

©2006 IEEE. Personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution to servers or lists, or to reuse any copyrighted component of this work in other works must be obtained from the IEEE.

Mapping and Modeling of Oil and Gas Relational Data Objects for Warehouse Development and Efficient Data Mining

Shastri L Nimmagadda and Heinz Dreher

Abstract— Oil and gas industries archive volumes of heterogeneous data. These companies, by virtue of their diverse operations, comprise of complicated organizational structure. The nature of organizational set-up with several operational units, often results with communication barriers among operational units. In order to effectively and efficiently perform oil and gas company's business activities, the flow of data and information must be consistent and sharing among its units. In order to improve information sharing among oil and gas company's personnel, heterogeneous data from various sources are integrated. Data warehouse is a solution, in which, oil and gas data entities, identified as class objects, are used for multidimensional modeling. Relational data structures constructed using these class objects, are stored in a warehousing environment to minimize the complexity of heterogeneous data and enhances power of data integration and information sharing among different operational units.

I. INTRODUCTION

Oil and gas industry as a system deals with different operational units, each with different types of data. This complex system possesses several sub-systems such as exploration, drilling, production and marketing which can be viewed as class objects with common boundaries and attributes. Each sub-system is further composed of several sub-classes. In order to reduce the complexity in data structuring, concepts of object relational, logical dimensional data modeling and data warehousing have been introduced. Data warehousing is widely accepted database technology in many industries, such as banking, transport, health and insurance industries. In recent years, oil and gas industries have realized the significance [8, 10] of these warehousing and data mining technologies.

Practical aspects of data modeling have been discussed in [1] and [11], with examples of star and snowflake schema applications in [2]. Various class objects, programming syntaxes and coding procedures have been demonstrated in [3] and [13]. Metadata, using an object relational approach for data warehouse development has been detailed in [4]. Several data cube views have been interpreted through OLAP in [7, 12, 14], and issues relevant to data mining in

Manuscript received March 31, 2006. Shastri L Nimmagadda, Kuwait Santa Fe, Project and Petroleum Engineering Company, PO Box 9250, Ahmadi, Kuwait City, 61003, Kuwait. (phone: 965 398 3639; e-mail: NIMM@chevron.com).

Heinz Dreher, Curtin University of Technology, GPO Box U1987 Perth, Western Australia 6845. (e-mail: h.dreher@curtin.edu.au)

the industrial scenario have been exhaustively discussed in [5] and [15]. Data objects relevant to petroleum industry have been described in [6].

II. PROBLEM STATEMENT

Different object classes, sub classes and attributes of the oil and gas industry are identified. These objects and their relationships are intended to be mapped for multidimensional modeling and for the purpose of data warehouse development. One key dimension of the data warehousing is period, which is interpreted as an object class, has relation with other objects of the oil and gas company's data. These data associations will be mapped and thus dimensional models will be created. It is intended to design multidimensional data structuring for warehousing historical data of oil and gas industry.

III. METHODOLOGY

Using a relational database approach, object relational data structures are developed using oil and gas company exploration data. Star and snowflake schemas have been used to develop multidimensional data models to be described in the next section.

A. Star Schemas

Several fact and dimension objects have been identified from the oil and gas companies. In the star structural data model, associated object dimensions surround several object facts in a star style.

The basic idea of the star schema is to retain the multidimensional capability of the cube while providing the flexibility of smaller data storage.

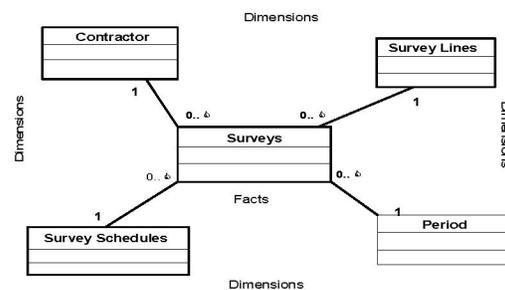


Fig. 1: A star schema data model of "surveys" object

As shown in Fig. 1, the star schema represents a dice of four object dimensions (contractor, survey lines, survey schedule and survey period) with survey object facts.

The fact object class refers to every associated object dimension, which is to be placed in a cell of the cube. The dimensions in our example are:

- 1) The contractor object class describes each individual contractor of the contracting company (chosen by the oil & gas company) identified by name, ID, contractors not obliged to the terms and conditions are not part of the data warehouse.
- 2) Survey schedule is the schedule of surveys to be conducted in a basin
- 3) Survey lines possess several survey lines laid for this survey
- 4) Period is the start of survey (including ending date of survey)
- 5) Type of survey conducted in a type of basin

The survey object fact describes here a single survey engaged by a contractor at a period, choosing a specific survey with number of survey lines. The analytical space can be the whole cube, or the analytical space can be sliced according to the dimension in to smaller pieces. Each dimension is described according to an object class, which is specified according to a business subject. This is most important for the success of data warehouse design. Typical users are exploration manager, exploration database analyst, or a marketing engineer. The fact table itself is another object, represented again by a class object. The fact table refers to every dimension. The association between the fact tables and dimension tables is always one to many, which means each fact is associated with exactly one unit of the single dimension, and each unit of the dimension (each contractor, period etc...) can be associated with any number of facts (including 0) Transforming the object model into the data model implies an implementation of the star schema as shown in Fig. 1. As shown in Fig. 2, automatically generated primary and foreign key constraints are hidden in the display. The dimensions of the star schema are represented as independent tables.

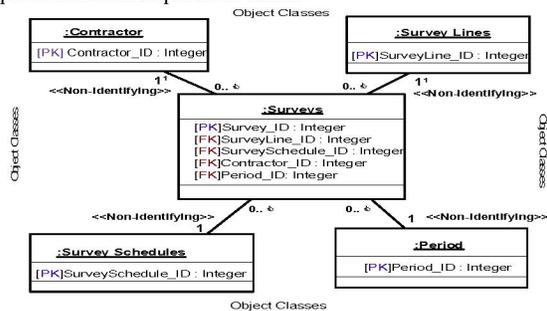


Fig. 2: Implementation of star schema for “survey/contractor” problem

In the implementation model, primary keys of dimensions are generated when the object model is transferred to the data model. The facts of the oil and gas data thus refer to the dimensions using the key migration from the migration tables. When the data models are composed, foreign keys are identified. Slicing and dicing in star schemas have a limitation (selection) of dimensions. It is a run-time issue, not a modeling one, but the model has to recognize the need of it.

Alternatively, all the links or associations among dimension and fact tables can also be stored in separate tables that logically created, called associative schemas. During design and implementation stages, these associative schemas are triggered for mapping dimensional and fact data tables.

B. Snowflake Schemas

The basic star schema does not satisfy all the needs of data mining. Composite dimensions are used such as period of survey in the example above. The analyst investigates the patterns of data according to day, week, month, quarter, year etc... In such a case, during the mapping process, dimension relations have to be normalized. Redundancy of dimension tables is needs to be addressed to reduce the complexity of data slicing. The schema that is derived with such a process is called a snowflake. As shown in Fig. 3, the period dimension is normalized to days, weeks, months, quarters, and year. For every additional normalized dimension: day, week, month, quarter and year, slices can be made from the cubes. For this reason the dimension of period is pre-selected using one of the normalized tables, which is a simple addition to the data mining queries.

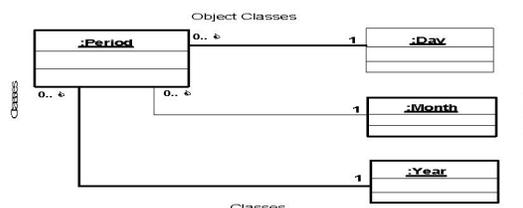


Fig. 3 Normalized period dimension

The resulting snowflake with normalized dimension is as shown in Fig. 4.

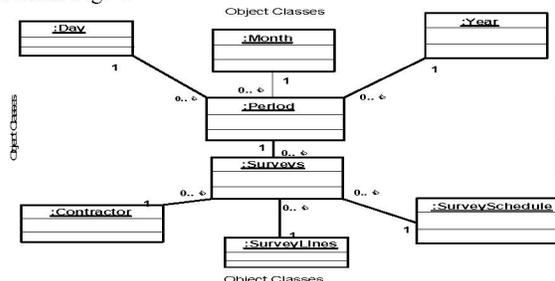


Fig. 4: A snowflake schema with normalized period dimension associated with survey facts

Of course, all the dimensions may be normalized like the period example. This results in simplifying more complex data schemas. The implementation schema (data model) developed out of this snowflake is shown in Fig. 5. Again, the generated constraints are hidden in our model display. Slicing in a snowflake is possible not just at the basic period dimension, but also at normalized days, week, month, quarter and year.

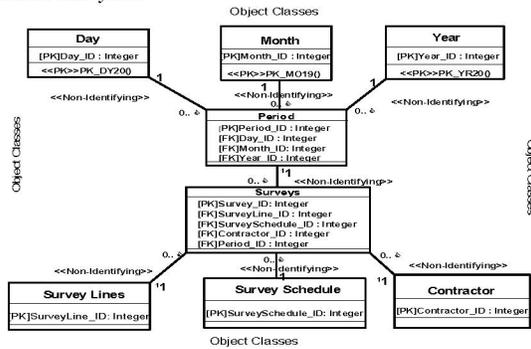


Fig. 5: A snowflake schema model with normalized period dimension and association with “Surveys” object

C. Many to-many relationships

On every survey or investigation, there may be many survey lines. There can be several of them if the survey is extended on offshore and transitional zones from onshore areas. In such a case, one-to-many relationships will not be there between survey facts and survey line dimensions, but many-to-many associations exist. However such an association cannot be implemented in the star schema. A special form of a snowflake schema implements the necessary structure of the data to comprehend this need. Our model will be changed to many-to-many associations between the fact and the dimension tables. This is just a change in the cardinality of the association as shown in Fig 6.

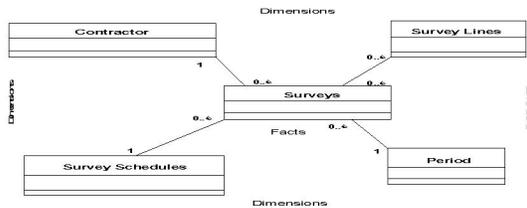


Fig. 6: A star schema with many-to-many relationship dimension of “Survey Lines”

The many-to-many association cannot be implemented with a relational database, but another kind of snowflake schema can be used to implement these relationships. As shown in Fig. 7, many-to-many relations are implemented in which an additional dimension refers to the dimension survey type and the survey facts object class. Relationships are identified and used to resolve many-to-many

associations. Documenting such a relationship must be made in practice for the snowflake schemas. The simple object view gives the opportunity for understanding the concept, whereas general data view informs the implementation stage.

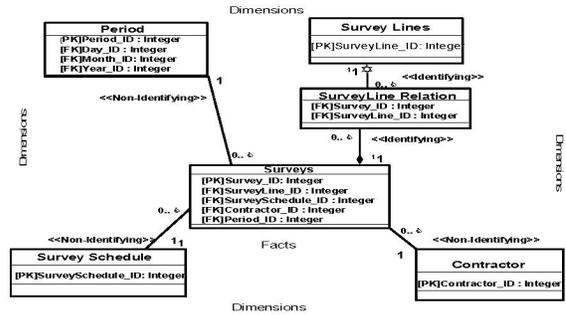


Fig. 7: A snowflake data model with normalized period dimension associated (many-to-many) with surveys object

D. Object class hierarchies

Data mining discovers both corporate and operational information from the oil and gas company data that is hidden at the bottom of the operational sub-systems. Demography data can be built based on the contractor dimension in different hierarchies. Contractor is grouped according to the city, or state or zip code, then according to country as shown in Fig 8.

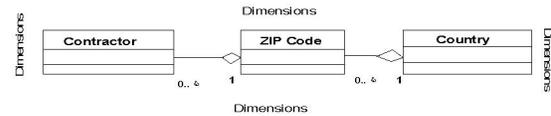


Fig. 8: The hierarchy of “contractor” object

The hierarchy is specified using an aggregation which defines the containment. The country contains the zip codes and the zip code will link to composite attribute, such as address of multiple contractors. Using aggregation, the dimension table is used at any level as defined in Fig. 9. Contractor, zip code, or country attributes can slice the analytical space.

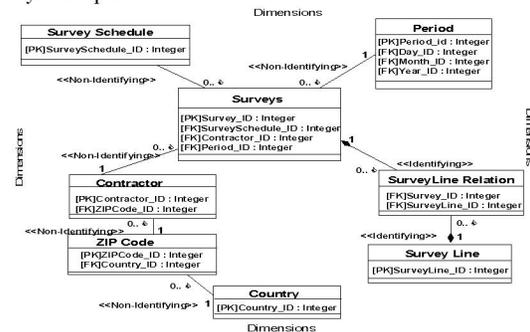


Fig. 9: A snowflake schema implementing aggregates of “contractor” object

IV. ANALYSIS AND DISCUSSIONS

The basic star schema creates a multidimensional space (often called dice), using the basic capabilities of a relational database utilities [12]. One needs to understand a multidimensional space. A multidimensional analysis space is depicted in the Fig. 10. A geometrical dice is an example of three dimensional space with all three dimensions of the same size. Imaging a cube with each object class dimension of three units, we get $4 \times 4 \times 4 = 64 = 256$ cells of equal structure. The multidimensional analysis space (or a data warehouse dice) differs just in details from a geometrical space.

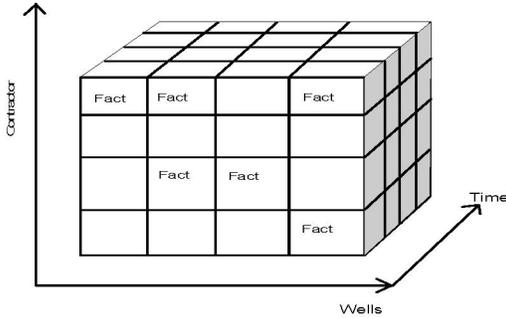


Fig. 10: A dice with dimensions wells, contractor, time

However, the dimensions are not just limited to just three. It is not easy to handle a cube with several object class dimensions, which often result in most of the implementations, limited to six or seven object dimensions. However, one should never expect a good graphical representation of more than three object dimensions. All object dimensions are not of the same size and unit. The size can differ from few units to several millions of units. The units can be period (day, quarter, month or year), contractor, wells or an exploration department, with several fact tables (see Fig. 10). The data cube needs much memory to store all the facts. For this reason, we design multidimensional schemas in a warehouse environment, optimizing storage capacity and preserving the flexibility of data structuring.

The data warehouse developed using the relational objects of the oil and gas data items, has several architectural views. With the design of data structures designed and application developed, for oil and gas company, these architectures possess great flexibility. An example of architecture with the positioning of data warehouse and data marts is shown in Fig. 11.

The purpose of mapping objects of oil and gas company data is to keep the operational managers active and sharp in their managerial decision support to act promptly. Million dollar decisions are taken based on the information processed by data warehouse with accountability and it provides an added value in return, tens of millions of dollars saved. Data objects from different warehouse marts, such as exploration, drilling and production are integrated as

examined in Fig 12.

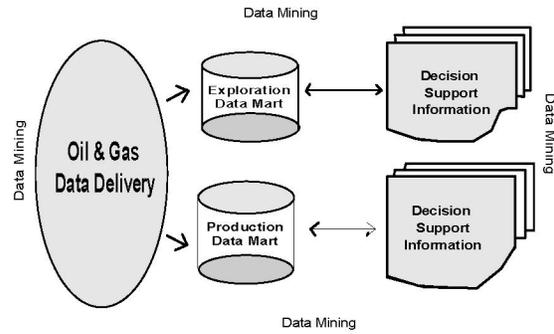


Fig. 11: Position of the data mart within an oil and gas company

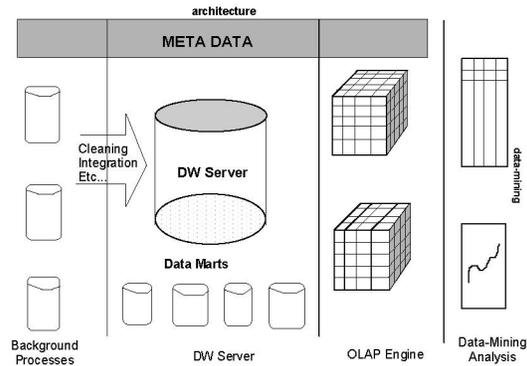


Fig. 12: Integration of data objects from different marts

On Line Analytical Processing (OLAP) [7] has capability to look into oil company's data object models in depth, map and present them in interpretable ways by decision makers.

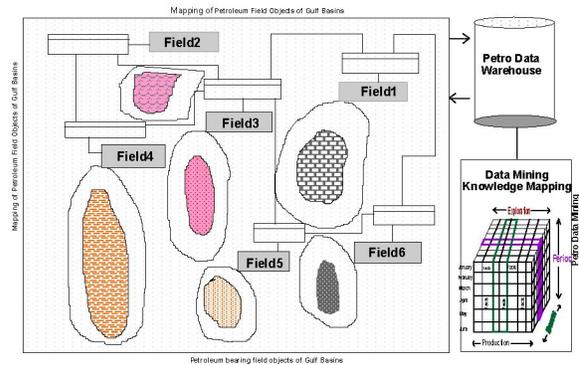


Fig. 13: Interoperability of data objects among petro-fields

Two of such situations discussed in this paper, such as, contractor-surveys, operator-wells problems have been processed and interpreted as shown in Figs. 16, 17a and 17b. Oil-field base data objects are reused (interoperability)

among multiple petroleum fields (Fig. 13), so that knowledge built from data integration, is interpreted in terms of drillable exploratory or development location as given in an OLAP model as demonstrated in Fig 14.

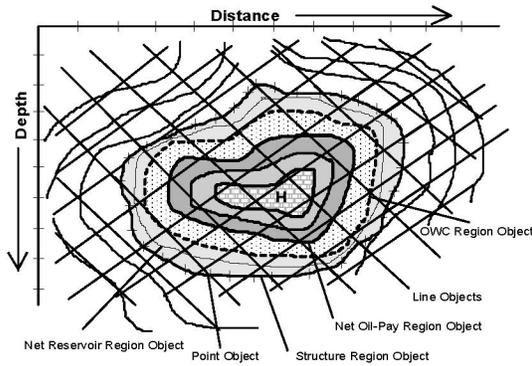


Fig. 14: OLAP model for petroleum exploration

Exploration managers are responsible for providing the valued processed exploration information, so that critical decisions made in planning new exploratory or development wells in the frontier oil bearing sedimentary basins are accurate and precise. Fig.15 exhibits the interoperability of petroleum data objects among basins, when data objects are conceptualized using petroleum ontology [9].

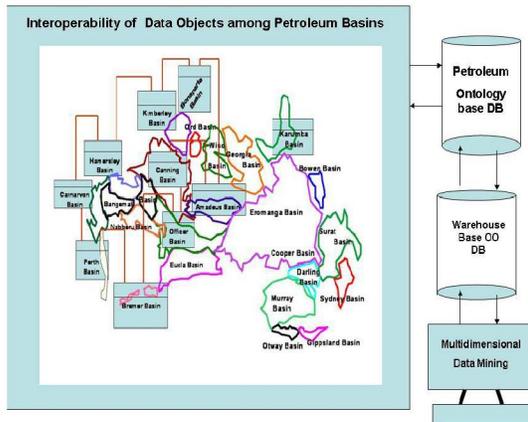


Fig. 15: Interoperability of petroleum data objects among petroleum bearing sedimentary basins

All the survey information processed by OLAP is presented by different combination of aggregated views as shown in Fig. 16.

Well facts stored in the oil and gas data warehouse or data marts, are processed by OLAP procedure [7] and presented in aggregated data views convenient to interpret and extract knowledge from the past historical oil and gas business data.

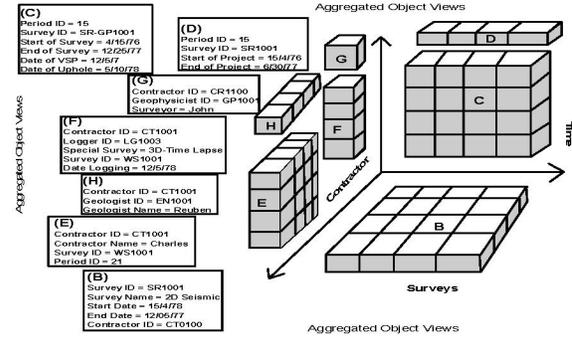


Fig. 16: Aggregated computed views of "survey" object

Multidimensional object data views for super class object "wells" are shown in Figs. 17a and 17b. All the available data and information are now integrated (Fig.16) and available in a centrally located enterprise data warehouse (EDW), making it easy for all managers to take timely decisions. Different data warehouse architectures may be tried creating several data marts that can map and process individual operational units' object classes and make them available to exploration managers. Similar data marts may be initiated for drilling, production, marketing, human resources and other support engineering class objects.

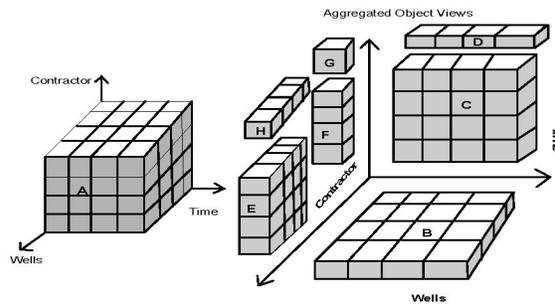


Fig. 17a: Example of 3D data cube showing three dimensions wells, contractor and time

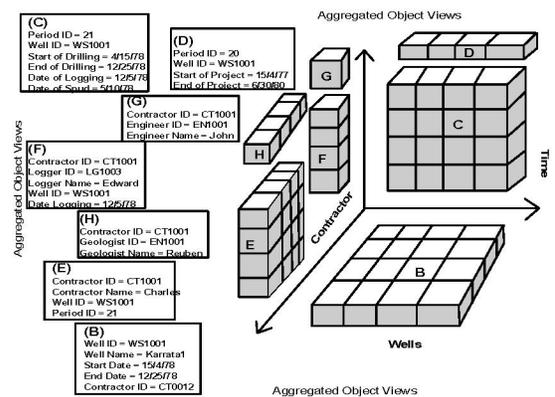


Fig. 17b: Computed aggregate views of "wells" object

V. SUMMARY AND CONCLUSIONS

The object class model describes an integrated multidimensional data structuring in a data warehouse environment. The object class models described in this paper can easily be implemented in an oil and gas industry. The data models are generated for specific classes of oil and gas data object classes. For surveys – contractor, wells - operator problems, star and snowflake multidimensional structuring appear to be consistent. It is appropriate to argue multidimensional structuring in a data warehouse environment offers better data integration functionality, the database administrator needs to tune the data warehouse for multi users. This paper analyzes the oil and gas company's business data object classes; develops logical models for implementing them in a warehouse. This paper establishes applicability and feasibility of object class data modeling and mapping procedures for oil and gas Company. Millions of facts and hundreds of dimensions have been identified from operations such as exploration, drilling, production and marketing classes, in which several other sub-classes, are interpreted as objects. This paper restricts to mapping of surveys and wells data objects that belong to super class exploration object. There is future scope and application of these logical modeling concepts in other areas of drilling, production, marketing class objects and integrating with several other object classes from other support services of oil and gas company, thereby drastically reducing the complexity of data items and enhancing the information sharing and data integrity among operational units of an oil and gas company.

REFERENCES

- [1] D'Orazio, R., and Happel, G. (1996). Practical Data Modelling for Database Design. The information Technology Series, John Wiley & Sons Australia Ltd., 180-280p.
- [2] Gornic, D (2000) Data Modelling for Data Warehouses, Rational Software White Paper, www.rational.com/worldwide.
- [3] Hoffer, J.A, Presscot, M.B and McFadden, F.R (2002). Modern Database Management, Sixth Edition, Prentice Hall, 260-520p.
- [4] Huynh, T.N. Mangisengi, O. and Tjoa, A.M., (2000). Metadata for Object-Relational Data Warehouse., Proceedings of the International Workshop on Design and Management of Data Warehouses., Stockholm, Sweden., June 5-6.
- [5] King, E. (2000). Data Warehousing and Data Mining, Computer Technology Research Corporation, 50-110p, <http://www.ctrcorp.com>.
- [6] Longley, I.M. Bradshaw, M.T. and Hebberger, J. (2001) *Australian petroleum provinces of the 21st century*, in Downey, M.W. Threet, J.C. Morgan, W.A (2001) *Petroleum provinces of the 21st century, AAPG Memoir 74*, pp.287-317.
- [7] Moody, L. D and Kortink, M.A.R (2003). From ER Models to Dimensional Models: Bridging the gap between OLTP and OLAP Design, Part1 and Part 2, *Business Journal Intelligence*, Summer Fall editions, Vol. 8(3), <http://www.tdwi.org>.
- [8] Nimmagadda, S.L, Dreher, H. and Rudra, A. (2005a) Data Warehouse Structuring Methodologies for Efficient Mining of Western Australian Petroleum Data Sources, published in the *proceedings of IEEE conference of INDIN 05, Perth, Australia*
- [9] Nimmagadda, S.L., Dreher, H. and Rudra, A. (2005a) Ontology of Western Australian petroleum exploration data for effective data warehouse design and data mining, published in the proceedings of the 3rd international IEEE conference on Industrial Informatics, held in Perth, Australia, August.
- [10] Nimmagadda, S.L, Dreher, H. and Rudra, A. (2005b) Warehousing of Object Oriented Petroleum Data for Knowledge Mapping, published in the *proceedings of 5th International Conference of IBIMA*, Cairo, Egypt
- [11] Ozkarahan, E., (1990). Database Management, Concepts, Design and Practice., Prentice Hall Publications., 15-200p.
- [12] Pujari, K. A. (2001) Data Mining Techniques, Universities Press, p. 7-67.
- [13] Rumbaugh, T., M. Blaha, W. Premerlani, F. Eddy, and W. Lorensen. (1991). Object-Oriented Modeling and Design. Englewood Cliffs, NJ: Prentice-Hall.
- [14] Smith, J.R. (2004). A Wavelet Frame Work for Adapting Data Cube Views for OLAP, *IEEE Trans. On Knowledge and Data Engineering*, Vol. 18(5), pp.552-565.
- [15] Zaima, A. and Kashner, J., (2003). A data mining premier for the data warehousing professional, *Business Intelligence Journal*, Vol 8(2). <http://www.tdwi.org>.