Some Reviews of Current Literature

J. R. Swenson
University of Toronto
Dept. of Computer Science

LUM, V.Y.
General performance analysis of key-to-address transformation methods using an abstract file concept.
Comm ACM 16, 10 (Oct. 1973) 603-612

This paper explains a systematic methodology which can be used to evaluate the performance of key-to-address transformation (hashing) techniques.

The hashing problem is as follows. Let S be an abstract set, the key space, and F a subset of S, the file space. F is generally much smaller than S in application. For example, if S is the set of all last "names" of 10 or fewer English characters and F is the set of those names in Toronto, then $S \approx 26^{10}$ and $F < 2 \cdot 10^6$. Now $S \geq 10^{13}$ $\therefore$ $S/F \geq 10^7$. Let D be another set, the address space. Customarily, D is of the same order of magnitude as F. Then a hashing function, T, is a map from S to D. T performs well relative to F if T maps F uniformly on D, i.e. if the cardinalities of the inverse image in F under T of elements of D are almost the same.

The problem is that we do not usually know sufficient characteristics of F to enable us to define T perfectly. This paper proposes a methodology to determine whether, given some characteristics of F, then some class of T's can be expected to work well on the average.

The discussion of the application of the theory is not as well written in comparison with the rest of the paper but none-the-less offers some indication of the manner in which the methodology can be applied. I am confused by the introduction of definition 7 and would have liked to have seen a better discussion of its importance.

The paper is carefully written and relatively easy to read. The simple conceptual explanation of what 'hashing' really is should be memorised by all subsequent writers on this subject. The bibliography is very brief and only mentions immediately useful references.

* * *

1. BRACCHI, G., A. FEDELI and P. PAOLINI
   A Language for a relational data base management system.
   Proc. of the 6th Annual Princeton Conf. on Information Sciences and Systems 1972

2. BRACCHI, G., A. FEDELI and P. PAOLINI
   A relational data base management system.
   Proc. of ACM 72 Annual Conf. 1972

3. BRACCHI, G. and P. PAOLINI
   Architecture of an on-line information management system.
   ONLINE72 Conference Proc. 1972

These three papers are highly repetitious. Clearly they have all been prepared from one report by cutting and pasting. (2) gives the most general survey while (1) and (3) are somewhat more detailed on complementary topics.

(2) discusses the general organization of MORIS, a DBMS based on the relational model of Codd, and illustrates COLARD, a language which enables a user to deal with the MORIS data base.

A brief introduction to various DBMS's is given along with the idea of relations. The authors desire to permit users to deal with hierarchical structures, but the system will only deal with relations in First [Codd] normal form. The overall architecture of MORIS is given in terms of design criteria and a schematic description of an organization which is claimed to fulfill the design. The language which allows operation on the data base is briefly illustrated.

A most curious aspect of the design is that users are expected to provide information as to the "most suitable physical representation" for the data structure being stored. Considering the early discussion on the desire for independence of data bases from particular user application programs, I question this feature of the design. No default is mentioned in case one wants the system to choose. However, the design does recognize that the internal (physical) storage structure should be allowed to be different for two data bases whose logical structure is the same without impacting any use of the logical structures.

In (1) more precise language is developed in order to help in defining the operations which may be performed on (non-normalized) relations. As well, the language which defines relations (the DDL) and which invokes operations on relations (the DML) is informally discussed. The language is self contained and more procedural.

In (3) the MORIS architecture is discussed in more detail than in (2).

These papers are easy to read and do raise some interesting thoughts on implementing relational models. A single technical report would have saved a lot of duplication though. Either

papers (1) and (3) should be read, or (2) by itself.

For collateral reading to these three papers, refer to Codd, Date and Hopewell, and Fillat and Kraning, which are referenced in these papers.

These papers are curious. No clue is given as to the state of implementation of the system or the size of the data base which the system is expected to handle. In Codd's papers we are presented with the theoretical aspects of the relational model and thus don't expect performance predictions. I would expect such predictions in these papers.

No data is given on the meaning by which queries are analyzed except for one remark which suggests that logical statements are placed in disjunctive normal form (3, p. 204). Another remark (1, p. 91) suggests that the analysis could be very dependent on the way in which the user initially presents the query.

One reason given for using the relational model is that this permits users to develop more complex data structures in a methodic and efficient way. Except for a brief reference to heirarchies, this ability is not demonstrated. Occasional reference is also made to the fast and uniform retrieval time possible in a relationally organized data base. This also is not substantiated.

JR Swenson