Lecture Notes in Computer Science 1920

Alberto H. F. Laender Stephen W. Liddle
Veda C. Storey (Eds.)

Conceptual Modeling – ER 2000

19th International Conference on Conceptual Modeling
Salt Lake City, Utah, USA, October 2000
Proceedings

Springer

Alberto H.F. Laender   Stephen W. Liddle
Veda C. Storey   (Eds.)

Conceptual Modeling – ER 2000

19th International Conference on Conceptual Modeling
Salt Lake City, Utah, USA, October 9-12, 2000
Proceedings

Springer

Series Editors

Gerhard Goos, Karlsruhe University, Germany
Juris Hartmanis, Cornell University, NY, USA
Jan van Leeuwen, Utrecht University, The Netherlands

Volume Editors

Alberto H.F. Laender
Universidade Federal de Minas Gerais
Departamento de Ciência da Computação
31270-010 Belo Horizonte - MG, Brasil
E-mail: laender@dcc.ufmg.br

Stephen W. Liddle
Brigham Young University
Marriott School, School of Accountancy and Information Systems
585 TNRB, P.O. Box 23087, Provo UT 84602-3087, USA
E-mail: liddle@byu.edu

Veda C. Storey
Georgia State University, College of Business Administration
Department of Computer Information Systems
Atlanta, Georgia 30302-4015, USA
E-mail: vstorey@gsu.edu

Cataloging-in-Publication Data applied for

Die Deutsche Bibliothek - CIP-Einheitsaufnahme

Conceptual modeling : proceedings / ER 2000, 19th International Conference on Conceptual Modeling, Salt Lake City, Utah, USA, October 9 - 12, 2000. Alberto H. F. Laender ... (ed.). - Berlin ; Heidelberg ; New York ; Barcelona ; Hong Kong ; London ; Milan ; Paris ; Singapore ; Tokyo : Springer, 2000
(Lecture notes in computer science ; Vol. 1920)
ISBN 3-540-41072-4

CR Subject Classification (1998): H.2, H.4., F.4.1, I.2.4, H.1, J.1

ISSN 0302-9743
ISBN 3-540-41072-4 Springer-Verlag Berlin Heidelberg New York

Springer-Verlag Berlin Heidelberg New York
a member of BertelsmannSpringer Science+Business Media GmbH

Printed in Germany

Typesetting: Camera-ready by author, data conversion by PTP-Berlin, Stefan Sossna

Preface

This volume provides a comprehensive, state-of-the-art survey of conceptual modeling. It includes invited papers, research papers, and abstracts of industrial presentations given at ER2000, the 19th International Conference on Conceptual Modeling, held in Salt Lake City, Utah. Continuing in its long tradition of attracting the leading researchers and practitioners in advanced information systems design and implementation, the conference provided a forum for presenting and discussing current research and applications in which the major emphasis was on conceptual modeling. The conference topics reflected this strong conceptual-modeling theme while recognizing important, emerging developments resulting from recent technological advances.

The call for papers for the research track resulted in the submission of 140 papers from researchers around the world. Of these, 37 were selected for inclusion in the program. The authors of these papers are from 14 countries. These papers represent a variety of topics including:

- Database integration
- Temporal and active database modeling
- Database and data warehouse design techniques
- Analysis patterns and ontologies
- Web-based information systems
- Business process modeling
- Conceptual modeling and XML
- Engineering and multimedia application modeling
- Object-oriented modeling
- Applying object-oriented technology
- Quality in conceptual modeling
- Application design using UML

Three internationally recognized scholars in the area of conceptual modeling also submitted papers and delivered keynote speeches:

- John Mylopoulos: *From Entities and Relationships to Social Actors and Dependencies*
- Salvatore T. March: *Reflections on Computer Science and Information Systems Research*
- Philip A. Bernstein: *Generic Model Management—Why We Need It and How to Get There*

In addition to the research papers and invited papers, the conference included two workshops, two pre-conference full-day tutorials, four short tutoriwas entitled "Conceptual Modeling Approaches for E-Business (eCOMO2000)," and the other was entitled "The World Wide Web and Conceptual Modeling (WCM2000)." The pre-conference tutorials allowed participants to learn the latest information about using conceptual-modeling techniques for software development using UML and for Internet site development using co-design. Scott N. Woodfield taught the UML software development tutorial, while Bernhard Thalheim and Klaus-Dieter Schewe taught the Internet site development tutorial. Short tutorials were taught by Avigdor Gal on utilizing ontologies in e-commerce, by Uma Srinivasan and John R. Smith on conceptual modeling for multimedia applications, by Il-Yeol Song on logical design for data warehousing, and by Benkt Wangler and Paul Johanneson on application and process integration. The industrial sessions were largely sponsored in conjunction with DAMA International and addressed pertinent issues relating to the application of conceptual modeling to solving real-world problems.

Authors who submitted their work and the program committee (PC) members and additional reviewers who carefully reviewed the papers and provided their professional assessments deserve appreciation and recognition. A number of the PC members participated in a rather lengthy virtual meeting to finalize the paper selection and diligently reviewed additional papers on demand, all in an effort to create the best program possible. The PC chairs, Alberto H.F. Laender and Veda C. Storey, labored diligently to provide the best possible research program for ER2000. This volume is a tribute to their efforts. Stephen W. Liddle deserves special thanks for creating the Web-based conference management software that enabled the online submission of papers and reviews, and that made the virtual PC meeting possible. He also worked closely with the PC chairs to manage the review process and assemble the conference proceedings.

Many others deserve appreciation and recognition. The steering committee and conference chair provided advice and vision. Bernhard Thalheim directed the search for timely workshops. Selected workshops were chaired by Heinrich C. Mayr (eCOMO2000) and Stephen W. Liddle (WCM2000) who both worked tirelessly with their program committees to obtain good papers for their respective workshops. (A companion volume, LNCS 1921, contains these papers.) Ling Liu enticed 15 submissions for short tutorials and then had to make hard decisions to cut this list of submissions to the four accepted for the conference. Terry Halpin worked diligently as industrial chair, eventually joining forces with Davida Berger, Vice President of Conference Services for DAMA International, to jointly come up with an outstanding industrial track. Il-Yeol Song continuously provided publicity for the conference, John F. Roddick worked on panels, Scott N. Woodfield handled local arrangements, and Stephen W. Liddle kept the Web pages up to date and artistically presentable. Yiu-Kai (Dennis) Ng (registration), Douglas M. Campbell (socials), Dan Johnson and Jennifer Shadel (treasurers), and several Brigham Young University students also helped make the conference a success.

ER2000 Conference Organization

Conference Chair
David W. Embley, Brigham Young University, USA

Program Co-Chairs
Alberto H. F. Laender, Federal University of Minas Gerais, Brazil
Veda C. Storey, Georgia State University, USA

Workshops Chair
Bernhard Thalheim, Brandenburg Technical University at Cottbus, Germany

Tutorials Chair
Ling Liu, Georgia Institute of Technology, USA

Industrial Chair
Terry Halpin, Microsoft Corporation, USA

Publicity Chair
Il-Yeol Song, Drexel University, USA

Panels Chair
John F. Roddick, Flinders University of South Australia, Australia

DAMA International Liaison
Davida Berger, Roche Labs Inc., USA

Local Organization Committee
Local Arrangements: Scott N. Woodfield, Brigham Young University, USA
Registration: Yiu-Kai (Dennis) Ng, Brigham Young University, USA
Webmaster: Stephen W. Liddle, Brigham Young University, USA
Social: Douglas M. Campbell, Brigham Young University, USA
Treasurer: Dan Johnson, Brigham Young University, USA
Treasurer: Jennifer Shadel, Brigham Young University, USA

Steering Committee Representatives
Chair: Bernhard Thalheim, BTU Cottbus, Germany
Vice-Chair and ER2000 Liaison: Tok Wang Ling, NUS, Singapore
Emeritus: Peter P. Chen, Louisiana State University, USA

Area Liaisons
Asia: Tok Wang Ling, National University of Singapore, Singapore
Australia: John F. Roddick, Flinders University of South Australia, Australia
North America: Sudha Ram, University of Arizona, USA
South America: José Palazzo de Oliveira, Federal University of Rio Grande do

Program Committee

Jacky Akoka, National Conservatory of Industrial Arts and Crafts, France
Hiroshi Arisawa, Yokohama National University, Japan
Paolo Atzeni, University of Rome, Italy
Terry Barron, University of Toledo, USA
Dinesh Batra, Florida International University, USA
Joachim Biskup, University of Dortmund, Germany
Mokrane Bouzeghoub, University of Versailles, France
Marco A. Casanova, Catholic University of Rio de Janeiro, Brazil
Tiziana Catarci, University of Rome, La Sapienza, Italy
Stefano Ceri, Milan Polytechnic, Italy
Roger H.L. Chiang, University of Cincinatti, USA
Wesley Chu, University of California at Los Angeles, USA
Yves Dennebouy, Swiss Federal Institute of Technology, Switzerland
Deb Dey, University of Washington, USA
Ramez Elmasri, University of Texas at Arlington, USA
Donal Flynn, University of Science and Technology in Manchester, UK
Antonio Furtado, Catholic University of Rio de Janeiro, Brazil
Paulo Goes, University of Connecticut, USA
Jean-Luc Hainaut, University of Namur, Belgium
Igor Hawryszkiewycz, University of Technology, Sydney, Australia
Matthias Jarke, Technical University of Aachen, Germany
Paul Johannesson, Stockholm University, Sweden
Yahiko Kambayashi, Kyoto University, Japan
Hideko S. Kunii, Ricoh Co. Ltd., Japan
Maurizio Lenzerini, University of Rome, La Sapienza, Italy
Stephen W. Liddle, Brigham Young University, USA
Ee-Peng Lim, Nanyang Technological University, Singapore
Tok Wang Ling, National University of Singapore, Singapore
Salvatore T. March, Vanderbilt University, USA
Claudia Bauzer Medeiros, University of Campinas, Brazil
Elisabeth Metais, University of Versailles, France
Takao Miura, Hosei University, Japan
Renate Motschnig-Pitrik, University of Vienna, Austria
John Mylopoulos, University of Toronto, Canada
Sham Navathe, Georgia Institute of Tecnnology, USA
Daniel O'Leary, University of Southern California, USA
Antoni Olivé, University of Catalunya, Spain
Maria Orlowska, University of Queensland, Australia
Jose Palazzo de Oliveira, Federal University of Rio Grande do Sul, Brazil
Guenther Pernul, University of Essen, Germany
Javier Pinto, Catholic University of Chile, Chile
Alain Pirotte, Catholic University of Louvain, Belgium

Sudha Ram, University of Arizona, USA
John Roddick, Flinders University of South Australia, Australia
Sumit Sarkar, University of Texas at Dallas, USA
Arne Solvberg, Norwegian Institute of Technology, Norway
Il-Yeol Song, Drexel University, USA
Vijayan Sugumaran, Oakland University, USA
Mohan Tanniru, Oakland University, USA
Toby Teorey, University of Michigan, USA
Bernhard Thalheim, Brandenburg Technical University at Cottbus, Germany
Olga De Troyer, Tilburg University, The Netherlands
Kyu-Young Whang, Korea Advanced Institute of Science and Technology, Korea
Michael Williams, Manchester Metropolitan University, UK

External Referees

Stephen Arnold
Maria Bergholtz
Stephane Bressan
Diego Calvanese
Cecil Eng Huang Chua
Angelo E. M. Ciarlini
Robert M. Colomb
Gillian Dobbie
Laurent Ferier
Thomas Feyer
Renato Fileto
Rakesh Gupta
Hyoil Han
Gaby Herrmann
Patrick Heymans
Ryosuke Hotaka
David Johnson
Masayuki Kameda
Gilbert Karuga
Hirofumi Katsuno
Hiroyuki Kitagawa
Birgitta Koenig-Ries
Takeo Kunishima
Dongwon Lee
Joseph Lee
Weifa Liang
Mihhail Matskin
Giansalvatore Mecca
Paolo Merialdo
Sophie Monties
Luciana Porcher Nedel
Sanghyun Park
Ilias Petrounias
Alysson Bolognesi Prado
Torsten Priebe
Ivan Radev
Srinivasan Raghunathan
Wasim Sadiq
Giuseppe Santucci
Klaus-Dieter Schewe
Torsten Schlichting
Joachim W. Schmidt
Junho Shim
Isamu Shioya
Roderick Son
Eng Koon Sze
Hideyuki Takada
Hiroki Takakura
Babis Theodoulidis
Hallvard Traetteberg
Meng-Feng Tsai
Vassilios S. Verykios

Tutorials

The Unified Modeling Language
Scott N. Woodfield, Brigham Young University, USA

Conceptual Modeling of Internet Sites
Bernhard Thalheim, Brandenburg Univ. of Technology at Cottbus, Germany
Klaus-Dieter Schewe, Massey University, New Zealand

Utilizing Ontologies in eCommerce
Avigdor Gal, Rutgers University, USA

Conceptual Modeling for Multimedia Applications and the Role of MPEG-7 Standards
Uma Srinivasan, CSIRO Mathematical and Information Sciences, Australia
John R. Smith, IBM T.J. Watson Research Center, USA

Logical Design for Data Warehousing
Il-Yeol Song, Drexel University, USA

Application and Process Integration
Benkt Wangler, University of Skövde, Sweden
Paul Johannesson, Stockholm University, Sweden

Workshops

eCOMO2000
International Workshop on Conceptual Modeling Approaches for E-Business
Chair: Heinrich C. Mayr, University of Klagenfurt, Austria

WCM2000
2nd International Workshop on the World Wide Web and Conceptual Modeling
Chair: Stephen W. Liddle, Brigham Young University, USA

See *LNCS vol. 1921* for the workshop proceedings.

Table of Contents

Invited Papers

Database Integration

Temporal and Active Database Modeling

Database and Data Warehouse Design Techniques

Analysis Patterns and Ontologies

Web-Based Information Systems

Business Process Modeling

Conceptual Modeling and XML

Engineering and Multimedia Application Modeling

Object-Oriented Modeling

Applying Object-Oriented Technology

Quality in Conceptual Modeling

Application Design Using UML

DAMA International Industrial Abstracts

23. Briand, L.C., Daly, J.W., Wüst, J.K.: A Unified Framework for Cohesion Measurement in Object-Oriented Systems. Empirical Software Eng., An Int'l J. 3 (1998) 65-117
24. Briand, L.C., Daly, J.W., Wüst, J.K.: A Unified Framework for Coupling Measurement in Object-Oriented Systems. IEEE Trans. Software Eng. 25 (1999) 91-121
25. Snoeck, M.: On a process algebra approach for the construction and analysis of M.E.R.O.DE.-based conceptual models. Ph.D. dissertation. Katholieke Universiteit Leuven (1995)
26. Snoeck, M., Dedene, G.: Existence Dependency: The Key to Semantic Integrity Between Structural and Behavioural Aspects of Object Types. IEEE Trans. Software Eng. 24 (1998) 233-251
27. Booch, G., Rumbaugh, J., Jacobson, I.: The Unified Modeling Language User Guide. Addison-Wesley (1999)
28. Poels, G., Dedene, G.: Measures for Object-Event Interactions. In: Proc. 33rd Int'l Conf. Technology of Object-Oriented Languages and Systems (TOOLS-33). Mont St. Michel, France (2000) 70-81
29. Brito e Abreu, F., Esteves, R., Goulao, M.: The Design of Eiffel Programs: Quantitative Evaluation Using the MOOD Metrics. In: Proc. 20th Int'l Conf. Technology of Object-Oriented Languages (TOOLS-20). Santa Barbara, Calif. (1996)
30. Poels, G.: On the use of a Segmentally Additive Proximity Structure to Measure Object Class Life Cycle Complexity. In: Dumke, R., Abran, A.: Software Measurement: Current Trends in Research and Practice. Deutscher Universitäts Verlag, Wiesbaden, Germany (1999) 61-79
31. Kitchenham, B., Pfleeger, S.L., Fenton, N.: Towards a Framework for Software Measurement Validation. IEEE Trans. Software Eng. 21 (1995) 929-944
32. Zuse, H.: A Framework for Software Measurement. Walter de Gruyter, Berlin (1998)
33. Roberts, F.S.: Measurement Theory with Applications to Decisionmaking, Utility and the Social Sciences. Addison-Wesley (1979)
34. Suppes, P., Krantz, D.M., Luce, R.D., Tversky, A.: Foundations of Measurement: Geometrical, Threshold, and Probabilistic Representations. Academic Press, San Diego, Calif. (1989)
35. Poels, G., Dedene, G.: Distance-based software measurement: necessary and sufficient properties for software measures. Information and Software Technology 42 (2000) 35-46
36. Poels, G., Viaene, S., Dedene, G.: Distance Measures for Information System Reengineering. In: Proc. 12th Int'l Conf. Advanced Information Systems Eng. (CAiSE*00), Stockholm (2000) 387-400
37. Briand, L., Arisholm, E., Counsell, S., Houdek, F., Thévenod-Fosse, P.: Empirical Studies of Object-Oriented Artifacts, Methods, and Processes: State of The Art and Future Directions. Tech. Rep. IESE 037.99/E, Fraunhofer IESE (1999)
38. Benlarbi, S., El Emam, K., Goel, N.: Issues in Validating Object-Oriented Metrics for Early Risk Prediction. In: Proc. 10th Int'l Symposium Software Reliability Eng. (ISSRE'99). Boca Raton, Florida (1999)
39. Basili, V.R., Briand, L., Melo, W.L.: A Validation of Object-Oriented Design Metrics as Quality Indicators. IEEE Trans. Software Eng. 22 (1996) 751-761
40. Briand, L., Daly, J.W., Porter, V., Wüst, J.: A Comprehensive Empirical Validation of Product Measures for Object-Oriented Systems. Tech. Rep. ISERN-98-07, Fraunhofer IESE (1998)
41. Briand, L.C., Morasca, S., Basili, V.R.: Defining and Validating Measures for Object-Based High-Level Design. IEEE Trans. Software Eng. 25 (1999) 722-743
42. Brito e Abreu, F., Melo, W.: Evaluating the Impact of Object-Oriented Design on Quality. In: Proc. 3rd Int'l Software Metrics Symposium (METRICS'96). Berlin (1996)
43. Benlarbi, S., Melo, W.L.: Polymorphism Measures for Early Risk Prediction. In: Proc. 21st Int'l Conf. Software Eng. (ICSE'99). Los Angeles (1999) 334-344

Measuring the Quality of Entity Relationship Diagrams

Marcela Genero [1], Luis Jiménez [2], and Mario Piattini [1]

[1]Grupo ALARCOS
[2]Grupo ORETO
University of Castilla-La Mancha
Ronda de Calatrava, 5
13071, Ciudad Real (Spain)
E-mail: {mgenero, ljimenez, mpiattin}@inf-cr.uclm.es

Abstract. Database quality depends greatly on the accuracy of the requirement specification and the greatest effort should focus on improving the early stages of database life cycle. Conceptual data models form the basis of all later design work and determine what information can be represented by a database. So, its quality has a significant impact on the quality of the database which is ultimately implemented. In this work, we propose a set of metrics for measuring entity relationship diagram complexity, because in today's database design world it is still the dominant method of conceptual modelling. The early availability of metrics allows designers to measure the complexity of entity-relationship diagrams in order to improve database quality from the early stages of their life cycle. Also we carried out a controlled experiment in order to analyse the existent relationships between each of the proposed metrics and each of the maintainability sub-characteristics. In order to analyse the obtained empirical data we propose a novel data analysis technique based on fuzzy regression trees.

1 Introduction

Database quality depends greatly on the accuracy of the requirement specification and the greatest effort should be focus on improving the early stages of database life cycle. In a typical database design a conceptual data model which specifies the requirements about the database is first built. The conceptual data model determines what information can be represented by a database [1]. So, its quality have a significant impact on the quality of the database which is ultimately implemented [2], and an even greater impact if we take into account the size and complexity of current databases. Improving the quality of conceptual data models will therefore be a major step towards the quality improvement of the database development. We will focus on entity-relationship (ER) diagrams because in today's database design world it is the dominant method of conceptual modelling [3].

In practice, evaluation of the quality of conceptual data models takes place in an *ad hoc* manner, if at all. There are no generally accepted guidelines for evaluating the quality of data models, and little agreement even among experts as to what makes a “good” data model [4].

In general we agree with Krogstie et al. [5] in the sense that "Most literature provides only bread and butter lists of useful properties without giving a systematic structure for evaluating them". Moreover these lists are mostly unstructured, use imprecise definitions, often overlap, and properties of models are often confused with language method properties [6]. In addition to this, these lists are not generally sufficient to ensure quality in practice, because different people will have different interpretations of the same concept. It is necessary to have quantitative and objective measures to reduce subjectivity and bias in the evaluation process.

Recently, some frameworks have been proposed which attempt to address quality in conceptual modelling in a much more systematic way [4-7]. The only papers that propose metrics for conceptual data models are those proposed by Eick [8], Gray et al. [9], Moody [10] and Kesh [11.

Although all of this metric proposal is a good starting point to think about quality in conceptual modelling in a numeric scale, most of them are open-ended, subjective and lack empirical and theoretical validation. Thus, there is a need for metrics and quality models that can be applied in the early stages of database design, and we are particularly concerned with that applied to ER diagrams, to ensure that that designs have favorable internal properties that will lead to the development of quality databases.

Before thinking about how to measure the quality of ER diagrams it is essential to define a quality model. This quality model must describe each of the characteristics that compose the concept of "quality". The ISO 9126 [12] proposed a quality model that can be applied to any artifact produced at any stages of the software development life cycle. To our knowledge, not all of the characteristics proposed in that standard are suitable for ER diagrams. We focus our work on one of the most important quality characteristics, maintainability. Maintainability is influenced by the following sub-characteristics:

- Understandability: the ease with which the conceptual data model can be understood.
- Legibility: is the ease with which the conceptual data model can be read, with respect to certain aesthetic criteria [13].
- Simplicity: means that the conceptual data model contains the minimum number of constructions possible.
- Analysability: the capability of the conceptual data model to be diagnosed for deficiencies or for parts to be modified.
- Modifiability: the capability of the conceptual data model to enable a specified modification to be implemented.
- Stability: the capability of the conceptual data model to avoid unexpected effects from modifications.
- Testability: the capability of the conceptual data model to enable modifications to be validated.

As most of these maintainability sub-characteristics are in turn influenced by complexity [14], our objective in this work is to provide a set of metrics for measuring ER diagram complexity (section 2).

As in other aspects of Software Engineering, proposing techniques and metrics is [illegible]

We have made a theoretical validation of some of the proposed metrics following Briand framework [19] in [20] (Briand et al., 1996) and Zuse's framework [21] in [22]. But, we can conclude that in software measurement there is not an agreement of how to make theoretical validation. Most authors have opposing ideas of which is the best way of making a theoretical validation.

In order to empirically validate the proposed metrics we will show in section 3 how we have performed a controlled experiment. The main objective of this experiment is to ascertain the existent relationships between each of the proposed metrics and each of the maintainability sub-characteristics. In order to analyse the obtained empirical data we propose a novel data analysis technique based on fuzzy regression trees. The results demonstrate what metrics are more relevant taking into account each of the sub-characteristic mentioned above. Finally the last section draws on our conclusions, and presents our future work.

2 Metrics for ER Diagram Complexity

As complexity is a multidimensional attribute it is not advisable to try to combine different aspects of this attribute into a single measurement unless you have a model or theory to support you [23]. Henderson-Sellers [24] distinguishes three types of complexity, among which he quoted "product or structural complexity", which is our focus when we refer to the concept of complexity.

In this section we will define a set of closed-ended metrics [25] to measure ER diagram complexity, taking into account its constituent parts, such as entities, attributes and relationships. These closed-ended metrics are more useful, because they are bounded (in this case in the interval [0,1]), they can be easily visualised by graphics, and their values are easily interpreted.

2.1 RvsE Metric

This metric measures the relation that exists between the number of relationships and the number of entities in an ER diagram.

We define this metric as follows:

$RvsE = \left(\frac{N^R}{N^R + N^E} \right)^2$	N^R is the number of relationships in the ER diagram. N^E is the number of entities in the ER diagram. Being $N^R + N^E > 0$.

When we calculate the number of relationships (N^R), we consider the IS_A relationships and the aggregation relationship. In the case, of IS_A relationships we consider one relationship for each child-parent pair. In aggregation relationships we consider one relationship for each part-whole pair. The number of relationships per entity in the ER diagram influences this metric. Intuitively, the greater the number of relationships the greater the complexity. RvsE metric is zero when there are no relationships, it takes the value 1 when there are a lot of relationships and very few

2.2 EAvsE Metric

This metric measures the relations that exist between the number of entity attributes and the number of entities in an ER diagram.

We define this metric as follows:

$EAvsE = \left(\frac{N^{EA}}{N^{EA} + N^{E}} \right)^2$	N^{EA} is the number of entity attributes within the ER diagram. N^E is the number of entities within the ER diagram. Being $N^{EA} + N^E > 0$.

When we calculate the number of entity attributes in the ER diagram (N^{EA}) we also consider composite and multivalued attributes (each of them is considered to take the value 1). The number of attributes per entity in the ER diagram influences this metric. Intuitively, the greater the number o attributes the greater their complexity.

EAvsE metric is zero when there are no entity attributes, it takes value 1 when there are a lot of entity attributes and very few entities.

2.3 RAvsR Metric

This metric measures the relations that exist between the number of relationship attributes and the number of relationships in an ER diagram.

We define this metric as follows:

$RAvsR = \left(\frac{N^{RA}}{N^{RA} + N^{R}} \right)^2$	N^{RA} is the number of relationship attributes within the ER diagram. N^R is the number of relationships within the ER diagram. Being $N^{AR} + NR > 0$.

When we calculate the number of relationship attributes in the ER diagram (N^{EA}) we also consider composite and multivalued attributes (each of them is considered to take the value 1).

In this metric when we calculate the number of relationships (N^R), we disregard IS_A relationships and aggregation relationships.

The number of attributes per relationship in the ER diagram influences this metric. Intuitively, the greater the number of attributes the greater their complexity.

RAvsR metric is zero when there are no relationship attributes, it takes value 1 when there are a lot of relationship attributes and very few relationships.

2.4 M:NRel Metric

The M:N Relationships metric measures the number of M:N relationships compared with the number of relationships in an ER diagram.

We define this metric as follows:

$M:N\,Rel = \frac{N^{M:NR}}{N^R}$	$N^{M:NR}$ is the number of M:N relationships within the ER diagram. N^R is the number of relationships within the ER diagram. Being $N^R > 0$.

In this metric when we calculate the number of relationships (N^R), we disregard IS_A relationships and aggregation relationships.

M:NRel metric scores zero when there are no M:N relationships, it scores 1 when there is a high percentage of M:N relationships.

2.5 1:NRel Metric

The 1:N Relationships metric measures the number of 1:N relationships (also include 1:1 relationships) compared with the total number of relationships in an ER diagram.

We define this metric as follows:

$1:N\,Rel = \frac{N^{1:NR}}{N^R}$	$N^{1:NR}$ is the number of 1:N relationships in the ER diagram. N^R is the number of relationships in the ER diagram. Being $N^R > 0$.

In this metric when we calculate the number of relationships (N^R), we disregard IS_A relationships and aggregation relationships.

1:NRel metric scores zero when there are no 1:N relationships, it scores 1 when there is a high percentage of 1:N relationships.

2.6 N-aryRel Metric

The N-ary Relationships metric measures the number of N-ary relationships (not binary) compared with the number of relationships in the ER diagram.

It is convenient for the number of N-ary relationships in an ER diagram to be minimal, because they contribute to increasing its complexity.

We define this metric thus:

$N\text{-}ary\,Rel = \frac{N^{N\text{-}aryR}}{N^R}$	$N^{N\text{-}aryR}$ is the number of N-ary relationships in the ER diagram. N^R is the number of relationships in the ER diagram. Being $N^R > 0$.

In this metric when we calculate the number of relationships (N^R), we disregard IS_A relationships and aggregation relationships.

This metric is zero when the ER diagram has no N-ary relationships, it takes the value 1 when all of the relationships are N-ary.

2.7 BinaryRel Metric

The Binary Relationships metric measures the number of Binary relationships compared with the number of relationships in the ER diagram.

We define this metric thus:

$Binary\ Rel = \frac{N^{BinaryR}}{N^{R}}$	$N^{N\text{-}aryR}$ is the number of Binary relationships in the ER diagram. N^R is the number of relationships in the ER diagram. Being $N^R > 0$.

In this metric when we calculate the number of relationships (N^R), we disregard IS_A relationships and aggregation relationships.

This metric is zero when the ER diagram has no Binary relationships, it takes the value when all of the relationships are binary.

3 Empirical Validation of the Proposed Metrics

We are interested in ascertaining if any relationship exists between each of the proposed metrics and each of the maintainability sub-characteristics: understandability, legibility, simplicity, analysability, modifiability, stability, and testability. With that objective, we have carried out and controlled experiment.

In the remaining of this section, we will present: the experimental design, how we collect the experimental data, the technique used to analyse the empirical data, and the results of the experiment.

3.1 Experimental Design and Data Collection

Sixteen subjects (database designers) participated in the experiment. The subjects were given twenty four ER diagrams taken from different books of database design [26-27]. Each diagram have enclosed a form which includes the description of maintainability sub-characteristics, such as: understandability, legibility, simplicity, analysability, modifiability, stability, and testability. Each subject has to rate each sub-characteristic using a scale consisting of seven linguistic labels. For example for the sub-characteristic understandability we proposed the following linguistic labels:

Extremely difficult to understand	Very difficult to understand	A bit difficult to understand	Neither difficult nor easy to understand	A bit easy to understand	Very easy to understand	Extremely easy to understand

After collecting the data, we assign to each label a number in the following way: The greater the expert´s rate the greater the difficulty to understand.

Extremely difficult to understand	Very difficult to understand	A bit difficult to understand	Neither difficult nor easy to understand	A bit easy to understand	Very easy to understand	Extremely easy to understand
7	6	5	4	3	2	1

All of the proposed metrics were automatically calculated using our metric tool called MANTICA [28].

At this point we have all the experimental data. The next step is to analyse them, which we will explain in the next subsection.

3.2 Data Analysis Technique and Results

Due to the nature of the software development process and products, one cannot expect to use in Software Engineering the same measurement data analysis techniques that are used in "exact" sciences, e.g., Physics, Chemistry, nor obtain the same degree of precision and accuracy [29]. So that, we will analyse the empirical data with a data analysis technique based on fuzzy regression trees with linguistic variables [30]. This approach provide models that allow us to discover the most relevant conceptual relationships between the data we are analysing, where the accuracy of this models is sacrificed in favour of its simplicity and easiness to understand.

In the next subsection we will describe the theoretical grounds and the methodology for building the regression fuzzy trees.

Induction Method for Building Fuzzy Regression Trees. This induction method is a generalisation of the classical regression approach.

Let $S=\{s_1, \dots, s^n\}$ be a set of data, which are defined by the value given by a set of variables $X=\{X^1, \dots, X^d, Y\}$, $s_i=(x^1_i, \dots, x^d_i, y_i)$. Let's suppose there is a function F which is only known at the points of S so that $F(s_i)=y_i$.. The objective of regression, which is defined classically as a parametric function F', is to minimise the distance between the sample output values y_i and the predicted value $F(s_i)$.

The regression problem differs from the classification problem in that the output variable can be continuous, where as in classification this is strictly categorical. From this perspective, classification can be thought to be a subcategory of regression. Recursive partition methods for classification problem such as ID3 decision tree have been applied, as a regression method by restrictions, in the CART program [31], developed in statistical research community.

The program CART (classification and regression tree) is based on the building of a tree structure where the regions are defined by possible answers to a set of questions raised about the variables that define the problem. This approach creates a set of disjoint regions $SR=\{r_1, \dots, r_p\}$ of the global domain of problem. These partitions are obtained according to the kind of questions and answers that we have formulated. Our method generalize the obtained kind of partitions, using the method proposed by Linares et al. [32] based on the fuzzy set theory [33]. We use Linares et al.´s idea [32] in order to built a fuzzy regression tree (FCART).

Hereafter, we will explain how to obtain the fuzzy regression tree. Let A_T be a fuzzy set defined over the set S of the node T, such as $A_T : S \rightarrow [0,1]$. In the root node this fuzzy is $A_{root} : S \rightarrow 1$. We define as output value at node T by the membership value of each point s_i, AT (s_i) and the output value in this point yi.

$$F''(T) = \frac{\sum_{i=1}^{n} A_T(s_i)^m * y_i}{\sum_{i=1}^{n} A_T(s_i)^m}$$

The error estimated is defined by

$$Err(T) = \frac{\sum_{i=1}^{n} (F''(T) - y_i)^2 * A_T(s_i)^m}{\sum_{i=1}^{n} A_T(s_i)^m}$$

Now, our problem is how to create the set of questions for the node T. The questions were formulated for each variable, obtaining a binary fuzzy partition for every one. We supposed a binary fuzzy partition for the fuzzy set of node T by the fuzzy set A^j_T of variable j, this is $Q^j_T = \{B(x), C(x)\}$. This partition originates two new nodes and two new fuzzy sets for them

$$A_{T_1}(s_i) = min(A_T(s_i), B(x_i^j))$$

$$A_{T_2}(s_i) = min(A_T(s_i), C(x_i^j))$$

IF variable^j IS A_{T_1} THEN .....

ELSE IF variable^j IS A_{T_2} THEN ..

Following the CART program, we defined the proportion for B(x) and C(x) in relation to fuzzy set A^j_T as

$$P(T_1) = \frac{\sum_{i=1}^{n} B(x_i^j)}{\sum_{i=1}^{n} A_T^j(x_i^j)}$$

The quality of the partition can be estimated through the equation

$$C(T, p^j) = Err(T_1) * P(T_1) + Err(T_2) * P(T_2)$$

where p^j is the fuzzy partition of variable j.

We will select the p^* which minimises the value of C(T, p^j). This approach to create the questions originates hierarchical fuzzy partition for each variable. The partition process stops when a stop criterion is raised. In this case we look for larger estimated error. So that stop criterion can be

$$ERROR = \max_{T \in \overline{T}} Err(T) \leq \varepsilon$$

Where $\overline{T}$ is the set of leaf nodes of the tree, and each one represent a region for our solution. The output value y_i' for a input value s_i is

$$F'(s) = \frac{\sum_{T \in \overline{T}} A_T(s)^m * F''(T)}{\sum_{T \in \overline{T}} A_T(s)^m}$$

3.3 Experiment Results

Due to the sake of brevity in this paper we used the induction method shown in the previous subsection, to show only the existent relationships between our metrics and the maintainability sub-characteristic, understandability. For the other sub-characteristics the process is similar.

We establish a fuzzy regression tree for the maintainability sub-characteristic: understandability. We use a learning set with X={set of our metrics} and Y={the values of understandability obtained in our experiment}. We have obtained 384 values (24 ER diagrams and 16 subjects) of this unknown function:

F(RvsE; EAvsE; RavsR; M:Nrel; 1:Nrel; N-aryRel; BinaryRel) = Understandability.

The obtained fuzzy sets, which define the fuzzy regression tree (see fig. 1), are labelled by a set of linguistic labels. This labelling process allows us to abstract the numeric values for building a conceptual linguistic model which is highly qualitative and more closed to human minds.

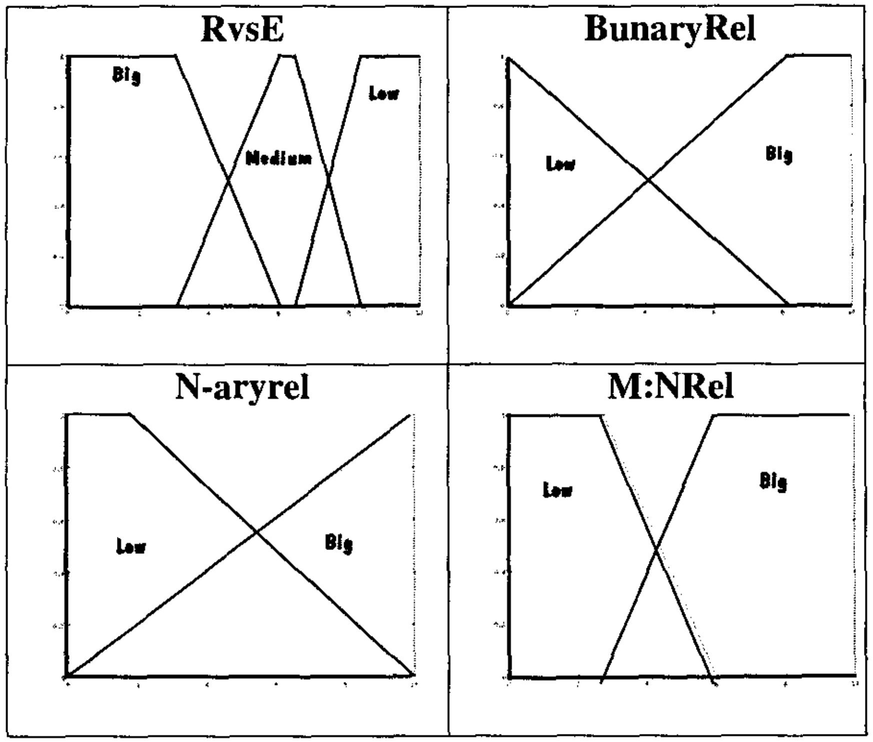

Fig. 1. Labels of fuzzy sets

We only show in fig. 1 the fuzzy representation of the values of RvsE, BinaryRel, N-aryRel and M:NRel metrics because after applying the induction method we have found that the others are not relevant for the understandability.

The obtained fuzzy regression tree is shown in fig. 2 by a nested structure if-then-else.

```
TREE
IF RvsE IS BIG THEN UNDERSTANDABILITY IS 6
ELSE IF RvsE IS MEDIUM THEN
            IF BinaryRel is LOW THEN UNDERSTANDABILITY IS 3
            ELSE IF N-aryRel IS BIG THEN UNDERSTANDABILITY IS 3
                 ELSE IF M:NRel IS LOW THEN UNDERSTANDABILITY IS
                     5
                     ELSE UNDERSTANDABILITY IS 4
     ELSE   IF BinaryRel is LOW THEN UNDERSTANDABILITY IS 2
            ELSE IF N-aryRel IS BIG THEN UNDERSTANDABILITY IS 3
                 ELSE IF M:NRel  IS LOW THEN UNDERSTANDABILITY
                     IS 3
                     ELSE UNDERSTANDABILITY IS 4.
```

Fig. 2. A Fuzzy Regression Tree

As we can see, when RvsE is "big" then understandability is 6 (the model is "very difficult to understand"), when RvsE is "medium" then understandability value ranges between 3 and 5 (the model is "a bit easy to understand" – the model is a "bit difficult to understand"), and when RvsE is "low" then understandability value ranges between 2 and 4 (the model is "a very easy to understand" – the model is a "neither difficult nor easy to understand"). This result show that understandability is very close at RvsE value and that Rbin, Rter and MNRel show more local value for this characteristic. Good values of understandability are assigned to low value of RvsE and in opposition bad values are assigned to big values of RvsE.

4 Conclusions

Due to the growing complexity of actual databases, continuous attention to and assessment of the conceptual data models is necessary to produce quality databases. Following this idea, we have presented a set of objective and automatically computed metrics for evaluating ER diagram complexity.

We want to highlight that our proposal cannot be considered as a final proposal. Instead, it is only a starting point and we require feedback to improve it. However, due to the lack of objective metrics for measuring conceptual model quality, it serves the purpose of getting database designers to think about the quality of their conceptual data models in numeric terms.

We have presented an induction method for building fuzzy regression trees that allow us to build models for ascertaining the existent relationships between the proposed metrics and the ER diagram maintainability sub-characteristics. So, for example, the metric RvsE could be used as an understandability indicator. This induction method is highly qualitative and very closed to human mind, and above all very ease to understand. Moreover, the models built through this induction technique can also be used as a prediction model by using approximate reasoning [34].

Although the results obtained in this experiment are encouraging, we are aware that it is necessary to undertake more experiments, and also more controlled ones when different values of all the metrics are considered in different conceptual models. Also "real" case studies taken from enterprises must be carried out, with the objective of assessing these metrics as predictors of maintenance efforts, and therefore, determining whether they can be used as early quality indicators.

We cannot disregard the increasing diffusion of the object-oriented paradigm in conceptual modelling. We think that object oriented models are more appropriate than ER diagrams to describe the kind of information systems built nowadays. We have also been working on metrics for measuring OMT [35] class diagrams [36] and also UML [37] class diagrams [38].

In our knowledge, few works have been done towards measuring models that capture the dynamic aspects of an OO software systems [39-40]. As is quoted in [41] this is an area which need further investigation. So that, as a future work we will define metrics for UML dynamic diagrams, such as state diagrams or activity diagrams.

Furthermore, we will not only address the sub-characteristics of maintainability, we also have to focus our research towards measuring other quality factors as proposed in the ISO 9126[12]. Also we will focus on dynamic aspects of object

Acknowledgements

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

References

1. Feng, J. The "Information Content" problem of a conceptual data schema and a possible solution. Proceedings of the 4^{th} UKAIS Conference: Information Systems-The Next Generation, University of York, (1999) 257-266
2. Shanks, G. and Darke, P. Quality in Conceptual Modelling: Linking Theory and Practice. Proc. of the Pacific Asia Conference on Information Systems (PACIS'97), Brisbane, (1997) 805-814
3. Muller, R. Database Design For Smarties: Using UML for Data Modeling. Morgan Kaufman, (1999)
4. Moody, D., Shanks, G. and Darke, P. Improving the Quality of Entity Relationship Models – Experience in Research and Practice. Proceedings of the Seventeenth International Conference on Conceptual Modelling (ER ´98), Singapore, (1998) 255-276
5. Krogstie, J., Lindland, O. and Sindre, G. Towards a Deeper Understanding of Quality in Requirements Engineering, Proceedings of the 7^{th} International Conference on Advanced Information Systems Engineering (CAISE), Jyvaskyla, Finland, June, (1995) 82-95
6. Lindland, O., Sindre, G. and Solvberg, A. Understanding Quality in Conceptual Modelling. IEEE Software, March, Vol. 11 N° 2, (1994) 42-49
7. Schuette, R. and Rotthowe, T. The Guidelines of Modeling – An Approach to Enhance the Quality in Information Models. Proceedings of the Seventeenth International Conference on Conceptual Modelling (ER ´98), Singapore, November 16-19, (1998) 40-254
8. Eick, C. A Methodology for the Design and Transformation of Conceptual Schemas. Proc. of the 17^{th} International Conference on Very Large Data Bases. Barcelona (1991)
9. Gray, R., Carey, B., McGlynn, N. and Pengelly A. Design metrics for database systems. BT Technology, Vol. 9 N° 4, (1991) 69-79
10. Moody, D. Metrics for Evaluating the Quality of Entity Relationship Models. Proceedings of the Seventeenth International Conference on Conceptual Modelling (ER ´98), Singapore, November 16-19, (1998) 213-225
11. Kesh, S. Evaluating the Quality of Entity Relationship Models. Information and Software Technology, Vol. 37 N° 12, (1995) 681-689.
12. ISO/IEC 9126-1. Information technology- Software product quality – Part 1: Quality model. (1999)
13. Batini, C., Ceri, S. and Navathe, S. *Conceptual database design. An entity relationship approach.* Benjamin Cummings Publishing Company. (1992)
14. Li, H. and Cheng, W. An empirical study of software metrics. IEEE Transactions on Software Engineering, Vol. 13 N° 6, (1987) 679-708

15. Fenton, N. and Pfleeger, S. Software Metrics: A Rigorous Approach. 2^{nd}. edition. Chapman & Hall, London (1997)
16. Kitchenham, B., Pflegger, S. and Fenton, N. Towards a Framework for Software Measurement Validation. IEEE Transactions of Software Engineering, Vol. 21 N° 12, (1995) 929-943
17. Schneidewind, N. Methodology For Validating Software Metrics. IEEE Transactions of Software Engineering, Vol. 18 N° 5, (1992) 410-422
18. Basili, V., Shull, F. and Lanubile, F. Building knowledge through families of experiments. IEEE Transactions on Software Engineering, Vol. 25 N° 4, (1999) 435-437
19. Briand, L., Morasca, S. and Basili, V.. Property-Based Software Engineering Measurement. IEEE Transactions on Software Engineering, Vol. 22 N° 6, (1996) 68-86
20. Genero, M., Piattini, M. and Calero, C. (2000). Formalization of Metrics for Conceptual Data Models. UKAIS 2000. Cardiff, 26-28 April, (2000) 99-119
21. Zuse, H. A Framework of Software Measurement. Walter de Gruyter, Berlin (1998)
22. Genero, M., Piattini, M., Calero, C. Serrano, M. (2000). Measures to get better quality databases. ICEIS 2000. Stafford, 4-7 July, (2000) 49-55
23. Fenton, N. Software Measurement: A Necessary Scientific Basis. IEEE Transactions on Software Engineering, Vol. 20 N° 3, (1994) 199-206
24. Henderson-Sellers, B. Object-oriented Metrics - Measures of complexity. Prentice-Hall, Upper Saddle River, New Jersey. (1996)
25. Lethbridge, T. Metrics For Concept-Oriented Knowledge bases. International Journal of Software Engineering and Knowledge Engineering Vol. 8 N° 2, (1998) 161-188
26. Ruiz, I. and Gómez-Nieto, M. Diseño y uso de Bases de Datos Relacionales. Ra-Ma, (1997) (in spanish)
27. De Miguel, A. and Piattini, M. Fundamentos y Modelos de Bases de Datos. Ra-Ma. (1997) (in spanish)
28. Calero, C., Pascual, C., Serrano, M. and Piattini, M. Measuring Oracle Database Schema. Computers and Computational Engineering in Control, (Cap. 42), World Scientific Engineering Society, (1999) 237-243
29. Morasca, S. and Ruhe, G. Guest Editors´Introduction: Knowledge Discovery From Empirical Software Engineering Data. International Journal of Software Engineering and Knowledge Engineering, Vol. 9 N° 5, (1999) 495-498
30. Zadeh, L. The Concept of Linguistic Variable and its Applications to Approximate Reasoning Part I. Information Sciences, Vol. 8, (1973) 199-249.
31. Breiman, L., Friedman, J., Olshen, R. and Stone, C.. Classification and Regression Trees. Wadsworth, Belmont, CA, (1984)
32. Linares, L., Delgado, M. and Skarmeta, A. Regression by fuzzy knowledge bases. Proceedings of the 4^{th} European Congress on Intelligent Techniques and soft computing. Aachen, Germany, September, (1996) 1170-1176
33. Zadeh, L. Fuzzy sets. Information and control, (1965) 338-353.
34. Sugeno, M. An Introductory Survey of Fuzzy Control. Information Sciences, Vol. 36, (1985) 59-83.
35. Rumbaugh, J., Blaha M., Premerlani, W., Eddy, F., and Lorensen, W. Object-Oriented Modeling and Design. Prentice Hall, USA. (1991)

36. Genero, M., Manso, Mª E., Piattini, M. and García, F. Assessing the Quality and the Complexity of OMT Models. 2nd European Software Measurement Conference - FESMA 99, Amsterdam, The Netherlands, (1999) 99-109
37. Booch, G., Rumbaugh, J. and Jacobson, I. The Unified Modeling Language User Guide. Addison-Wesley, (1998)
38. Genero, M., Piattini, M. and Calero, C. 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, (2000) 373-384 (in spanish)
39. Poels, G. On the use of a Segmentally Additive Proximity Structure to Measure Object Class Life Cycle Complexity. Software Measurement: Current Trends in Research and Practice. Deutscher Universitäts Verlag, (1999) 61-79
40. Poels, G. On the Measurement of Event-Based Object-Oriented Conceptual Models. 4th International ECOOP Workshop on Quantitative Approaches in Object-Oriented Software Engineering, June 13, Cannes, France. (2000)
41. Brito e Abreu, F., Zuse, H., Sahraoui, H. and Melo, W. Quantitative Approaches in Object-Oriented Software Engineering. Object-Oriented technology: ECOOP'99 Workshop Reader, Lecture Notes in Computer Science 1743, Springer-Verlag, (1999) 326-337.

Behavior Consistent Inheritance in UML

Markus Stumptner[1] and Michael Schrefl[2]*

[1] Institut für Informationssysteme, Technische Universität Wien,
mst@dbai.tuwien.ac.at
[2] School of Computer and Information Science, University of South Australia,
cismis@cs.unisa.edu.au

Abstract

Object-oriented design methods express the behavior an object exhibits over time, i.e., the object life cycle, by notations based on Petri nets or state charts. The paper considers the specialization of life cycles via inheritance relationships as a combination of extension and refinement, viewed in the context of UML state machines. Extension corresponds to the addition of states and actions, refinement refers to the decomposition of states into substates. We use the notions of observation consistency and invocation consistency to compare the behavior of object life cycles and present a set of rules to check for behavior consistency of UML state machines, based on a one-to-one mapping of a meaningful subset of state machines to Object/Behavior Diagrams.

1 Introduction

Object-oriented design methodologies such as OMT [15], OOSA [4], OOAD [2], OBD [6], and UML [16] differ from programming languages in that they represent the behavior of an object type not merely by a set of operations, which may be performed on instances of the object type, but instead provide a higher-level, overall picture on how instances of the object type may evolve over their lifetime, e.g., a hotel or car reservation, which must be requested before it can be issued and can be used only after being issued. Such a description is often referred to as "object life cycle". In this paper we consider a special variant of state charts, the state chart diagrams of the Unified Modeling language (UML), which has recently emerged as a prominent commercial object-oriented design notation. The behavior of an UML object class is represented by a state chart, which consists of states and state transitions. When mapped to OODB designs, transitions are mapped to operations and states to pre- and postconditions of these operations.

Object-oriented systems organize object types in hierarchies in which subtypes inherit and specialize the structure and the behavior of supertypes. These inheritance hierarchies provide a major aid to the designer in structuring the description of an object-oriented system, and they guide the reader who tries to understand the system by pointing out similarities between object types that are so connected. Informally, specialization means for object life cycles that the