DATABASES

Edited by
S. M. Deen
P. Hammersley

DATABASES

Proceedings of the 1st British National Conference on Databases held at Jesus College Cambridge, 13-14 July 1981

Edited by

S M Deen, *University of Aberdeen*
P Hammersley, *Middlesex Polytechnic*

PENTECH PRESS

London: Plymouth

First published 1981 by
Pentech Press Limited
Estover Road, Plymouth
Devon

ISBN 0 7273 0405 4

British Library Cataloguing in Publication Data
National Conference on Databases *(1st: 1981: Cambridge)*
Databases.
1. Data base management – Congresses
I. Title. II. Deen, S. M.
III. Hammersley, P.
001.64'42 QA76.9.D3

ISBN 0-7273-0405-4

Printed in Great Britain by Anchor Press, Tiptree, Essex
Bound by William Brendon and Son, Tiptree, Essex

DATABASES

Organisers

University of Aberdeen
University of Cambridge
Middlesex Polytechnic
The British Computer Society

Conference Committee

Chairman: S M Deen, University of Aberdeen
Secretary: P Hammersley, Middlesex Polytechnic
Member: J K M Moody, University of Cambridge
Member: M J R Shave, University of Bristol
Member: P M Stocker, University of East Anglia

Acknowledgement

The organisers wish to thank IBM (UK) Limited for their generous financial support.

PREFACE

Following the success of the International Conference on Databases (ICOD – 1) at Aberdeen, it was agreed to organise a second international conference (ICOD – 2) in 1983 (September 20-23), with a British national series of conferences (BNCOD) held annually in between. The BNCOD series is meant to focus primarily on British research work, although overseas papers are welcome. This conference, the first of the British national series (BNCOD – 1), is being organised jointly by the Aberdeen University Computing Science Department, The British Computer Society, the University of Cambridge and Middlesex Polytechnic. Its objective is to encourage database research in Britain by bringing together the researchers and other interested parties.

The papers for this conference were selected in two stages. Initially thirty papers were shortlisted on the basis of abstracts, and then the completed papers were refereed leading to the final selection of ten papers. The selection criteria used were research content, relevance to implementation, quality of work and the spread of subject areas. The selected papers can be divided into the following five groups:

1. Data Dictionary Facilities (two papers)
2. Query Facilities (two papers)
3. Query Languages (two papers)
4. DBMS Implementation (two papers)
5. Optimisation and Evaluation (two papers)

In addition to these papers a research review paper is also presented. As this is the first British national conference on database research it was thought that such a review paper would be useful, particularly to the new entrants in the field. The published conference proceedings contain these eleven papers in order of their presentation. An overview of the papers is given below.

The research review paper on the state of the art in database research (given by the chairman) addresses the mainstream activities of database research, highlighting some of the more interesting works and future trends, and with a large number of references. The next paper by F S

Zahran (London School of Economics) looks into the IBM and ICL data dictionary systems and presents a generalised model which would allow the user to extend the data dictionary database structure and its management software. A G P Brown, H G Cosh and D J L Gradwell of ICL describe their rapid application development system (RADS), which allows a quick development of non-procedural applications from user defined data structures, using an interactive data dictionary facility to store the definition of data programs.

P M D Gray (University of Aberdeen) presents a new relational operator called 'Group by' which enhances the capability of the relational algebra; this has been implemented as part of a relational algebra which is used to generate Fortran application programs for a Codasyl database. R M Tagg (independent consultant) summarises the findings of the BCS Query Language Group on the desirable features for query languages, and uses them to evaluate a number of current commercial implementations. J Longstaff, F Poole and J Roper (Leeds Polytechnic, Sheffied Polytechnic and Durham University) discuss an implementation of a 'natural' language interface based on three user modes which allow users to progress easily from the user mode one (for naive users) to user mode three (for more experienced users) where an SQL like language is supported. The paper by T Crowe, D R Hainline and R G Johnson (Thames Polytechnic) describes a relational query validation facility using reverse translation, which helps to reduce errors and, in particular guards against pitfalls such as connection and selection traps encountered in relational operations.

M A Gray (University of Cambridge) explores a general implementation technique, based on the mathematical concept of lattice, to represent imprecise values (including null values and values with error limits) in databases. J W R Glauert, T J King and M Robson of the same university present a paper on a relational database system that runs on a number of minicomputers using the Cambridge ring. S W Ho (University of Hong Kong) describes a mathematical algorithm to determine the optimal search sequence of files for complex queries where several files, particularly with replicated data, are involved. The paper by B J Lowndes and J W Martin (University of Liverpool) presents an evaluation study of four database products, viz. ROBOT, IDMS, RDMS AND RAPPORT; the main factors considered include ease of use (setting up a database, loading the database with data and unloading such as dumping the data) and efficiency (performing a join and retrieval on primary and secondary keys).

Turning from the presenters to the delegates the general response to the conference has been encouraging. The number of delegates attending is more than three times the number originally expected on the basis of past experience. To me it indicates a growing British interest in database research and I trust that this conference will give a further impetus in that direction. It may be noted that the next conference, BNCOD – 2, is expected to be held at the University of Bristol from 30 June to 2 July 1982; further information can be obtained from the conference chairman.

Finally I wish to thank The Computer Journal, Computing, the IUCC Bulletin and a number of other news magazines and their staff for giving publicity to the conference. I wish also to mention especially the staff of Middlesex Polytechnic, Aberdeen University and Jesus College,

Cambridge for their cooperation and assistance. Lastly I must express my thanks to all the referees (whom I cannot name for reasons of confidentiality) for doing a thorough and conscientious job within a very strict time limit.

S M Deen
Conference Chairman

Contents

THE STATE OF THE ART IN DATABASE RESEARCH

S M Deen

Department of Computing Science, University of Aberdeen

In this report a number of database research areas, namely: database architecture, data modelling, database design, database modification, run-time optimisation, end-user facilities, database machines and distributed databases are briefly examined. Current research efforts in those areas are outlined, highlighting some of the more interesting works. Research trends and future directions are also indicated. The presentation is primarily aimed at the intending database researchers.

1. INTRODUCTION

The term database as a means of storing a vast quantity of data was first coined in the sixties, with IDS (now IDS-1 of Honeywell) ushering the birth of modern databases in 1964. Research activity in databases also began in earnest at about that time, a number of papers on binary relational systems being published around 1967, the year when the Codasyl Committee renamed its List Processing Task Group as Data Base Task Group (DBTG). Bachman played pioneering roles in the developments of IDS and DBTG proposals. Codd's paper on the relational model was published in 1970, and was followed in quick succession by a number of other works, notably late Senko's DIAM model in 1973. The progress was so rapid that today the database technology is recognised 'as one of the seven major areas of research and study in computer and information science and engineering' [ACM TODS, vol 4, p261, 1979].

In order to present the state of the art in this rapidly growing branch of information technology, we shall divide it into the following 8 subject-areas:

1. Database Architecture
2. Data Modelling
3. Database Design
4. Database Modification
5. Run-time Optimisation
6. End-User Facilities
7. Database Machines
8. Distributed Databases

The choice of this categorisation, although reflects author's preferences, has not been without some fine-tuning problems, as some issues and contributions fit more than one category. The 'Database Abstraction' model of Smith and Smith, and the SDM model of Hammer and McLeod, could be viewed as user schemas and hence covered under End-User Facilities rather than under Data Modelling where they have made a great impact. Equally the functional approaches of Buneman and Shipman, could have been designated as modelling tools rather than as end-user facilities. To resolve this, we have categorised all works specifying end-user languages as End-User Facilities. A similar delineation problem has been encountered in other areas as well, particularly in database design and modification, which cover some common ground despite addressing different issues.

In the following eight sections, we would explore these subject areas, indicating issues, current state of research and in some cases future directions. A short conclusion is also presented.

2. DATABASE ARCHITECTURE

Under database architecture we wish to discuss the schema levels and associated facilities, along with the role of a dictionary. Major contributions to this area have been made by bodies such as ANSI/SPARC[1], CODASYL DDLC[2], BCS/DBAWG[3], BCS/DDSWP[4]. Academic and research contribution has been less spectacular.

Early databases were thought to be the direct extension of large files, with a single level description, without making any distinction between logical and physical levels. No separate user views were possible. IDS-I and TOTAL are examples of this period. The Codasyl DBTG Report of 1971 [5] recognised the need for a two level architecture, with separate schema and subschema, as a means to provide data independence. The Device Media Control Language identified there, was seen as a language for physical storage allocation rather than a third level of data description. It was the ANSI/SPARC proposal of 1975 [1] where a three-level data description for greater data independence was advocated (figure 1).

The external schemas are seen as language-independent user views of data, on which language-dependent user views such as the present day Codasyl subschemas would map. They are also expected to have some independence from the conceptual schema in data definition. However it is not entirely clear what ANSI/SPARC meant by a conceptual schema. The minimal subset of its content on which most experts are likely to agree is:

Logical data description
Description of data relationship
Description of integrity constraint
Description of privacy constraint

All storage dependent attributes of data, including access paths, are intended to be described in the internal or storage schema. Its basic content could then be

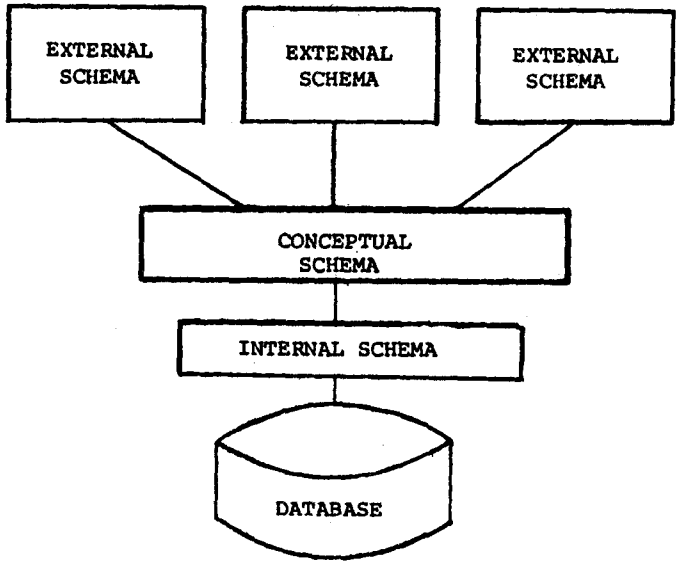

Figure 1

Storage Strategy

Storage space allocation
Storage description of data and relationships
Data placement strategy
Treatment of overflows

Access Paths

Key specification (primary and secondary keys)
Indexing techniques (several could be supported)
Chaining specification

Miscellaneous

Data compression techniques if any
Data encryption techniques if any.

The storage schema itself is independent of physical devices in the sense that all media space are described in terms of pages rather than physical units of tracks and cylinders. This physical allocation could be made by a Device Media Control Language. Any given storage schema should work for any physical device, but clearly efficiency would vary.

The conceptual schema is meant to reflect the nature of the enterprise and hence its content is expected to be more stable than that of the storage schema, the latter reflecting the usage of data which changes more frequently. One should be able to modify the storage schema periodically to improve database performance, without affecting the conceptual schema. The user programs are guaranteed data independence through those three levels. The ANSI/SPARC proposal however allows for direct linking of external schemas to the internal schema, if efficiency rather than data independence is preferred.

If the argument of stability is applied to the content of the conceptual schema as listed earlier, then we see the following relationship:

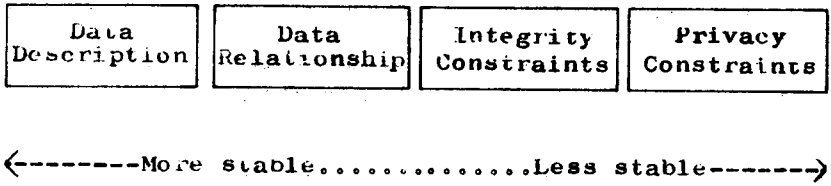

Figure 2

Should we resolve the conceptual schema into 4 others to provide greater facility for modification? There is a good case for separating the privacy constraints into an access-control schema. The separation of integrity constraints is however more debatable.

As it reflects the nature of an enterprise, the conceptual schema is regarded by some as a model independent view of data, the model dependent view being the Global Implementation Schema or Global Schema for short. One can then envisage

Conceptual External Schemas (CES) from which Implemetation External Schemas (IES) are derived. The union of the Conceptual External Schemas would then be the Conceptual Schema. We have then:

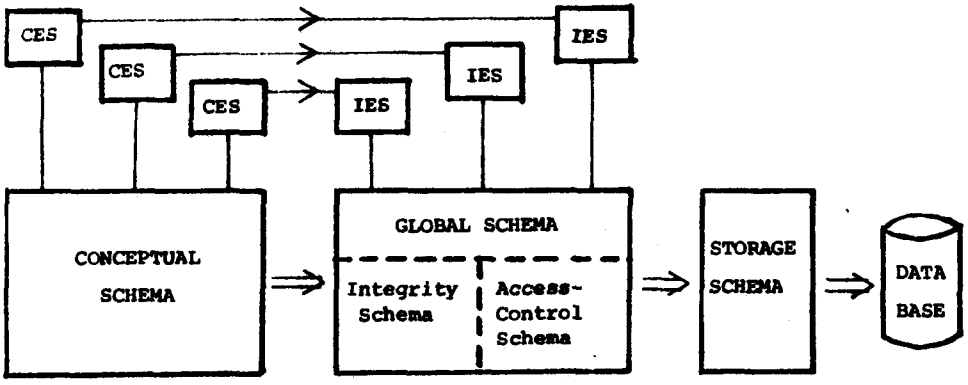

Figure 3

It can also be argued that a database and hence the global schema could represent only a subset of the data described in the conceptual schema. However in the rest of this paper we would assume a conceptual schema to be model-dependent (rather than model-independent) and identical to an ideal global schema, unless otherwise indicated.

A data dictionary [4,6] is now-a-days regarded as an integral part of a database architecture. It is supposed to service the users, designers, DBA and the control system in an all-pervading manner, by holding all the necessary information for the purpose. It is this 'holder' where we might store the definitions of data, their meanings, integrity and privacy requirements, authorisation status, source and object versions of application programmes and schemas, usage statistics, back-up facilities, routines, utilities, design tools, administrative and runtime control information, and so on. As a 'holder' the title 'data dictionary' appears to be a misnomer. However, if a data dictionary is to do all these, it would be very complex, and might turn out to be as complex as the database itself with an unacceptably high overhead. The problem therefore is the definition of what it could or should be, what it should do and how best to design, maintain and use it.

3. DATA MODELLING

Research in data modelling gained momentum after the publication of Codd's paper on the relational model, with a further impetus from Senko's DIAM model [21] and ANSI/SPARC'S conceptual schema [22]. The great debate of the early seventies, on the relative merits of the Codasyl and relational models [23] also helped to develop a clearer

understanding of their strengths and weaknesses. Bachman suggested an improvement of the Codasyl model by a Role model [24], and Codd has proposed an extension of the relational model to capture more meaning [25]. To overcome the limitation of the binary relational approach, we now have a new concept of 'irreducible relations' [6], which permits use of relations of higher degrees than binary only if absolutely necessary, such as to represent Data with 3 domains. Other workers adopted different approaches. Apart from the wide academic interest that the data modelling has aroused, it has also caught the imagination of IFIP and ISO, both of which have set up working groups. The IFIP/WG2.6 is designing a conceptual schema (the ENALIM model of Nijssen [27,28,29]) and the ISO/TC97/SC5/WG3 is studying data models with a view to future standardisation [30].

The problem of data modelling is often equated to the problem of designing a conceptual schema. But, as noted earlier, there is no agreement as to its content, although a conceptual schema is generally viewed as an information model representing a portion of the Universe. However, the Universe as it exists, is meaningful to us only through perception, and therefore what we model in a conceptual schema is a selected part of the perceived reality, sometimes referred to as a Universe of Discourse (UoD) [27]. It is assumed that a good model of the UoD would not only allow more meaningful and useful data description, but also be able to support the user views of all other data models as local or external schemas. This last aspect which Nijssen [27] calls the coexistence approach, is the primary objective of all unified or canonical data models. The first aspect is however double-edged, since we do not know how far we should over-load the structure of a model in order to make data more useful and meaningful. There is a point of diminishing return, as too complex structures, particularly those based on unfamiliar abstract concepts, are likely to be difficult for most users to comprehend. In any event, improvement in data modelling is likely to be continuous and evolutionary.

We shall give below an overview of some of the modelling approaches under the following headings:

Role model
Recent relational approaches
Other approaches

Some interesting characteristics of data that affect modelling approaches can be seen in [31].

3.1 Role Model

In the basic Codasyl model data are described as record types which can be related to each other by set types, characterised

by one owner record type and one or more member record types. Bachman has introduced in his Role model [24], as an extension of the Codasyl facilities, the concept of role and role records. An entity has an entity record which contains all the attribute values of this entity. However an entity may play one or more roles, and therefore the entity record should be logically subdivided into role records, one per role of that entity. Thus a person may play roles as an employee, as an employer, as a car owner; and if so, in Bachman's schema we must describe three role records (with data redundancy where necessary), all of which can be collectively viewed as a person record. A given role however can be played by more than one entity type. A role employer can be played by person and organisation, a role vehicle by car, bus, lorry etc. A set type in the role model is defined between two role types, one as owner such as employer and the other as member such as employee. Both owner and member roles in a given set type can belong to many different entity record types. Bachman's model does not consider the coexistence between relational and Codasyl models. Some of the limitations of the Codasyl model documented by Mercz in [32] would probably apply also to the Role model.

3.2 Recent Relational Approaches

The original relational model as proposed by Codd is considered inadequate for meaningful data description. We would list its limitations as follows:

(i) inability to describe data meaningfully

(ii) inability to support sequencing information as in the case of representing bus routes, each being defined by a variable number of bus stop names [38]

(iii) inability to define groups (e.g. to group day month year as Date)

(iv) difficulty to support semantic integrity among groups of tuples.

Codd [25] has recently proposed an extension of the relational model to remove the first obstacle by subdividing a fully normalised relation into meaningful sub-relations. This would help (i) and perhaps (iv). The binary relational model of Bracchi [33] is theoretically interesting and elegant, but does not relieve the problems (ii), (iii), and (iv). In his Entity-Relationship model, Chen [34] used domain to represent Bachman's role concept, and special relations to support 1:n relationship. The latter is meant to facilitate the transformation of 1:n relationships to m:n relationships if needed; and hence is considered to be a better construct. Domains and attributes are seen to have m:n relationships,

since the values of a domain (role) can appear as several attributes, and vice versa. Chen's model is claimed to be a unified data model. That a Codasyl subschema can be supported as a user schema in a unified (or canonical) data model based on relations is shown by Deen in [35].

A higher level approach based on the relational model is provided by Smith and Smith [36]. It consists of a hierarchy of meaningful abstract levels, each level being derived either by grouping related but dissimiliar objects into an aggregate object of a higher type, or by grouping similar objects into a generalised object of a higher type. Similar concepts have been used also in the TAXIS language [Section 7].

3.3 Other Approaches

The last few years have seen a number of alternative approaches on data modelling. We shall briefly mention a few of them below.

Nijssen's ENALIM model [27,28,29] is based on what he calls a conceptual approach. It describes data in very abstract form, with a heavy emphasis on integrity constraints, which typically can represent 80% of the conceptual schema description. The final version of the model is yet to be released, possibly as an IFIP/W2.6 publication. His co-worker in the ENALIM model, Falkenberg [37] has also developed another model called Object-Role model. Although there are some similarities in the basic concepts with the ENALIM model, Falkenberg's model is much less abstract. It shares with ENALIM the concept of 'object-role' pairs, each object such as 'salary' playing a role such as 'salary of ...'. The pairs are collected into meaningful associations. Objects can be grouped into hierarchies by Type declarations. The model claims to support a full set of integrity constrants and also the handling of the time dimension with its complex implications (see also [44]).

Deen's [38] Rolentity model (REM) is claimed to be progmatic and is based on the concepts of role (as in the Role model) and type. Each entity-type is viewed as a set of rolentity-types, one for each role of the entity-type, each rolentity-type having a set of relations. Relations can be shared between rolentity-types, and data replications are permitted. A given property-type may be represented by different attribute-names, and an association type can be defined between two rolentity-types. A full range of constraints, including the predictable commitment units can be specified, but the impact of time dimension is handled only partially. Objects described in the conceptual schema, including relations, roles, attributes, property-types, entity-types, rolentity-types, and so on, can be logically grouped into meaningful hierarchies by Type declarations, for