Reengineering Relational Databases to Object-Relational Databases Using Collection

M. Bahaj, I. Cherti, A. Jounaidi, Y. Khourfi

1Professor, Department of Mathematics and Computer Sciences, University Hassan I Settat, Morocco
2Professor, Department of Mathematics and Computer Sciences, University Hassan I Settat, Morocco
3Phd, Department of Mathematics and Computer Sciences, University Hassan I Settat, Morocco
4Phd, Department of Mathematics and Computer Sciences, University Hassan I Settat, Morocco

mohamedbahaj@gmail.com, iliacherti@yahoo.fr, jounaidiadi@gmail.com, ykhourdfi@gmail.com

ABSTRACT

This paper discusses the approach to migrate Relational Databases (RDBs) into Object-Relational Databases (ORDBs). Our main effort is to establish models for preserving collection in aggregation, composition and association relationships into ORDB. In our approach we have developed a prototype to create ORDB from relational database. This prototype extracts schema metadata from a database then transforms it into a semantic enrichment of relational database to facilitate the migration process, after the system generates the structure a set of User-Defined-Types (UDTs) and a set of typed tables. The experimental results show that the approach is feasible, efficient and implementable with higher degree of accuracy.

Keywords: Relational databases, Object-relational databases, Migration.

1. INTRODUCTION

To solve the problems of sharing and reusing information in the information system, we propose an approach of rule-based to construct target databases from source databases.

The object-relational databases are hybrid type of databases, which use the best facilities of its predecessors (relational and object-oriented databases) [8], [9].

Strong ORDB allows users to define hierarchies of data types and to introduce the concept of a typed table, which is similar to a class in an OODB. A typed table is created on the basis of a user-defined type (UDT), which provides a way to define complex types using the concept of encapsulation. UDT and their corresponding typed tables can be formed in class hierarchies.

Many works have been conducted to map the Data Definition Language (DTD) and the XML Schema into the Object-Relational (OR) Schema [2]-[4], [10], [11].

Different researches are investigated in RDB migrations to ORDB focusing on different domains [1], [4]-[6]. Most existing approaches are restricted by a range of assumptions and characteristics such as the respect of 3rd Normal Form (3NF), the integrity constraints and hierarchy constraints [2].

The migration is based on the creation of the New Data Model (NDM) that stocks the RDB in a form of a structured table, in order to create a navigational model.

This review differs from comparable approaches, presents an algorithm for automatically constructing object-relational database from relational database. The methodology preserves the constraints of collection in aggregation, composition and association relationship. The algorithm captures important semantic properties.

We extend the algorithm in [5] by proposing different mapping methods for association, composition and aggregation hierarchy. Second, the logical model is mapped into the physical implementation using collection in ORDB.

A main contribution of this review is to describe an approach that allows us to consider a semantic model enriched with the database, process it in relational systems without manual intervention and bring out a model of an object-relational databases.

The goal of this review is to propose models that can preserve collection for composition, aggregation, inheritance and association relationships in RDBs into ORDB. We perform two mapping steps. First is the mapping from the logical model to canonical data model (CDM).

Second, the CDM is mapped into the physical implementation using SQL in ORDB. For this purpose we use the collection data types [7], [8].

In this article we focus on associations, aggregation and composition design, in the framework of methodology in SQL4. This reason is that it supports a collection data type, array and MULTISET (array is recommended if the maximum number of elements is known; if the number of values is unknown or very uncertain, it is recommended to use MULTISET) which is specially appropriated to implement aggregations and compositions.

The study presents a prototype with implementation and validation which enables integration of relational databases to an object-relational database.

The remainder of the paper is organized as follows. Section II presents generating canonical data model from RDB. It explains the formal notation with which the transformation rules are defined. Tool support for the migration process is sketched in Section III. It contains the...
concept for schema transformation migration. The implementation of the prototype is described in Section IV.

2. SEMANTIC ENRICHMENT OF RELATIONAL DATABASE

Generation canonical data model (CDM) from RDB

The CDM produced is enriched with the RDB’s constraints (Domain Constraints, Content Constraints, Cardinality Constraints…) and data semantics that may not have been explicitly expressed in the metadata source.

The CDM is defined as a set of classes:

CDM: = \{ C : C = (Cn: class name, Acdm: Attribute List, Rel: relationships) \}

Where each class C has a set of attributes (Attributes List) and a set of relationships Rel.

Table 1: The CDM DATA

<table>
<thead>
<tr>
<th>Cn</th>
<th>An</th>
<th>T</th>
<th>L</th>
<th>N</th>
<th>E</th>
<th>Reltype</th>
<th>Dir</th>
<th>Dir C</th>
<th>Dir As</th>
<th>C</th>
<th>InvAs</th>
</tr>
</thead>
<tbody>
<tr>
<td>CUSTOMER</td>
<td>CustNum</td>
<td>Int</td>
<td>PK</td>
<td>25</td>
<td>N</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Street</td>
<td>Char</td>
<td></td>
<td>40</td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>City</td>
<td>Char</td>
<td></td>
<td>40</td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Zipcode</td>
<td>Int</td>
<td></td>
<td>5</td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Phone</td>
<td>Int</td>
<td></td>
<td>10</td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>CustName</td>
<td>Char</td>
<td></td>
<td>40</td>
<td>N</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>OrderNum</td>
<td>Int</td>
<td>FK</td>
<td>25</td>
<td>Y</td>
<td>Asso</td>
<td>PURCHASE</td>
<td>ORDER</td>
<td>OrderNum</td>
<td>1..*</td>
<td>OrderNum</td>
</tr>
<tr>
<td>CUSTOMER_ASSOCIATION</td>
<td>Id</td>
<td>Int</td>
<td>PK</td>
<td>25</td>
<td>N</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Description</td>
<td>Char</td>
<td></td>
<td>255</td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>CustNum</td>
<td>Int</td>
<td>FK</td>
<td>25</td>
<td>N</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Percentage</td>
<td>Int</td>
<td></td>
<td>25</td>
<td>Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PURCHASE_ORDER</td>
<td>OrderNum</td>
<td>Int</td>
<td>PK</td>
<td>25</td>
<td>N</td>
<td>Comp</td>
<td>ORDERLINE</td>
<td>ITEM</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Shipdate</td>
<td>Date</td>
<td></td>
<td>N</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Tocity</td>
<td>Char</td>
<td></td>
<td>40</td>
<td>N</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Tostreet</td>
<td>Char</td>
<td></td>
<td>40</td>
<td>N</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Tozip</td>
<td>Int</td>
<td></td>
<td>5</td>
<td>N</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ORDERLINEITEM</td>
<td>LineNum</td>
<td>Int</td>
<td>PK</td>
<td>25</td>
<td>N</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>ProductNum</td>
<td>Int</td>
<td>FK</td>
<td>25</td>
<td>N</td>
<td>Asso</td>
<td>PRODUCTS</td>
<td>ProductNum</td>
<td>1..*</td>
<td>ProductNum</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Quantity</td>
<td>Int</td>
<td></td>
<td>25</td>
<td>N</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>OrderNum</td>
<td>Int</td>
<td>Fk</td>
<td>25</td>
<td>N</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PRODUCTS</td>
<td>ProductNum</td>
<td>Int</td>
<td>PK</td>
<td>25</td>
<td>N</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Description</td>
<td>Float</td>
<td></td>
<td>10.2</td>
<td>N</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>STORE</td>
<td>Location</td>
<td>Int</td>
<td>PK</td>
<td>25</td>
<td>N</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Capacity</td>
<td>Int</td>
<td></td>
<td>25</td>
<td>N</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Street</td>
<td>Char</td>
<td></td>
<td>40</td>
<td>N</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>City</td>
<td>Char</td>
<td></td>
<td>40</td>
<td>N</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Zipcode</td>
<td>Int</td>
<td></td>
<td>5</td>
<td>N</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>STOCK</td>
<td>ProductNum</td>
<td>Int</td>
<td>FK</td>
<td>25</td>
<td>N</td>
<td>Asso</td>
<td>PRODUCTS</td>
<td>ProductNum</td>
<td>1..*</td>
<td>ProductNum</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Location</td>
<td>Int</td>
<td>PK</td>
<td>25</td>
<td>N</td>
<td>Asso</td>
<td>STORE</td>
<td>Location</td>
<td>1..*</td>
<td>Location</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Quantity</td>
<td>Int</td>
<td></td>
<td>25</td>
<td>N</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Date</td>
<td>Date</td>
<td></td>
<td>N</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PERSON</td>
<td>PersonId</td>
<td>Int</td>
<td>PK</td>
<td>25</td>
<td>N</td>
<td>Inherts</td>
<td>CUSTOMER</td>
<td>CustNum</td>
<td>1..1</td>
<td>CustNum</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Discount</td>
<td>Int</td>
<td></td>
<td>25</td>
<td>N</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMPANY</td>
<td>Type</td>
<td>Int</td>
<td>PK</td>
<td>25</td>
<td>N</td>
<td>Inherts</td>
<td>CUSTOMER</td>
<td>CustNum</td>
<td>1..1</td>
<td>CustNum</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Taxes</td>
<td>Int</td>
<td></td>
<td>25</td>
<td>N</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Attributes List Acdm = \{ a := (an, t, tag, l, n, d) \}

Where an : name of the attribute, t :type of the attribute, tag: primary key(PK) | foreign key(FK) ) | key(PF), l: length of the attribute; n denotes whether or not an accepts nulls and d: the default value of the attribute.

Rel: = \{ rel : (Reltype, dirC, DirC.c, DirAs, c, InvAs) \}

Where Reltype: type of the relationship, dirC: the name of the referenced table, c and dirC.c: the cardinalities of the rel, dirAs: a set containing the attribute names representing the relationship from the dirC side and InvAs: denotes a set of inverse attribute names representing the inverse relationship from the C side.
The methodology for classification of types associations
Denote
Att(C): a set of attributes defined in class C
Transform recovered functional dependency
C1(rel1.dirAs, Att(C1)) and DirC=(rel2.dirAs, Att(C2),rel1.InvAs) with functional dependency , then reltype is classified association one to many.

C1(rel1.dirAs,Att(C1)), C2=(rel2.As, Att(C2)) and C3= (rel1.InvAs, rel2.InvAs, Att(C3)) with functional dependency , then reltype is classified association many to many.
Identification of compositions
We identify the composition as a class that represents a multi-valued attribute.

3. TRANSLATING CDM INTO OBJECT-RELATIONAL SCHEMA
We deal with different kind of relationships. There are four class relationships which are associations, inheritance, composition and aggregation.

The proposed algorithm Define the function mapAttrType(ORDB, attr.t ) translates the CDM data type of an attribute attr.t into an equivalent data type in ORDB.

Algorithm ORDBSchema (cdm: CDM) return ORschema
Foreach class C ∈ CDM do
  Foreach relationship rel ∈ C:REL do
    If rel:relType = 'Asso' then
      If rel:c = (0..* | 1..*) && rel:dirC.c=1 then // create Table C without foreign key
        Procedure NFK(C)
      Else if rel:c = * && rel:dirC.c=* then
        // create Table dirC
        // Add a foreign key as a MULTISET
        addAttribute PK_C MULTISET|ARRAY
        (PK(C) data_type);
      Else if rel:relType = 'Aggr' then
        // Create Type C (CREATE TYPE C AS OBJECT)
        Procedure CreateTable(C)
        // add a foreign key
        addAttribute PK_dirC data_type;
        // add all attribute of associate table C’ to C as MULTISET ROW
        addAttribute <<AssociatesMN>:name
        MULTISET|ARRAY ROW
        (foreach attribute attr ∈ C'=mapassociation(C,dirC): ACDM do
          if attr:tag not = "PF" then
            attr.n mapAttrType(ORDB, attr.t )
          end if
        endforeach)
        // create table dirC
        Procedure CreateTable(dirC)
        End If
      End If
    End If
  End Foreach
End Foreach
Add attribute “C” MULTISET|ARRAY (C Type)

Else if rel:relType = ‘Comp’ then

// Create Table C
Procedure CreateTable(C)
addAttribute “dirC” MULTISET|ARRAY ROW (foreach attribute attr ∈ dirC: ACDM do
attr.n  mapAttrType(ORDB, attr.t ),
end foreach)

Else if rel:relType = ‘Inherts’ then

// Create Type C1 (CREATE TYPE C AS OBJECT)
Procedure CreateType (C)
// Create type C2 under Type C
Procedure CreateUnderTable (dirC, C)
End if
End Foreach
End Foreach
End Algorithm

Fig. 2 : The Algorithm of Translating CDM into ORDB

4. IMPLEMENTATION

// Algorithm for build ORDB from CDM
public void ORDBSchema(ArrayList<Cdm> cdm){

// Foreach class C ∈ CDM
for (Cdm C : cdm) {

// Foreach relationship rel ∈ C:REL
for (Rel Rel : C.Rel) {

// If rel:relType = ‘Asso’ then
if (Rel.RelType.toUpperCase().equals("ASSO")) {

if ((Rel.C == "*" ) && (Rel.DirC_C == "1")){

this.NFK(C.Cn, cdm);
this.addMultiset(Rel.C.Cn);
this.createTable(Rel.DirC, cdm);
}

else if ((Rel.C == "*" ) && (Rel.DirC_C == "*" )) {

this.createTable(Rel.DirC, cdm);
this.addMultisetRow(C.Cn, Rel.DirC, cdm);
}

} else if(Rel.RelType.toUpperCase().equals("AGGR")){

this.createType(C.Cn, cdm);
this.createTable(Rel.DirC, cdm);
this.addMultisetType(C.Cn, Rel.DirC, cdm);
}

else if(Rel.RelType.toUpperCase().equals("COMP")){

this.createTable(C.Cn, cdm);
this.addMultisetRow(Rel.DirC,C.Cn, cdm);
}
}

} else if(Rel.RelType.toUpperCase().equals("INHERTS")) {

this.createTypeAsObject(Rel.DirC, cdm);
this.createUnderTable(C.Cn, Rel.DirC, cdm);
}
}
}
this.getCodeSql();

Fig. 3 : The Source Code of the Algorithm

CREATE TYPE CUSTOMER AS OBJECT
( CustNum Int(25) NOT NULL CONSTRAINT CustNum_pk PRIMARY KEY,
  Street Char(40),
  City Char(40),
  ZipCode Int(5),
  Phone Int(10) DEFAULT ‘0661616161’,
  CustName Char(40) NOT NULL,
  OrderNum MUTISET(Int(25)),
  CUSTOMER_ASSOCIATION MUTISET(CUSTOMER_ASSOCIATIONType),
  NOT FINAL;
)

CREATE TABLE PURCHASE_ORDER
( OrderNum Int(25) NOT NULL CONSTRAINT
  OrderNum_pk PRIMARY KEY,
  ShipDate Date NOT NULL,
  ToCity Char(40) NOT NULL,
  ToStreet Char(40) NOT NULL,
  ToZip Int(5) NOT NULL,
  ORDERLINEITEM MUTISET
    (ROW( LineNum Int(25) NOT NULL
      CONSTRAINT LineNum_pk PRIMARY KEY,
      Quantity Int(25) NOT NULL,
      OrderNum Int(25) NOT NULL)));

CREATE TYPE CUSTOMER_ASSOCIATIONType
( Id Int(25) NOT NULL CONSTRAINT Id_pk PRIMARY KEY,
  Description Char(255),
  custNum Int(25) NOT NULL,
  Percentage Int(25));

CREATE TABLE PRODUCTS
( ProductNum Int(25) NOT NULL,
  Description Char(255) NOT NULL,
  Price Float(10,2) NOT NULL,
  LineNum MUTISET(Int(25)),
  STOCK MUTISET
    (ROW( Location Int(25) NOT NULL,
      Quantity Int(25) NOT NULL,
      Date date NOT NULL)));

CREATE TABLE STORE
( Location Int(25) NOT NULL CONSTRAINT
  Location_pk PRIMARY KEY,
CREATE TABLE PERSON UNDER CUSTOMER
    (personId Int(25) NOT NULL,
     personId_pk PRIMARY KEY,
     Discount Int(25) NOT NULL);

CREATE TABLE COMPANY UNDER CUSTOMER
    (   Type Int(25) NOT NULL,
        Taxes Int(25) NOT NULL);

Fig. 4 : Final Result of Translating

5. CONCLUSION
In this paper, a set of rules is presented to generate object-relational database from relational database. Our work includes the implementation of automatic translation tool based on the semantic enrichment of relational database. This approach can restore the semantics of relational database schema. In the next paper we plan to extend the set rules described here with new rules more complex RDB modeling constructs.

REFERENCES


AUTHOR PROFILES

M. Bahaj He is a full Professor in Department of Mathematics and Computer Sciences from the University Hassan 1st Faculty of Sciences & Technology Settat Morocco. He is co chairs of International Conference on Software Engineering, Databases and Expert Systems (SEDEXS’12) , NASCASE’11. He has published over 60 peer-reviewed papers. His research interests are intelligent systems, Ontologies Engineering, Partial and differential equations, Numerical Analysis and scientific computing.

I. Cherti He is a full Professor in Department of Mathematics and Computer Sciences from the University Hassan 1st Faculty of Sciences & Technology Settat Morocco. His research interests are intelligent systems, partial and differential equations. He has organized various national and international events.

A. Jounaidi, He is Phd student in the Department of Mathematics and computer sciences, Faculty of Sciences & Technology of Settat, University Hassan I, Settat, Morocco. His area of interest includes ORDB, web ontologies and semantic web.

Y. Khoudifi, He is Phd student in the Department of Mathematics and computer sciences, Faculty of Sciences & Technology of Settat, University Hassan I, Settat, Morocco. His area of interest includes ORDB.

http://www.scientific-journals.org