

USING SAS WITH RAQL IN DATABASE COURSES

David Burrage Michael Gilman
McGill University School of Computer Science

ABSTRACT

The relational approach to data management is fundamental to university database courses. Unfortunately, very few relational systems are available at a price most universities could justify solely for teaching purposes.

RAQL is a relational query language which operates in a SAS environment providing a low cost relational teaching system. The language of RAQL is based on the syntax used in the popular textbook by C.J. Date - Introduction to Database Systems, Edition Two, Chapter 6. RAQL offers all the facilities in Date's book, and much more, and is implemented so as to run at reasonable expense on any teaching database.

Although no knowledge of SAS is necessary to use RAQL, it was found that the data management features of SAS reinforced and complemented the teaching of the relational approach. SAS used with RAQL at McGill University has completely transformed the undergraduate database courses.

INTRODUCTION

This article is not an argument for the relational approach for which a case has been strongly made by Codd(1), rather it is a case for the use of a database system **with a query language** on introductory database courses.

If a choice is available, a database instructor must decide whether to introduce data manipulation using a query language or using a language, based on subroutine CALLs, imbedded in a host language. Having used both methods, we found that the improvement in the course through the use of the query language was striking.

USING A DATA MANIPULATION LANGUAGE WITHIN A HOST LANGUAGE

Our first experiences of database teaching involved the use of an educational CODASYL type database with FORTRAN as host language, and also a commercial hierarchical database system with COBOL.

The problems associated with the use of these systems were:-

1. the students did not feel comfortable with these systems
2. the manuals were expensive to buy and not easily available to borrow
3. debugging was difficult

4. much classroom explanation was necessary before a student could begin assignments
5. students were too concerned with problems in the host language and job control language
6. the JCL required was complex and beyond the experience of most students
7. the creation by the students of their own schemas and databases led to many unforeseen problems

In general, the students were 'turned off' by their first database experiences.

USING A QUERY LANGUAGE

RAQL was the query language used. This new language, based on relational algebra, brings a full relational capability to SAS. With RAQL, the manipulation of the database, the SAS data sets, becomes more precise and systematic. For details on RAQL and the use of RAQL in the SAS data management environment see (2),(3) and (4). The syntax of RAQL follows that of Date(5).

It was found that the problems associated with the use of host languages for database processing disappeared. In fact, apart from instruction on the relational operators, the RAQL User Manual was all the students required in order to begin solving retrieval problems. Such is the simplicity of RAQL syntax and the job control statements. The RAQL translator program checks for over fifty possible error conditions in the queries, an essential aid to debugging.

OTHER BENEFITS

The fact that a query language is available for practical work allows an instructor to discuss the general subject of query languages in a more meaningful way. A survey of other query languages is instructive. The number of query languages available is increasing, so it is useful for students to be aware of other methods, features and syntaxes.

RAQL is a stand alone system because a user can create, manipulate, update, copy, purge and retrieve from SAS data sets using RAQL statements only. However, a user can also easily interleave SAS statements with the RAQL statements. An instructor familiar with SAS can introduce many SAS features which enhance RAQL. The SAS procedures CONTENTS and DATASETS are among those which are relevant to database courses. Students are particularly delighted to see how easily graphics are obtained using PLOT and CHART.

The fact that SAS data sets are easy to create means that students can construct their own relational databases.

Database design is a natural subject to introduce at this point. On the course, projects were assigned in which students created, updated and retrieved from their own data bases. Many students felt that this was one of the most useful parts of the course.

Another topic which naturally arises through the use of RAQL is the consideration of the efficiency of the solution of the query. One rough measure of the 'cost' of the solution is the number of sorts required. All of the commonly used relational operators require sorts with the exception of SELECT and TIMES. RAQL possesses features such as SORTED ON and -PROJECT which can be used to reduce 'cost'. The problem of the optimization of the solution to the query can be raised at this point.

CONCLUSION

We suggest that students should be introduced to database processing through query languages. Not only is the introduction less painful than using other methods but the use of the query language itself provides a natural means of introducing students to other database problems and concepts.

FOR FURTHER INFORMATION

For further information, please contact Michael Gilman, McGill University, School of Computer Science, Burnside Hall, 805 Sherbrooke St. West, H3A 2K6, Tel: (514) 392-4566.

REFERENCES

- (1) CODD, E.F. Relational Database: A Practical Foundation For Productivity, Comm of the ACM, Feb. 82, Vol.25, No.2.
- (2) BURRAGE D. and GILMAN M. "RAQL - An Evolution in SAS Data Management", SUGI Conference Proceedings 1983.
- (3) GILMAN M. and BURRAGE D. RAQL User Manual, McGill University, Montreal, Canada 1982.
- (4) VAN ROSSUM T. "Construction and Management of a Medical Research Database Using RAQL and SAS." SUGI Conference Proceedings 1983.
- (5) DATE C.J. Introduction to Database Systems, 1977, Second Edition, Addison Wesley