45th EOQ CONGRESS 2001

19 - 21 SEPTEMBER 2001
İSTANBUL
TURKEY

LUTFİ KIRDAR INTERNATIONAL CONVENTION & EXHIBITION CENTER

EUROPEAN ORGANIZATION FOR QUALITY

TSE TURKISH NATIONAL COMMITTEE

Quality: *The Bridge to Global Competition*

PROCEEDINGS

TABLE OF CONTENTS

AUTHOR INDEX

commercial considerations. The selected assessment attributes strongly influence the set of necessary assessment evaluation modules. Obviously, each evaluation module must be associated with at least one attribute and conversely each attribute must be assessed by evaluation modules. The choice of evaluation modules for assessing a particular piece of software will be based on the type of available information. For example, the programming language used may restrict the number of available tools for static analysis.
On the other hand, an evaluation module might not be explicitly related to any higher level quality characteristics. Its result may be used in the assessment of various characteristics. However, the definition of evaluation modules using other evaluation module results will provide some structuring elements.
It may be that no complete set of evaluation modules exists that would completely assess to the required level for a particular software product. In this situation there may be a case for constructing a new evaluation module or rejecting certification of the product.

7 Conclusion

The report summarizes the ideas and views of several evalaution and certification initiatives as they appeared during the 90ies. The models and methods used for assessment and certicification might need further refinement and the case studies being carried out in phase two may also indicate the need for modifications.

References

[Azu91] Azuma, M. *Japanese contribution on Quality Metrics*, ISO / IEC JTC1 / SC7 / WG3, International Organization for Standardization, May29, 1991.
[ISO9126] Draft international standard ISO / IEC DIS 6126, *Information technology - Software product evaluation - Quality characteristics and guidelines for their use*, International Organization for Standardization, International Electrotechnical Commission, January 5, 1991.
[HCW91] Hausen, H.-L.; Cacutalua, N.; Welzel, D.; *A Method of Software Assessment and Certification-Draft-*, Arbeitspapiere der GMD, No. 571, GMD Sankt Augustin, September 1991 or SCOPE Report SC.91/101/GMD.hlh/T2.1.2.1./FOR.MOD/RP/IO2, December 1991.
[McCa77] McCall, J.A.; Richards, P.K.; Walters, G.F. *Concepts and Definitions of Software Quality* Factors in Software Quality, Vol.1, Springfield, Va.:NTIS, November 1977.
[Sco90] Concortium SCOPE, *Certification Model - final report*, SC.90/031 /ECT.jb/T1.1.1.2 /Rp/04, 1990.
[Wel91] Welzel, Dieter *Mapping Software Characteristics onto Metrics and Methods* - An Example, SCOPE Working Paper SC.91/015/GMD.dw/T2.1.1.1/MAP.CHAR/WP/I01, October 1991.

METRICS FOR DATABASE MODELS

Mario Piattini, Coral Calero, Marcela Genero,
Grupo ALARCOS
University of Castilla-La Mancha

ABSTRACT

Nowadays, in a global and increasingly competitive market, quality is a critical success factor for all economical and organisational aspects, and especially in Information Systems (IS). It is important that software applications can be evaluated for every relevant quality characteristic using validated metrics. Software engineers have been putting forward huge quantities of metrics for software products, processes and resources. Unfortunately, almost all the metrics proposed are focused on programs, disregarding data-related quality. In this paper we present different metrics to analyse the quality of databases, measuring the complexity of entity relationship diagrams and relational database schemas.

1. INTRODUCTION

Nowadays, in a global and increasingly competitive market, quality is a critical success factor for all economical and organisational aspects, and especially in Information Systems (IS). Developing and selecting high quality software applications is fundamental. It is important that the software applications can be evaluated for every relevant quality characteristic using validated metrics.

Taking into account that databases are the core of information systems, it is clear that their quality has a significant impact on the quality of the information system which is ultimately implemented. Database quality deals with different aspects: DBMS quality, data model quality (both conceptual and logical) and data quality, see figure 1. We only refer in this work to data model quality.

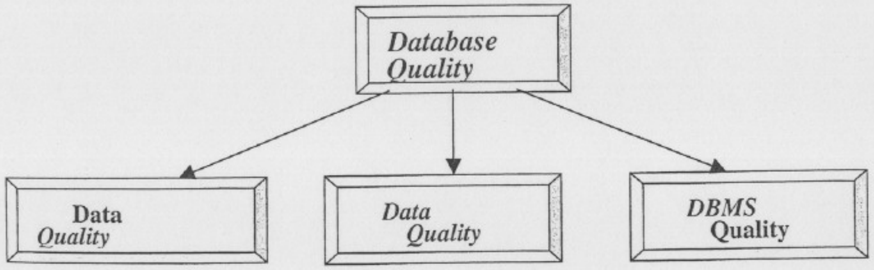


Figure 1. Different aspects of database quality.

Metrics are good mechanisms for improving the quality of a software product. A lot of metrics for software products, processes and resources can be found in the related literature (Melton, 1996; Fenton and Pfleeger, 1997). Unfortunately, almost all the metrics proposed are focused on programme, disregarding data-related quality (Sneed and Foshag, 1998). This neglect could be explained as databases have developed until recently just a secondary role with minor contribution to the quality of the overall system. Nowadays, databases are introduced in most of the important IS, becoming their essential core, so assuring the quality of databases means assuring the quality of the Information Systems.

Following the ISO/IEC 9126 quality model (ISO, 1999), several characteristics can be identified in software quality: functionality, reliability, usability, efficiency, maintainability and portability. Taking into account that maintenance arrange between 60 and 90 percent of life cycle costs (Card y Glass, 1990; Pigoski, 1997), we focus our work on maintainability. ISO/IEC 9126 distinguishes five subcharacteristics for maintainability: analysability, changeability, stability, testability and compliance (see figure 2). Analysability, changeability and testability are in turn influenced by complexity (Li y Cheng , 1987). However, a general complexity measure is "the impossible holy grail" (Fenton, 1994). Henderson-Sellers (1996) distinguishes three types of complexity: computational, psychological and representational, and for psychological complexity he considers three components: problem complexity, human cognitive factors and product complexity. The last one is our focus.

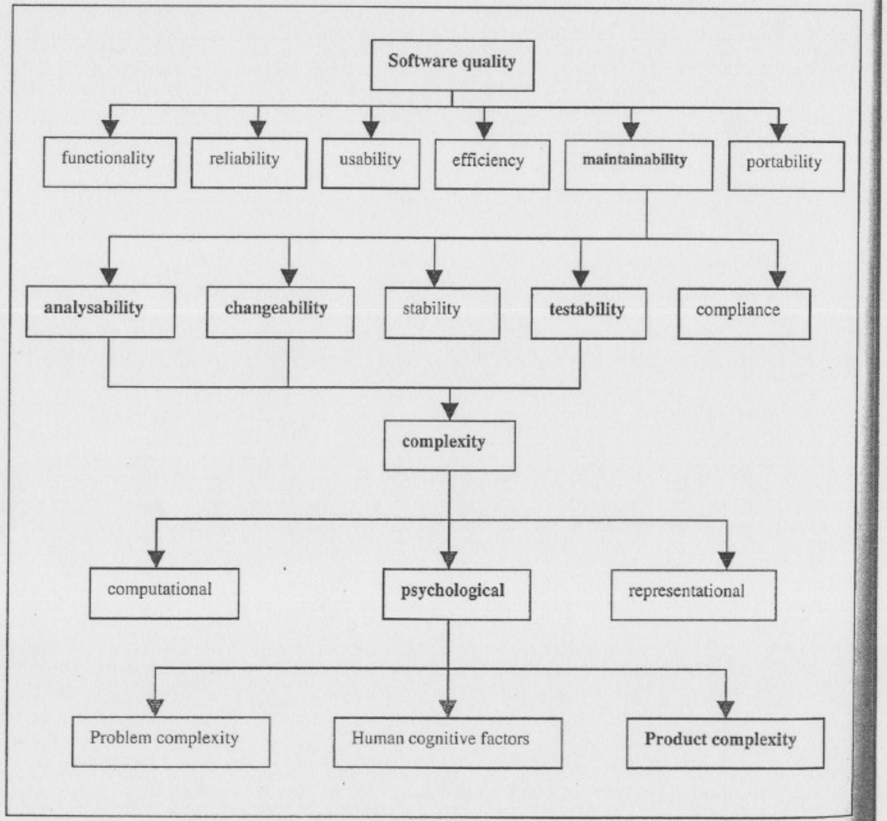


Figure 2. Relation between products complexity metrics and software quality

So, the metrics we define are for measure internal attributes of data model, which can characterised their complexity which can help to assess its maintainability (the external attribute).

As we have said previously, we cover both conceptual and logical data model quality. The conceptual data model specifies the requirements about the database and is the first product within the database life cycle. Moreover conceptual data models determine what information can be represented by an information system (Feng, 1999). So that, it is really important to build them as "good" as possible because the cost of correcting a mistake introduced in the first stages of software development grows up exponentially when the project advances.

We will focus on entity relationship (ER) diagrams (Chen 1976; Teorey, 1999) because in today's database design world it is still the dominant method of conceptual modelling (Muller, 1999).

The logical model organizes the entities and relationships identified in the previous step, in several tables. We will focus on the relational model because relational DBMSs are the most used today, so if we have some metrics for assuring relational databases quality, we will be able to control the quality of a high percentage of the current IS.

The metrics we have proposed for E/R diagrams and for relational databases will be presented in the next section.

2. PROPOSED METRICS

2.1. Metrics for conceptual modelling

Here we present the definition of Genero et al. (2000a)'s metrics for measuring ER diagrams structural complexity. These metrics can be classified into the following categories: Entity metrics, Attribute metrics and Relationship metrics.

Entity metrics

NE METRIC. Is the total the number of entities within the ER diagrams.

Attribute metrics

NA METRIC. Is the total number of attributes that exist within the ER diagrams, taking into account both entity and relationship attributes. In this number we include simple attributes, composite attributes and also multivalued attributes, each of which take the value 1.

DA METRIC. An ER diagrams is minimal when every aspect of the requirements appears once in the diagram, i.e. an ER diagrams is minimal if it does not have any redundancies. One of the sources of redundancies in the ER diagrams is the existence of derived attributes. An attribute is derived when its value can be calculated or deduced from the values of other attributes. The Derived Attributes metric as the number of derived attributes existing in the ER diagrams.

CA METRIC. Is the total number of composite attributes within an ER diagrams.

MVA METRIC. Is total number of multivalued attributes within the ER diagrams.

Relationship metrics

NR METRIC. Is the total number of relationships within the ER diagrams, taking into account only common relationships.

M:NR METRIC. Is the total number of M:N relationships within the ER diagrams.

1:NR METRIC. Is the total number of 1:N relationships (including also 1:1 relationships) within the ER diagrams.

N-ARYR METRIC. Is the total number of N-ary relationships (not binary) within the ER diagrams.

BINARYR METRIC. Is the total number of binary relationships within the ER diagrams.

NIS_AR METRIC. Is the total number of relationships IS_A (generalisation/ specialisation) that exist within the ER diagrams. In this case, we consider one relationship for each child-parent pair within the IS_A relationship.

REFR METRIC. Is the total number of reflexive relationships that exist within the ER diagrams.
RR METRIC. Another source of redundancy in an ER diagrams is the existence of redundant relationships. We define the Redundant Relationship metric as the number of relationships that are redundant in the ER diagrams.

These metrics are open-ended metrics, i.e. they are not bounded in an interval. Close-ended metrics (ratio metrics) could also be useful, such as the following: RR/NR; DA/NA; M.NR/NR; 1:NR/NR, BinaryR/NR, N-AryR/NR, etc.

2.2. Metrics for relational schemas
Traditionally, the only indicator used to measure the "quality" of relational databases has been the normalization theory, upon which Gray et al. (1991) propose to obtain a normalization ratio. But we think that normalization is not enough to measure the quality of relational databases, so we propose the following three metrics in addition to normalization (Calero et al, 2000):

NT, the number of tables in the database
NA, the number of attributes in all the tables of the database
NFK, the number of foreign keys in the database
DRT (Depth of the referential Tree), defined as the longest referential path between tables in the schema database

All these metrics have been validated formally and empirically (Calero et al., 2000) demonstrating their utility as database quality factors.

3. EXAMPLE
In this section we present a metrics application example of an hospital database. We will present the conceptual and the relational schemas with the corresponding values for the presented metrics. In figure 3 the conceptual schema is shown.

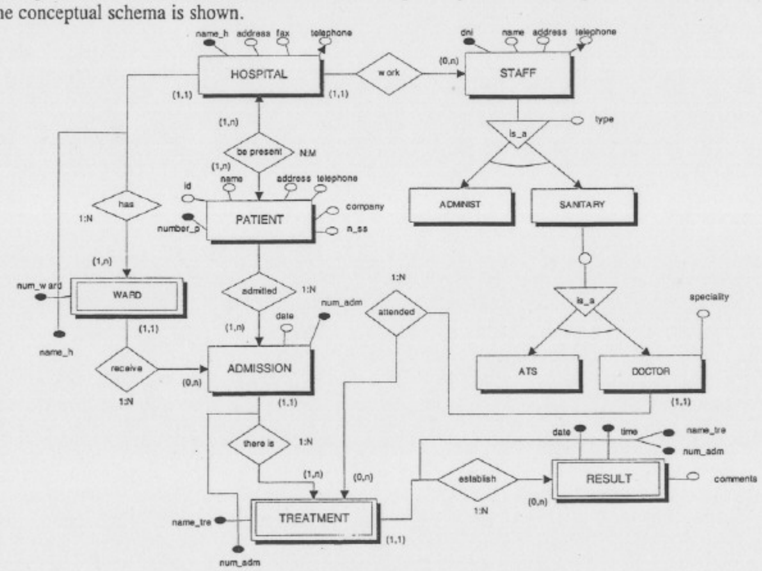


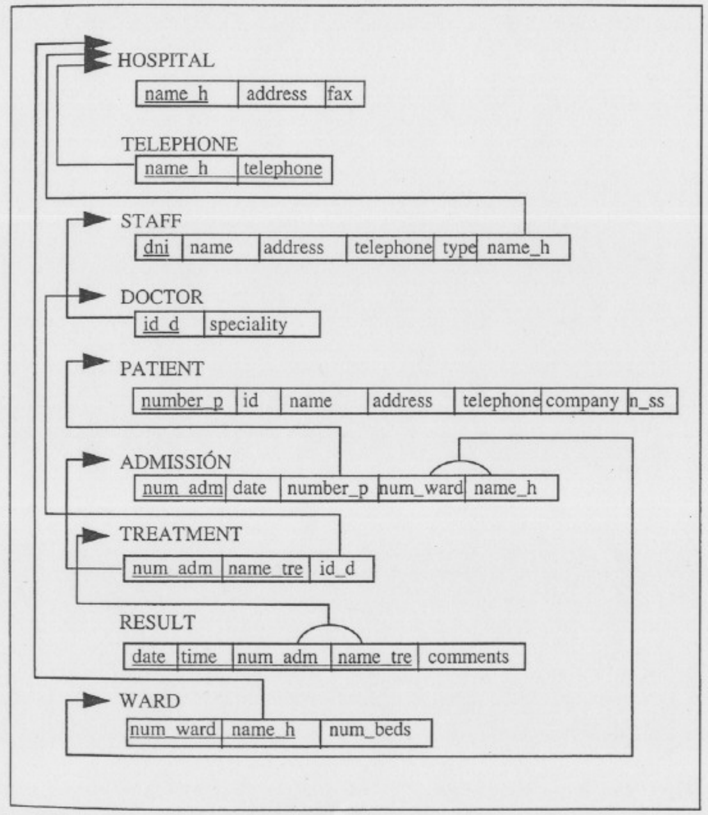
Figure 3. Conceptual schema of the example

3.1. Conceptual schema
The values of the metrics are:

NE METRIC = 11	NA METRIC = 28	DA METRIC = 0
CA METRIC = 1	MVA METRIC = 0	NR METRIC = 7
M:NR METRIC = 1	1:NR METRIC = 6	BINARYR METRIC = 7
N-ARYR METRIC = 0	NIS_AR METRIC = 4	REFR METRIC = 0
RR METRIC = 0		

3.2 Relational schema

In figure 4, we represent the tables with the corresponding primary and foreign keys.
Figure 4.- Relational database of the example



The values of the metrics are:
NT = 9 NA = 36 RFK = 9 DRT = 4

4. CONSEQUENCES FOR QUALITY DATABASE DESIGN

In the previous sections we have presented the metrics that can be applied to the E/R model and to the relational model. But, how can these metrics help us in designing and in improving the quality?.
On the one hand, let us think in the situation of a totally new database design. In this case, we can obtain from the requirements of our future system, more than one alternative of conceptual modelling. Usually the designer selects one of them based on his/her own experience. This method, although used, is not the best one because decisions are taking attending only subjective considerations. For avoiding these situations we can apply the proposed metrics. From the conceptual schema more than one relational schema can be derived, depending on the transformation rules selected. This selection is made by the designer following his/her intuition and his/her experience. As in the conceptual modelling, a big part of the design is based in the designer background. So, metrics can help in the selection of the best option independently of the designer subjective factors.
On the other hand, let us think in an existing database. Usually, databases as other software product must be submitted to a maintenance process. As a result of the maintenance process, some changes can be necessaries in the relational schema (reorganization, restructuring, partition, etc.). In this situation, different alternatives for implementing these changes could be considered and the proposed metrics can be applied for guiding the designer in choosing the best one.

Summarizing, metrics can drive designers along the different steps of a database design or in the database maintenance, assuring database quality.

5. CONCLUSIONS AND FUTURE WORK

Due to the growing complexity of IS, continuous attention to and assessment of the data models are necessary to produce quality information systems. Following this idea, we have presented a set of objective and automatically computed metrics for evaluating the complexity of E/R diagrams and relational schemas.

The proposed metrics can be used for guiding the designer in the database design process. Also the metrics can be used to improve database maintainability. Through the metrics the designer can select the best design option. As a result of the metrics usage, quality databases will be obtained.

In our research group we are defining more metrics for both conceptual and logical models. We have already performed some research regarding OO conceptual modelling (Genero et al., 1999; Genero et al., 2000b). In Calero et al (1999) metrics for object-relational databases can be found and in Díaz et al (2001) metrics for active databases are presented.

ACKNOWLEDGMENTS
This research is part of the MANTICA project, partially supported by CICYT and the European Union (1FD97-0168).

REFERENCES

Calero, C., Piattini, M., Ruiz, F. and Polo, M. (1999) Validation of metrics for Object-Relational Databases, *International Workshop on Quantitative Approaches in Object-Oriented Software Engineering* (ECOOP99), Lisbon (Portugal), 14-18 June
Calero, C., Piattini, M., Genero, M., Serrano, M. and Caballero, I. (2000). Metrics for Relational Databases Maintainability, *UKAIS2000*, Cardiff, UK.
Card, D.N. y Glass, R.L. (1990). *Measuring Software Design Quality*. Englewood Cliffs. USA.
Chen, P. (1976). The Entity-Relationship Model: toward a unified view of data. ACM Transactions on *Database Systems*, 1, 9-36.
Díaz, O., Piattini. M. and Calero, C. (2000). Measuring active databases maintainability. Accepted for pubblication in *Information Systems Journal.*
Feng, J. (1999). The "Information Content" problem of a conceptual data diagram and a possible solution. *Proceedings of the 4th UKAIS Conference: Information Systems-The Next Generation*, University of York, 257-266.
Fenton, N. 1994. Software Measurement: A Necessary Scientific Basis. *IEEE Transactions on Software Engineering*, 20 (3), 199-206.
Genero, M., Manso, Mª E., Piattini,M., García, F. (1999). Assessing the Quality and the Complexity of OMT Models. *2nd European Software Measurement Conference - FESMA 99*, AmstER diagramsam, The Netherlands, 99-109.
Genero, M., Piattini, M. and Calero, C. (2000a). Formalization of Metrics for Conceptual Data Models. *UKAIS 2000*. Cardiff, 26-28 April.
Genero, M., Piattini, M. and Calero, C. (2000b). Métricas para jerarquías de agregación en diagramas de clases UML. *Memorias del Jornadas Iberoamericanas de Ingeniería de Requisitos y ambientes de Software, IDEAS'2000,* Cancún, México, 5-7 Abril, 373-384.
Gray, R., Carey, B., McGlynn, N. and Pengelly A. (1991). Design metrics for database systems. BT *Technology*, 9(4).
Henderson-Sellers, B. (1996). *Object-oriented Metrics - Measures of complexity*. Prentice-Hall, Upper Saddle River, New Jersey.
ISO/IEC 9126-1.2 (1999). Information technology- Software product quality – Part 1: Quality model. Lethbridge
Li, H. and Cheng, W. (1987). An empirical study of software metrics. *IEEE Transactions on Software Engineering*, 13 (6), 679-708.
Muller, R. (1999). *Database Design For Smarties: Using UML for Data Modeling*. Morgan Kaufman.
Pigoski, T.M. (1997). *Practical Software Maintenance*. Wiley Computer Publishing. New York, USA.
Teorey, T. (1999). *Database modeling and design*. Third edition. San Francisco, Morgan Kaufmann.