An application of CODASYL techniques to research in the humanities

L.D. Burnard

1.

1 The Codasyl database management system proposals have been much discussed but little used in Universities, [1] although several implementations of them are now widely used in industry and government. For this project we used IDMSX an extended version of the original Cullinane IDMS at present being developed by ICL for their new 2900 range. [2] It provides what is, I believe, currently the largest subset of the facilities proposed by the various Codasyl reports.

2 All database management systems include at least two major components, one for data description and the other for data manipulation. A Codasyl schema describes the data to be stored in terms of things called records, fields and sets — names with which no-one is content. A field is a named elementary piece of data (e.g. PERSON-NAME) and a record is a named collection of such fields which always co-occur and which we wish to treat as a unit. Occurrences of such records may be grouped together into sets. The records making up a set will be of at least two different types, and will include one occurrence of one type, known as the owner record, and one or more of different type(s), known as the set members. Apart from the restriction that owner and member records

must be of different types, and that a set may have only one owner, there is no limitation on the complexity of structures that may be built from these objects . A database consists of one or more areas of storage space (also called realms) in which record and set occurrences of different types are stored according to various schema-defined criteria. For example, records may be stored randomly through a realm, accessible only by pre-defined key, or (if they are members of a set) they may be stored as close as possible to their owner record. Sets may be implemented by means of chains of pointers stored with the records, or by separately stored indexes or pointer-arrays. The logical order of member records within a set is defined in the schema. The most noteworthy differences between a Codasyl-style database and a conventional data set are thus firstly that an area may contain records of many types (and of course occurrences of a particular record type may be found in different areas), and secondly that extensive cross-referencing between record occurrences is an integral part of the system.

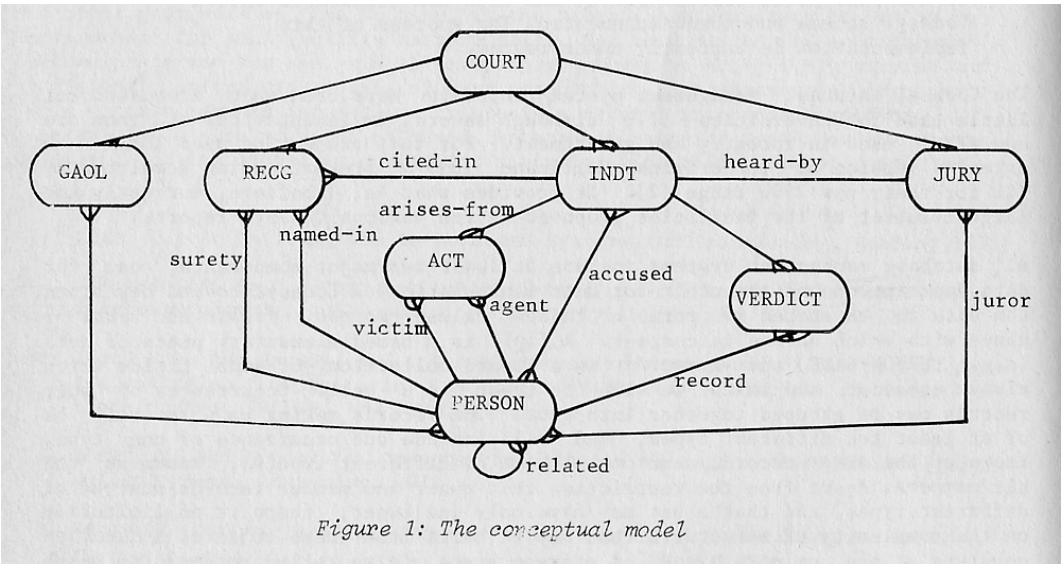
3 Data is retrieved from, and added to, the database by means of user programs (which may be written in Cobol or Fortran), or by utilities such as a query language processor. In either case, all data manipulation is performed in terms Of the records, fields and sets specified in the schema, using the CodasYl-defined data manipulation language.

4 The data to be processed by our system derives from the manuscript records of assize courts and quarter sessions held in Hertfordshire during the early years of the 17th century. [3] The most important of these include the commissions issued to the judges authorising the holding of a court , the gaol delivery calender which summarises its activities and includes lists of all those currently held in gaol; lists of recognsizances, that is, bonds entered into by individuals to fulfil certain conditions, usually to appear (or to ensure the appearance of another) to answer a charge; and lists of indictments.

5 The indictment, a formal accusation made by members of the community (the grand jury) on behalf of the king, includes information about the persons accused, the criminal acts committed and the results of the judicial process. It should be stressed that our major research focus is on the social context in which so-called criminal acts occur and are judged rather than on the criminal acts themselves. Our data does not record crime but crime prosecuted.

6 The process of data analysis, particularly when the full complexities of a social context are to be modelled, is neither simple nor definitive. The model builder mustbe ready constantly to revise earlier assumptions in the light of subsequent analysis. Our current entity model went through four or five evolutionary stages before reaching the state shown below.[4]

Figure 1. The conceptual model.



7 In this network, the nodes represent entities, that is, informational categories represented in the data,which we consider of interest. The vague subjectivity of this definition is intentional for the problems of formally identifying entities are far from trivial (Kent 1976). We adopted the pragmatic approach of classing as entities only those categories the instances of which could be uniquely identified. Thus, an instance of the COURT entity is uniquely identified by some particular combination of its defined attributes (type, date, reference number etc., as listed below the figure). The arcs in figure I represent relationships between entity instances, which are considered as objects in the model in their own right. Thus the particular court to which it is attached is not an attribute of the JURY entity (nor is the jury an attribute of the court), although relationships may be thought of in that way. The crowsfoot symbol is used to indicate that more than one instance of the entity it touches may be involved in the relationship. Thus, several JURYs may be associated with one COURT. A crowsfoot at both ends indicates a many-to-many relationshop, as, for example, that between JURY and PERSON (a jury consists of more than one person, and one

person may serve on more than one jury). Some relationships have clear semantic content; the two relationships between ACT and PERSON, for example, correspond with the two different ways a person may be related to an act: as its victim or as its performer. This formalism allows us to represent information which is semantically quite complex in a fairly natural and straightforward manner. Thus, an indictment is "heard-by" a jury, "arises-from" a number of acts, and specifies one or more people as "accused." For any recognisances "cited-in" the indictment there will be other people related by the "surety" relationship. A relationship may exist amongst instances of the same entity: in our data we often find (or conjecture) such relationships as "spouse," "employee," "sibling" etc between two PERSONS. Each of these should properly be represented by a different arc, but for convenience we summarise them all in the single relationship labelled "related." it should be noted that no relationship is assumed in our conceptual model between the entities VERDICT and ACT. This separation is justified by the observation that phrase such as "guilty to the value of twelve pence" (where no item worth twelve pencehas been stolen) occur at least as often as phrases such as "guilty on the first count." The temptation to link such a verdict with one of the several acts relating to an indictment should be resisted. As with the granting of benefit of clergy or remission due to pregnancy, it seems that these so-called "partial verdicts" were a common method of tempering justice with mercy in an age when all but the pettiest of crimes carried the death penalty.

8 Having thus represented in our conceptual model all the entities and relationships present in the data, we next consider the processes whereby they are created, destroyed and accessed, that is, the flow of information in and out of the system. Such considerations are obviously of the highest importance for a System processing highly dynamic data (e.g. airline reservations, stock control) but may safely be ignored when dealing with data which like ours has remained static for several centuries. The information flow required out of the system will become of importance, however, when we consider its Codasyl implementation.

9 At this stage therefore we note any attributes which may be useful as secondary keys during analysis, although it is often difficult to find an attribute which does not potentially qualify as a key. Indeed the requirement to prioritize (and hence prejudge) the usefulness of attributes as discriminants seems to run counter to the spirit of scientific enquiry.

10 Figure represents an early stage in mapping the conceptual model of figure 1 onto a Codasyl implementation model (also known as an extended Bachman diagram). In this figure, boxes represent Codasyl record types, and arrows represent Codasyl sets, going from owner to member. While many relationships can simply be replaced by sets, and most entities by records, special action is necessary for many-to-many and reflexive relationships, as summarised below.

Figure 2. Mapping rules for reflexive and many-to-many relationships.

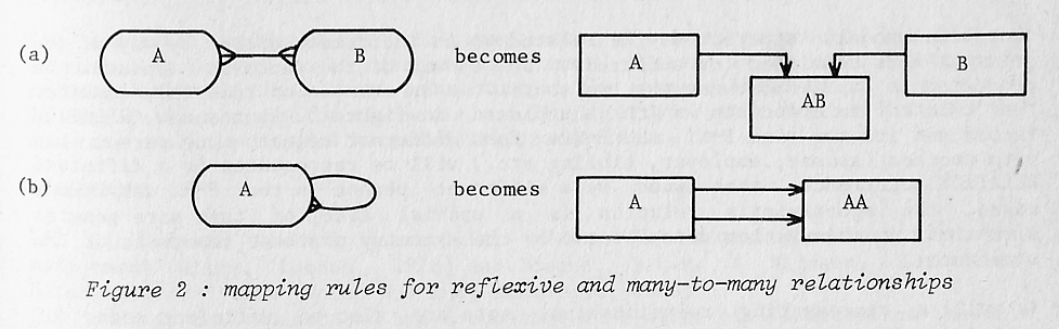


Figure 2 : mapping rules for reflexive and many-to-many relationships

11

12 As an example of rule (a) consider the "juror" relationship which is replaced by the J-JP set, JP intersection record and P-JP set. There will be one JP record occurrence for each member of a particular jury, each connected to the appropriate PERSON occurrence by an instance of the P-JP set; for each occasion On which some person has been a juror, there will be a JP record connected to the appropriate JURY occurrence by an an instance of the J-JP set. Although the JP record itself holds no information, the analogous GP record is used to hold information about the role the person plays in this gaol delivery (remanded on bail, escaped, pardoned etc.). Similarly the intersection record needed to represent the many-to-many accused relationship becomes a convenient place to store the verdict for each of the possibly many people named in one indictment, or for each of the possibly many indictments against one person. As in the conceptual model, a person may be related to an indictment either by way of the criminal acts committed (or suffered), or by way of the sentence imposed. No connexion is implied between the two routes. As an example of rule (b), consider the "related" relationship which is replaced in figure 3 by the new RELATION record and its two sets P-RI and P-R2. Each different relation one person has with another (spouse, employer, sibling etc.) will be represented by a different RELATION occurrence,

each owned by a different person in the P-R2 set. In a sense, the agent/victim relation is a special case of this more general situation, an observation demonstrated by the symmetry of these two parts of the structure.

Figure 3. Implementation model.

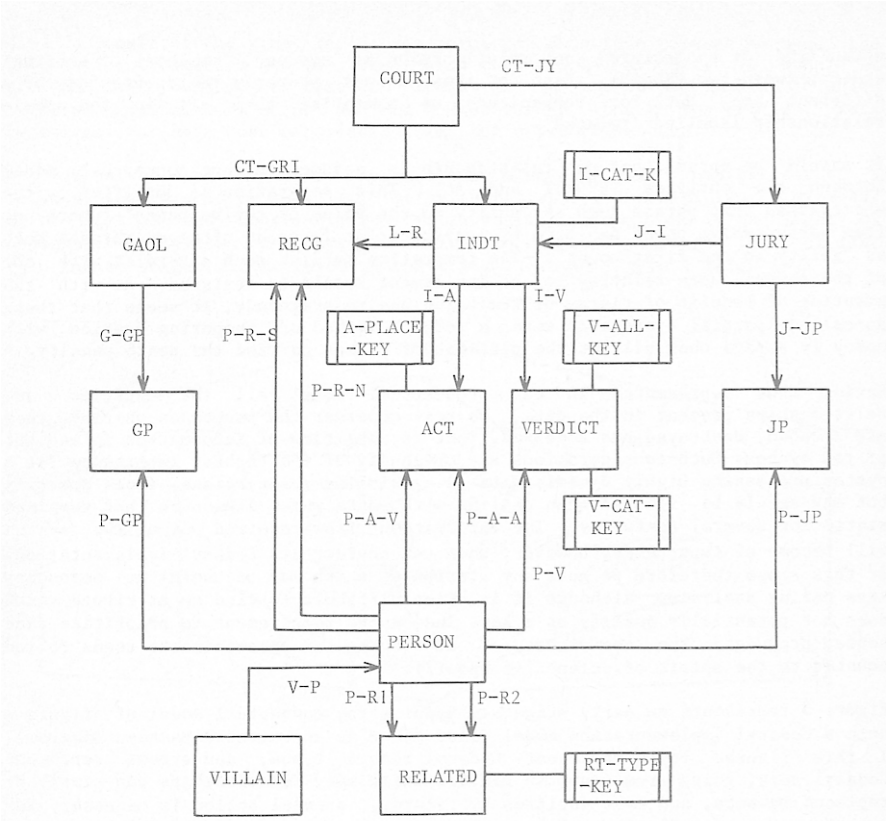


Figure 3 : Implementation model (first stage)

13 As well as representing relationships, sets are also an efficient means of optimising frequently-used access paths. The V-P set for example simply connects all PERSON occurrences for which there has ever been an indictment. Without it, we would have to inspect all of them, rejecting those which are not owners in a P-V or P-A set occurrence, which, (while simple and feasible) is too slow for a frequently-required process. The owner record in the V-P set is a dummy in which we store some summary statistics about the attributes of the set members. If at a later date the need arose to optimise access, perhaps to all people who have been victims of a crime, a similar set could be added. Such structural changes may be made by means of a Restructure Utility

supplied with IDMSX. A similar mechanism is used in the record key indexes provided with IDMSX. These (represented by the smaller boxes in figure 3) are system-defined sets connectin to other occurrences of records with the same value for some specified field or combination of fields. Thus all criminal acts performed at the same place may be retrieved using the A-PLACE-KEY, all people related in a certain way using the RT-TYPE-KEY, all indictments for a particular category of crime using the I-CAT-KEY and so on.

14 Our implementation of the system has now reached the stage of loading in test data and evaluating the usability of the system. The data is first transcribed into a (comparatively) easily punched form by Peter Lawson, who uses human judgement as a historian to eke out the notorious stupidity of the computer when faced with such equally notorious problems as the identification of possible relations between people (Wrigley, 1973) and the disambiguation of variant spellings and aliases. To assist him in the latter task, an index of names is maintained (at)resent outside the database, though it will be integrated at a later stage), which is updated by the first of the two programs used to load the database. Essentially a data validation program written in SNOBOL4, this checks the iltitiched data for a wide range of errors and inconsistencies and reformats those parts of it which can currently be loaded into the database. This task is performed by the second database load program, which is written in Fortran, using the Codasyl-defined data manipulation language.

15 Retrieval of information is carried out by an interactive query processor provided by ICL for IDMSX databases, called DATA DISPLAY. This facility allows the non-programmer (who must, however, have a good understanding of Godasyl structures) to retrieve and display at a terminal individual records or parts of them, using the navigation paths defined in the schema. All records on one path may be displayed, or a selection of them defined by simple logical operations on the contents of records. The simple and English-like query language includes facilities for defining macro commands to abbreviate complex queries. From being a convenient method of program testing and debugging, it seems probable that this utility will become the standard procedure for ad hoc enquiries to establish the possible utility of more fundamental analysis, which will be (carried out by specially-written Fortran DML programs. The next stage will be to provide a means of interfacing the database to the standard statistical package SPSS. Work currently going on at Queen Mary College London suggests this might be possible using an ICL product called FAME; we have yet to evaluate this possibility.

16 For academic applications, the complexity of data structures which may be supported by Codasyl style database management systems is at least as strong a recommendation as their known popularity in the commercial world. If the user is to take full advantage of the power and flexibility of such systems a formal and accurate analysis remains an essential prerequisite.

BIBLIOGRAPHY

Wrigley, E.A. Identifying people in the past (E.Arnold, London, 1973)

———. Data and reality (North Holland, Amsterdam, 1978)

NOTES

1 On Codasyl systems see Olle, T.W. The Codasyl approach to database management (Wiley, London, 1978) and Douque, B.C.M. & Nijssen, G.M. (eds) Database description (North Holland, Amsterdam, 1975)

2 IDMSX is fully described and documented in a suite of ICL Technical Manuals (TP 6923–6). The account given here of entity modelling owes much also to the ICL Data Dictionary System.

3 I am indebted to the historian working on this project, Mr Peter Lawson (Worcester College) for explaining to me the vagaries of English legal history. Most of the courts records have been published and partially indexed by HMSO

4 I am grateful to Jim Pimpernell of ICL for invaluable assistance during the initial stages of this project.

ABSTRACT

This paper describes work using a Codasyl type database management system for research into a large and complex body of data deriving from seventeenth century court records. A conceptual model of the real world represented by the data was translated into a Codasyl schema and then implemented. The success of this implementation is currently under review.

AUTHOR

L.D. BURNARD

Oxford University Computing Service