THE HYPERGRAPH-BASED DATA STRUCTURE :

A NEW APPROACH TO DATA BASE MODELLING AND APPLICATION

François BOUILLÉ

Institut de Programmation
Université P. et M. CURIE
4, Place Jussieu
75230 Paris cédex 05
FRANCE

ABSTRACT

The HBDS model is a type of data structure essentially built upon the set theory and the hypergraph concept. Four abstract data types are defined, respectively named : class, object, attribute and link. Its architecture is composed of an arborescence, an hypergraph, a forest and two multigraphs. In the inner part of the system, links are transformed into hidden classes and objects, allowing to consider them as hierarchized. Extensions of the model are proposed in order to reduce the number of links: hyperclass, hyperlink, multilink and hypermultilink. One of the main features of the HBDS is the presence of a skeleton structure which makes the structure immediately understandable, without depending upon the number of the components.

The HBDS may be used in data base modelling and design and is able to ensure almost all tasks and properties like protection, simultaneous shareability and distribution, with which special hidden structures are associated.

The HBDS may be applied to business data with several advantages compared with other models : trees, nets, boxes, which appear as subsets of it.
But, its main possible applications essentially concern the very large scientific data bases with complex-structured data.

The HBDS allows a new scientific approach of data base design and applications by giving us a very general model of data structure which may be used everywhere.

KEY-WORDS :
Abstract data type, arborescence, attribute, class, concentrical abstraction, data base, data structure, deadlock prevention, distribution-structure, graph theory, hidden component, hyper/class /graph /link, Hypergraph-Based Data Structure, object, protection-structure, skeleton-structure, shareability-structure, Shareable Data Structure Unit.

H. J. Schneider (ed.), *GI — 7. Jahrestagung*

INTRODUCTION : Towards an universal data structure model ?

There are several classical models of data structure, like trees (more exactly arborescences = rooted trees), nets, boxes.... Is one of them better than the others ? It is generally said that it depends upon the problems to solve. Thus, it would be better to name them a problem structure than a data structure. In fact, the problems, to be solved, must sometimes use some aspects of the data structure which are represented by trees, sometimes by nets, and sometimes by boxes. Finally, these three representations are nothing but some subsets of a complete data structure, a synthesis which is generally not taken into account.
Then, there are two possible attitudes : trying to realize this synthesis, or building the most general model on mathematical basis without considering the other models presently proposed. We have choosen the second case and we shall verify that the synthesis nevertheless is effectively obtained.

Some main principles have been adopted :

- a data structure only concerns the data, not the problems ; if correctly built, it can be used to solve almost all problems.
- a data structure is not conceived for a particular programming language, but a programming language is choosen to process the structure because it is well adapted, and not the contrary.
- a data structure must not be confused with a storage structure, as it is often the case ; files, records, etc. have not their place in the terminology here.

This paper presents the result, which is the Hypergraph-Based Data Structure, presently little known ; as yet, it has been used for complex scientific data structures, where other models could not be applied. It is also the kernel of a new kind of data base which is protected, simultaneously shareable, portable and distributed. Firstly, we sum up the HBDS ; then, we describe the approach to data base design thus allowed ; finally, we show some applications in sciences and business.

THE HYPERGRAPH-BASED DATA STRUCTURE : "HBDS"

According to the set theory, a set is composed of elements which have properties and may present relations or not (BOUR 39). Though a property is nothing but a particular relation, we keep this distinction. Using the abstract data type concept (LISK 74), we consider four types respectively named : class, object, attribute and relation. They must be associated with distinctive graphical concepts. Graph theory is generally used, but cannot correctly represent the difference between set and element. So, we include here the hypergraph concept (BERG 70), as the main component of the structure.