Hierarchical Approach to Data Extraction using UML 2.0

Prof. (Dr.) Payal Pahwa
CSE & IT
Bhagwan Parshuram Institute of Technology
Delhi, India

Geetika Chaudhary
Khyati Jain
Neha Sharma
Ruchira Gupta
IT, Bhagwan Parshuram Institute of Technology
Delhi, India

Abstract - The concept of data warehousing includes different algorithms and tools for combining data from multiple sources into a single repository called data warehouse. Various ETL tools are used to extract the data from these multiple sources. The paper proposes an object oriented approach to model the process of data extraction as part of extraction, transformation and loading process. The hierarchies of each data element can be explicitly defined, thus highlighting the data granularity and hence simplifying the data extraction process. A case study to illustrate the concept has been depicted using University Management System. The approach has been modelled using Unified Modelling Language 2.0. University Management system has been used as a case study to illustrate this modelling concept.

Keywords - Data Warehouse, Unified Modelling Language, Extraction Transformation Loading, Object Oriented Paradigm.

I. INTRODUCTION

A data warehouse is a subject oriented, integrated, non-volatile and time variant collection of data in support of management’s decisions[1][7] . The main aim of the data warehouse is to facilitate the decision making process.

Extraction-transformation-loading (ETL) processes are responsible for the extraction of data from heterogeneous operational data sources, their transformation which involves conversion, cleaning and their loading into a warehouse. ETL functions form the prerequisites for the data warehouse information content. Thus, the ETL activities are the most time-consuming and human-intensive, thereby making them a crucial factor in the success of data warehouse development projects. Therefore, a correct framework of these processes is essential for the improvement of data quality. However, not much work has been done in this area.

ETL processes can be represented pictorially by mapping them to UML (Unified Modelling Language) diagrams. In this paper an approach based on the use of Unified Modelling Language and Object Oriented Paradigm (OOP) has been presented to accomplish the modelling of data extraction process. UML is a standardized general-purpose modelling language in the field of software engineering, managed and created by Object Management Group[8]. The Unified Modelling Language is used to specify, visualize, modify, construct and document the artifacts of an object-oriented software-intensive system under development. UML offers a standard way to visualize a system's architectural blueprints[2]. Using UML we can pictorially map the data extraction process in the data warehouse development and make it more clear and simplified.

We have used object oriented methodology in addition to UML to extend the data extraction process and show the various levels of granularity from where the source data is being accessed. We have adopted UML as our modelling language owing to its wide acceptance and the possibility of using and extending various complementary diagrams for modelling various system aspects.

II. LITERATURE REVIEW

[3] presents a framework for the design of the data warehouse back-stage. Here the key observation was that the task fundamentally involved dealing with the specificities of information at very low levels of granularity including transformation rules at the attribute level. Specifically, [3] presented a disciplined framework for the modelling of the relationships between sources and targets in different levels of granularity, including coarse mappings at the database and table levels to detailed inter-attribute mappings at the attribute level. In order to accomplish this goal UML has been extended to model attributes as first-class citizens.

In [4] an object-oriented approach to accomplish the data extraction modelling of extraction transformation-loading process has been proposed. The data extraction scenario consists of data staging area, heterogeneous information sources, wrappers, monitors, integrator, and source identifier. All the above mentioned entities have been modelled using Unified Modelling Language 2.0. Banking system has been used as an application to illustrate the modelling.
Krishna and Sreekanth [5] have discussed a web-based framework model for representing extraction of data from one or more data sources and use transformation business logic and load the data in the data warehouse. A new feature of entire loading process of data movement between source and target system has also been made visible to the users. In addition a reporting capability to log all successful transformations is provided.

[6] proposes, the use of UML 2.0 extensibility mechanisms to provide security in the conceptual modelling of the data warehouse. Among the extensibility mechanisms incorporated are Stereotypes, tagged values and constraints etc.

All the above discussed approaches do not consider certain object-oriented concepts like aggregation, composition, generalisation and association which can be further used to simplify and to make the process of data extraction more transparent. This paper focuses on these aspects.

III. UML MODELLING AND ETL

ETL basically refers to extraction transformation and loading process. ETL functions reshape the relevant data from the source systems into useful information to be stored in the data warehouse. If the source data is not extracted correctly, cleansed and integrated in the proper formats, query processing would yield incorrect results. ETL functions form the pre-requisites for the data warehouse information contents. ETL encompasses the area of data acquisition and data storage. These are back-end processes that cover the extraction of data from the source systems. They include all the functions and procedures for changing the source data into the exact formats and structures appropriate for storage in the data warehouse database. After the transformation of data, these processes consist of all the functions for physically moving the data into the data warehouse repository.

Data extraction forms the initial step of the ETL process. The data is extracted from various disparate sources for further transformation and loading into the data warehouse. In this paper the extension mechanisms of UML (stereotypes, constraints, tagged values, notes etc) along with object-oriented concepts of generalisation, aggregation, composition and association have been incorporated to propose an efficient way of data extraction. Here we are extending the data sources and classifying every data source into different categories. This categorical representation helps in accessing every specific detail of the data sources. This enables us to define various hierarchies present in the source data and hence allows the users to access the data at the lowest level.

Data Granularity

Data Granularity models the level of detail present in the data sources. The lower the level of detail the finer the data granularity. Rolling up and drilling down of data becomes more efficient by defining the data grain at each level of hierarchy. We have categorized the data sources to exhibit the fine-grained data source which helps in efficient extraction of relevant and detailed data from different databases. By using object-oriented concepts we have defined relationships between various data entities which help us to understand the interdependence between these data entities on every level of hierarchy.

For elaborating our concept we have used the case study of University Management System in this paper. We have used various UML diagrams - class diagram, use case diagram and sequence diagram to explain the Extraction, Transformation and Loading processes within a University.

IV. INTRODUCING OBJECT-ORIENTED FEATURES IN UML

A. Class diagram

The data sources for the University Management System can be the operational database, archived data source external data source or internal data source. Here we have considered only the operational databases. The operational data sources have been categorised to form hierarchies of data sources. Traversing the hierarchies in a top-down fashion provides us with the summary data at the higher levels and detailed data as we move towards the lower levels of the hierarchy. Using this hierarchical approach, every data entity can be traced back to identify its ultimate source which helps in authentication of data.

The data sources connect to Wrapper component. This component connects the source and extracts data from it. The monitor module thereafter detects the changes in the data (if any) and further reports them to the integrator module. If any new data source comes into picture in the ETL process or any modification of data take place, the source identifier passes the modified data to the integrator. Later the information is loaded into the Data Staging Area (DSA) by the integrator. DSA is that area where the data extracted from various heterogeneous data sources is stored. Therefore, it is called intermediate data storage. This intermediate data acts as an input for the cleansing and transformation process of the ETL, and finally the output of the cleansing and transformation processes is loaded into the data warehouse [4].

The class diagram (Figure 6.) reflecting all the enhanced features is shown below. The major classes are wrapper, monitor, integrator, source identifier and DSA. The methods and attributes of each and every class are mentioned clearly in the diagram. Relationship between each class is governed by a set of cardinalities which maps one class to another as shown in the class diagram.

The class diagram as shown below is composed of OOP features like composition, aggregation, generalisation etc. which help us to understand the hierarchies better. The
UML extensible mechanisms like notes and constraints have also been incorporated in the class diagram.

1) **Composition**: Object composition is a way to combine object(s) into another object, implying ownership. In this paper, the concept of composition (Figure 1.) has been used to show how the database of one department is composed of the data extracted from the databases of its sub-departments, as shown in the figure. It simplifies the process of retrieving summary data as well as detailed data from each level of granularity.

2) **Aggregation**: The concept of aggregation is very similar to that of composition, the difference being that aggregation does not imply ownership. Here the concept of aggregation (Figure 2.) has been implemented using a bottom-up approach where the data is travelling from the subclass ‘person’ to its parent classes – ‘CSE Class’, ‘MBA Class’ and ‘Electricals-Electronics Class’. It helps in capturing all the details present in different levels of hierarchy efficiently.

3) **Generalisation**: Generalisation of a concept is the extension of the concept to a less specific criterion. The concept of generalisation (Figure 3.) follows a bottom-up approach. This paper proposes the use of this concept in the process of data extraction as shown below. It helps in defining the hierarchies of data sources which is further used to get a clear picture of data classification.

4) **Association**: An Association is a channel between classes through which messages can be passed. The concept of association (Figure 4.) has been shown between the student class and faculty class. This helps in creating a direct relationship between the student and the faculty class, making the data extraction faster and easier.

**B. Use case diagram**

Similarly, the use case diagram (Figure 5.) for the University Management System has been made. Object oriented features and UML extensibility mechanisms (tagged values, relationships) have also been incorporated.

**C. Sequence diagram**

The sequence diagram (Figure 7-9, The Sequence Diagram Fig. 8 is in continuation with Fig. 7 and the sequence diagram Fig. 9 is in continuation with Fig. 8) has been elaborated to depict the way in which the data travels from the lowest levels of hierarchy to higher levels as independent data entities. The transaction of the data entities between various sources is explicitly and clearly reflected in the sequence diagram.

---

**Figure 1. Composition**
Figure 5. Use-Case Diagram
Figure 6. Class Diagram
Figure 7. Sequence Diagram 1

Figure 8. Sequence Diagram 2

Figure 9. Sequence Diagram 3
V. CONCLUSION

It is observed that the process of data extraction can be simplified and can be made more transparent by incorporating object oriented features while mapping the extraction process to UML diagrams. A picture says more than words. UML enables one to get a pictorial view of the entire extraction process which makes it more suitable for analysis. The data repository of the University Management System can be categorised and further elaborated to lower levels of granularity, highlighting data grain at each level of hierarchy. The object oriented features of generalisation, aggregation, composition and association have been incorporated. These features help in identifying and establishing the relations between various data sources, thereby making the process of data extraction more reliable. Solving queries has been made easier because data sources at every level of granularity can be identified and targeted directly.

REFERENCES


