

wide interest.

The paper provides a very good description of many early developments in the area of data-base management systems. It is one of the few papers to appear in the published literature that treats the subject of data-base management systems from an historical view. "Evolution of Data-Base Management Systems" is important and should be read by all workers in the field.

References

- (1) M. Kochen "Adaptive Mechanisms in Digital Concept Processing" in Discrete Adaptive Processes - Symposium and Panel Discussion, AIEE, New York, 1962, pp. 50-58.
- (2) R.E. Levien and Maron, M.E. "A Computer System for Inference Execution and Data Retrieval" CACM 10 (11) (Nov. 1967), 715-721.
- (3) J.L. Kuhns "Logical Aspects of Question-Answering by Computer", Proc. Third International Symposium on Computer and Information Sciences (COINS IV), December 1969, Academic Press, New York 1969.
- (4) J. Minker and Sable, J.D. Relational Data Systems Study, Final Report, Auerbach Technical Report AVER-1776-TR-1 July 1970 submitted to Rome Air Development Center. Also RADC-TR-70-180. Final Technical Report September 1970 AD-720-263, ED-049-645.

Jack Minker

RELATIONAL DATA-BASE MANAGEMENT SYSTEMS

Donald D. Chamberlain

Computing Surveys, Vol 8, No 1. pp 43-66

Chamberlain has succeeded where many others have failed. The phrase "Relational Data-Base" has attained an aura of mystique about it, largely due to the new terminology introduced, and the inability on the part of early authors to simplify the concepts involved. The author, in this paper, has carefully defined the terminology in terms of common nomenclature, and has successfully conveyed the basic concepts of the relational model, and

its implementation, in an easy to understand and intuitive approach.

After reviewing some basic concepts, Chamberlain proceeds with a simple set of definitions for the terms used in the relational model. He follows with an intuitive presentation of the concepts of normalization succeeding in imparting the gist of the theory without using the precision of rigorous definitions. A survey of the user interfaces developed for the relational model is presented, including an evaluation of the various languages proposed, followed by a summary statement of the advantages of the relational approach to data base management. As presented by Chamberlain, relations have four primary advantages:

- o Simplicity
- o Data Independence
- o Symmetry
- o Strong Theoretical Foundation

Another area where the relational data model may be applicable, not fully explored by Chamberlain, is as a data base design methodology. Some researchers feel that applying the concepts of normalization to data base design, with conventional DBMS's can reap advantages related to data independence.

Chamberlain concludes with a good survey of current and past implementations of the relational model. Also of significance is an excellent bibliography based on a bibliography compiled by E.F. Codd.

Bernard K. Plagman

Review: "CODASYL Data-Base Management Systems", by R.W. Taylor and R.L. Frank, in Computing Surveys, Vol. 8, No. 1, March 1976.

"CODASYL Data-Base Management Systems" "...presents in tutorial fashion the concepts, notation, and data-base languages that were defined by the CODASYL Data Description Language and Programming Language Committees. Data structure diagram notation is explained, and sample data-base definition is developed along with several sample programs." [from the abstract] In doing this, the authors of this paper have undertaken a difficult task, and have carried it off fairly well. Since the CODASYL specifications incorporate facilities for tuning, resource allocation, error indication, etc., in addition to those for specification of data structures, the CODASYL languages can be difficult to explain in a simple way, and thus difficult to prepare concise tutorial material for.

The paper begins by presenting concepts of data structure design, oriented toward the use of data structure diagrams. This presentation describes certain of the features of the CODASYL approach in a particularly useful way. The identification of the CODASYL languages with the definition of data structure diagrams, the identification of CODASYL sets with specific access rules instead of a particular implementation technique, and the use of a matrix technique in describing set occurrences instead of the usual "rings", serve to provide useful insight into the properties of the structures being defined without confusing the logical structures with their implementation.

The paper next presents the CODASYL DDL description of a "Presidential" data base, a good example from the point of view of data structure description, as it contains both a many-to-many relationship and a cycle. The paper is thorough in describing the more important features of the DDL definition of this data base. Some readers may find difficulty due to the way the explanation of these features is tied to the example. For instance, during the description of the data base definition, the individual options of the SET SELECTION clause are discussed as they are used in the example. However, a general discussion of the overall concept of set selection does not appear until the "Advanced Features" section later in the paper. While the later discussion is quite good, this method of presenting language features seems in some cases to break up the discussion unnecessarily. A similar situation exists with respect to discussion of the "currency" concept in the paper. Readers who encounter this difficulty might choose to turn immediately to the "Advanced Features" section to get the more complete discussion of such features, rather than taking this section in its turn.

The paper then describes several sample programs which make use of the CODASYL COBOL Data Manipulation Language (DML). Due to the procedural nature of the DML (about the same as that of COBOL itself), this involves a fairly low-level discussion of individual program steps, and is thus not easy to read. However, this seems to be almost unavoidable.

In connection with the DML, a topic which I would like to have seen explicitly discussed in the paper, was the development of query languages for CODASYL data structures (this topic is discussed in the companion paper by Michaels, Mittman, and Carlson). While such languages are not currently in CODASYL-approved specifications, a number have been developed, and at least one of them (the Univac DMS-1100 query language) is available commercially. The topic of query languages seems an important part of the "model debate", as presentations using only the COBOL DML seem to reinforce the mistaken impression that CODASYL structures "inherently" involve record-by-record access. The reference list at the end of the paper includes several papers (M2, M3, and M11) on this general topic.

The "Advanced Features" section of the paper, in addition to providing further discussion of concepts used in the previous sections, contains some interesting discussion of "data independence". The authors make the important point that in the CODASYL languages the data administrator often has a choice as to whether a facility is used in a manner which will enhance data independence, or will make a program dependent on that facility in the interests of efficiency. The paper concludes with a useful section describing which features of the CODASYL specifications tend to be available in current implementations, and a fine reference list.

This paper is a worthwhile addition to the tutorial literature on the CODASYL data-base specifications. Combined with the material on the CODASYL specifications in texts such as those by Date or Martin, it should provide a good introduction to this class of data-base management systems.

Frank Manola
Naval Research Laboratory