Association for Information Systems
AIS Electronic Library (AISeL)

AMCIS 2001 Proceedings

Americas Conference on Information Systems (AMCIS)

December 2001

The Impact of Domains of Knowledge on the Quality of Database Conceptual Models

Alain Villeneuve
Université de Sherbrooke

Diane Strong
Worcester Polytechnic Institute

THE IMPACT OF DOMAINS OF KNOWLEDGE ON THE QUALITY OF DATABASE CONCEPTUAL MODELS

Alain O. Villeneuve
Université de Sherbrooke
avilleneuve@adm.usherb.ca

Diane M. Strong
Worcester Polytechnic Institute
dstrong@wpi.edu

Abstract

We investigated the role of database designers' business and database knowledge in producing high quality conceptual designs. We collected think aloud protocols from experienced designers and students, as well as measures of internal schemata and script knowledge structures. Both domains of knowledge contributed to the technical quality, while neither contributed to the business quality of the conceptual models.

Keywords: Database, conceptual modeling, expertise, knowledge domain, model quality

Introduction

Since their inception, databases have evolved dramatically and are now ubiquitous in organisations. The 1999 database market hit $13 billion with relational database technology accounting for more than $11 billion (Gantz, 1999). In business environments, applications are being integrated through a common database. Improvements in the reliability of databases and information systems require more attention to good database design practice and a better understanding of the designer (Brooks, 1987).

The database design process is typically defined as a four-step process, requirements analysis, conceptual design, logical design, and physical design (Elmasri and Navathe, 2000). The output of requirements analysis and conceptual design is a conceptual model comprised of entities and attributes (Chen, 1976). Understanding requirements is the cornerstone of all good designs since errors or misconceptions introduced during conceptual modelling are expensive to fix. Poor understanding of the requirements has been found to be an important contributor to implementation failures and extensive maintenance (Curtis, Krasner and Iscoe, 1988). Designers carry the burden of eliciting requirements from various users to integrate different needs into a coherent whole. There is still much to learn about the designer, especially how she/he processes information during the requirements stage of the database design. Understanding requirements enjoins the designer to understand the user's world in addition to mastering database design methodology and techniques. This suggests that two domains of knowledge pertain to successful database designs: the business domain, which facilitates the understanding of requirements, and the database domain, which eases the preparation of a sound conceptual database model.

Theoretical Framework

We posit that two domains of knowledge, business knowledge and database knowledge, each comprised of two knowledge structures, contribute to conceptual database model quality, as shown in the research model (Figure 1). The two knowledge structures, schemata and scripts, together help the database designer understand the requirements of a problem, prepare a sound conceptual model of the database, and propose a solution of high quality.

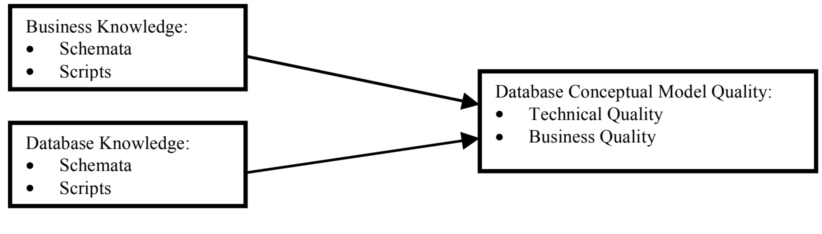

Figure 1. Research Model

Knowledge Structures

Schemata are knowledge structures that categorize and represent knowledge (Rumelhart, 1980; Reimann and Chi, 1989). They contain variables, are organized in an attribute-value format, and bear an indication of the superset to which they belong (Smith, 1989). A schema can have fixed parts (always assumed true for instances of the concept the schema represents) and variable parts that can take on different values. Variables have default values and a range of context dependent permissible values.

Experts' and novices' schemata differ in terms of structure and distinctiveness (Murphy and Wright, 1984). In the case of experts, an attribute can belong to more than one schema while novices' schemata are more independent and concrete. Experts' schemata tend to be organized in many more layers than are those of novices. Experts' higher level schemata are less discriminating since they are more abstract and tend not to contain concrete values for their attributes; they contain ranges of permissible values as opposed to the mostly concrete values of the attributes of novices' schemata.

Scripts have temporal and spatial dimensions and they apply to the interpretation of events taking place at the moment of their instantiation. Scripts are long causal chains of conceptualizations that have occurred in that order many times before (Schank and Abelson, 1977). This contrasts with schemata which are static representations of concepts. Scripts, as causal chains, help in selecting schemata to represent the situation at hand. For instance, the scripts corresponding to the following two situations do not activate exactly the same schemata: 'I have been to a restaurant last night' and 'I have been to the cafeteria last night' (Abelson, 1981; Choo, 1989; Gioia, Donnellon and Sims, 1989). The script for the latter would include a schema not included in the former: a platter or a tray used for self-service.

Schemata and scripts are seen as complementary structures. Schemata categorize concepts but do little to support action. This is the purpose of scripts. Scripts depict a concept in action, and so support behavioral change. Scripts are used to adjust a solution to the problem. Thus, schemata are understood to be static representations of concepts (declarative knowledge) whereas scripts describe the processes of putting concepts into action (procedural knowledge).

Domains of Knowledge

Schemata and scripts play an important role in solving database design problems. The schemata and scripts used during the requirements understanding process cover two different knowledge domains: business-oriented and database-oriented knowledge. The process of understanding the user's world has more to do with business application knowledge than with database knowledge (Elmasri and Navathe, 2000). The designer uses business knowledge to understand business processes, entities, and their attributes, whereas the designer uses database knowledge to understand the technical solution to the problem. For example, database design knowledge provides rules and principles that guide the designer in the application of techniques for uncovering dependencies (Diederich and Melton, 1988; Lee, 1987). Identification of dependencies requires the designer to push his analysis of requirements further since users are not likely to provide such technical information. Without appropriate database knowledge, the designer is unlikely to pursue his/her inquiry to a level that will promote identification of dependencies. Both domains of knowledge, business and database, are important in the database design process.

Model Quality

Model quality is conceived along two dimensions: technical quality, corresponding to compliance with database design and normalization rules, and business quality, corresponding to how well the model supports stated objectives and business processes. Past database design research (Batra, Hoffer and Bostrom, 1990; Shoval and Even-Chaime, 1987) has mainly addressed technical quality, i.e., the principles of good database design. Business quality of the conceptual model has mostly been ignored, implicitly assuming that if the model is technically sound, business quality follows.

Hypotheses

A designer who is knowledgeable about the application domain can build on past experiences and existing knowledge to uncover more facets of users' needs. Business-oriented knowledge aids in understanding the requirements from a business perspective, and thus augments the level of comprehension acquired during analysis. It enables the designer to see subtleties in the entities and thus provide a more comprehensive view of the user's world. Better organized business schemata should provide for a better understanding of the entities present in the user's world while the use of appropriate business scripts should provide for a better understanding of the business processes to be supported by the model. Scripts support recognition and selection of appropriate schemata for interpretation of the situation at hand. Through the use of strong scripts, designers see schemata in their normal time sequence (Choo, 1989; Gioia and Poole, 1984), which helps to identify dependencies, thus favoring the correct placement of the attributes. For example, running a business-script to simulate database retrievals can help to guide the validation of the solution. Thus, we posit:

H1: Business-oriented knowledge will contribute positively to model quality

Database-oriented knowledge will help the designer produce a technically sound solution. Database-oriented knowledge is required for producing database models that follow design conventions, which includes issues around entities and entity types, uniqueness of each piece of data, normalization, etc. This knowledge also encompasses rules for coping with design issues such as complex relationships among entities. For example, ternary relationships can be represented either as a ternary relationship or as a group of binary relationships. These two approaches, however, each give rise to different functional constraints which may limit the use of the database (Elmasri and Navathe, 2000; Batra et al., 1990).

Database-oriented scripts help the designer produce a solution of higher quality by encouraging the use of design rules, and by facilitating a good understanding of relationships among entities. Database-oriented scripts help in defining and handling the degrees of the relationships to be included in the model thus impacting how relationships are represented in the final model. For example, the use of appropriate scripts enables the designer to recall how to handle ternary relationships and how to map them into the final solution, which in turn contributes to a higher quality design. Database-oriented scripts support the understanding of the behavior of the database management system, which contributes to the completeness of the design. For example, simulating database retrievals (e.g., through SQL statements or otherwise) and ensuring that all pieces of information retrieved are present and accessible in the model are examples of database-oriented scripts. Validating the solution by means of playing appropriate scripts can then lead to improvements in quality. Therefore,

H2: Database-oriented knowledge will contribute positively to model quality

Method

We employed a quasi-experimental design for this study. Participants were given a business case and asked to develop a conceptual database model. For ease of comparison among resulting models, participants were instructed to produce their final solution in relational data model format, though they could use any conceptual modelling methods they deemed appropriate. Participants were required to think aloud while working on the case.

Data Collection

The sample included 29 practitioners of varying degree of database work experience and 27 MIS graduate students, some of whom had work experience. Of the 56 participants, 21 were female. A questionnaire was used to capture demographics of the participants (see Table 1), control variables and self-assessments of quality. Cronbach's alpha (Cronbach, 1951) for the scales ranged from 0.80 to 0.94 and were all within acceptable ranges (Nunnally, 1978).

Table 1. Demographics

	Total Sample		**Practitioners**		**Graduate Students**	
	Mean	SD	Mean	SD	Mean	SD
Age	33.95	7.22	38.55	7.03	29.16	3.46
Work experience	10.98	7.58	15.24	7.26	6.41	4.79
Database work experience	3.92	5.01	7.12	5.08	0.48	1.21

A computer program, called MemTree, was used to capture participants' schemata organization. The program presented each participant with pairs of terms for which she/he entered a similarity judgement value ranging from 1 to 9 (1 for very similar items, 9 very dissimilar) for each pair. MemTree generated the pairs of terms separately from two lists of terms. The list for the database domain comprised 17 terms drawn from database design textbooks, while the list for the business domain comprised 23 terms drawn from the case material. The same ordering of pairs was used for all participants. The program processed each domain of knowledge independently.

A complex business case was used as the main stimulus for this study. The case required support for purchasing, order-entry, billing, shipping, receivables, inventory control, inventory management, cost accounting, and production planning and scheduling. The case, which was thoroughly tested in earlier research (Villeneuve and Strong, 1993), illustrates a complex, unfamiliar business (producer of granite funeral monuments and construction slabs). The requirements information was designed to simulate the information a designer would obtain from interviews, e.g., it was complex, not pre-processed into entities, attributes, or reports, and was written from a business process perspective.

A microcassette tape recorder was used to capture each participant's think aloud verbal reports while he/she was working on the case. An observer used a digital watch and preprinted observation sheets to record real time observations on the working behavior of the participant. Since participants in past think aloud research tended to report their behavior and not necessarily their thoughts, or did not verbalize enough to produce a usable protocol (Ericsson and Simon, 1993; Mackay and Elam, 1992; Vessey and Galletta, 1991), two problems with no resemblance to the main case were used to warm up participants.

Measures

Schemata were measured using Johnson Hierarchical Clustering (Johnson, 1967) to produce memory trees from the participants' similarity assessments collected by the MemTree program. From the memory trees, the sum of the path lengths was used as a raw score of schemata organization: long paths indicate lack of organization (concepts are not integrated), short paths moderate organization, and moderate paths high organization (subtleties between concepts are captured by the participant). A categorical variable was used to reflect the level of schemata organization (low, moderate, high). The number of layers of abstraction was used as a second measure within the tree along three categories (low, moderate, high). Separate variables were prepared for each domain of knowledge: business (BS) and database (DB).

Scripts were identified from the verbal protocols. Since scripts represent sequences of events, episodes were analyzed to identify such sequences. Sequences were then classified as either business-oriented or database-oriented. Counts by category were then prepared (business and database) to represent script use. Furthermore, since scripts vary in terms of length and content (elaborateness), a categorical variable was derived to measure scripts' elaborateness (low, moderate, high).

Quality was assessed along two dimensions: technical and business. An error-based approach was used to assess technical quality (ratio of errors to attributes): counts were derived from errors such as unique identifier errors (e.g., not a superkey, must be present and sufficient, etc.); aggregation errors; derivation errors; normalisation errors. Raw scores were then expressed as a percentage of the total number of attributes present in the proposed model to account for varying model sizes and scopes.

Business quality (business score) was measured by assessing how well the model supports business objectives and needs. Each objective was assessed on a 10-point scale and the sum of all objectives was used as a measure of business quality. Five random samples of four solutions each were independently assessed by groups of graduate students knowledgeable about business issues and about the case; all rs were high and significant at $p < .01$.

Results

Table 2 summarizes the basic results, while results from analysis of variance (ANOVA) are presented in Tables 3 and 4. The ANOVAs compare similar knowledge structures to one another (i.e., schemata to schemata, scripts to scripts).

Technical Quality (Ratio of Errors to Attributes)

When comparing the schemata knowledge structures (Table 3), both knowledge domains were significant for explaining technical quality, thus providing support for the hypotheses. The interaction between the two knowledge structures, however, was the most important factor in explaining the dependent variable with an overall effect of $\eta^2_{alt} = .26$ and respectable statistical power $\beta = .07$. The (high BS - moderate DB) combination made the most errors when compared to other combinations. This suggests that advanced business knowledge may be detrimental to the process of designing databases. Since designing databases requires identifying attributes of entities (mostly low-level knowledge) and since high-level schemata are less discriminating than low-level schemata, the individual may be struggling during the process. Database knowledge may take over when it reaches a high level only, and until then the individual is unsure about what route to take.

Table 2: Means and Standard Deviations for Solution Quality

	Low		Moderate		High	
Business Schemata: Organization						
	Low (n=20)		Moderate (n=18)		High (n=18)	
Technical Quality	18.85_a	(6.60)	20.26_a	(8.69)	23.98_a	(14.36)
Business Quality	22.35_a	(13.64)	20.94_a	(14.74)	14.83_a	(16.35)
Business Schemata: Abstraction						
	Low (n=17)		Moderate (n=12)		High (n=27)	
Technical Quality	19.84_a	(9.74)	24.14_a	(15.43)	20.24_a	(7.81)
Business Quality	20.82_a	(14.98)	20.42_a	(18.07)	18.22_a	(14.00)
Business Scripts: Use						
	Low (n=20)		Moderate (n=17)		High (n=19)	
Technical Quality	22.64_a	(10.69)	20.35_a	(13.09)	19.71_a	(6.97)
Business Quality	17.20_a	(11.64)	21.18_a	(18.49)	20.37_a	(15.19)
Business Scripts: Elaborateness						
	Low (n=17)		Moderate (n=17)		High (n=22)	
Technical Quality	24.17_a	(11.06)	17.70_a	(4.40)	20.98_a	(12.45)
Business Quality	13.65_a	(11.55)	$20.29_{a,b}$	(13.29)	23.36_b	(17.57)
Database Schemata: Organization						
	Low (n=19)		Moderate (n=18)		High (n=19)	
Technical Quality	19.86_a	(9.56)	24.19_a	(13.13)	18.98_a	(7.52)
Business Quality	19.16_a	(15.59)	15.33_a	(11.28)	23.74_a	(16.94)
Database schemata: Abstraction						
	Low (n=17)		Moderate (n=22)		High (n=17)	
Technical Quality	21.64_a	(12.50)	19.92_a	(9.56)	21.60_a	(9.41)
Business Quality	15.00_a	(12.06)	23.77_a	(16.57)	18.41_a	(14.79)
Database Scripts: Use						
	Low (n=17)		Moderate (n=20)		High (n=19)	
Technical Quality	26.73_a	(13.81)	17.83_b	(7.53)	19.08_b	(7.16)
Business Quality	12.47_a	(13.05)	24.30_b	(15.44)	$20.68_{a,b}$	(14.49)
Database Scripts: Elaborateness						
	Low (n=18)		Moderate (n=18)		High (n=20)	
Technical Quality	24.22_a	(9.25)	19.02_a	(8.42)	19.75_a	(12.42)
Business Quality	14.39_a	(12.40)	21.94_a	(15.09)	21.85_a	(16.53)

Notes:
Means on the same row that do not share subscripts differ at $p < .05$ by the Fisher least significant difference test.
Numbers in parentheses are standard deviations.
Technical quality is the ratio of errors to attributes; Business quality is the business score defined previously.

Table 3. Analysis of Variance for the Effects of Schemata on Technical Quality

	df	F	η^2_{alt}	β
DB schemata organization (A)	2	4.23*	.14	.65
BS schemata organization (B)	2	5.19**	.18	.65
A x B	4	4.09**	.26	.07
Error	47	(81.90)		

When comparing the scripts knowledge structures (Table 4), the use of database scripts and the interaction between the use of business scripts and their elaborateness were significant, providing support for the hypotheses. Moderate and heavy use of DB scripts reduced substantially the number of errors over light use of such scripts, while low use of highly elaborate BS scripts also reduced the number of errors over moderate and heavy use of such scripts. Too much reliance on highly elaborate BS scripts was detrimental to quality. Heavy use of moderately elaborate scripts, however, were as appropriate as low use of highly elaborate BS scripts, suggesting a tradeoff in the designer's solving strategy.

Table 4. Analysis of Variance for the Effects of Scripts on Technical Quality

Source	df	F	η^2_{alt}	β
DB scripts use (A)	2	6.20**	.24	.11
BS scripts use (B)	2	2.04	-	-
DB scripts elaborateness (C)	2	0.79	-	-
BS scripts elaborateness (D)	2	0.49	-	-
A x C	4	2.41	-	-
B x D	3	2.92*	.18	.62
Error	40	(85.61)		

Business Quality (Business Score)

None of the schemata-based models or the scripts-based models were significant for business quality. High business schemata organization, however, consistently produced solutions of the poorest business quality. The use of database scripts was, in general, associated with better business quality. Therefore, the data did not support the hypotheses when considering business quality.

Discussion

Overall, the research model received partial support. Especially in the area of technical quality (ratio of errors to attribute), it was clear from the results that no single domain, when taken alone, was sufficient to explain solution quality. In the case of business quality, however, one model approached significance, $F(4, 51) = 2.29$, $p = .07$, with database schemata abstraction as an explanatory variable. Additional analysis is underway to examine cross-over knowledge structures from the two domains of knowledge. Preliminary results reveal that there exists some threshold where one domain, while dominant below the threshold, is taken over by the other domain above that threshold. Such results are encouraging.

This study makes several contributions. It informs us about the role of different knowledge domains in the process of developing databases. The results indicate that it is not sufficient to train designers in the sole area of database techniques and methods as is common in current curriculum models (Davis, Gorgone, Couger, Feinstein, and Longenecker, 1997). Designers must also be knowledgeable about the target application domain. The research also informs us about the role of knowledge structures during conceptual modelling thus contributing to a theory of expertise. Results also suggest that it may be appropriate to revise our design methodologies along with training methods to promote a better understanding of the user's world and, by extension, to develop systems of higher quality.

Future work should aim at better understanding the specific role of each knowledge structure and should address the specifics of the how and when a knowledge domain contributes to quality. Future research should also consider addressing measurement issues and content analyze verbal protocols.

References

Abelson, R. P. "Psychological status of the script concept," *American Psychologist* (36:7), July 1981, pp. 715-729.

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

Brooks, F. P. Jr. "No silver bullet: Essence and accidents of software engineering," *IEEE Computer* (20), April 1987, pp. 10-19.

Chen, P.P. "The Entity-Relationship Model -- Toward a Unified View of Data," *ACM Transactions on Database Systems* (1:1), January 1976, pp. 9-36.

Choo, F. "Cognitive scripts in auditing and accounting behavior," *Accounting, Organizations & Society* (14:5), 1989, pp. 481-493.

Cronbach, L. J. "Coefficient Alpha and the Internal Structure of Tests," *Psychometrica* (16:3), September 1951, pp. 297-334.

Curtis, B., Krasner, H. and Iscoe, N. "A field study of the software design process for large systems," *Communications of the ACM* (31:11), November 1988, pp. 1268-1287.

Davis, G. B., Gorgone, J. T., Couger, J. D., Feinstein, D. L., and Longenecker, H. E. "IS '97 Model curriculum and guidelines for undergraduate degree programs in information systems," *Database* (28:1), Winter 1997, pp. 1-94.

Diederich, J., and Milton, J. "New Methods and Fast Algorithms for Database Normalization," *ACM Transactions on Database Systems* (13:3), July 1988, pp. 339-365.

Elmasri, R., and Navathe S. B. *Fundamentals of database systems (Third edition)*, The Benjamin/Cummings Publishing Company Inc., Redwood City, CA, 2000.

Ericsson, K. A., and Simon, H. A. *Protocol analysis: Verbal reports as data (Revised edition)*, The MIT Press, Cambridge, MA, 1993.

Gantz, J. "Will relational databases leave dinosaur tracks?," *Computerworld* (33:19), May 10 1999, pp. 32-33.

Gioia, D. A. and Poole, P. P. "Scripts in organizational behavior," *Academy of Management Review* (9:3), July 1984, pp. 449-459.

Gioia, D. A., Donnellon, A. and Sims, H. P. Jr. "Communication and cognition in appraisal: A tale of two paradigms," *Organization Studies* (10:4), 1989, pp.503-529.

Johnson, S. C. "Hierarchical Clustering Schemes," *Psychometrika* (32), 1967, pp. 241-254.

Lee, T. L. "An Information-Theoretic Analysis of Relational Databases-Part I: Data Dependencies and Information Metric," *IEEE Transactions on Software Engineering* (SE-13:10), October 1987, pp. 1049-1061.

Mackay, J. M., and Elam, J. J. "A comparative study of how experts and novices use a decision aid to solve problems in complex knowledge domains," *Information Systems Research* (3:2), June 1992, pp. 150-172.

Murphy, G. L., and Wright, J. C. "Changes in conceptual structure with expertise: Differences between real-world experts and novices," *Journal of Experimental Psychology: Learning Memory and Cognition* (10:1), 1984, pp. 144-155.

Nunnally, J. *Psychometric theory (Second edition)*, McGraw-Hill, New-York, NY, 1978.

Reimann, P. and Chi, M. T. H. "Human expertise," In *Human and Machine Problem Solving*, K. J. Gilhooly (ed.), Plenum Press, New York, NY, 1989, pp. 161-191.

Rumelhart, D. E. "Schemata: The building blocks of cognition," In *Theoretical Issues in Reading Comprehension*, R. Spiro, B. Bruce, and W. Brewer (eds.), Lawrence Erlbaum Associates, Hillsdale, NJ, 1980, pp. 33-58.

Schank, R. C., and Abelson, R. P. *Scripts, plans, goals, and understanding: An inquiry into human knowledge structures*, Lawrence Erlbaum Associates, Hillsdale, NJ, 1977.

Shoval, P., and Even-Chaime, M. "Data base schema design: An experimental comparison between normalization and information analysis," *Database* (18:2), Spring 1987, pp. 30-39.

Smith, E. E. "Concepts and induction," In *Foundations of Cognitive Science*, M. I. Posner (ed.), MIT Press, Cambridge, MA, 1989, pp. 501-526.

Vessey, I., and Galletta, D. "Cognitive fit: An empirical study of information acquisition," *Information Systems Research* (2:1), January 1991, pp. 63-84.

Villeneuve, A. O., and Strong, D.M. "Database Requirements: Schemata as an Explanatory Construct in Database Conceptual Modeling," *Proceedings of the Third Workshop on Information Technology and Systems*, Orlando, FL, 1993, pp. 57-66.