TOWARDS CONSTRUCTIVE AXIOMATIC SPECIFICATIONS

C.S. dos Santos*, A.L. Furtado+
J.M.V. de Castilho*, S.E.R. de Carvalho+

*Universidade Federal do Rio Grande do Sul
+Pontifícia Universidade Católica do Rio de Janeiro
Brasil

The main goal of our efforts is the specification of data bases whose structure and behaviour are restricted by semantic integrity constraints. Research proceeds along the following stages:

1. Choose a data model;
2. Establish a taxonomy of integrity constraints;
3. Specify the data model formally;
4. Develop a methodology for specifying conceptual schemas for different applications, based on the data model;
5. Provide a conceptual characterization of query and update operations.

In the sequel we give a brief account of our conclusions up to the time of writing.

As a data model, we adopted the entity-relationship model [CHE], which treats primitive data objects in an appropriately intuitive way, extended with three constructors: sum, product (analogous to generalization and aggregation [SMI]) and correspondence [SA1]. Correspondences permit the construction of derived types whose instances are cosets of entities indexed by attribute-values or by related entities; they generalize the notions of inversions, CODASYL sets and partitioned relations [FKE].

Figure 1 below shows an academic data base with students, courses and teachers as primitive entity-sets and student-records and course-pre-requisites as derived entity-sets obtained through the correspondence constructor.

A taxonomy of integrity constraints is useful during the phase of conceptual schema design as a checklist and part of a framework [HAM], calling attention to relevant characteristics of constraints.

To determine the taxonomy, we considered several aspects, such as origin, substance (i.e. intrinsic characteristics), form of specification and mode of application. Integrity constraints originate in the enterprise or in its environment (natural requirements, regulations), or they can be imposed by the data model chosen or by implementation restrictions. Two of the many categorizations related to substance are state-span (i.e. whether

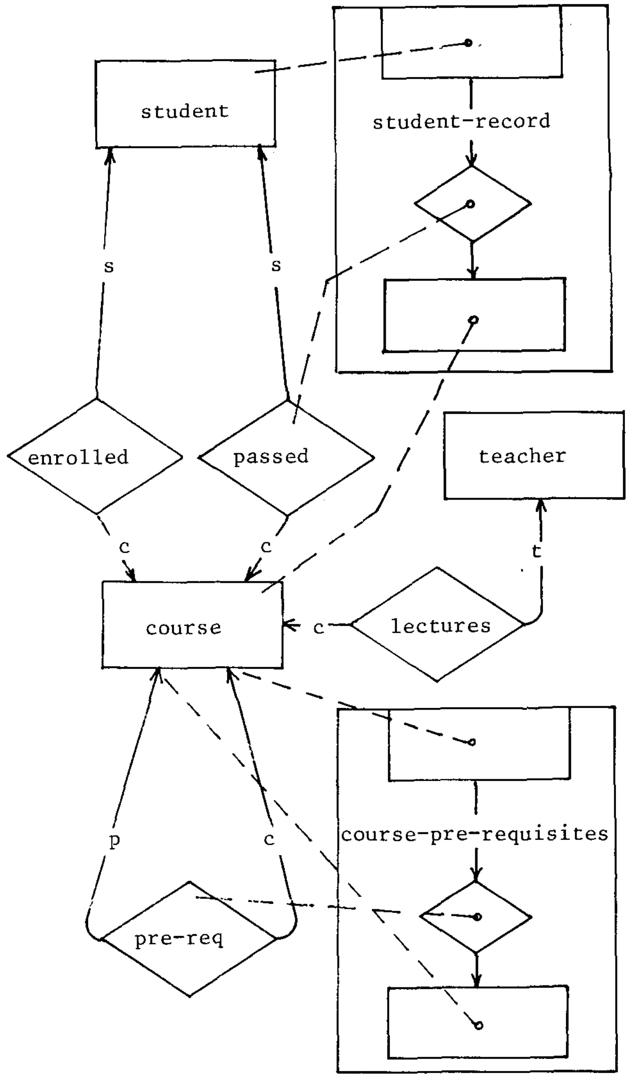

Figure 1 - Conceptual schema in the extended model.

the constraint refers to just one or to a succession of states, or to circumstances - such as time - associated with the occurrence of the transition between states), and composition of the constrained objects (primitive or constructed, the latter being indirectly affected via certain inheritance rules). As to form of specification, constraints need not always be explicitly declared in the schema specification; they may be implicit in the data model adopted or they may be a logical consequence of other constraints. Among the categories for mode of application are activation (manual or automatic) and inspection strategy (direct or indirect, using for example stored aggregate values).

The model is specified in a many-sorted first-order predicate calculus based on the state space approach [NIL] and on axiomatic specification techniques [GUT,PEQ,BRO].

The data model specification takes the form of axioms expressing constraints at the schema level and at the contents (or instances) level. An example of a constraint at the schema level is the requirement that, to declare correspondences indexed by attribute-value, the index must come from one of the declared attributes of the indexed entity-set. In general, constraints at the schema level express consistency requirements among declarations.

Constraints at the contents level refer to pre-requisites and effects of operations and to co-requisites expressing interdependencies among predicates. For example, for linking entities via a given relationship in the indicated roles, we have the pre-requisite that they must not already be so related in the current state; the effects are that they become related after the operation (intended effect) and that counters associated with the number of links in the given relationship-set and with the number of such links incident to each of the entities will be incremented (triggered effect). An example of co-requisite is that entities can be related only at states where they all exist in the appropriate entity-sets.

The specification of the conceptual schema of a given application involves declaring the particular primitive and constructed data objects and, usually, additional integrity constraints which are added to - and cannot conflict with - those of the data model. The incorporation of such constraints may be done through additional pre-requisites, triggered effects and co-requisites. In the example data base of figure 1 we might have a constraint like:

> a student cannot enroll in a course that he has already passed (and hence already appears in his record).

This constraint can be implemented as a co-requisite of the enrollment relationship. Note that it does not contradict the already mentioned general requirements for relationships.

We are investigating how constraints formulated statically can be enforced dynamically by adding appropriate pre-requisites and triggered effects to update operations [FSA]. Data bases whose manipulation is disciplined in this way are correct (that is, consistent with the specified integrity constraints) by construction.

In fact, using the axiom system developed for a particular application, we were able to demonstrate how update transactions preserving the constraints may be synthesized through a theorem-proving process [SA2]. We also presented a unified treatment of queries and updates, where queries were shown to be executable by analysing the generated current state through the same theorem-proving techniques [MAI].

It has been shown that algebraic definitions can lead to the symbolic manipulation of data types [GUT,GOG]. We have recently translated an axiom system, corresponding to a reasonably powerful submodel of our data model, into a set of functions for manipulating data bases in canonical term algebra representation (as in [PEQ]). This translation was extended with another set of functions corresponding to the axioms of a specific application implemented (in the precise abstract data type sense [GUT]) in terms of the model. The methodology for effecting this translation (and its further translation into SNOBOL functions) is quite straightforward and provides a useful tool for the early testing of specifications, and for the study by simulation of their behavioural characteristics. We also plan to apply these ideas to the study of ANSI/SPARC external schemas.

To summarize, our research is based on the belief that it is convenient to use formal tools to represent conceptual models of real world data organizations. Such representations are useful, for example, to rigorously define the semantics of a given model, and to compare models. Moreover, once the fundamental model is represented, different applications of that model to different real life situations can be conveniently accomodated within the same representation.

References

[BRO] Brodie, M.L. - "Axiomatic definitions of data model semantics" - IFSM T.R. 41 - Univ. Maryland (1979).

[CHE] Chen, P. - "The entity-relationship model - towards a unified view of data" - ACM-TODS- Vol. 1, No. 1 (1976).

[FKE] Furtado, A.L. and Kerschberg, L. - "An algebra of quotient relations" - Proc. SIGMOD (1977).

[FSA] Furtado, A.L., Santos, C.S. and Castilho, J.M.V. - "Static and dynamic specification of constraints" - Information Systems 6, 1 (1981).

[GOG] Goguen, J.A. and Tardo, J.J. - "An introduction to OBJ: a language for writing and testing formal algebraic program specifications - Proc. Specifications of reliable software (1979).

[GUT] Guttag, J.V., Horowitz, E. and Musser, D.R. - "The design of data type specifications" - Proc. Conference on Software Engineering (1976).

[HAM] Hammer, M.M. and McLeod, D. - "A framework for data base semantic integrity" - Proc. Conference on Software Engineering (1976).

[MAI] Maibaum, T.S.E., Santos, C.S. and Furtado, A.L. - "A uniform logical treatment of queries and updates" - T.R. db018001 - PUC/RJ (1980).

[NIL] Nilsson, N.J. - "Problem solving methods in artificial intelligence" - McGraw-Hill (1971).

[PEQ] Pequeno, T.H.C. and Veloso, P.A.S. - "Do not write more axioms than you have to" - Proc. International Computer Symposium (1978).

[SA1] Santos, C.S., Neuhold, E.J. and Furtado, A.L. - "A data type approach to the entity-relationship model" - Proc. Int. Conf. on the entity-relationship approach to systems analysis and design - P. Chen (ed.) (1979).

[SA2] Santos, C.S. and Furtado, A.L. - "Synthesis of update transactions" - T.R. db107901 - PUC/RJ (1979).

[SMI] Smith, J.M. and Smith, D.C.P. - "Database abstractions: aggregation and generalization" - ACM-TODS - Vol. 2, No. 2 (1979).