



UNIVERSITI PUTRA MALAYSIA

**DEVELOPMENT OF AN ASSERTION MODEL OF INTEGRITY
CONSTRAINTS IN OBJECT-ORIENTED DATABASES**

BELAL MOHAMMAD ZAQAIBEH

FSKTM 2006 11

**DEVELOPMENT OF AN ASSERTION MODEL OF INTEGRITY
CONSTRAINTS IN OBJECT-ORIENTED DATABASES**

By

BELAL MOHAMMAD ZAQAIBEH

**Thesis Submitted to the School of Graduate Studies, Universiti Putra Malaysia,
in Fulfilment of the Requirements for the Degree of Doctor of Philosophy**

June 2006



DEDICATION

To the memory of my Grandfather,

To my Parents: Mohammad and Ne'mat,

To my Wife: Maisa and my Son: Mohammad,

To my Brothers and Sisters.

Belal



Abstract of thesis presented to Senate of Universiti Putra Malaysia in fulfilment of the requirement for the degree of Doctor of Philosophy

**DEVELOPMENT OF AN ASSERTION MODEL OF INTEGRITY
CONSTRAINTS IN OBJECT-ORIENTED DATABASES**

By

BELAL MOHAMMAD ZAQAIBEH

June 2006

Chairman: Associate Professor Hamidah Ibrahim, PhD

Faculty: Computer Science and Information Technology

Object-Oriented Databases (OODBs) have been designed to support large and complex programming projects. The data accuracy, consistency, and integrity in OODBs are extremely important for developers and users. Checking the integrity constraints in OODBs is a fundamental problem in database design. Existing OODB Management Systems (OODBMSs) lack to a capability of an ad-hoc declarative specification of enforcing and maintaining integrity constraints that are appeared among attributes in association, composition, and inheritance hierarchies' relationships.

A critical problem in the existing OODBs is that they cannot support User-Defined Constraints (UDCs) that can be defined in classes with composition (logical or physical composition) and inherence (single or multiple inheritance) hierarchies. Integrity constraints in the current



OODBMSs are maintained either by disallowing and rolling back transaction or modifying operations that may produce a violation. The constraints must be maintained in the backward direction along the class composition hierarchy as well as in the forward direction.

In this work an Assertion Model of Integrity Constraints (AMIC) is proposed. The AMIC keeps the derivation path along with the attributes' relationships that are derived from association, composition, and inheritance hierarchies. The AMIC techniques are designed to implement the needed functions that are collecting the attributes' relationships and checking the integrity constraints. Moreover, AMIC keeps UDCs with their relationships in both single classes and multilevel classes (intra-class and inter-class). Furthermore, the AMIC can maintain constraints in a single object and a set of distributed objects (intra-object and inter-object). Therefore, this makes the new model extendable and can be integrated with any existing constraints' service.

A new technique called Detection Method (DM) is designed to check the Object Meta Data (OMD) to detect the constraints violation before it occurs. The AMIC is designed for both Centralized Integrity Maintenance (CIM) and Application-Oriented Integrity Maintenance (AOIM). The AMIC can also enforce and maintain structural and logical integrity constraints, in addition to enforce and maintain redundant, inconsistent, and duplicate constraints.

Abstrak tesis yang dikemukakan kepada Senat Universiti Putra Malaysia
sebagai memenuhi keperluan untuk ijazah Doktor Falsafah

**PEMBANGUNAN MODEL PENERAPAN KEKANGAN INTEGRITI
DALAM PANGKALAN DATA BERORIENTASI OBJEK**

Oleh

BELAL MOHAMMAD ZAQAIBEH

Jun 2006

Pengerusi: Profesor Madya Hamidah Ibrahim, PhD

Fakulti: Sains Komputer dan Teknologi Maklumat

Sistem pangkalan data berorientasikan objek (OODB) direka untuk menyokong projek pemrograman yang kompleks dan berskala besar. Ciri-ciri seperti keketepatan data, konsistensi pangkalan data serta kewibawaan data dalam OODB amat penting bagi pengguna dan pembangun perisian. Penyediaan kewibawaan dalam OODB merupakan dilema asas yang dihadapi oleh pereka pangkalan data. Ini disebabkan sifat terwujud OODB yang tidak berupaya untuk memproses spesifikasi deklaratif yang bertujuan mengekalkan kewibawaan konstren-konstren yang ada kaitan dengan hierarki persekutuan, komposisi dan pewarisan. Sifat sedia ada OODB inilah yang menyebabkan ialah ia tidak berupaya untuk menyokong kekangan atau konstren yang tertakrif dalam komposisi sesebuah

kelas (komposisi secara logik mahupun fizikal) ataupun pewarisan (tunggal mahupun berganda) dengan sebaiknya.

Buat masa ini, konstren kewibawaan dalam OODB disokong melalui penggulungan balik sesuatu transaksi, tidak membenarkan transaksi tersebut ataupun melalui pengubahsuaian operasi tersebut. Cara-cara sedemikian besar kemungkinan boleh menyebabkan keadaan pangkalan data menjadi tidak konsisten kerana pemeliharaan konstren dilakukan dalam dua arah bertentangan (hadapan dan belakang) di dalam hierarki komposisi kelas.

Dalam tesis ini, Model Penerapan Kewibawaan Konstren (AMIC) dicadangkan untuk menyokong konstren takrifan pengguna. AMIC mengekalkan laluan penerbitan serta sifat perhubungan yang diperoleh dari hierarki persekutuan, komposisi dan pewarisan. AMIC digunakan untuk melaksana fungsi-fungsi yang mengumpul sifat perhubungan serta memeriksa kewibawaan konstren. AMIC juga mengekalkan konstren serta hubungan mereka dalam kelas tunggal dan berganda (antara dan intra kelas). Selebihnya, AMIC juga dapat mengekalkan hubungan sedia ada di dalam sesebuah objek dan juga antara objek-objek tertabur. Justeru itu, prestasi akan bertambah baik dan membolehkan model ini diluaskan dan disepadukan dengan sebarang perkhidmatan konstren yang sedia ada.

Teknik baru yang dinamakan Kaedah Pengesanan (DM) digunakan untuk mengenalpasti Objek Meta Data (OMD) bagi tujuan mengesan

pencabulan terhadap pangkalan data sebelum ia berlaku. AMIC di cipta untuk menyokong kewibawaan terpusat dan juga kewibawaan berorientasikan aplikasi. AMIC juga dapat mengekalkan kewibawaan struktur dan logik untuk model-model yang berasaskan OODB. AMIC juga dapat mengekalkan konstren yang lewah, tak konsisten dan salinan.

ACKNOWLEDGEMENTS

In the name of ALLAH, I heartiest would like to thank my supervisor Associate Professor Dr. Hamidah Ibrahim, for her incredible guidance, continuous support, and encouragement. Always having time for me and readily providing her technical expertise. I owe more than I can ever repay, her valuable directions and suggestions are very helpful in my research. Only has the successful completion of this work become possible due to her supervision, she is the first person to thank for making my PhD. program at the Universiti Putra Malaysia a very enjoyable experience. Her high stand of diplomatic power and professionalism set a great model for me to follow.

To my thesis committee members, Associate Professor Dr. Hj. Ali Mamat and Associate Professor Dr. Hj. Md. Nasir Sulaiman, I would like to express appreciation for their insightful comments, questions, criticisms, and suggestions on the work. Their critical appraisals of my papers and presentations are extremely valuable for the improvement in my thinking also their kindness and willingness to help is unforgettable. Their great patience and continual encouragement over the past several years have been indispensable for the completion of this work.

In addition, I would like to express my gratitude to Associate Professor Dr. Abdul Azim, for his critical appraisals and constructive comments of my presentations are extremely valuable for the



improvement in my research, also my special thanks to Associate Professor Dr. Ramlan Mahmud, he taught me how to do research.

This research is partially supported by an IRPA fund number (04-02-04-0797-EA001), which is sponsored by Malaysian Ministry of Science, Technology, and Innovation. Thanks to the Universiti Putra Malaysia and Malaysian Government for the support.

I would like to thank many people I have met during my stay in Malaysia for their help, enjoyable discussions and some good times.

Finally, I would like to express my love and deepest thanks to my parents, brothers and sisters, who have loved and supported me throughout my life. And also to my wife and my son, they made my life here in Malaysia enjoyable and memorable.

BELAL ZAQAIBEH

June 2006



TABLE OF CONTENTS

	Page
DEDICATION	ii
ABSTRACT	iii
ABSTRAK	v
ACKNOWLEDGEMENTS	viii
APPROVAL	x
DECLARATION	xii
LIST OF TABLES	xvi
LIST OF FIGURES	xvii
LIST OF ABBREVIATIONS	xx
CHAPTER	
I INTRODUCTION	
Background	23
Problem Statements	27
Research Objectives	30
Research Scope	30
Research Methodology	31
Contribution of Research	33
Organization of the Thesis	35
II LITERATURE REVIEW	
Introduction	36
Preliminary	38
Object-Oriented Databases	41
Object-Oriented Data Modeling	49
Object Constraint Language	58
Optimization Method	59
Integrity Constraints	61
Constraints in Relational Databases	63
Intra-Relational Constraints	65
Inter-Relational Constraints	66
Constraints in Object-Oriented Databases	68
Intra-Class Constraints	68
Intra-Object Constraints	69
Inter-Object Constraints	70
Inter-Class Constraints	72
Integrity Constraint Processing Methods	75
Integrity Constraint Enforcement	76
Integrity Constraint Maintenance	82
Constraint Simplification Technique	90
Enforcement Strategies	93



	Constraint Management Architecture	94
	Constraint Rule Languages	98
	Discussion	101
	Conclusion	103
III	THE ASSERTION MODEL OF INTEGRITY CONSTRAINTS	
	Introduction	105
	Preliminary	106
	AMIC Features	107
	Structure Format of OALIC	108
	OALIC Grammar	110
	The AMIC Framework	113
	The AMIC Architecture and Components	116
	Compile-Time Model	117
	Run-Time Model	122
	Object Meta Data	126
	Constraint Optimization Class	127
	Constraint Knowledge Class	128
	Knowledge Base Class	130
	Detection Method	131
	Domain Optimization	133
	Summary	134
IV	ENFORCING AND MAINTAINING INTEGRITY CONSTRAINTS	
	Introduction	137
	Preliminary	137
	The Maintenance Base	138
	Constraint Violation	139
	Violation Detection	141
	Constraint Maintenance in CTM	144
	Maintaining Intra-Class Constraints	145
	Maintaining Inter-Class Constraints	155
	Composition Hierarchy	155
	Inheritance Hierarchy	160
	Association	165
	Integrity Enforcement in RTM	167
	Inserting Object	173
	Deleting Object	176
	Updating Object	177
	Summary	179
V	RESULTS AND DISCUSSIONS	
	Introduction	181
	Implementation	181
	Evaluation of the AMIC	183
	Comparison with Assertion Rule Languages	184



	Comparison with Constraint Management	
	Architecture	188
	Comparison with Enforcement and Maintenance	
	Approachs	191
	Summary	192
VI	CONCLUSION AND FUTURE WORK	
	Introduction	194
	Contribution	195
	Suggestion for Future Work	197
	REFERENCES	200
	APPENDICES	209
	BIODATA OF THE AUTHOR	235



LIST OF TABLES

Table		Page
2.1	The Development of Integrity Constraints	39
2.2	The UML Modeling Diagrams	57
2.3	Common Multiplicities in Class Diagram	58
2.4	Types and Operations Supported by OCL	59
2.5	Weaknesses for the Previous Studies	102
3.1	Types of Constraint Operands Supported by OALIC	111
4.1	Collection of Constraints for Composition Hierarchy	157
4.2	Collection of Constraints for Inheritance Hierarchy	162
5.1	The Main Procedures in AMIC Technique	182
5.2	Comparison using Constraints Detection Direction	185
5.3	Comparison using Trigger	186
5.4	Comparison using Hierarchy Relationships	186
5.5	Comparison using Expression with Binary Operands	187
5.6	Comparison using Constraint Type	187
5.7	Comparison using User-Defined Methods	188
5.8	Comparison of Constraint Management Approaches	190
A.1	Basic Types in OCL	212
A.2	Operations on Predefined Types	212
A.3	Type Conformance Rules	213
A.4	Valid Expressions	213



LIST OF FIGURES

Figure	Page
2.1 Object-Oriented Database Features	42
2.2 Types of Inheritance	46
2.3 Structure and Relationship on an Object Book	50
2.4 The Description of the Book and Author Classes	50
2.5 Relationships among Complex Objects	51
2.6 Objects Inter-Connection	52
2.7 The Syntax Tree	60
2.8 The Optimized Syntax Tree using DAG	60
2.9 Violated Relations	64
2.10 The Framework of Processing a User Request	75
2.11 Constraint Management	95
2.12 Intermediate Translator for Constraint Management	96
2.13 Application and Database Methods	96
2.14 Triggers and Methods	97
2.15 ARL Rule Format	99
2.16 TQL Syntax	99
2.17 TQL* Syntax	100
3.1 The General Structure of OALIC Format	109
3.2 The Brief OALIC Grammar	111
3.3 A Hierarchy Model	112
3.4 The AMIC Framework	114
3.5 The AMIC Architecture	117
3.6 The CTM Architecture in AMIC	118



3.7	The RTM Architecture in AMIC	123
3.8	The OMD Structure	127
3.9	An Instance of OMD ^{CK}	128
3.10	An Instance of OMD ^{KB}	130
3.11	The DM Heading	131
3.12	The Heading of the Overloaded DM	131
3.13	DM Instance	132
4.1	Composition Hierarchy	140
4.2	The Sal() Method	141
4.3	Source Code for a Single Class	146
4.4	UML for an Independent Class	146
4.5	An Instance of OMD for Intra-Class Constraint	147
4.6	The Domains for UDCs	149
4.7	Verifying Constraints using Maple	153
4.8	The Maintained OMD ^{CO} in the OMD	154
4.9	The UML Diagram for a Composition	156
4.10	The OMD for a Composition Hierarchy	157
4.11	Inconsistent Domains	159
4.12	A Redundant Domain	159
4.13	Single Inheritance between Child and Person	160
4.14	UML Diagram Showing the Added Constraints	161
4.15	OMD ^{CO} in a Single Inheritance	162
4.16	The OMD ^{KB} for a Single Inheritance	163
4.17	The UML Diagram Representing Association	166
4.18	The OMD for Association	167
4.19	Classes and their Constraints	170



4.20	The OMD ^{CO} for Child Database	171
4.21	The OMD ^{KB} for Child Database	172
A.1	The Types Defined in the OCL Standard Library	217
A.2	A University Database	222
A.3	The Optimized Domains in the OMD ^{CO}	223
A.4	The Person Information in the OMD ^{KB}	223
A.5	The Employee Information in the OMD ^{KB}	224
A.6	The Degree Information in the OMD ^{KB}	224
A.7	The Alumnus Information in the OMD ^{KB}	225
A.8	The Student Information in the OMD ^{KB}	225
A.9	The Staff Information in the OMD ^{KB}	226
A.10	The Faculty Information in the OMD ^{KB}	226
A.11	The Student_Assistant Information in the OMD ^{KB}	227
A.12	The Graduate_Student Information in the OMD ^{KB}	227
A.13	The Undergraduate_Student Information in the OMD ^{KB}	228
A.14	The Research_Assistant Information in the OMD ^{KB}	228
A.15	The Teaching_Assistant Information in the OMD ^{KB}	229
A.16	Composition, Association, and Inheritance Relationships	230
A.17	The Optimized Domains in the OMD ^{CO}	231
A.18	The Department Information in the OMD ^{KB}	231
A.19	The Overtime Information in the OMD ^{KB}	232
A.20	The Employee Information in the OMD ^{KB}	232
A.21	The Person Information in the OMD ^{KB}	233
A.22	The Child Information in the OMD ^{KB}	233
A.23	The Programmer Information in the OMD ^{KB}	234



LIST OF ABBREVIATIONS

ADT	Abstract Data Type
AC	Antecedent Constraint
AID	Attribute ID
ALICE	Assertion Language for Integrity Constraint Expression
AMIC	Assertion Model of Integrity Constraints
AOIM	Application-Oriented Integrity Maintenance
ARIEL	ARray-orIEnted Language
ARL	A constraint Rule Language
BLOBS	Binary Large Objects
BPP	Backward Propagation Problem
CA	Constraint Analyzer
CAD	Computer-Aided Design
CAM	Computer-Aided Manufacturing
CASE	Computer-Aided Software Engineering
CC	Constraint Checker
CCG	Constraint Code Generator
CID	Class ID
CIM	Centralized Integrity Maintenance
CM	Constraint Maintenance
CTM	Compile-Time Model
CO	Constraint Optimizer
CP	Constraint Parser
DAG	Directed Acyclic Graph
DBMS	Database Management System
DE	Dependency Evaluation



DID	Domain ID
DM	Detection Method
EBNF	Extended Backus-Naur Form
EC	Engineering Changes
ECA	Event Condition Action
EH	Error Handler
ER	Entity Relational
GCS	Greatest Consistent Specialization
GIS	Geographic Information System
HiPAC	High Performance ACtive
IMPR	Integrity Maintenance Production Rule
IRules	Integration Rules
LAR	Limited Ambiguity Rule
LIC	Logical Integrity Constraint
OALIC	Object Assertion Language for Integrity Constraints
OCL	Object Constraint Language
ODE	Object Data and Environment
ODMG	Object Data Management Group
OID	Object Identifier
OIS	Office Information System
OM	Optimization Method
OMD	Object Meta Data
OMD ^{CK}	OMD Constraint Knowledge
OMD ^{CO}	OMD Constraints Optimization
OMD ^{KB}	OMD Knowledge Base
OMG	Object Management Group

OODB	Object-Oriented Database
OODBMS	Object-Oriented Database Management System
OODM	Object-Oriented Data Model
RAID	Related Attribute ID
RCID	Related Class ID
RDB	Relational Database
RDBMS	Relational Database Management System
RDM	Relational Data Model
RMS	Rule Management System
RTM	Run-Time Model
SC	Supplement Constraint
SIC	Structural Integrity Constraint
SIS	Semantic Integrity Subsystem
SQL	Structured Query Language
TQL	Terminology Query Language
UA	Update analyzer
UC	Update Checker
UDC	User-Defined Constraint
UDT	User-defined Data Types
UE	Update Enforcer
UI	User Interface
UM	Update Maintenance
UML	Unified Modeling Language



CHAPTER 1

INTRODUCTION

1.1 Background

Integrity constraints refer to the expression of integrity validity and do not include the enforcement or the maintenance part. The term integrity covers consistency (data is well organized in accordance with the requirements of a data model) and validity (all invalid data is excluded from the database).

The proper handling of integrity constraints is essential to any data storage and management. Handling integrity constraints is an essential premise to managing semantically rich data (Formica, 2002; Rao, 1994). In Object-Oriented Databases (OODBs), checking the integrity constraints is a fundamental problem in the database design (Formica, 2002). The automated verification of constraints and their enforcement provided by current OODB Management Systems (OODBMSs) is limited (Formica, 2002; Eick and Werstein, 1993) due to the user participation is required.

Maintaining constraints that are scattered in applications is called Application-Oriented Integrity Maintenance (AOIM) (Do *et al*, 1997; Eick and Werstein, 1993). Centralizing the management of integrity constraints by extending database systems to have a dedicated

component for constraint enforcement is called Centralized Integrity Maintenance (CIM) (Do *et al*, 1997; Eick and Werstein, 1993; Urban and Desiderio, 1992).

OODBMSs do not have adequate support for certain types of constraints especially the ones defined in a class composition and inheritance hierarchies (Bagui, 2003; Formica, 2002; Do *et al*, 2002, 1997; Choi *et al*, 1997; Junmang, 1997; Do and Choi, 1994). The integrity constraints must be maintained in the backward direction along the class composition and inheritance hierarchies as well as in the forward direction.

The class composition hierarchy is represented by IS-PART-OF relationship, and class inheritance hierarchy is represented by IS-A relationship (Graham, 2001; Brown, 2001; David and Embley, 1998). The Object-Oriented Data Model (OODM) can support three types of relationships between classes, which are:

- Composition hierarchy (logical or physical composition) is a relationship between two classes where the instances of one class are in some way attributes, methods, and constraints of the other.
- Inheritance hierarchy (single or multiple inheritance) is a relationship between superclasses and subclasses. A superclass may have any number of subclasses, which subclasses inherit attributes and methods of superclass. This means all global attributes, methods, and constraints