ARE DATA TYPES UNIVERSAL MODELLING CONCEPTS FOR DATA BASE SYSTEMS?

HERBERT WEBER

UNIVERSITY BREMEN, GERMANY

Modelling encompasses two intellectual tasks: (1) the perception of some reality and (2) the representation of the perceived reality in accordance to a given model (obviously is a representation once again some kind of reality and hence subject to another representation in accordance to another model or notation). Because of human beings limited capabilities to perceive and completely and correctly represent reality any representation is an abstraction of the reality.

A representation in accordance to a given model is of course constrained by that model. Any facet of reality not representable with the model cannot be represented at all. Moreover any model constrains at the very same time the perception capabilities as well. Experience verifies that knowledge of a model tends to force the modeller to model according to this model even though it may not result in the most accurate representation. These two facts speak against an universal model but makes for a variety of models for different purposes. It is common practice for the representation of the reality in data bases to apply different models for the conceptual description of data bases or to apply different kinds of models for the physical description of data bases, etc.

Obviously a precise definition of a model is a prerequisite for accurate modelling. Even though models in use may have been properly defined the modelling process itself may be unsystematic and vague. It may - and probably will - be considered by many people as an art. It is the thesis of this remark that modelling may be made systematic and that the notion of type may aide in the modelling as a universal modelling concept independent of the model in use.

This notion of type which seems to be suitable as a universal modelling concept considers types as classes of objects together with collections of operations on objects of the respective class. This notion encompasses as a second ingredient a notion for the composition of types from more primitive types and consequently the notions of object and operation compositions. This notion of types supports in any modelling since it enforces

- the classification of things in the observed reality
- the analysis of the mutability of things in the observed reality
- the analysis of the structure of things in the observed reality independent of a particular model.

For these characteristics the type notion supports modelling in the following ways:

(1) The representation in accordance to a data model: The data model itself (e.g. the relational data model) may be considered as a parameterized type (all relations in a relational data base may then be considered to be of type "relation" /Web 76, Web 78/)

(2) The representation of data structures: structuring models for data like aggregation or generalization may be represented as type compositions /SSW 80/

(3) The representation of dynamic properties of data: The operations and some applicability constraints associated with a type may be considered the representation of the

dynamic properties of data /Web 79, SSW 80/

The type notion also provides means for a mathematical formal definition of classes of objects, their mutability or invarient properties and of the structure of objects. This formal definition may be based on algebraic specifications of data types introduced in /Zi 74, GTW 76/ and can then be applied in the verification of the consistency and correctness of data bases /EKW 78, EKW 79/

Based on these characteristics the type concept supports three types of abstractions in the modelling of data base system: (1) it supports the abstraction in the modelling of things in reality in terms of a given model and (2) it supports the abstraction in the modelling of a given representation by another representation in terms of another model and, finally, (3) it supports the abstraction in the modelling of a given representation by a mathematically formal meta notation.

References

/EKW 78/ Ehrig, H., Kreowski, H.-J., Weber, H.: Algebraic Specification Schemes for Data Base Systems, Proc. 4th VLDB Conf, Berlin 1978, 427-440, earlier version in Technical Report HMI-B266, HMI Berlin, 1978

/EKW 79/ Ehrig, H., Kreowski, H.-J., Weber, H.: New Aspects of Algebraic Specification Schemes for Data Base Systems, Springer Lecture Notes, 1980

/GTW 76/ Goguen, J.A., Thatcher, J.W. Wagner, E.G.: "An Initial Algebra Approach to the Specification, Correctness, and Implementation of Abstract Data Types, IBM Research Report RC 6487 (1976)

/SSW 80/ Scheuermann, P., Schiffner, G., Weber, H.: Abstraction Capabilities and Invariant Properties Modelling within the Entity-Relationship Approach, in

Chen, P.P. (ed) Entity-Relationship-Approach to Systems Analysis and Design, North Holland, 1980

/Web 76/ Weber, H.: The D-Graph Model of Large Shared Data Bases: A Representation of Integrity Constraints and Views on Abstract Data Types, IBM Research Report RJ 1875 (27024) 1976

/Web 78/ Weber, H.: A software Engineering View of Data Base Systems. Proceedings 4th VLDB conference, 1978

/Web 79/ Weber, H.: Modularity in Data Base System Design, in

Weber, H., Wasserman, A.I.: Issues in Data Base Management, North Holland, 1979

/Zil 74/ Zilles, S.N.: Algebraic Specification of Data Types, Proj. MAC Report 11, MIT, Cambridge, Mass (1974) 25-28