FEATURES OF A CONCEPTUAL SCHEMA

BY

D. Tsichritzis*

Computer Science Department

University of Toronto

A B S T R A C T

Data Base Management Systems (DBMS's) can be divided in three main categories according to the basic data organization offered to their users. The three approaches are: hierarchical, network and relational. In future DBMS's all views may have to be offered to please a diverse population of users. Acknowledging this requirement a common facility has often been proposed under the name of a conceptual schema. The conceptual schema comprises a common denominator for all three DBMS approaches.

We propose a set of facilities appropriate for a conceptual schema. The basis of our proposal relates to the following requirements for a conceptual schema.

1) Enough flexibility to support different views of data
2) Adequate structuring and optimization tools
3) A framework for common integrity constraints for the data base
4) Absence of all purely physical properties of data

We propose four basic objects for the definition of structure among data. They correspond to four elementary facilities that any DBMS should have.

1) Record types. They define data pools, where data is stored
2) Selectors. They select appropriate data in the data pools
3) Links. They connect different data pools
4) Expressions. They combine the previous facilities

We propose a Data Definition Language which provides both intention and generation statements for record types, selectors, links and expressions. An intention statement specifies the constraints by which a particular object should abide. An intention statement may specify, for instance, that a particular data item is a key. It may constrain a selector to select records without null data items. It may specify that a certain link between two record

---

*On leave with the Technische Universitat Berlin during the spring semester 1975. This work was supported in part by the National Research Council of Canada. Author's address: Computer Science Department, University of Toronto, Toronto M5S 1A7 Canada.

types establishes a 1:N relationship. Depending on the time of its application, an intention statement is a constraint on the data base, or a hypothesis, or both. Intention statements enable us to specify common integrity constraints on the data base irrespective of the user's view.

A generation statement specifies some properties according to which a basic object can be generated. It can be a loading facility for a record type. It can be a complex boolean qualification for a selector. It can be a property establishing a relationship between two record types for a link. A generation statement can define, for instance, a fast access path to a set of records. It can define a new path which semantically makes sense, or it can serve as a reminder of a complicated relationship.

We give a set of features for specifying intention and generation statements. We also show with examples their uses. For instance, we specify with these statements the properties of a hierarchical data base. In this way other users can access the hierarchical data base using, for instance, relational commands. The intention statements will preserve both the integrity and the hierarchical properties of the data base.

We give constructs for both the definition and the application of intention and generation statements. Definition involves stating the nature of the statement, naming it and checking it syntactically and semantically. However, the statement is not enforced on the data base. Its application will result in executing the statement on the data base. In the case of an intention statement, it will impose an additional constraint on the data base. In the case of a generation statement, it will result in the construction of a pointer structure implementing, for instance, an access path.

We outline a set of features which enable the definition of different approaches and organizations of data. We can express, for instance, notions of a key, a DBTG set, a hierarchical definition tree, a relational join, etc. The properties expressed with these features are both inter-record and intra-record.

We feel that a system based on the proposed features can provide an environment for the implementation of all three views of data organization: hierarchical, network and relational. In addition, it will provide a general environment for specifying properties and integrity constraints for the data base.

References

1. Abrial, J.R., "Data Semantics" in "DATA BASE MANAGEMENT", eds. Climbie and Coffeman, North Holland, 1975.

2. Bachman, "The Programmer as Navigator", CACM 16, No. 11,(November 1973), pp. 653-689.

3. Bernstein, P. and Schmid, H., "A Multi-level Architecture for Relational Data Base Systems", Boston Conference on Very Large Data Bases.

4. Bernstein, P., Swenson, R. and Tsichritzis, D., "Functional Dependencies and Relations", Proceedings 1975 SIGMOD Conference, San Jose, California.

5. Boyce, R.F., Chamberlin, D.D., King, W.F. and Hammer, M.M., "Specifying Queries as Relational Expressions", Proceedings of ACM SIGPLAN/SIGIR Interface Meeting on Programming Language and Information Retrieval 1, Gaithersburg, Maryland, November 1973.

6. Bracchi, G., Paolini, P. and Pelagatti, G., "Data Independent Descriptions and the DDL Specifications" in "DATA BASE DESCRIPTION", eds. Douge and Nijssen, North Holland 1975.

7. CODASYL Data Base Task Group Report, April 1971.

8. Codd, E.F., "A Relational Model of Data for Large Shared Data Banks", CACM 13, No. 6 (June 1970), pp. 377-387.

9. Codd, E.F., "Further Normalization of the Data Base Relational Model", Courant Computer Sciences Symposia, Vol. 6, Data Base Systems Prentice-Hall New York, May 1971.

10. Codd, E.F., "Relational Completeness of Data Base Sublanguages", Courant Computer Science Symposia, Vol. 6, Data Base Systems, Prentice-Hall, New York 1971.

11. Codd, E.F., "Normalized Data Base Structure: A Brief Tutorial", Proc. 1971 ACM SIGFIDET Workshop on Data Description, Access and Control, San Diego, November 1971.

12. Codd, E.F., "A Data Base Sublanguage Founded on the Relational Calculus", Proc. 1971 ACM SIGFIDET Workshop on Data Description, Access and Control, San Diego, November 1971.

13 EDBS Users manual, CSRG Report No. 40, University of Toronto.

14. IMS, IBM Corp. IMS/360 General Information Manual form #GH201260.

15. Klebanoff, J., Lochovsky, F. and Tsichritzis, D., "Teaching Data Base Concepts using APL", Proceedings APL 75, Pisa, June 1975,

16. Nijssen, G.M., "Data Base Management Languages and Systems", Proceedings Symposium on Programmer Languages and Compiler Construction. Antwerb, April 1975.

17. Schmid, H. and Swenson, R., "Relational Semantics", Proceedings 1975 SIGMOD Conference, San Jose, California

18. Senko, M., "DIAM with FLORAL" in "DATA BASE DESCRIPTION", eds. Dough and Nijssen, North Holland, 1975.

19. Tsichritzis, D., "A Network Framework for Relation Implementation" in DATA BASE DESCRIPTION", eds. Dougue and Nijssen, North Holland, 1975.