to be kept safely for at least 10 years.

It was already known that several groups of scientists had a desire to use data from a variety of sources but were thwarted by the computing difficulties involved. Although most groups were experienced data handlers and appreciated the inefficiency of using different systems, they all had significant investments in software specially developed for their own projects. Conversion costs in adopting any common system were regarded as prohibitive. On the other hand, groups about to undertake projects which would involve large volumes of data would clearly benefit from the availability of common facilities.

As a result of informal contacts it was possible to define a set of common data manipulation tasks:

(i) Data collection and/or storage involving a group in the task of accepting raw data from its source, and storing that data in the preferred computer system: the data medium and format were determined by the originator (another research group, an external organisation or a data recording device). The mechanisms for data transmission ranged from the simple use of a processor-to-processor link or the development of a specialised software translator between alien character sets and formats.

(ii) Improving integrity of data: different methods were employed ranging from the visual inspection of tabulated listings and interactive editing of data to the development of software packages which automatically identified all doubtful data items in the file. Whichever method was used the research group always required access to data at its most basic level, the data item.

(iii) The data conversion task including the reformatting, sorting, merging and selection of data records: in general, applications software was developed to perform any data conversion. Such software was written by a research group to satisfy its immediate objective; the program itself and the data files on which it could operate were regarded as inseparable. Naturally, when data conversion requirements changes, that is when the formatting of the input file or the output file changes, the application program frequently needed to be at best modified, or at worst re-written.

(iv) The presentation of data: this fell into one of the following forms:

(a) A set of tabulations
(b) Approximate graphs produced by line-printers
(c) High quality graphs produced on graph-plotters.

(v) Application software for the analysis of data: programs made data requests to the George operating system to retrieve data from the storage hardware. Thus the operating system made a significant contribution to data manipulation by the user, as well as facilitating the development of his software.

4 THE NEED FOR A DATA-BASE

A project team was set up to consider matters more formally and to establish the feasibility or otherwise of introducing a technical data-base for use on an Establishment-wide basis. Staff came from two sources - a section of Aerodynamics Department concerned with large amounts of data arising from wind tunnel tests, and from Instrumentation and Trials Department (then Mathematics and Computation Department) who, as providers of the Central Computing Facility, were well aware of the size of the central file store and the problem of the management of the rapidly growing volume of data.

It was decided that a feasibility study contract should be placed to be followed, if feasibility was demonstrated, by a further contract for either the supply of a suitable commercial data base, or for the design, development and supply of a data base written specially for RAE. The contract for the feasibility study was won by TRIAD Computing Systems Ltd and the feasibility study commenced in August 1976.

To provide a framework for the feasibility study the following stipulations were made:

(i) The purpose of a technical data base system must be to provide improved facilities to research staff and management.

(ii) It must be evident that a significant overall reduction in programming effort would be obtained in such activities as the display and processing of data.

(iii) There must be better exploitation of hardware resources.

(iv) The accessibility of the data must be improved, both to the group responsible for data collection and to other groups who may wish to use the data, so as to encourage the wider exploitation of the data.

(v) There must be an improvement in the availability of management information in forms suitable to the recipients, which would enable the resources of the Establishment to be utilised more effectively.

In their report[1] TRIAD produced the following broad conclusions based on an examination of 15 research groups:

(i) Most groups devoted considerable effort to the detailed software aspects of data handling. This was in contrast to the effort devoted to organising the data in a way which would enable it to be accessed easily in the future.

In general these two aspects were not distinguished, leading to duplication of software effort even within the same group.

(ii) Considerable commonality of logical data structure was observed. Scientific groups fell into two classes, and a group concerned with management data formed a third class. It was noted however that all three classes of data structure could readily be accommodated within a single data handling framework.

(iii) The total load on computing hardware resources was relatively high, confirming that any data-base system must be designed to operate as efficiently as possible. It was noted that the efficiency with which hardware resources were used varied significantly between groups.

(iv) Little use was being made of data produced by any research group outside the group itself. It was confirmed that the difficulties lay not in the lack of desire to exchange data but in the computing difficulties involved.

(v) The high cost of producing the data argued strongly for a storage and retrieval system permitting maximum value to be derived from it. The ability to correlate data produced on different occasions was clearly desirable but it was found that correlation was in general performed only within a single experiment.

(vi) Most groups produced special programs to process and display data and there was little interchange of that type of software.

The study thus clearly confirmed the early conclusions and the need for a common data handling framework.

5 THE TYPE OF DATA BASE NEEDED

After the decision to proceed had been taken the first task was to describe, in conceptual terms, the type of data base required. This was done as follows:

(i) The data base user would be able to store data in a structure natural to the data collected. He would be absolved from all detailed software considerations, but would need to consider carefully the structure of the data itself and the ways in which it was to be used. Whilst straightforward data structures could be stored using a standard protocol, the handling of more complex or novel structures would require skill and experience. This skill and experience would be in data handling (arguably, part of the experimenter's stock in trade) rather than in software design and development.

(ii) An application programmer would, within certain limits, be able to write a program easily to use just that part of the data structure of interest to him. The data base system would then satisfy the application program's need for a particular data structure by retrieving the desired part of the actual data. This should encourage the production of software which would be both simple and widely applicable.

(iii) So far as the data-handling aspects were concerned, both data and programs would be readily portable between the different computer configurations (providing that the data base system was available on all of them). The normal considerations of storage media and language compiler compatibility would obviously still apply.

(iv) The data base system (supported by the operating system for the host computer) would provide full facilities to protect the data against loss or damage.

6 A TYPICAL SCIENTIFIC PROBLEM NEEDING A DATA BASE SOLUTION

Before describing the data base design which resulted from the foregoing work, it may be of interest to describe an actual scientific problem which faces Dr D.S. Woodward, an aerodynamicist at RAE concerned with research into high-lift systems, and which is typical of these applications amenable to data base applications.

A high-lift system changes the curvature of a wing surface in a streamwise direction and opens up a number of slots running spanwise in order to increase the lifting capability at landing and take-off so that these phases of the flight can be accomplished at a safe speed with the required payload. Thus in the cruise portion of the flight the aerofoil section through the wing looks as described in Fig 1. At landing however, it looks as it is shown in Fig 2, that is it consists of many separate bodies.

In order to understand how these systems work aerodynamically it is necessary to know how the local pressure varies at a large number of points on each of these bodies. Usually this is done experimentally in a wind tunnel and produces large quantities of data, which need to be processed in a number of different ways in order to extract a proper understanding of the flow. The quantities of data are such as to prohibit the use of elementary storage media such as punched paper tape or cards. Therefore long before the aerodynamicists considered anything as advanced as a data base they sought ways of handling the data in better forms.

As FORTRAN programmers they settled on a file design based on the concept of multi-dimensional arrays, arranged as shown in Fig 3. The data structure produced by an experimental run would consist of (1) an 80-character title which identified the contents of the file and could easily be made unique; (2) a fixed data block, which is of no interest in the context of this Report; (3) a three-dimensional (3-D) array containing the co-ordinates of the points where the pressures are measured (since surface pressure is measured by means of a small hole, drilled at 90^{o} to the surface and connected to a pressure sensor (transducer) by a small pipe running below the surface of the body, each measurement point is called a 'hole'. For each hole there are two co-ordinates and an integer body number*); (4) the main array of pressures - again a 3-D array - one pressure for each combination of hole number, body number and test condition.

The file design catered for a maximum of six bodies (the five illustrated plus one for contingency) and a maximum of 100 pressures per body, that is, about 25% greater than one might reasonably expect on one body.

During a run, readings from the transducers attached to the holes appear in the form of voltages which are recorded on some suitable medium, say 8-hole paper tape. This output is later processed by a computer program which applies calibrations, compensates for drift in the readings of two known calibration pressures, and converts to the non-dimensional numbers used by aerodynamicists. The post-processed data is stored (filed) on magnetic tape.

The system works well and those aerodynamicists who use the system are able to interpolate the data, evaluate local loads by integration, and do even more complex analysis in a matter of one or two days instead of the weeks it had taken previously.

Some time after this system had been completed, Aerodynamics Department investigated the reason for under-utilisation of its automatic graph plotting facility - off-line graph plotters fed by punched tape or cards. Users maintained that it was too time-consuming to produce the right data in the right order to make its use worthwhile for anything but the largest and most stereotyped of job. The conclusion drawn was that if most data were stored in some

---

* Aircraft are, of course, three-dimensional objects and hence, strictly, each hole should be specified by three co-ordinates. However for convenience in analysing the data it is usual to keep one of these co-ordinate values constant, and it is, therefore, not recorded in this data structure. The schema presented in section 10 is intended to cope with the general situation and hence contains all three spatial coordinates X, Y, Z, in its BODYCOORDS record.

clearly defined format on a computer medium permitting random access then the data items to be plotted could be assembled easily and quickly and it would then be worthwhile to use a plotter for much smaller jobs.

With the background of the filing system for the high-lift data this seemed an important conclusion that could be accommodated relatively simply by something like six standard file designs to cover the data for the whole of Aerodynamics Department. This proved to be naive thinking for when it came to designing these six 'standard' files the following difficulties arose:

(i) the size of the array dimensions required in different experiments varied to a large extent - whereas an array size of six bodies x 100 pressures was adequate for high-lift work, other experiments demanded, say, 20 bodies x 30 pressures. Clearly a file that could be more than half-empty all the time was extremely wasteful.

(ii) it was desirable to add additional dimensions to the basic file of the original system. Thus for instance in the original system 'cuts' could be made across the data array to produce the variation of pressure either on one body over the incidence range, or across all the bodies at one incidence, but in a developed system it could well be of equal interest to look at the variation on one body, at one incidence over a range of Reynolds or Mach numbers. Even these simple considerations added two more possible dimensions to the array in the simple file design. Once again the possible size of the file was becoming very large with a high probability that a large part of it would be empty.

(iii) As well as the local pressure measurements described above, measurements are also made simultaneously of the attitude of the model and the overall forces and moments acting on it. The original system catering for pressures only was developed in that form because in the data logging system of that time, pressures and forces were measured and recorded separately. Because the quantity of the overall force data was relatively small it could be handled manually and stored as line-printer output in a conventional filing system. However this was a serious shortcoming since, in analysis, pressure and attitude or overall force data need to be related. A developed system had to be able to deal with the storage and retrieval of both types of data, together or separately, since in other types of test only one or the other might be measured.

From this it was abundantly obvious that some more flexible concept of the data was required.

7 THE DESIGN CONCEPTS OF THE DATA BASE, AND MANAGEMENT OF THE PROJECT

To return to the main theme an early decision to be made was whether to use one of the commercially available data base systems (DBS) or to adopt an in-house approach. There are aspects of the experimental and theoretical analysis of a scientific data bank which differ from the commercial applications for which the majority of DBS are intended. For instance, there is the need to introduce a new data collection into the data base and keep it physically separate from similarly structured and named data items and records; some DBS could not provide this feature.

On a more basic level, there were some DBS which did not include support for floating point numbers or variable length records in their definition. A review of DBS packages which were available for one or more of the RAE computers showed that no DBS product was available for more than one computer system. The recommendations of the CODASYL Data Base Task Group (DBTG) were followed by a number of suppliers but short-listing a number of possible systems showed that they were incompatible with one another and would not support directly the requirement of portability of data and/or software.

Further examination identified those packages which were not suitable for such reasons as lack of full logical and physical data independence and the inability to interface to specified host languages. Finally two were left. Both followed CODASYL DBTG but neither could satisfy RAE requirements in full without extensive enhancements.

It was difficult to estimate accurately the manpower and machine resources required for conversion, even if a contractual agreement could be reached with the supplier of the package. It was concluded that the problem involved in producing versions of a DBS to run on hardware types for which it was not intended would be severe, and expensive to implement. In addition annual maintenance and usage levies would be payable together with fees for multiple site usage.

The in-house solution thus looked attractive, and the design of a custom-built DBS would allow full technical and contractual control by RAE. The features of the system would meet the RAE initial requirement and enhancements of modifications could be made more readily as RAE would be in full possession of the source code and documentation at all levels.

The decision was made therefore to have a custom-built system, and the preparation of the system design documentation began. It would have been possible to produce a design for a dedicated system which would satisfy a particular RAE requirement exactly, but it was felt that it was preferable to have a more generalised design which could be adapted as future requirements arose.

It has already been stated that interchange of data was already taking place between scientists at RAE and elsewhere - albeit with some difficulty. It was appreciated that the need for interchange could only steadily increase. Therefore it was also decided that the custom-built system should follow the recommendations of the CODASYL DBTG as closely as possible.

The system was structured into five major components comprising:

(i) A schema, sub-schemas and logical area definitions together with supporting software (these terms are described in section 8).

(ii) The logical to physical access mappings and software supporting the physical data base.

(iii) The data manipulation language (DML).

(iv) Utilities, such as on-line access and dictionary systems.

(v) Applications, such as graphical analysis programs.

A development schedule was then produced for implementing the first four components on an ICL 1904A computer under George 3. Later the system would also be available on an ICL 1906S under George 4.

Three important aspects of the development are worth mentioning:

(i) The portability of the DBS had already been identified as a primary requirement. To this end the DBS had to be developed almost exclusively using a high-level language for which compilation systems for anticipated machine types were available. At the time only one programming language, BCPL, could fill this requirement and so was chosen as the development language.

(ii) Although TRIAD were to undertake the development of the system it was envisaged that in time RAE would maintain and modify the system. A high standard of documentation was thus called for. The standard was devised and monitored by C. Graff of Computation Division, Instrumentation and Trials Department, RAE. In addition to comprehensive program and module descriptions, the need for four manuals was identified:

(a) System Design Report - a pre-implementation document detailing, at a high level, all aspects of the system design.

(b) Concepts and Facilities Guide - an introductory manual for users giving a high-level description of the system and its facilities.

(c) Programmers' Guide - a users' programming reference manual.

(d) System Manual and Specifications Manual - for use in system modification.

(iii) A self-monitoring capability was to be integrated into the DBS. This feature, one not found in many commercial packages, would be used by RAE when tuning was found to be necessary. The DBS could be directed to trace access paths and to identify the frequency of access to data structures, resident secondary store resident tables, and area records during the execution in main store of a particular application. Using the information, resource overheads could be quantified and inefficient implementation sequences modified.

The decision was taken to limit the scope of the implementation only to the basic DBS and its utilities. Applications software was to be developed by the research scientists themselves. Use of the data base system would therefore lead to a standardisation of data and software modules resulting eventually in mutually compatible data structures and names, and more general purpose software available to all. The data dictionary facilities were expected to play a key role in standardisation procedures.

One of the factors which can influence the success of any DBS is the provision of protection features. A number of commercial systems include features which have been successfully proven. However such features are usually expensive in terms of the use of machine resources, and a comprehensive security system can severely limit the throughput of a batch system or the response times for on-line access. For performance reasons, together with the fact that the George 3/4 operating system can be used to dump all amended files daily, it was decided that RAE DBS need not include its own security features.

Privacy of information is an important and controversial topic for some computer users. Consequently privacy features can also be found in some commercial packages. Again such features are costly in system performance and resource usage. It was decided that the file protection checks provided by the

George operating system were satisfactory and no additional features were to be provided by the DBS.

It was clear that the project would be large enough to warrant strict project control. It was agreed that the prime responsibility for this should rest with Instrumentation and Trials Department. However it was also clear that for such a novel and evolutionary task a fair degree of lattitude was necessary at the working level. Control was effected by establishing weekly progress meetings chaired by H.E. Taylor; major technical issues were either resolved by C. Graff or referred by him to the monthly formal project management meetings chaired by T.R.H. Sizer. Additionally fixed budgets of spend on computer time were set, and adhered to by the TRIAD programming team.

8 FEATURES OF THE RAE DBS

As far as possible the DBS follows the recommendations of the CODASYL DBTG. The DBS user therefore defines the data structure in terms of Data Description Languages (DDLs). The highest level of structure is the schema, which completely defines a single data-base; the schema allows the user to collect together logically related data items and assign them to a named record type and also to define relationships between record types in terms of set types. The overall structure that can be represented is a network (or 'plex'). Records associated with a particular set will in general have a logical order imposed upon them (such as being sorted on a nominated key) and users can 'navigate' through this in order to locate required records just as they can navigate through the structure as a whole. An important feature supported by the DBS is Secondary Key indexing; users may define an arbitrary number of search keys to be associated with each set. This means that the system is implemented using an inverted file structure. Individual records may be located on primary key value, secondary key value or via their logical position in a particular set.

A subschema DDL is available which allows users to define their local view of the data. In order to 'bind' a particular subschema to a schema (that is to set up rules which will enable the DBS to relate one view to the other), a schema is initially 'compiled' using a stand-alone utility called the 'schema compiler'; the subschema is bound to the schema dynamically by means of an appropriate call to a DBS library routine. As well as allowing a user to select particular records and set types of interest, the subschema allows him to define new relationships formed by partitioning and concatenating schema record types; it is also possible to define new data items, for example a new data aggregate.

The schema and subschema only allow users to define data structure; the data itself is held in host operating system files called 'realms'. Since a user will, in general, only want to store a selected subset of schema records in a realm, and, in particular, only selected items from these records, a third and final DDL is available, called Realm DDL, which effectively allows users to set up a 'filter' between a realm and the schema to which it conforms; without the filter, many null value data items will have to be stored with a corresponding waste of file store space. When a user wishes to view a realm through a specified schema/subschema binding, he is said to 'open' the realm; essentially, the binding process described above between schema and subschema is extended to include the realm. At any time, a particular realm may be bound to more than one schema/subschema combination. The realm description also includes a certain amount of information concerning the physical form of the data, for example the expected number of records and their average size.

The concept of separating realm descriptions (which relate to physical descriptions of the data) from schema and subschema descriptions of the data is considered to be an extremely desirable feature of the system which is not always found in other CODASYL systems. Realms are accessed by user programs invoking a FORTRAN subroutine call to one of a set of procedures collectively referred to as a Data Manipulation Language (DML). Currently, FORTRAN is the only language which can host such calls (although a COBOL interface is envisaged).

The DML routines provide for location, storage, retrieval, modification and deletion of records, as well as navigation through the data.

A number of stand-alone utilities are also associated with the system. These include:

(i) Data Interrogation which allows interactive data base access without the necessity for a user-written program.

(ii) Data Dictionary which allows the user to determine logical structure of schemas and realms.

(iii) Create Realm which allows users to create empty realms.

Examples of a schema, subschema and realm are illustrated by Figs 4a&b.

9 ACCEPTANCE TESTS

It was clearly necessary to design a set of acceptance tests which would both test the facilities provided by the DBS and give some guide as to its performance. This task was undertaken by C. Graff[2,3,4] and I.M. Cummings[5,6]. The detailed specification of each test, which included schemas, subschemas, realm descriptions, program descriptions, GEORGE macros and data to be used were produced by RAE and subsequently used by TRIAD. The tests were split into two categories - Facilities and Performance.

9.1 Facilities tests

The facilities tests were themselves split into several categories. The first tested the schema compiler verifying that all legal syntactical forms of the schema Data Description Language (DDL) could be processed, and that all illegal forms were handled correctly. A non-trivial schema was used to test for legal validation; and a series of illegal schemas was produced, each based on the legal schema, but containing a deliberately introduced error to test the error detection capability of the compiler. The second category tested the subschema compiler in an analogous manner.

The third category tested the utility programs associated with realm creation. As before, a test using a legal realm description was performed followed by a series (25 in all) of illegal realm descriptions. Tests were also performed which altered the record density in a realm.

The fourth category tested the DML procedures. A series of FORTRAN programs tested data base, realm, record and set accessing procedures; corresponding programs to test error handling were also specified.

The final category of tests dealt with the utilities: Data Dictionary, Data Interrogation and Schema Deletion.

9.2 Performance tests

The performance tests were designed to compare the performance of non-DBS programs with the performance of equivalent programs incorporating DBS software.

Two application programs from Aerodynamics Department were chosen for these tests. The first program performs editing of wind-tunnel experiment data stored on magnetic tape, and the second performs a series of calculations using the data from the magnetic tape. These programs were converted by replacing magnetic tape and disc file accessing instructions by calls to DML routines, the data to be used being transferred from the magnetic tape to a realm in the data base.

The two suites of programs were run in a dedicated George 3 environment and the following noted for each program run:

(i) central processor time clocked,

(ii) program size.

In addition, the number of realm transfers (disc accesses) were noted for the DBS programs.

The acceptance tests were run by TRIAD during their system testing phase and subsequently run by RAE after the system had been submitted for formal acceptance tests. The results are shown in Table 1.

Table 1

Results of performance tests

Edit program, non-MDMS		Edit program, MDMS	
Program time	0.08 CPU seconds	Program time	1.02 CPU seconds
Size	11 K words	Size	68 K words
		Realm transfers	3395
Calculation programs, non-MDMS		**Calculation program, MDMS**	
Program time	3.54 CPU seconds	Program time	4.55 CPU seconds
Size	17 K words	Size	67 K words
		Realm transfers	3492

This Table illustrates that there is a price to be paid for the facilities a DBS provides. In particular, disc channel utilisation is greatly increased. It is for the individual user to assess the benefits conferred (such as ease of data access, data storing, data structuring etc) against the cost (such as increased program size, longer program elapsed time due to data transfer etc).

10 A SCIENTIFIC DATA APPLICATION

Returning now to the problem described in section 6, a schematic logical data structure can be set up which shows how the system copes with the basic problems encountered by the aerodynamicists in trying to generalise their specialised filing system for high-lift data. The schema assumes the form:

EXPT SERIES ID
↓
EXPT ID → BODY COORDS
↓
DATA POINT
↓
PRESSDIST

Record name		Description
EXPT SERIES ID	contains data items -	TUNNEL ID, MODEL NO, BALANCE ID, START DATE, FINISH DATE
BODY COORDS	contains data items -	BODY ID, NO. OF HOLES, (HOLE NO. X,Y,Z) repeated NO. OF HOLES times
EXPT ID	contains data items -	RUNNO, MACH NO, REYNOLDS NO, SLAT CONFIGURATION, FLAP CONFIGURATION, TAIL CONFIGURATION, U/C CONFIGURATION, INTAKE CONDITIONS, BLOWING PARAMETERS, SWEEP, RUDDER, AILERON, ELEVATOR, ANGLES, DATE, etc
DATAPOINT	contains data items -	DATAPOINT NO, α, C_L, C_D, C_M, C_Y, C_N, C_l
PRESSDIST	contains data items -	NO. OF HOLES (HOLE NO, CP) repeated NO. OF HOLES times

where: α = incidence
C_L = lift coefficient
C_D = drag coefficient
C_M = pitching moment coefficient
C_Y = side force coefficient
C_N = yawing moment coefficient
C_l = rolling moment coefficient
C_P = pressure coefficient

In most cases all this data will not be relevant - the relevant data items are declared in the Realm Description when the data is first introduced to the data base. Thus a general schema can be produced which is applicable to a large body of experimenters, each of whom utilises only those parameters which are pertinent to his application. When the data is accessed by an application program, the data is located and retrieved by name so absolving the experimenter from any need to know how the data is stored. Furthermore generalised plotting programs can be developed since the data they need to access can be specified by name at run time.

The schema indicates that five record types, each containing many data item types, are sufficient to describe all the experimental data which may be held in the data base. The linking arrows indicate one-to-many relationships between records of different types. For example, in the data base, there will be many occurrences of records conforming to the EXPT ID record type; a collection, or set, of these occurrences will be 'owned' by one occurrence of a record conforming to the EXPT SERIES ID record type. Thus using these relationships the programmer is able to select the set of records required by selecting its owner record.

For example, if the programmer wishes to access the pressure values for DATAPOINT No. = 1, RUNNO = 2 and TUNNEL ID = 1234 these are the logical steps he must take.

(i) FIND an occurrence of EXPT SERIES ID for TUNNEL ID = 1234 (establishes corresponding EXPT ID record occurrences).

(ii) FIND an occurrence of EXPT ID for RUNNO = 2 (establishes corresponding DATAPOINT record occurrences).

(iii) FIND an occurrence of DATAPOINT for DATAPOINT No. = 1 (establishes corresponding PRESSDIST record occurrences).

(iv) Process PRESSDIST record occurrences.

11 ENHANCEMENTS

Provision is made in the basic design for enhancements to be made. Some have already been implemented in the light of users' experience; others are already under discussion. The three of significance which have been implemented are:

(i) The facility to search for a record specifying up to three secondary key values.

048

(ii) The facility to specify that a secondary key value to be used in a search can be within certain bounds, ie ± one part in 10^N, where N has a maximum value of 11.

(iii) The facility of index sequential realm organisation which allows records to be 'clustered' as they are stored. For applications where records are to be retrieved in the order in which they are stored, this facility greatly enhances the system performance.

Other enhancements of a more minor nature, such as finding the record occurrence in a set occurrence with the minimum/maximum value of a secondary key, have also been implemented.

12 CONCLUSIONS

At the time of writing the DBS has been released for 10 months. Apart from the original application foreseen by D.S. Woodward, about a dozen further applications have been identified by Instrumentation and Trials Department. They range over many RAE Departments and are concerned with both numeric and non-numeric work. Four pilot data bases have been set up and are being currently used; another two are in an advanced stage of development.

Current applications range from the monitoring of extramural contracts to the analysis of combat trial data. Initial indications indicate that data can be more readily structured and accessed under a DBS; a greater length of time will be required to assess more intangible benefits such as reduced maintenance cost of software systems.

It is clear that the advent of data base at the RAE will be of considerable benefit to a wide range of users and, to increase its value, there are two development areas currently being planned. The first involves implementing the system on PDP-11 computers. The second is concerned with enhancing the performance of the system, primarily in the area of secondary key indexing.

The method of project control described in section 7 can be deemed successful: in an overall planned time-scale of 18 months the project was only 2 weeks late.

Data base concepts are not easy to master and the programmer needs to do far more studying before embarking on a program than he would have to do, for example, before using a magnetic tape handling package. However, once the data base concepts have been grasped, the programmer should produce well-structured programs which reflect the logical organisation of the data on which the operations are being carried out.

In practice the acronyms 'DBMS' and 'DBS' are not used; the system is known in the RAE as the Multiple Database Management System (MDMS).

Acknowledgment

Acknowledgment is due to Dr K. McKenzie a director of TRIAD Computing Systems Ltd who, as well as coordinating activities on the project within TRIAD, provided helpful and constructive criticism of the Report in draft form.

REFERENCES

No.	Author	Title, etc
1	A.E. Taylor	Feasibility study and report. TRIAD Computing Systems Ltd
2	C. Graff	Acceptance tests for the RAE scientific data base: complex schema/subschema findings. RAE Technical Memorandum Math-Comp 7904 (1979)
3	C. Graff I.M. Cummings	Acceptance tests: utilities. RAE Technical Memorandum in preparation
4	C. Graff I.M. Cummings L. Piggott	Acceptance tests: DML procedures. RAE Technical Memorandum in preparation
5	I.M. Cummings	Acceptance tests for the RAE scientific data base: schema and subschema compiler tests. RAE Technical Memorandum Math-Comp 7905 (1979)
6	I.M. Cummings	Acceptance tests for the RAE scientific data base: the performance tests. RAE Technical Memorandum Math-Comp 7906 (1979)

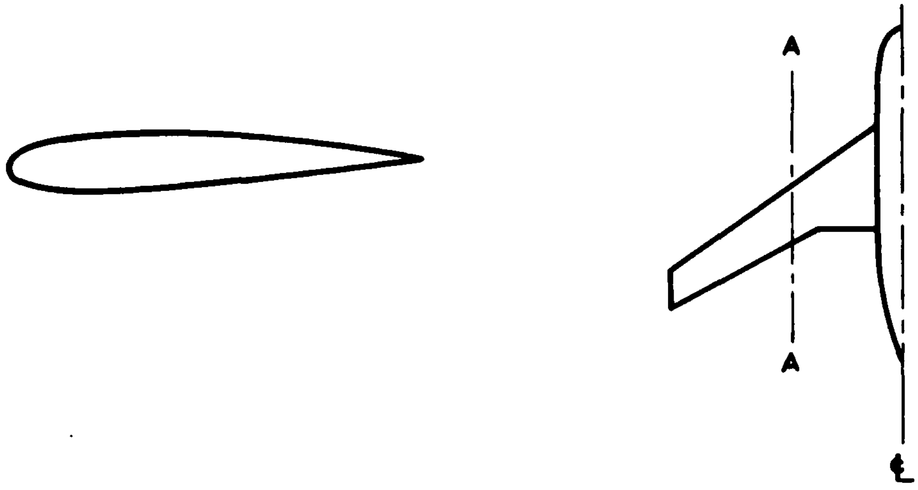


Fig 1 Section A–A on wing

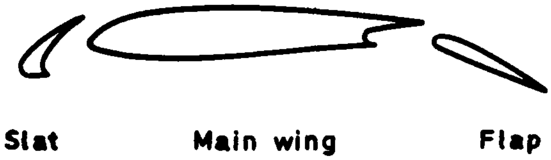


Fig 2a High lift system deployed

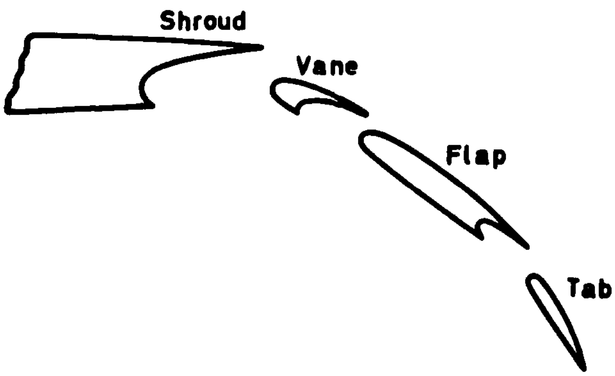


Fig 2b Deployed triple slotted flap

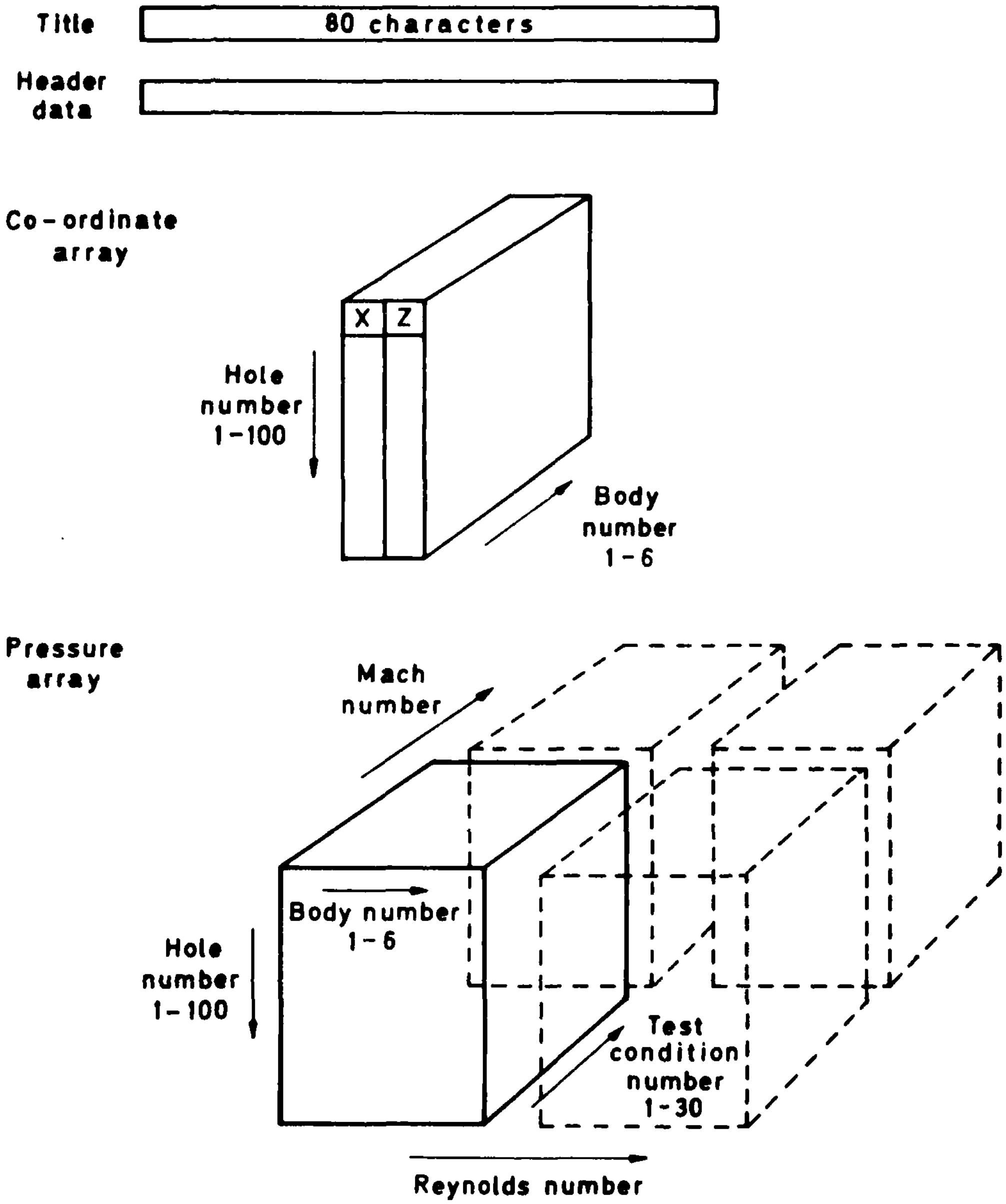
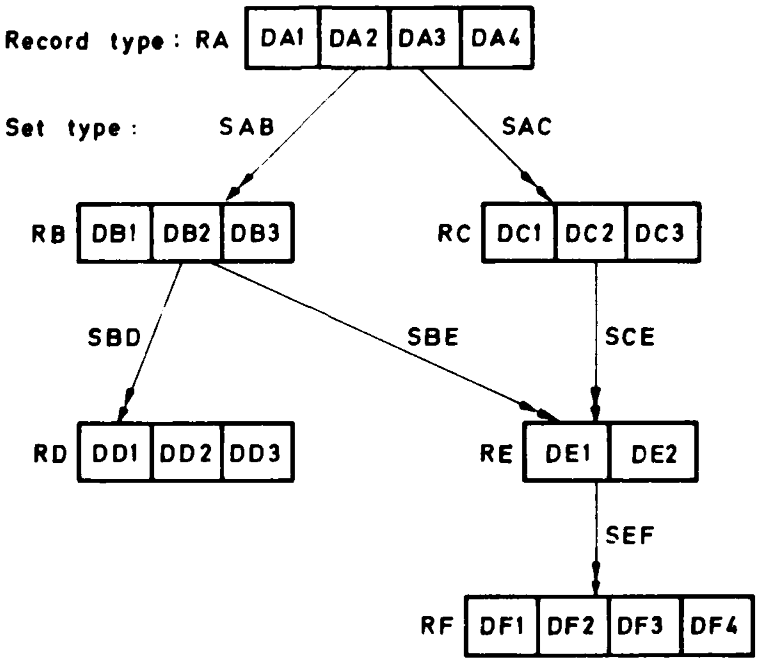


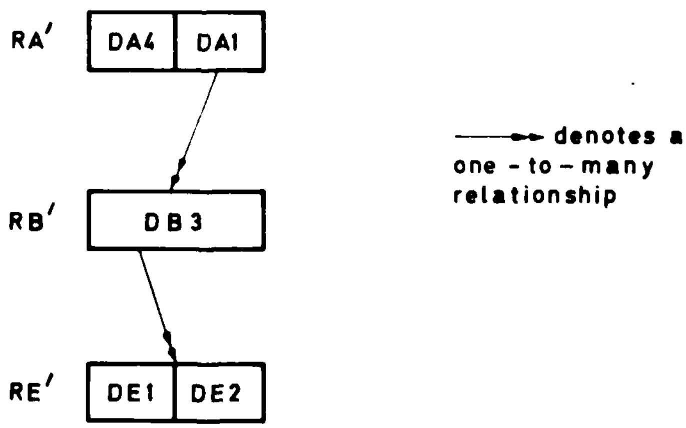
Fig 3 Fortran-based file design for aerodynamic data

TR 80048

An example of a schema



A possible subschema based on the above schema



TR 80048

Fig 4a

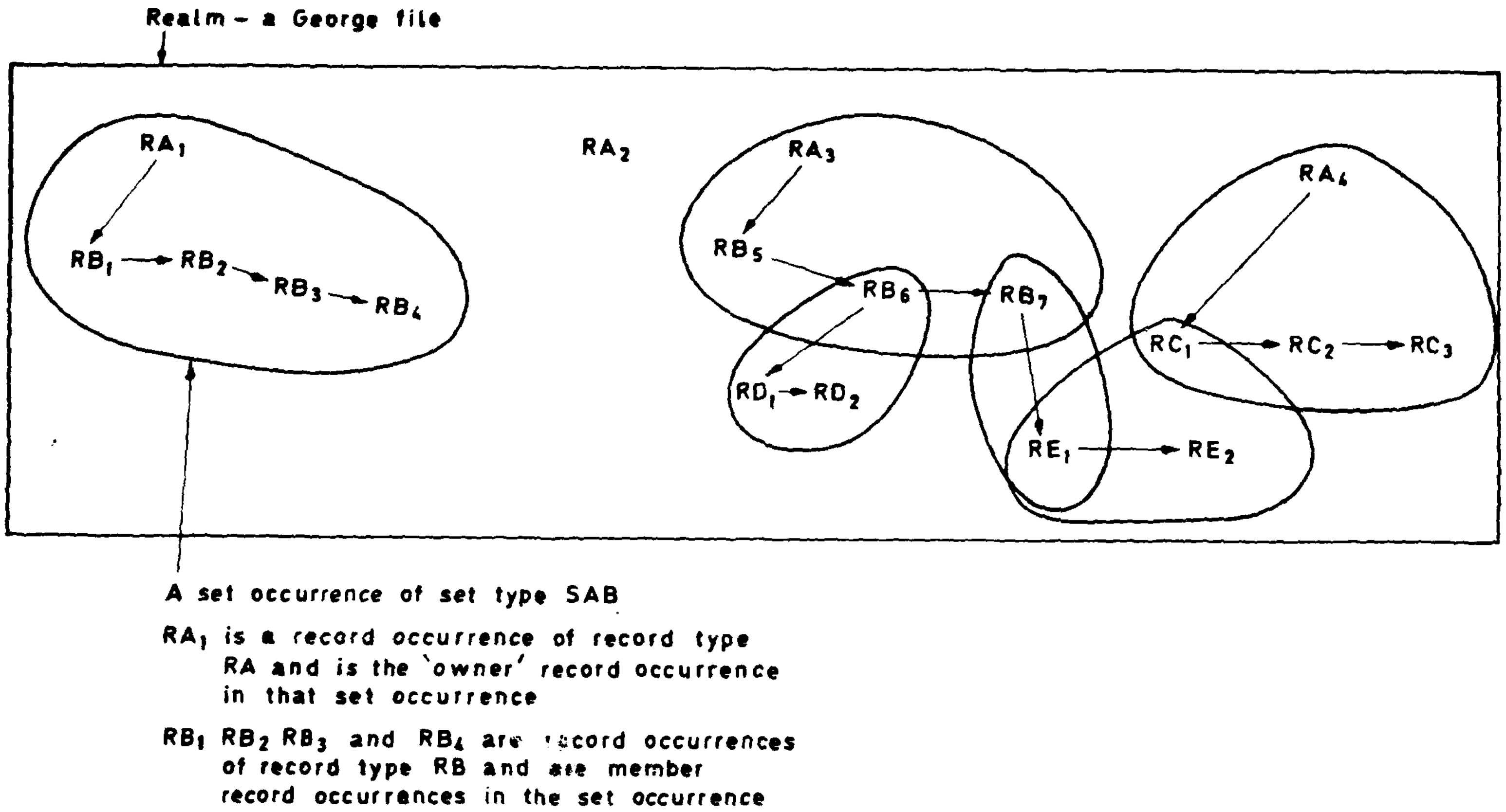


Fig 4b Data stored conforming to the schema

D

FI

6-