Association for Information Systems
AIS Electronic Library (AISeL)

AMCIS 1997 Proceedings

Americas Conference on Information Systems (AMCIS)

8-15-1997

Modeling Relationships Using The Relational And Object-Oriented Data Models

Atish P. Sinha
University of Dayton, sinha@udayton.edu

Iris Vessey
Indiana University, ivessey@indiana.edu

Modeling Relationships Using The Relational And Object-Oriented Data Models

Atish P. Sinha,
University of Dayton, Dayton, OH 45469-2130;
Ph: (937) 229-2247; email: sinha@udayton.edu

Iris Vessey,
Indiana University, Bloomington, IN 47405;
Ph: (812) 855-3485; email: ivessey@indiana.edu

Abstract

We compare the performance of naive data modelers in modeling association, generalization, and aggregation relationships with the relational and object-oriented data models. We first develop research hypotheses based on the properties of expressiveness, minimality, and unique semantic interpretation to analyze the effectiveness of the two models. We then test our hypotheses in an experiment with 22 naive modelers. The findings of our study support the notion that, to be effective, a data model should satisfy these three properties.

Introduction

Considerable research has been conducted into the relative effectiveness of different data modeling formalisms (e.g., Batra et al. 1990). Such studies have most often compared the relational model, which is a logical data model, to a semantic data model such as the entity-relationship (E-R) model. The only study to date, with which we are familiar, that examined comparable representations is that of Kim and March (1995), which compared the E-R model and Nijssen's information analysis methodology (NIAM), both of which are semantic data models.

This study compares performance with two logical (or implementation) data models: the relational data model (RDM), currently the most popular logical data model, and the object-oriented data model (OODM), which is becoming increasingly popular because of its ability to model complex real-world objects and semantic relationships. Specifically, we compare the ability of naive data modelers to capture the major types of relationships that are needed in today's world of data modeling using the RDM and OODM.

Research Hypotheses

We apply the properties of *expressiveness, minimality,* and *unique semantic interpretation* to analyze the effectiveness of the RDM and the OODM for logical design. A data model should be *expressive* enough to distinguish among different types of data, relationships, and constraints; it should be *minimal* in that every concept in the model should have a distinct meaning with respect to every other concept; and it should have a *unique interpretation*, implying that each modeling construct can be interpreted in only one possible way (Navathe 1992). Our analysis focuses on the effectiveness of the two models in representing the three types of relationships that are essential to capture the semantics of a real-world application, viz., association, generalization, and aggregation. Figure 1 presents the RDM schema, while Figure 2 presents the OODM schema, based on ORION (Kim 1990), for a University database.

Association. We address association in the form of binary 1:N and M:N relationships. In an RDM schema, a 1:N relationship is represented unequivocally by including the primary key of the "1-side" relation as a foreign key in the "N-side" relation. No other concept in the model can be used and, therefore, the RDM is minimal with respect to representing 1:N relationships. In the OODM, the 1:N relationship is modeled by having an attribute of the class on the N-side ("advisor" of PHD_STUDENT) reference the class on the 1-side (FACULTY). Because only domain references can be used to represent the relationship, the OODM is minimal. However, because the relationship is not highlighted in both directions (what we refer to as "construct asymmetry"), we expect some modelers will represent it in both directions, by adding a multi-valued attribute called "advisees" within FACULTY. But that would result in violations of relationship integrity and cardinality constraints (Cattell 1994), because the OODM adopted for this study does not employ *inverse references*. We therefore state the following hypothesis:

H1: More modelers represent 1:N associations correctly using the RDM than using the OODM.

The RDM is also minimal for representing an M:N association because it can be represented only by breaking the M:N relationship into two 1:N relationships, one from each participating relation, to form a third relation. The OODM is also minimal for representing M:N relationships because the association is specified only using references. However, there are two possible ways of correctly representing those references: "courses" could be represented as a multi-valued attribute within FACULTY, or "faculty" could be represented as a multi-valued attribute within COURSE (binary relationships each in one direction). As with the 1:N association, because of construct asymmetry, some users might represent the M:N association in both directions, thus violating relationship integrity constraints. We therefore state the following hypothesis:

H2: More modelers represent M:N associations correctly using the RDM than using the OODM.

Generalization. The RDM is not expressive with respect to generalization. Generalization can be captured indirectly by creating separate relations for a superclass and its subclasses, in which case the primary key in each subclass relation becomes a foreign key referencing the superclass relation. Using the foreign key construct to represent both generalization and association might confuse modelers because it does not have a unique interpretation. We therefore expect that some RDM users will not represent the generalization relationship fully, i.e., they will not show all the superclass-subclass relationships. The OODM, on the other hand, is expressive because it allows explicit representation of generalization relationships through the **:superclasses** keyword. We state the following hypothesis:

H3: More modelers fully represent generalization relationships using the OODM than using the RDM.

Aggregation. There is no construct in the RDM diagram to represent aggregation, resulting in lack of expressiveness. However, aggregation can be represented indirectly by disaggregating the wholes into their parts (thus disambiguating the whole-part semantics) using regular association relationships. But then the resulting schema does not have a unique interpretation. The OODM, on the other hand, is expressive because it possesses an explicit construct to represent aggregation. We therefore state the following hypothesis:

H4: More modelers fully represent aggregation relationships using the OODM than using the RDM.

Methodology

The participants were 22 MBA students enrolled in a course on database management, where they received instruction and completed exercises in each of the data models. They were assigned randomly and equally to one of the two data models, and instructed to use the model to design a schema for a University problem (see Figs. 1 and 2). The independent variables are "data model" (RDM or OODM) and "relationship modeled" (1:N, M:N, generalization, aggregation), and the dependent variable is the number of modelers who represented a relationship correctly/fully.

Results and Discussion

Fisher's exact test (see Fleiss 1981) was used to determine if there was any association between the data model used and the number of modelers who correctly/fully specified a relationship. We report one-tailed p values for the tests.

We hypothesized that more modelers would correctly specify the two association relationships using the RDM than using the OODM. Six RDM subjects (55%) compared with only two OODM subjects (18%) correctly specified the 1:N association, and Hypothesis H1 was weakly supported ($p = .092$). The corresponding numbers for the M:N association were eight (73%) and three (27%), respectively, supporting Hypothesis H2 ($p = .043$).

We hypothesized that more modelers would fully specify the generalization relationships using the OODM than using the RDM. We tested three generalization hierarchies separately: STAFF, FACULTY, and GA. Seven OODM users (64%) fully specified the STAFF generalization, and nine the FACULTY generalization (82%), compared with zero and two users (18%) for the RDM. Hypothesis H3 is supported for the STAFF generalization ($p = .002$), as well as for the FACULTY generalization ($p = .004$). The GA generalization proved to be problematical using both models. Only two from each group showed both parents, EMPLOYEE and PHD_STUDENT. Therefore, there is no support for Hypothesis H3 with respect to multiple-parent generalization. Overall, more modelers fully specified generalization relationships using the OODM than using the RDM ($p = .000$), supporting Hypothesis 3.

Finally, Hypothesis H4, that more modelers would fully specify aggregation relationships using the OODM than using the RDM, was supported ($p = .018$). None of the RDM users fully disaggregated all the components, compared with five OODM users (55%).

The findings of our study support the notion that, to be effective, a data model should satisfy the properties of expressiveness, minimality, and unique semantic interpretation. As predicted, the OODM is superior to the RDM for modeling generalization and aggregation relationships. Also, it appears that the equivocality caused by the asymmetry in representing association relationships using the OODM resulted in inferior performance compared to the RDM. The current set of ODMG (object database management group) standards (see Cattell 1996), which requires that a relationship is specified in both directions using *inverse references*, addresses the asymmetry problem and eliminates the equivocality posed to modelers. Future research is needed to empirically investigate the effects of such enhancements to the OODM, which, industry analysts predict, will usher in the next generation of database systems.

References

Batra, D., Hoffer, J. A., and Bostrom, R. P., "Comparing Representations with Relational and EER Models," Communications of the ACM, 33(2) (February 1990), 126-139.

Cattell, R. G. G., Object Data Management: Object-Oriented and Extended Relational Database Systems, Addison-Wesley, Reading, MA, 1994.

Cattell, R. G. G. (Ed.), Object Database Standard: ODMG-93, Release 1.2, Morgan Kaufmann, San Francisco, CA, 1996.

Fleiss, J. L., Statistical Methods for Rates and Proportions, 2nd. Ed., Wiley, 1981.

Kim, W., Introduction to Object-Oriented Databases. Cambridge, Mass: The MIT Press, 1990.

Kim, Y. and March, S., "Comparing Data Modeling Formalisms," Communications of the ACM, 38 (6) (June 1995), 103-115.

Navathe, S.B., "Evolution of Data Modeling for Databases," Communications of the ACM, 35 (9) (1992), 112-123.

Employee (Emp_Id, Name, Age, Salary)

Staff (Emp_Id)
Fk Emp_Id --> Employee
Faculty (Emp_Id)
Fk Emp_Id --> Employee
Course (Crse_Code, Crse_Title)
Assignment (Emp_Id, Crse_Code)
Fk Emp_Id --> Faculty
Fk Crse_Code --> Course
GA(Emp_Id, Name, Stipend, Hrs_Per_Week)
Fk Emp_Id --> Employee
Fk Name --> Phd_Student
Secretary (Emp_Id, Type_Speed)
Fk Emp_Id --> Staff
Administrator (Emp_Id)
Fk Emp_Id --> Staff
Clerk (Emp_Id)
Fk Emp_Id --> Staff
Assistant(Emp_Id, Num_Yrs_Tenure)
Fk Emp_Id --> Faculty
Associate (Emp_Id, Promn_Date)
Fk Emp_Id --> Faculty
Full (Emp_Id, Promn_Date)
Fk Emp_Id --> Faculty
Phd_Student (Name, Advisor)
Fk Advisor --> Faculty
Stu_Perf (Name)
Fk Name --> Phd_Student
Crse_Perf (Name, Num_Crses, Gpa)
Fk Name --> Stu_Perf
Exam_Perf (Name, Written, Oral)
Fk Name --> Stu_Perf
Thesis_Perf (Name, Title, Prop_Date, Def_Date)
Fk Name --> Stu_Perf

Fig. 1. RDM Schema

(**make-class** EMPLOYEE **:attributes** ((emp_id **:domain** string) (name **:domain** string) (age **:domain** integer) (salary **:domain** real)))	(**make-class** ASSOCIATE **:superclasses** (FACULTY) **:attributes** ((promn_date **:domain** date))) (**make-class** FULL

(**make-class** STAFF
:superclasses (EMPLOYEE))

(**make-class** FACULTY
:superclasses (EMPLOYEE)
:attributes
((courses **:domain** COURSE
multivalued)))

(**make-class** COURSE
:attributes
((crse_code **:domain** string)
(crse_title **:domain** string)))

(**make-class** GA
:superclasses (EMPLOYEE,
PHD_STUDENT)
:attributes
((stipend **:domain** real)
(hrs_per_week **:domain**
integer)))

(**make-class** SECRETARY
:superclasses (STAFF)
:attributes
((type_speed **:domain**
integer)))

(**make-class**
ADMINISTRATOR
:superclasses (STAFF))
(**make-class** CLERK
:superclasses (STAFF))

(**make-class** ASSISTANT
:superclasses (FACULTY)
:attributes
((num_yrs_tenure **:domain**
integer)))

:superclasses (FACULTY)
:attributes
((promn_date **:domain** date)))

(**make-class** PHD_STUDENT
:attributes
((name **:domain** string)
(perf **:domain** STU_PERF)
(advisor **:domain**
FACULTY)))

(**make-class** STU_PERF
:attributes
((crse_perf **:domain**
CRSE_PERF)
(exam_perf **:domain**
EXAM_PERF)
(thesis_perf :**domain**
THESIS_PERF)))

(**make-class** CRSE_PERF
:attributes
((num_crses **:domain** integer)
(gpa **:domain** real)))

(**make-class** EXAM_PERF
:attributes
((written **:domain** char)
(oral **:domain** char)))

(**make-class** THESIS_PERF
:attributes
((title **:domain** string)
(prop_date **:domain** date)
(def_date **:domain** date)))

Fig. 2. OODM Schema