International Journal of Information & Computation Technology.
ISSN 0974-2239 Volume 2, Number 1 (2012), pp. 27-36

http://www. ripublication.com

A Multidimensional Model for Automatic Schema Generation process by Integrating OLAP with Data Mining Systems

V. Mani Sarma[1], D. Rambabu[2], Prof. P. Premchand[3] and Dr. V. Purnachandra Rao[4]

[1]Associate Professor, Nalla Malla Reddy Engineering College, Hyderabad, Andhra Pradesh, India
E-Mail: manisharma.vittapu@gmail.com.
[2]Associate Professor, Holy Mary Institute of Technology & Science, Hyderabad, Andhra Pradesh, India
[3]Dean, Dept. of Computer Science and Engineering, Osmania University, Hyderabad, Andhra Pradesh, India
[4]Principal, Murthy Institute of Technology and Science, Hyderabad, Andhra Pradesh, India

Abstract

Data in a data warehouse is organized in a multidimensional model. This multidimensional model helps in faster query processing and efficient OLAP operations for data analysis and decision making. In this paper, we introduce a framework which proposes design methodologies to map the relational database into a multidimensional model. The process starts with first cleaning the relational database and then categorizing the attributes of this cleansed relational database into metrics and dimensional attributes by applying the proposed set of mapping rules.

A multi-dimensional model of data supports decision making in a much better way, and the aim of the project is to build a tool that helps companies to create their own multi-dimensional model from a collection of relational databases, and to make a web-based environment supporting flexible views of the multi dimensional model.

Keywords: Data Warehouse, Extraction Transformation and Loading, multidimensional model, star schema.

Introduction

A database is a data repository which includes a collection of entity sets in the form of rows and columns, and each of these entity sets contains any number of entities of the same type. A relational database is a shared repository of data [1]. These databases are used for day to day transaction processing and are also known as OLTP (On-Line Transaction Processing) systems. These systems do not fulfill the out-of-the-ordinary tasks of information processing and data analysis, therefore to overcome this drawback and to introduce added advantages from a data repository, data-warehouse concept was presented. Data-warehouse is a subject oriented, integrated, non-volatile and time variant collection of data in support of management's decision [2][3]. Data-warehouse has a broader scope and is fully equipped with the latest technologies supporting the decision making process.

The ETL (Extract, transform and load) process works at the backend for the design of data warehouse. ETL is a process in database usage and especially in data warehousing that involves:

- Extracting data from outside sources.
- Transforming it to fit operational needs (which can include quality levels).
- Loading it into the end target (database or data warehouse).
- The main source of the data is cleansed, transformed, loaded and made available to managers and other business professionals for data mining, online analytical processing, market research and decision support. The ETL process basically takes place in the data staging area where the data is integrated and cleansed. Data cleaning, refers to the process of removing errors and inconsistencies from the data so as to improve the quality of the data. In a data-warehouse data comes from different sources and is therefore available in various different forms and formats. The syntax and semantics are different for each of the multiple data source. So, if the data is stored as it comes from the data sources, it would create inconsistency in the data-warehouse, leading to confusion, and hence degrading the quality of the data.
- Multidimensional models like cubes, hyper cubes, star schema, snowflake schema etc are used for representing the data in a data warehouse. In our approach, we use star schema for this purpose. A star schema comprises of two major elements-fact (areas of interest for making strategic and analytical decisions for example: sales) and dimension (a base for fact analysis, for example: time, employee etc)
- [2]. Each fact contains a set of numerical attributes called metrics E.g. number of products sold, profit etc. and each attribute within a dimension is called dimensional attribute. E.g. price of a product, customer salary etc.
- The concept of metadata is also being focused upon in this paper. Every database has a Data Dictionary which stores metadata about the structure of the database, in particular the schema of the database. For example, Oracle Designer stores the design in Oracle Repository, which serves as a single point of metadata for the application. Metadata describes all the pertinent aspects of the data in the database fully and precisely. While building a data warehouse, one

requires metadata about the source systems, source to target mappings and data transformation rules. The metadata can then be used to generate forms and reports. [1][2]

- Data-warehouse design and development proposed earlier required ad-hoc methodologies. The star schema is one such logical representation which supports the conceptual modeling phase in the data-warehouse design. The earlier proposed work was based on the mapping of UML diagrams to obtain a multidimensional model (star schema or multi-dimensional ER model). We have formulated an approach to simplify the transformation of relational database schema to multidimensional form. We retrieve the metadata of relational database and use it to design the star schema of the corresponding relational database. The column names fetched from the metadata are segregated to figure out dimensional attributes and metrics. For this, we suggest a set of mapping rules. Before applying the mapping rules, the data in relational database is cleansed. A cleaning algorithm has been devised for cleaning the name and email id fields.

The Proposed Framework

The ETL framework (Figure 1) so proposed is based on a user friendly approach. Data from all the relational databases is gathered and integrated at one place which we call the integrated relational database. Since the data comes from disparate relational databases, we need to have all the data in a consistent format. We propose the following cleaning algorithm for this purpose. This algorithm has been devised for the name field and the email field.

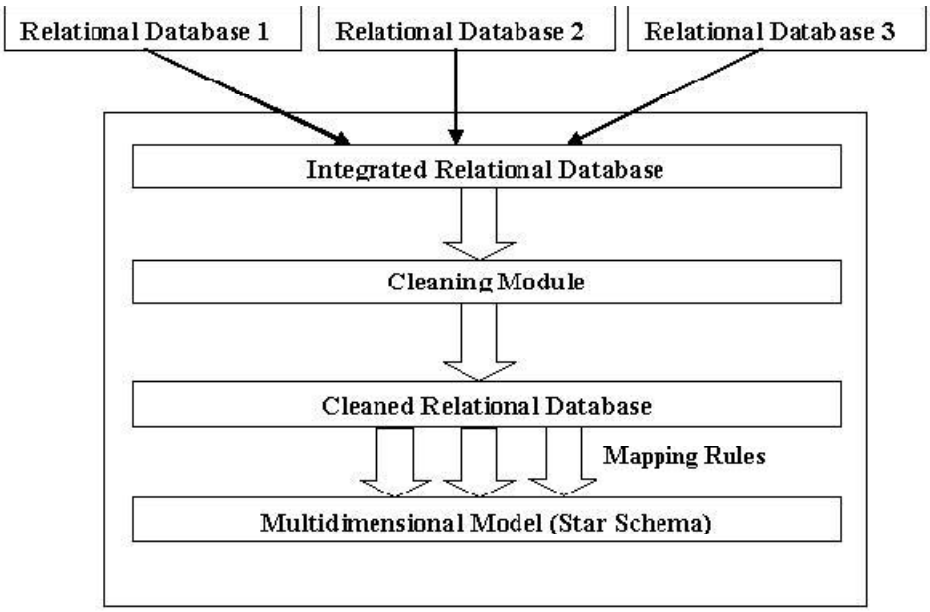

Figure 1: Proposed Framework

Data Cleaning

Data Cleaning is a process used to determine inaccurate, incomplete, or unreasonable records from a record set, table, or database and then improving the quality through correction of detected errors and omissions.

Data cleaning is important for the efficiency of any data dependent organization. In any organization incorrect data can be costly. Incorrect or inconsistent data can lead to false conclusions and misdirected investments. The job of data cleaning is to ensure that the data within a system is correct, so that the management is able to use this data. [12]

In this paper, an approach for data cleaning is proposed. Here, two data fields namely-Name and Email are primarily cleaned. Formats are set for both the fields and they are cleaned on the basis of those formats only.

E-mail Address field cleaning

The paper proposes a pattern matching algorithm for the cleaning of the E-mail id field. This algorithm works on the basis of comparison and string matching. All the e-mail ids are compared with the e-mail id format which is defined below. The pattern matching algorithm searches for irregularities in the e-mail ids and invalidates those e-mail ids which do not comply with the required format. In this way, the inconsistent and invalid database entries can be easily searched in the vast data pool and then can be removed later or fetched again from the corresponding sources.

The format of the e-mail field is:

[_a-z0-9-]+(\.[_a-z0-9-]+)*@[a-z0-9-]+(\.[a-z0-9-]+)*(\.[a-z]{2, 3})

It basically implies that the user name in the email address can consist It basically implies that the user name in the email address can consist of alphabets, digits or some special characters followed by „@“ sign which is compulsory. After the „@“ sign, domain name is specified to which email would be sent. Domain name consists of at least one „dot“ and at most two „dots“ and after the „dot“, specified domain needs to have minimum of two characters. This pattern is to be matched with all the e-mail ids for validating them.

Name field cleaning

The name field encompasses – salutation, first name, middle name, and last name. All these fields combine to form the name field. It is critical to clean this field because it may have been entered in different formats in different databases and tupples. Here we suggest an algorithm to clean all the names by grouping them under different categories based on the types of salutations. These categories are further divided into groups based on different possible formats for that particular category. Finally, all the names are cleansed and stored in one particular format. The salutations according to which names are categorized can be – Mr., Ms., Dr., Prof., Mrs. and Miss. and each name under every category may be written in a different format due to which the category is further divided into groups.

Methodology to Design Multidimensional model

A multidimensional model is defined by a fact table and several dimension tables. The first thing that the user has to select from the relational model is the measure attribute. As it has been explained, the measure attribute is the purpose of the cube. It is what the user wants to know, an attribute from the relational model that will be save into the fact table of the multidimensional model. Once the measure attribute is selected the user has to be asked about what is the information the he want to relate with the measure attribute. This information is needed to define the dimensions. Then the grain attribute has to be selected. A grain attribute is selected from the relational model. Once the grain attribute is selected, the tool has to find a path between the gain attribute and the measure. A grain attribute may have more than one path to the measure attribute. The user chooses the path or the paths that he wants for the multidimensional model. Each path represents a concept of information. Then each couple consisted of path and grain attribute represent different information that the user want to store in the multidimensional model. Therefore for each couple of grain attribute and path, that the user has selected, a new dimension is going to be created. In order to create the fact table are needed the measure and the grain attributes of each selection. It is because the fact table is composed for these elements.

Algorithm Multidimensional_Modelling
Input:

R: { (a_i, a_j) } set of foreign key

T: { t_j } set of tables of the relational model relations

Output:

FT = ({g}, {m})

D = {d}

AS = { a }

Variables used in the algorithm

G: variable used to store a grain attribute

m: variable used to store the measure attribute.

P: {pi}: set of path in construction.

They start in the grain attribute g.

p: a path.

Begin:

```
moreGrains: boolean:= true;
AS = {}
m:= select_measure_attribute_from_relational_model();

// Dimensions
while (moreGrains) loop
g:= select_grain_attribute_from_relational_model();
```

```
// P: {p} set of path from the measure grain attribute to the measure attribute
P = makePathsGrainToMeasure(T, R, g, m);
for all p∈ P loop
// d: new dimension
d:= makeDimension(T, R, AS, g);
D:= Ḍ∪ { d }
P:= Ṗ∪ { p }
end loop;
moreGrain:= user_answer();
end loop;
// Fact Table
for all d _ D loop
g':= grain_attribute_of(d);
FT:= ({g}∪ {g'}, {m})
end loop
end algorithm;
```

System Implementation

Architecture of the system

The system design in this thesis in order to resolve the problem of the multidimensional database is the one who is show in the Figure 5. 1. This system is composed by different components: a relational database, the thesis tool, a Data Warehouse, an OLAP server, and a web environment. The *relational database* is a database stored in a database manager. This database manager has to be a PostgreSQL manager.

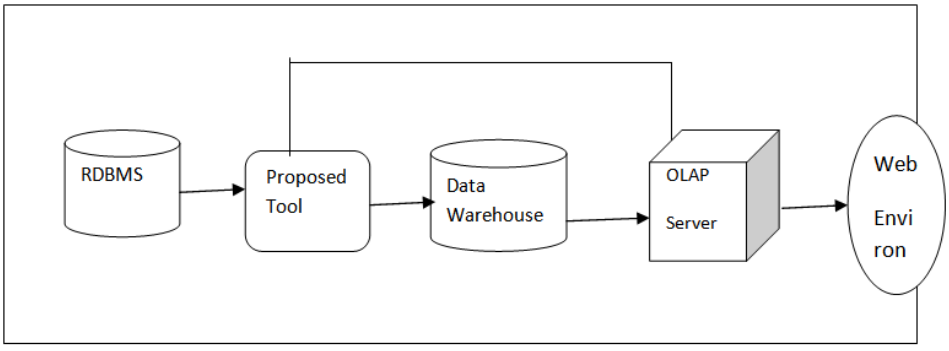

Figure 5.1: System Architecture

Object design

The purpose of this section is specifying the objects design of the thesis tool. In order to make the design of the thesis tool has been used an UML specification. It is going to be presented the collection of packages and the relations between those, which have

been created for the object design of the tool. The Figure 5.2 is the diagram of the packages of the system. It is composed by the package MultidimesionalBuilder that is associated with the package Multidimensional_construction in order to build the multidimensional database. And with the package Olap_Access in order to make a connection with a multidimensional database already created. The description of each package is the next one: The *Relational Model* package represents a database relational model. It is the package used only to represent the data extracted from the operational source. The *Multidimensional Model* package represents the multidimensional model. It contains all the information about the construction of the multidimensional database, and the information about the multidimensional model.

The *OLAP Model* package creates the OLAP schema for a multidimensional data model.

The *Multidimensional Construction* package has the intelligence to construct all the model.

The *MultidimensionalBuilder* class is the one that join all the activities that are describe in the multidimensional Construction package.

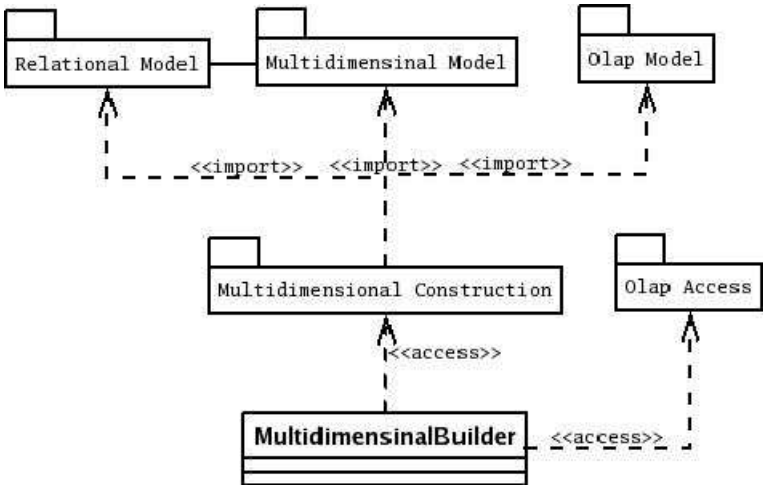

Figure 5.2: Package diagram

Package Multidimensional Model

The package multidimensional model represents a multidimensional data model. As the Figure 5.4 shows, it is compose by the classes Dimension, Fact_table, AttributeMulti, Semantic and Multidimensional_Model. The class Dimension represents a dimension. It has a name and is compose for a grain attribute, a set of attributes and a path that defines the hierarchy of the dimension. The class Fact_Table is composed by a measure attribute. The class AttributeMulti represents an attribute from the multidimensional model. It references a attribute from the relational model. And has semantic information. The class Semantic represents the semantic information of an attribute. It is compose by other attribute, and is specify the type. At the end the class Multidimensional_Model represents the data model. It is compose by

a set of dimensions a a Fact_Table organize the multidimensional model and allow to create a start schema and a transformation schema.

	customer_id	street	city	customer_name	dob	age	email_id	gender	act_no
	cust[illegible]	lutyens	delhi	Mrs. Sonia gandhi	19[illegible]-21 [illegible]	41	sg2011@gmail.com	female	1
	cust0002	lutyens	delhi	Mrs.Indira Gandhi	1973-09-23 00:00:00	38	ig2011@gmail.com	female	2
	cust0003	vasant vihar	delhi	dr. Menaka Kumari	1964-08-24 00:00:00	45	mg2011@gmail.com	female	3
	cust0004	rohini	delhi	mr.Rakesh kumar Sharma	1987-11-01 00.00.00	23	rg2011gmail.com	male	4
	cust0005	vasant vihar	delhi	Mr. Varun Kumar Jain	1950-00-10 00:00:00	61	vg2011gmail.com	male	5
	cust0006	greater kailash	delhi	Miss. vandana Kumari Luthra	1980-08-19 00:00:00	31	ng2011@gmail.com	female	6
	cust0007	greater kailash	delhi	Prof.Raj Mittal	1960-08-02 00:00:00	51	jg2011@gmail.com	male	7
	cuct0008	chandni chowk	dolhi	Mr. Rahman chaudhary	1961 08 20 00:00:00	50	hg2011@gmail.com	malo	8
	cust0009	lutyens	delhi	mr. rahul Kumar aggarwal	1976-08-15 00:00:00	35	ug2011@gmailcom	male	9
	cust0U1U	lutyens	delhi	ms. Priyanka Gupta	1965-10-31 00:00:00	46	pg2U11@gmail.com	female	1U

Figure 4: Table with erroneous entries

	customer_id	street	city	customer_name	dob	age	email_id	gender	act_no
	cust0001	lutyens	delhi	Mrs. Sonia Gandhi	1970-08-21 00:00:00	41	sg2011@gmail.com	female	1
	cust0002	lutyens	delhi	Mrs. Indira Gandhi	1973-09-23 00:00:00	38	ig2011@gmail.com	female	2
	cust0003	vasant vihar	delhi	Dr. Menaka Kumari	1964-08-24 00:00:00	45	mg2011@gmail.com	female	3
	cust0004	rohini	delhi	Mr. Rakesh Kumar Sharma	1987-11-01 00:00:00	23	invalid email_id address	male	4
	cust0005	vasant vihar	delhi	Mr. Varun Kumar Jain	1950-08-10 00:00:00	61	invalid email_id address	male	5
	cust0006	greater kailash	delhi	Miss. Vandana Kumari Luthra	1980-08-19 00:00:00	31	ng2011@gmail.com	female	6
	cust0007	greater kailash	delhi	Prof. Raj Mittal	1960-08-02 00:00:00	51	jg2011@gmail.com	male	7
	cust0008	chandni chowk	delhi	Mr. Rahman Chaudhary	1961-08-20 00:00:00	50	hg2011@gmail.com	male	8
	cust0009	lutyens	delhi	Mr. Rahul Kumar Aggarwal	1976-08-15 00:00:00	35	invalid email_id address	male	9
	cust0010	lutyens	delhi	Ms. Priyanka Gupta	1965-10-31 00:00:00	46	pg2011@gmail.com	female	10

Figure 5: Table with correct entries

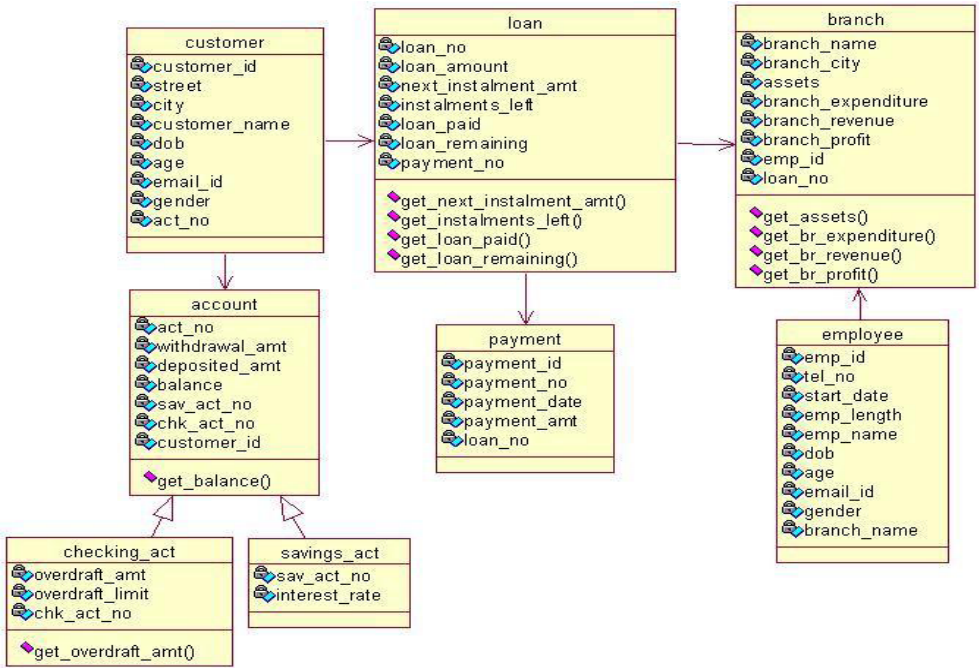

Figure 6: Bank Management System Class Diagram

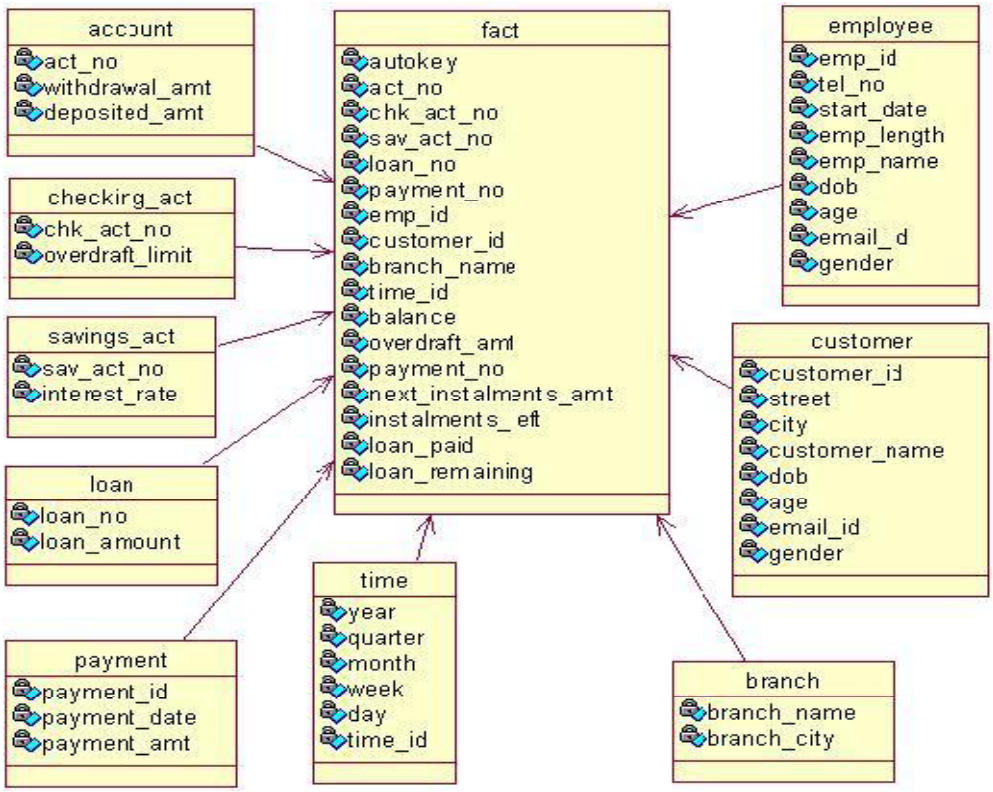

Figure 7: Star Schema

Conclusion

The goal of this paper is the creation simple model to create multidimensional database from relational database. It model allows the creation of the semi-automatic support for multidimensional databases. The model has been designed in the most simplify way. The simplicity of the model is for two reasons:

- The first one is that this model has been thought to help user that are non experts in multidimensional data model to create their own multidimensional database. For that reason if it is created a complicated model, the user has to know more things about multidimensional databases and this is not the goal of the thesis.
- The other reason is that this tool is not trying to replace the multidimensional experts. It is a simple model for simple databases. A construction of a complicated multidimensional model need more analysis, and this tool may be used to oriented.

The goal of the informatics is to organize the data in order to make it accessible for everyone. This Paper has tried to attain the same goal for the multidimensional databases. These databases organized the data to help users in a different way than the operational ones, but with the same data, the difference is the organization.

References

[1] Elmasri, R., Navathe, S.B.: Fundamentals of Database Systems. Addison Weasely Pub Co. ISBN 0201542633 (2000).
[2] Paulraj Ponniah, Data Warehousing Fundamentals, Wiley India Pvt. Ltd., Reprint 2008.
[3] Inmon, W.H., Building the Data Warehouse (2nd Edition), New York: Wiley, 1996.
[4] María, A. and Castillo, G., „Semi-Automatic Support for the Design of Multi-Dimensional Databases", Kongens Lyngby, 2007, www.imm.dtu.dk, ISSN 0909-3192, Chapter 3.
[5] Cabibbo, L. and Torlone, R., „A Logical Approach to Multidimensional Databases"
[6] Moody, D.L., Kortink, M.A.R. "From Enterprise Models to Dimensional Models: A Methodology for Data Warehouse and Data Mart Design", 2nd International Workshop on Design and Management of Data Warehouses (DMDW 2000), Stockholm, Sweden in June 2000.
[7] Pahwa, P., Taneja, S. and Jain, S., Design of a Multidimensional Model Using Object Oriented Features in UML. UTIT, International Conference on Upcoming Trends in IT, (Mrach 26, 2011) at PCTE Ludhiana, Punjab, India.
[8] Hang-Hai, D., & Erhard, R.: Data Cleaning: Problems & Current Approaches. IEEE bulletin of the technical committee on Data Engineering, 24, 4 (2000).
[9] Pahwa, P., Chaudhary, G., Jain, K., Sharma, N. and Gupta, R., „Hierarchical Approach to Data Extraction using UML 2.0", Proc. of the International Conference on Advanced Computing and Communication Technologies (ACCT 2011), Copyright © 2011 RG Education Society.