Transforming MS-Excel Questionnaire Responses into Structured Data for HDBMS

Alexander P. Harris[1], Olivia J. Martinez[2]
[1]Department of Computer Engineering, University of Oxford, Oxford, UK
[2]Lecturer, Department of Electrical Engineering, University of Cambridge, Cambridge, UK

ABSTRACT

The purpose of this paper is to identify the structure of storing responses of questionnaires from MS-Excel into the hierarchical data model. To explore the structure of responses which is stored in MS-Excel datasheet, help has been taken of a questionnaire designed by a business management researcher. The simple excel response sheet is structured into HDBMS by using the concept of hierarchical representation of data. A comparative study of structure of responses of questionnaires in MS-Excel and HDBMS data model is made. It has been found that the hierarchical database model is a better option for storing responses of questionnaires

Keywords: Questionnaire, Hierarchical Database Management System (HDBMS), Virtual Parent-Child relationship, Integrity Constraints, DDL and DML for HDBMS.

I. INTRODUCTION

A. Questionnaire

A questionnaire is one type of data collection method, which contains formalized set of structured type of questions and to obtain responses from respondents. Respondents were asked to mark their responses. Usually novices store their responses in the structured excel sheet. Here an attempt is made to associate relational data model with the storage of questionnaire responses. Business Management researcher whose case study was considered for research had used excel response sheet for storing responses. It is redefined by using relational data model approach.

B. Hierarchical Database Management System

The hierarchical data model organizes data in a tree like structure. There is a hierarchy of parent and child data segments. This structure implies that a record can have repeating information, generally in child data segments. The hierarchical model employs two main data structuring concepts: records and parent-child relationships. A record is a collection of field values that provide information on an entity or a relationship instance. Records of the same type are grouped into record types. A record type is given a name, and its structure is defined by a collection of named fields or data items. Each field has a certain data type, such as integer, real, or string.

A hierarchical database schema consists of a number of hierarchical schemas. Each hierarchical schema (or hierarchy) consists of a number of record types and PCR types.

A hierarchical schema is displayed as a hierarchical diagram, in which record type names are displayed in rectangular boxes and PCR types are displayed as lines connecting the parent record type to the child record type.

C. Properties of Hierarchical Schema

A hierarchical schema of record types and PCR types must have the following properties:

1. One record type, called root of the hierarchical schema, does not participate as a child record type in any PCR type.
2. Every record type except the root participates as a child record type in exactly one PCR type.
3. A record type can participate as parent record type in any number (zero or more) of PCR type.
4. A record type that does not participate as parent record type in any PCR type is called a leaf of the hierarchical schema.
5. If a record type participates as parent in more than one PCR type, then its child record type are ordered. The order is displayed from left to right in a hierarchical diagram.

In the implementation of hierarchical data model the pointers are normally from a parent record to a child record only. The definition of a hierarchical schema defines a tree data structure. In the terminology of tree data structures, a record type corresponds to a node of the tree, and a PCR type corresponds to an edge (or arc) of the tree. We use the terms node and record type, and edge and PCR type, interchangeably.

D. Virtual Parent-Child Relationship (VPCR)

An M: N relationship may be handled in the hierarchical model by allowing duplication of child record instances. The solution to the problem of M: N relationship- the concept of Virtual (or Pointer) Record Type or Virtual Parent-Child Relationship (VPCR).

A virtual (pointer) record type VC is a record type with no data, but containing a logical pointer to a particular physical record. Instead of replication, we keep a single copy of a record and virtual records. Virtual record VC plays the role of a "Virtual Child" pointing to a "Virtual Parent" in a "Virtual Parent-Child Relationship" (VPCR). Each record occurrence VC points to exactly one record occurrence of VP.
To illustrate the concept of Hierarchical data model, the following case study has been considered.

II. CASE STUDY

The management researcher Mrs. Soma Sharma, Research scholar of RTM Nagpur University, in her Ph.D. thesis "A Study of Micro Finance as a tool for alleviating urban poverty with reference to Nagpur City" has done her work in MS-Excel response sheet for storing responses.

In this paper the best way of representing given responses in the hierarchical data model is proposed. The questionnaire contains 35 structured questions which are multiple choice questions filled from respondents and whose responses were stored in excel which are to be organized in tree like structure using hierarchical data model.
***Multiple-Choice Questions*:** In this type of questions the management researcher provides a choice of answers and respondents are asked to select only one alternative from the given choices.

A. Problems/Limitations of MS-Excel sheet

The researcher used MS-Excel sheet for storing responses. With reference to the case study considered for research, storing responses in the MS-Excel datasheet is not an efficient way. It has the following major limitations:

1. Data types and default values of data types are not supported in excel.
2. Field width or size of field is not supported in excel.
3. Validation rules and validation text cannot be applied.
4. Text formatting is not supported.
5. MS-Excel does not give the facility of data query processing.
6. Data present in an MS-Excel does not support referential integrity rules for establishing relationship between data.
7. There are limitations in data storage size. Eg. In the new version of MS-Excel 2007, the limitations are 1,048,576 rows by 16,384 columns. [a][b].
8. Looking at the above limitations or problems of MS-Excel datasheet. We propose the partitioning questionnaire and use of hierarchical data model for representing responses.

B. Partition of Questionnaire into Records

In hierarchical data model data can be organized using records and parent-child relationship. Here to store the responses of respondent to the questionnaire the following record types are used:

1. **RespondtDetail**
2. **RespondtFmDetail**
3. **MicroFinance**
4. **SHG**
5. **SHGLoan**

A parent-child relationship establish in between two record types. It is one-to-one (1:1) or one-to-many (1: N) relationship. A Hierarchical data model for questionnaire data is as follows:

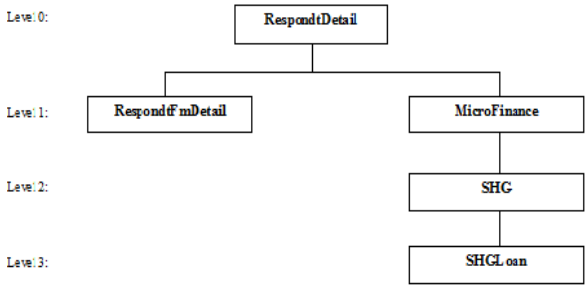
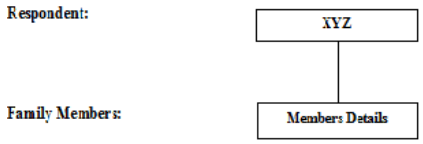
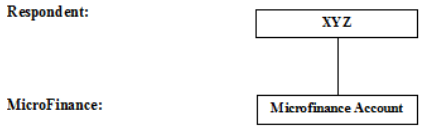


Figure shows a simple hierarchical diagram for hierarchical schema with five record types and four PCR types. Field names can be displayed under each record type name but in some diagram display only the record type names without fields. PCR type represented in hierarchical schema by listing the pair (Parent record type, Child record type) between parentheses. There are four PCR types present in the given hierarchical diagram. They are (RespondtDetail, RespondtFmDetail), (RespondtDetail, MicroFinance), (MicroFinance, SHG) and (SHG, SHGLoan).

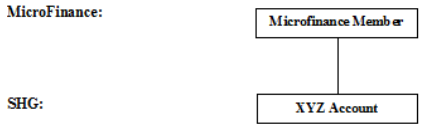
Each occurrence of the (RespondtDetail, RespondtFmDetail) PCR type relates one respondent record to the record of family members who belong to a respondent.



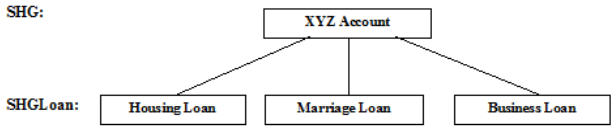
Each occurrence of the (RespondtDetail, MicroFinance) PCR type relates one respondent record to the exactly one record of microfinance.



Each occurrence of the (MicroFinance, SHG) PCR type relates a member of micro finance to the record of SHG type.



Each occurrence of the (SHG, SHGLoan) PCR type relates a member of SHG can have one or more SHG loan.



III. INTEGRITY CONSTRAINTS AND DATA DEFINITION IN HIERARCHICAL MODEL

A. Integrity Constraints

A number of built-in **inherent constraints** exist in the hierarchical model whenever we specify a hierarchical schema. These include the following constraints:

1. No record occurrences except root records can exist without being related to a parent record occurrence. This has the following implications:

- A child record cannot be inserted unless it is linked to a parent record.
- A child record may be deleted independently of its parent; however, deletion of a parent record automatically results in deletion of all its child and descendent records.
- The above rules do not apply to virtual child records and virtual parent records.

2. If a child record has two or more parent records from the same record type, the child record must be duplicated once under each parent record.

A child record having two or more parent records of different record types can do so only by having at most one real parent, with all the others represented as virtual parents.

B. Data Definition in Hierarchical Model

A hierarchical data definition language **(HDDL),** which is not the language of any specific hierarchical DBMS but is used to illustrate the language concepts for a hierarchical database. The HDDL is used to demonstrate how a hierarchical database schema can be defined. To define a hierarchical schema, we must define the fields of each record type, the data type of each field, and any key constraints on fields. The following definitions for the record types RespondtDetail, RespondtFmDetail, MicroFinance, SHG, SHGLoan.

Schema Name = Respondent_Details
Hierarchy = Hierarchy1, Hierarchy2
record
Name = RespondtDetail
Type = root of Hierarchy1
Data items=
RSPD_ID: int;
RSPD_NO: int;
RSPD_Nm: string;
RSPD_Age: int;
Key = RSPD_ID
end

record
Name = RespondtFmDetail
Parent = RespondtDetail
Child Number = 1
Data items =
RSPD_ID: int;
RSPD_Fm_ID: int;
No_Fm_mem: string;
Monthly_Income: int;
Key = RSPD_Fm_ID
Key = RSPD_ID
end

record
Name = MicroFinance
Type = root of Hierarchy2
Data items =

MF_ID: int;
RSPD_ID: int;
MF_aware: string;
MF_reference: string;
Key = MF_ID
Key = RSPD_ID
end

record
Name = SHG
Parent = MicroFinance
Child Number = 1
Data items =
SHG_ID: int;
MF_ID: int;
SHG_mem_status: string;
Key = SHG_ID
Key = MF_ID
end

record
Name = SHGLoan
Parent = SHG
Child Number = 2
Data items =
Loan_ID: int;
MF_ID: int;
SHG_ID: int;
Avail_Loan: string;
Loan_Amt: int;
Key = Loan_ID
Key = MF_ID
end

From the above definition of hierarchical schema of questionnaire data, each record type is declared to be of type root or a single (real) parent record type is declared for the record type. The data item of the record are then listed along with their data types. Data items declared under the *Key* clause are constrained to have unique values for each record. Each *Key* clause specifies a separate key; in addition, if a single *key* clause lists more than one field, the combination of these field values must be unique in each record. The *child* clause specifies the left to right order of a child record type under its parent record type.

IV. DATA MANIPULATION LANGUAGE FOR HIERARCHICAL MODEL

Hierarchical Data Manipulation Language (HDML**),** which is a record at-a-time language for manipulating hierarchical databases. The commands of the language must be embedded in a general purpose programming language called the host language**.** The DBMS maintains a pointer to the current record. Subsequent database commands proceed from the current record and may define a new current record, depending on the type of command.

A. GET Command

The HDML command for retrieving a record is the GET command. The syntax of this command is:

GET FIRST <record type name> [WHERE<condition>]

By using this command, which always starts searching the database from the beginning of the hierarchical sequence until it finds the first record occurrence of <record type name> that satisfies <condition>.

For Example: “Retrieve the record of first Respondent in the hierarchical sequence whose name is XYZ”.

$GET FIRST RespondtDetail WHERE RSPD_Nm = ‘XYZ’;

The DBMS uses the condition following WHERE to search for the first record in order of the hierarchical sequence that satisfies the condition and is of the specified record type.

B. UPDATE Command

The HDML commands for updating a hierarchical database are shown in following table:

RETRIEVAL	
GET	Retrieve a record into the corresponding program variable and make it the current record.
RECORD UPDATE	
INSERT	Store a new record in the database and make it the current record.
DELETE	Delete the current record (and its sub tree) from the database.
REPLACE	Modify some fields of the current record.
CURRENCY RETENTION	
GET HOLD	Retrieve a record and hold it as the current record so it can subsequently be deleted or replaced.

The INSERT command inserts a record into the database. The newly inserted record also becomes the current record for the database, its hierarchical schema, and its record type. The record is inserted in the hierarchical sequence in the order specified by any ORDER BY fields in the schema definition.

To insert a child record, we should make its parent, or one of its sibling records, the current record of the hierarchical schema before issuing the INSERT command.

To delete a record from the database, we first make it the current record and then issue the DELETE command. The GET HOLD is used to make the record the current record, where the HOLD key word indicates to the DBMS that the program will delete or update the record just retrieved.
For Example: “Delete all Respondent details whose age is less than 30”

```
$GET HOLD FIRST RespondtDetailWHERE RSPD_Age< 30;
whileDB_STATUS=0 then
begin
writeln (RSPD_Nm);
$DELETE RespondtDetail;
$GET HOLD NEXT RespondtDetailWHERE RSPD_Age< 30;
end;
```

V. BENEFITS OF HIERARCHICAL DATA MODEL

When questionnaire data represent in HDBMS, it gives better performance than MS-Excel datasheet. It has some advantages are given below:

1. Data must possess a tree like structure.
2. Navigating among the records in a hierarchical database is very fast because the parent/child relationships are implemented with pointers from one data record to another.

Example: By knowing the multiple child record pointers or the single parent record pointer to get the "next" record easily.

VI. PROBLEMS/LIMITATION OF HDBMS DATA MODEL

1. The structure of hierarchical data model is inflexible because of its rigid structure.
2. We cannot define new linkages between records once the tree is established.
3. There is no ALTER TABLE command in hierarchical data model when we want to add new field into the table.
4. When we want to add new relationship between two records than we build completely new and possibly redundant database structure exist.
5. Hierarchical data model cannot be implemented easily because of its complex structure.

VII. CONCLUSION

From the above case study we conclude that:

1. HDBMS data model is a better fit in representing response of questionnaire data as compare to MS-Excel datasheet.
2. By using tree-like structure the originality of the response sheet is maintained.
3. Navigation between the records is very fast because parent-child relationship is implemented with pointers from one data record to another. It is not happened with MS-Excel.
4. Storing data in hierarchical data model and defining relations among the records will produce the transactional data which proves to be easy to handle which is not supported by MS-Excel.
5. Supporting referential integrity by using parent/child relationship which is not present in MS-Excel.
6. To define data types for each member variable of record with their size this is not possible in case of MS-Excel.
7. In HDBMS for each member variable of record with their size can be defined but this is not possible in case of MS-Excel datasheet.
8. HDBMS support many-to-many relationship by using virtual parent/child relationship (VPCR).

VIII. ACKNOWLEDGMENT

I would like to thanks Dr. Vijay Khandal for their valuable contribution and support in the preparation of this paper.

REFERENCES

[1] http://spreadsheetpage.com
[2] http://wiki.answers.com/Q/Total_number_of_row_and_column_of_m.s.excel.