TRIBHUVAN UNIVERSITY

INSTITUE OF ENGINEERING

PULCHOWK CAMPUS

DEPARTMENT OF ELECTRONICS AND COMPUTER ENGINEERING

Warehouse based Intelligent Banking Transaction Analysis System

BY

Bibek Subedi (16206)
Dinesh Subedi (16212)
Jivan Nepali (16217)
Laxmi Kadariya (16218)

TRIBHUVAN UNIVERSITY

INSTITUTE OF ENGINEERING

PULCHOWK CAMPUS

DEPARTMENT OF ELECTRONICS AND COMPUTER ENGINEERING

The undersigned certify that they have read, and recommended to the Institute of Engineering for acceptance, a project report entitled "Warehouse based Intelligent Banking Transaction Analysis System" submitted by Mr. Bibek Subedi, Mr. Dinesh Subedi, Mr. Jivan Nepali & Mr. Laxmi Kadariya in partial fulfilment of the requirements for the Bachelor's degree in Computer Engineering.

Supervisor, Prof. Dr. Subarna Shakya

Asst. Dean

Institute of Engineering, Pulchowk Campus

Internal Examiner, Dr. Arun Timilsina

Head

Department of Electronics & Computer Engineering, Pulchowk Campus

---

External Examiner, Roshan Regmi

Head

Information Technology, NMB Bank Ltd.

DATE: August 30, 2013

COPYRIGHT

ACKNOWLEDGMENT

We owe our deepest gratitude towards the department of Electronics and Computer Engineering, IOE for providing us the wonderful opportunity to undertake this project.

It is our honor for us to express respect and gratitude to our project supervisor Prof. Dr. Subarna Shakya for providing us with the constant support and feedback related to the project. It was his valuable suggestions that helped us to cope up with the emerging obstacles during the development of this project.

We are truly thankful to Yomari Inc. Pvt. Ltd. for their faith upon us and tremendous support and motivation by our mentor Er. Yam Bahadur Gurung.

Last but not the least, we would like to thank all our teachers and colleagues for their direct and indirect help related to our project.

Any kind of suggestion or criticism will be highly appreciated and acknowledged.

Thanking you.

Bibek Subedi (16206)

Dinesh Subedi (16212)

Jivan Nepali (16217)

Laxmi Kadariya (16218)

ABSTRACT

The data warehousing & data mining have changed the decision making process in modern day business environment, which basically equip the business companies to reach their customers with the right product and right offer at the right time. This project is mainly concentrated to provide solutions to ATM card fraud detection, customer churn behaviour prediction, yearly/quarterly financial reporting, trend analysis, geo-demographic analysis & time-series predictive analysis in a banking system. The project has been implemented with a completely warehouse based business intelligence tools with some of the data mining algorithms implemented during reporting phase for churn prediction and ATM fraud detection.

The banks have to publish their quarterly/yearly financial reports & it has been customary to the banks to analyze the transaction behaviour of their customers using business intelligence solutions to gain the overall insights & attract newer customers to their services. Our system is designed as a solution to these requirements. It provides information with tabular as well as with graphical visualization to clearer the concept. Additionally, the system has been designed to be flexible enough to incorporate visualizations with different performance indicators.

Keywords: Banking Analysis, Banking Data Warehousing, Financial Reporting & BI, ATM Fraud Detection, Churn Prediction, Time-series Analysis in Banks

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

ATM	Automated Teller Machine
BI	Business Intelligence
CART	Classification and Regression Tree
ETL	Extract, Transform and Load
OBIEE	Oracle Business Intelligence Enterprise Edition
ODBC	Open Data Base Connectivity
OLAP	OnLine Analytical Processing
OLTP	OnLine Transaction Processing
ORE	Oracle R Enterprise
SQL	Structured Query Language
SVM	Support Vector Machine
TCP/IP	Transmission Control Protocol / Internet Protocol
WIBTAS	Warehouse based Intelligent Banking Transaction Analysis System

1 INTRODUCTION

1.1 Background

In Nepal, the number of banking customers are increasing day by day. As the customers' number increases, the number of transactions will also increase and more transactions as well as customer's data will be added into the bank's database. This results into difficulty in managing the transaction and keeping the sound relationship with each customer. The customer dissatisfaction leads into the continuous loss and even collapse of the organization. So managers and executives of organization must be able to know the transaction behavior of his customer and must maintain a family relationship with all the customers. It costs very high if the managers use traditional approach without using the new tools and technology. Our system visualizes and prepares report for the churn behavior, fraud detection and customer relationship management (CRM) in a banking system.

Customer is the heart and soul of any organization. The era of globalization and cut throat competition has changed the basic concept of marketing, now marketing is not confined to selling the services to the customers, but the objective is to reach to the hearts of the customers so that they feel belonging towards the organizations and hence should remain the loyal customers. But, the ever growing databases make it difficult to analyze the data and to forecast the future trends. The solution lies in the use of Data Mining tools for predicting the transaction behavior of the customers. [1]

The fraudulent behavior in ATM transaction are increasing each day as the number of customer and transactions increases. Customers are losing their money although they don't withdraw money themselves. The groups of hackers and even authorized bank personnel are involving in the fraudulent ATM activities. Recently many banks of Nepal including Siddhartha Bank, Nabil Bank, Nepal Investment Bank, Laxmi Bank suffered from the fraudulent banking transactions. [2] This makes banks to lose their reputation to customers and heavy loss of money.

Customer satisfaction is challenging task for very bank. Different customer has different sort of interest. The bank should offer various schemes to their customers based on their interest and needs. If the bank fails on full filling its customers need, then it is obvious that the customer will cease the relationship with the bank. Acquiring the new customer is far costlier than retaining the existing customer. [3] So it is really challenging problem for every bank to analysis the transaction behavior, personal interest and needs and updates their policy accordingly.

The personal customer profile and their transaction history is main data source of this project. It analyses the transactions over past several years and generates various kinds of report including trend analysis, time series analysis and customer segmentation. Based on the customers ATM transaction history, the project predict the possible fraud transaction and generates different types of reports. Similarly it predicts the customer churn behavior based on the past churned customer behavior using data mining tools and techniques.

1.2 Problem Statement

According to Nepal Rastra Bank there are total 32 Commercial banks, 88 Development banks and 69 Finance companies in Nepal in 2012 [4]. Every Commercials banks and almost all development banks use ATM service. This number is increasing every year. So there is a very competitive market in banking area of Nepal. Lots of Transactions occurs day by day and each day a massive data is collected. The banks are facing a big problem in managing their customer because the bank cannot talk with each customer and ask for their problems individually. This may make customers to feel like the bank is neglecting them. So, we discover that there is a need of Customer Relationship Management (CRM) in Nepalese banking sector.

There are many customers in Nepal who transfer their transaction from one bank to another frequently. The banks are unaware about the reason why their customers are churning. There are different churners like switchers, movers, loyalists, and diehards. Different categories of customer churn for different reasons. The lack of ATM machine in different locations, unreliable interest rates, lack of services like e-banking, SMS based banking etc. are some of the problems.

The ATM fraud is another problem in banking sector of Nepal. Many fraud cases are arising day by day. Himalayan bank ATM fraud is latest example in which the bank is suspected to have lost more than 5 million rupees [2]. The ATM skimmers use different tricky methods to get the debit card numbers and personal identification number (PIN) from the innocent users.

1.3 Motivation

Banking has become an important feature which renders service to the people in financial matters, and its magnitude of action is extending day by day. It is the major institutional system which carries out the financial flow within the economy. Recently the action of financial transaction has been increased in a rapid manner which results in acquiring huge amount of daily data. Moreover with more and more banks being increased, huge competition is resulted among them.

The traditional analysis with this huge data is not appropriate to implement new strategy and plans. Today, the world runs in technology. Technology is meant to be for all, but in reality it never always gets to the reach of all. Sometimes it‘s the gap between the technical experts and the domain experts; sometimes it‘s the cost. It is of less pain to know that technology was inaccessible because it was too expensive to afford. But it‘s a true tragedy to know of instances where technology was far-fetched despite of desperate demand. It‘s the responsibility of all the technocrats to understand the sentiments of these fields and contribute to bridge the gap.

Banking has huge daily transactional data. Information can be mined out of the data. Our effort is to demonstrate how information can be mined out of dumped data. With this information, organization can provide appropriate service to appropriate costumers at appropriate time and acquire the existing costumer within itself and get the organizational goals.

1.4 Objectives

i. To help keep and provide efficient visualization quarterly, half and financial report about ATM and debit transaction of Bank.

ii. To identify the patterns of fraudulent behavior in ATM withdrawal transactions- location-wise and time-wise.
iii. To help analyze and prediction about their customer churn behavior and assist in decision making process.

1.5 Scope

i. Scope of this project is to build the central repository of data for Bank to deliver business intelligence, save time, enhance data query and consistency.
ii. This software provides analysis of customer transaction, predict the customer behavior and identify the fraudulent patterns in ATM withdrawal transaction. It can't integrate the data of multiple banks and compare the result.
iii. This software would address the issue of changing report format and change in rules and regulation provided by bank.
iv. This application can be used as decision support, information and analysis system, and client can't add, delete and update the records.
v. Target client of this application is banks and other financial institutions helping them to retain their existing customer and attract new customers to their services.

1.6 Overview of the Report

In Chapter one, the main concept of application has been covered. Here, the contextual problems in the banking sector have been discussed. The main objectives of the application have been mentioned and its scopes clarified.

Chapter two discusses about the literature review. In this chapter, the related past & ongoing research in the field of banking data warehousing & data mining has been mentioned.

In Chapter three, theoretical background relating ATM frauds, customer churn behaviour & different analyses in banking such as trend, geo-demographic & time-series analyses have been covered.

In Chapter four, technical background has been covered with different anomaly detection algorithms, CART algorithm & OBIEE.

In Chapter five, System Analysis has been discussed. Functions that are required to be performed by our system, specific user requirements and other non-functional requirements like reliability, performance and security requirements are listed and then described in detail. Feasibility assessment has also been covered, including operational, technical and economic grounds.

In Chapter six, System Design has been covered. In this chapter, Construction as well as description of Use case, Deployment, Component and Sequence diagram has been done. System Architecture has also been described in detail. Schema Diagram and Class Diagram have been included too.

Chapter seven discusses about the implementation issues of the project including data collection phase, ETL, data warehouse design & implementation of financial & BI as well as ATM fraud detection & churn prediction.

Chapter eight covers different types of testing such as black box, performance, alpha testing etc. It also discusses about the some of the problems faced while conceptualizing & realizing the project.

In Chapter nine, Results and Conclusions part has been covered. The limitations and future extendibility of our application has been mentioned, along with conclusion.

2 LITERATURE REVIEW

Several banks have been established throughout the world with specific motto and aim .These banks have been using several software to be in the right track of success. Different research and papers have been published regarding the data warehousing and means to improve the several aspects in the banking transaction. In order to implement any strategic plan the banks don't directly circulate it, rather it first study the current situation with the help of available data and visualize the different trends and issue a plan and strategy to achieve the goal in particular field. Due to ongoing process of globalization and increasing competition, models which could enable the fast and rapid analysis tool is very important. Banks now use the data warehouse to support strategic business goal such as compliance, cost reduction and increased profitability. [5] Data warehouse in banking system is a state of art tool that provides the data that employees need for faster, more effective decision and customer service analysis. The data warehouse provides information and analysis that supports customer service and knowledge, regulatory requirement and risk management, and analytics by product, channel and geography. Employing data warehouse provides new view of customer information which supports marketing campaigns, business decision making and service approaches. [6]

Banking is the vast and critical field. "A small leak will sink a great ship" is very meaningful in the banking area. Banks have several branches in different places. Huge number of transaction takes place daily. Today one don't need a gun to rob a bank any more. Bank robber's most powerful weapon is fraudulent activities and behavior which banking system is unable to detect. Thus Fraud pattern detection has become an essential need for every financial institution today. Analytics is a way to actually detect fraudulent patterns of behavior efficiently. [7] The output of the analytics includes a customer score, which determines how the activity corresponds to the customers' actual behavior, and a transaction fraud score, which determines the fraudulent nature of the transaction. Furthermore, Bank requires a large database to save and handle the data. So it is very important to manage the bank's data warehouse and business intelligence to support the daily data. Further when any strategic analysis have to be made, manual analysis is impossible.so it is very necessary to design a software which can easily handle the large amount of data available and visualize the data pattern. Such software saves the

time and helps getting towards the designed goal. Further data warehouse supports the banks to be more data-driven to support new initiatives, support its strategy and governance requirement. Different banks in India and throughout the world are using the banking data and trends analysis tools and software to analysis the situation and getting benefit through it. Today data warehouse is operationally entrenched in the banks, serving all departments with data, insight and business value in customer service, regulatory requirements and risk mitigation and power analytics. [8]

So finally our project is a step recognizing the growing use of the data warehouse throughout the world. It is a report visualization, fraud pattern detection and churn behavior prediction tool for end users. It supports financial reporting, fraud pattern detection, churn prediction and provides a single view of all bank's information. Such tools can be very useful in Nepali banks and can utilize its feature to support its goal.

3 THEORETICAL BACKGROUND

3.1 ATM Card Transactions & Frauds

Nepalese use ATM cards several times a day at ATMs & POS terminals in Nepal & abroad to see their account information and make withdrawals. While most of these transactions are problem free, there have been a growing number of cases of ATM card frauds in recent years using Visa and Master Cards. Because of this, many of the Nepalese banks have lowered their daily ATM withdrawal limit.

According to the Himalayan Times Daily, the investigation about ATM card fraud in NIBL conducted by CIB, Nepal revealed that fraudsters had withdrawn Rs. 122,000 with fake ATM cards from 16 different accounts of Nepal Investment Bank Limited. Police also said they had withdrawn money from Nabil, Siddhartha and Sunrise Bank of Nepal. Police also informed they had seized 28 fake debit cards & one ATM card of Punjab National Bank. [9]

In order for fraud to occur, a thief needs both PIN and the magnetic stripe information on the back of ATM card. The PIN is not stored on the card's magnetic stripe. So, if a card is stolen or duplicated, the thief has to find some way to get the PIN. Common methods used to steal or duplicate cards and obtain the PIN are:

- Easily Identifiable PINs
- Surf & Pick Pocket
- Card Jam
- Skim & Clone

Once the fraudster gets fake ATM card(s), then the transaction behaviour of the particular account number changes drastically. So, based on the change in transaction behaviour, our SVM-enabled system analyzes the transactions & assigns fraud percentage to it.

3.2 Banking Customer Churn Behavior

Churn in banking refers to a customer ceases his or her relationship with a bank. Reducing customer churn is a key business goal of every online business. [10] The ability to predict that a particular customer is at a high risk of churning, while there is still time to do something about it, represents a huge additional potential revenue source for a bank. Besides the direct loss of revenue that results from a customer abandoning the business, the costs of initially acquiring that customer may not have already been covered by the customer's spending to date. Furthermore, it is always more difficult and expensive to acquire a new customer than it is to retain a current paying customer. In order to succeed at retaining customers who would otherwise abandon the business, our system will make the managers and executives of bank to be able to predict in advance which customers are going to churn. [11] Our project throws light on the underlying technology and the perspective applications of data mining in predicting the churn behavior of the customers and hence paving path for better customer relationship management in today's competitive banking environment.

3.3 Trend Analysis

Trend Analysis is the practice of collecting information and attempting to spot a pattern, or trend, in the information. Trend analysis is often used to predict future events. Trend analysis is one leg of analytic triangle and is used for financial surveillance and monitoring, for forecasting, for program evaluation, for policy analysis. [12]

Trend analysis is based on the idea that what has happened in the past gives traders an idea of what will happen in the future. In trend analysis financial statement of the banks are compared with each other for the several years, months etc. Trend analysis has the great advantage that it is possible to forecast the future cash flow based on data available of the past. With the help of trend analysis, we can track the variances to add performance.

3.4 Geo-Demographic Analysis

In the highly competitive market of the banking industry, it has become difficult for each player to compete in the market for a long-term period with the same products/services. Thus development of a suitable marketing strategy over time is required. Right marketing strategies will helps companies to achieve marketing and corporate objectives which in turn will create a competitive advantages in the market. Market analysis helps bank identify customer needs, preferences and find new marketing opportunities.

Market analysis is analysis of the customers to identify customer needs, characteristics, and behaviour. Market analysis is a process of analyzing customer on the basis of several factors geographic, demographic, psychographic, and behavioral. [13]

Geographic analysis divides a broad market into geographic units such as continents, nations, region, and cities. The philosophy behind geographic analysis is that people who live in the same area are bound to share similar culture and experience the same experience which makes them to acquire similar needs.

Demographic analysis is the process of analyzing a customer using demographic variables such as age, sex, marital status, family type/size, family life cycle, occupation, etc.

3.5 Time-series Predictive Analysis

The ability to compare business performance with previous time periods is fundamental to understanding a business and such analysis of comparison of performance over previous time is time series analysis. Time comparison enable businesses to analyze data that spans multiple time periods, providing a context for the data.

A time series is a sequence of data points, measured typically at successive points in time spaced at uniform time intervals. Time series analysis comprises methods for analyzing time series data in order to extract meaningful statistics and other characteristics of the data. Time series data have a natural temporal ordering. [8] This makes time series analysis distinct from other

common data analysis problems, in which there is no natural ordering of the observations (e.g. explaining people's wages by reference to their respective education levels, where the individuals' data could be entered in any order).

4 TECHNICAL BACKGROUND

4.1 Bash Shell Scripting to address ETL

Bash shell scripting is used to validate the files. In the case of validating the files, we have used commands like grep, awk, sed, find and other. It is also used to transfer files to server. Mostly, it is used to connect the oracle server and uploading the data to the database.

4.2 Data Warehouse Schema

We have used Snowflake warehouse schema for the warehouse design. This schema is diagrammed with each fact surrounded by its associated dimensions and those dimensions are further related to other dimensions, branching out into snowflake pattern. Use of snowflake diagram makes the database less redundant. [14] The design basically contains two types of table viz., dimension table and fact table.

4.2.1 Dimensions

We first identified the dimension of our requirements. As our objective is based on the fraud detection, churn analysis and CRM, we identified the dimensions like branch, time, customer, location, ATM, account etc. and the dimension tables were created. Dimension tables, also known as lookup or reference tables, contains the relatively static data in the warehouse. Dimension tables store the information that is normally used to contain queries. Dimension tables are usually textual and descriptive and we can use them as the row headers of the result set.

4.2.2 Facts

Next we identified the facts for our requirements. We identified and processed three facts table. Three facts tables are based on three primary objectives of our project. The general fact table contains the banking transaction information from which we generated different aggregates and showed them in the report. The next fact table is specific toward ATM card fraud detection and

contains the result of fraud pattern prediction. The last one is specific toward customer churn prediction and contains the result of churn prediction.

4.3 On-Line Analytical Processing

Online analysis and processing (OLAP) is part of the broader category of business intelligence, which also encompasses relational database, report writing and data mining. Typical applications of OLAP include business reporting for sales, marketing, management reporting, business process management (BPM), budgeting and forecasting, financial reporting. [15]

OLAP enables users to analyze multidimensional data interactively from multiple perspectives. OLAP consists of three basic analytical operations: consolidation (roll-up), drill-down, and slicing and dicing.

Consolidation involves the aggregation of data that can be accumulated and computed in one or more dimensions. For example, all month's withdrawal are rolled up to the year withdrawal to anticipate withdrawal trends.

By contrast, the drill-down is a technique that allows users to navigate through the details. For instance, users can view the month withdrawal by year withdrawal.

Slicing and dicing is a feature whereby users can take out (slicing) a specific set of data of the OLAP cube and view (dicing) the slices from different viewpoints.

4.4 Oracle Administration Tool to address Metadata Repository Design

4.4.1 Oracle BI Administration Tool

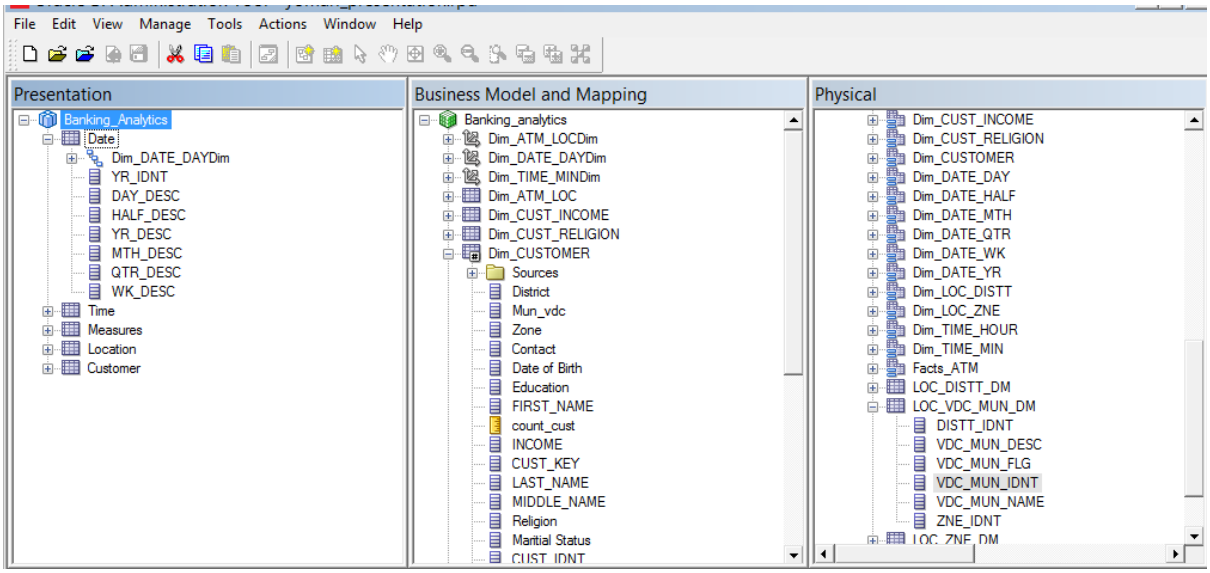

Figure 4.1: Oracle Administration Tool Showing three layers, namely physical layer, business model & mapping layer and presentation layer, in repository creation

The administration tool is used for

- Import metadata from databases and other data sources
- Simplify and reorganize the imported metadata into business models
- Structure the business model for presentation to users who request business intelligence information via Oracle BI end user tools.

We can develop a repository using Oracle Administration Tool. [16] The main window of the tool provides graphical representation of the three layers of a repository.

4.4.1.1. Physical Layer

The Physical layer defines the data sources to which Oracle BI Server submits queries and the relationships between physical databases and other data sources that are used to process multiple data source queries. The recommended way to populate the Physical layer is by importing metadata from databases and other data sources. The data sources can be of the same or different varieties. We can import schemas or portions of schemas from existing data sources. Additionally, we can create objects in the Physical layer manually. When we import metadata,

many of the properties of the data sources are configured automatically based on the information gathered during the import process. After import, we can also define other attributes of the physical data sources, such as join relationships, that might not exist in the data source metadata. There can be one or more data sources in the Physical layer, including databases, flat files and XML documents.

4.4.1.2. Business and Mapping Layer

The Business Model and Mapping layer of the Administration Tool defines the business, or logical, model of the data and specifies the mappings between the business model and the Physical layer schemas. This layer is where the physical schemas are simplified to form the basis for the users' view of the data. The Business Model and Mapping layer of the Administration Tool can contain one or more business model objects. A business model object contains the business model definitions and the mappings from logical to physical tables for the business model. The main purpose of the business model is to capture how users think about their business using their own vocabulary. The business model simplifies the physical schema and maps the users' business vocabulary to physical sources. [17] Most of the vocabulary translates into logical columns in the business model. Collections of logical columns form logical tables. Each logical column (and hence each logical table) can have one or more physical objects as sources. There are two main categories of logical tables: fact and dimension. Logical fact tables contain the measures by which an organization gauges its business operations and performance. Logical dimension tables contain the data used to qualify the facts.

4.4.1.3. Presentation Layer

The Presentation layer exposes the business model objects in Oracle BI user interfaces so that users can build analyses and dashboards to analyze their data.

4.4.2 Creating a Repository Using the Oracle BI 11g Administration Tool

The repository creation is the preliminary step in the creation of dashboard reports. Oracle BI metadata Repository is created using Oracle BI Administration Tool. During the repository creation, metadata is imported from data sources, simplified and reorganized into business model, and then structured the business model for presentation to users who request business intelligence information via Oracle BI user Interface.

4.4.2.1. Building the Physical Layer of a Repository

To build the physical layer of a repository, we perform the following steps:

- Create a New Repository
- Import Metadata
- Verify Connection
- Create Aliases
- Create Physical Keys and Joins

4.4.2.2. Building the Business Model and Mapping Layer of a Repository

To build the Business Model and Mapping layer of a repository, we perform the following steps:

- Create a Business Model
- Examine Logical Joins
- Examine Logical Columns
- Examine Logical Table Sources
- Rename Logical Objects Manually
- Rename Logical Objects Using the Rename Wizard
- Delete Unnecessary Logical Objects
- Create Simple Measures

4.4.2.3. Building the Presentation Layer of a Repository

To build the Presentation layer, we perform the following steps:

- Create a Subject Area
- Create Presentation Tables
- Create Presentation Columns
- Rename Presentation Columns
- Reorder Presentation Columns
- Creating Analysis and dashboard

4.5 Oracle Business Intelligence Enterprise Edition to address BI

The following description provides the basic building blocks of the Oracle BI which have been used as the front-end of our system.

4.5.1 Oracle BI server Architecture

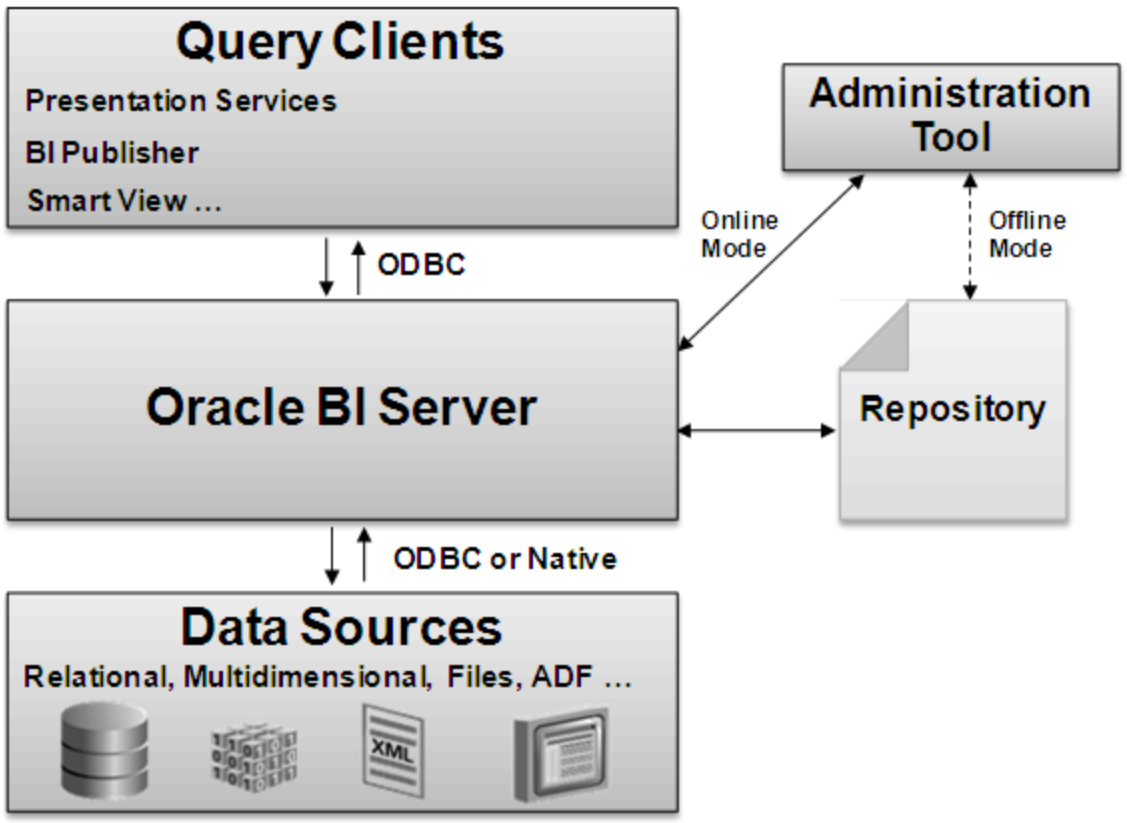

Figure 4.2: Oracle BI Server Architecture

Oracle BI Server is a high-impact query, reporting and analysis server that provides services to the other components of the Business Intelligence Suite such as Answers, Dashboards, Data Mining, Reporting, and Analytic Applications. [15] The main function of the BI Server is to process inbound SQL requests against a virtual database model, build and execute one or more physical database queries, process the data and then return it to users. The BI Server is one part of the Oracle BI Enterprise Edition Plus product family, and presents itself to query tools as one or more databases in a simple relational (star schema) model, that can then point to a much more complex set of relational, multidimensional, file and XML data sources.

4.5.2 Oracle BI Presentation Services

Oracle BI Presentation Services provides the framework and interface for the presentation of business intelligence data to web clients. It is a stand-alone process and communicates with Oracle BI server using Open Database Connectivity (ODBC) over TCP/IP. It consists of a set of query clients that includes the Analysis Editor and Interactive Dashboards.

The Analysis Editor enables users with the appropriate permissions to build and modify analysis that provides end users with the ability to explore and interact with information. An analysis can present information in various formats, such as tables, pivot tables, graphs, maps, and gauges; the result of an analysis can be enhanced by adding calculated items and drilling. Prebuilt analysis can be used out of the box or modified to suite business's information needs. Analysis are saved in the Oracle BI Presentation Catalog and integrated into Oracle Business Intelligence Dashboard. The recipient of analysis can format the analysis result, output the results to another data format, save the results, and share the analysis results with other users

.

4.5.3 Oracle BI Server

Oracle BI Server is the core server behind Oracle Business Intelligence. It is an optimized query engine that receives analytical requests, intelligently accesses multiple physical data sources, generates SQL to query data in data sources, and then structures the results to satisfy the requests. It also handles requests from a variety of front ends, including Oracle BI applications as well as third party tools. Oracle BI server allows a single information request to query multiple data sources, providing information access to members of the enterprise and to suppliers, customers, prospects, or any authorized user with web access.

4.5.4 Data Sources

Data sources are the physical sources where business data is stored. They can be in any format, including transactional databases, OLAP databases, text files, multidimensional data sources such as Essbase, spreadsheets, and so on. A connection to the data sources is created and then

used by Oracle BI Server. The data source connection can be defined to use native drivers or ODBC.

SQL is generated by Oracle BI Server against the data sources by using the data source connection, information from the repository, and database specific features and parameters. As a result Oracle BI Server is not just a SQL generator; it determines the best source and the optimal way to access data.

4.5.5. Oracle BI Repository

Oracle BI Server stores metadata in repositories. The Oracle BI Administration tool has a graphical user interface that allows server administrators to set up these repositories.

4.6 Anomaly Detection Algorithms

The goal of anomaly detection is to identify cases that are unusual within data that is seemingly homogeneous. Anomaly detection is an important tool for detecting fraud, network intrusion, and other rare events that may have great significance but are hard to find. We have used the anomaly detection algorithms to detect the anomalous/fraudulent transactions using ATM cards. [18]

Out of the several algorithms available for anomaly detection, we've used the Support Vector Machine based anomaly detection algorithm, which is currently used by the Oracle as Oracle Data Mining.

Anomaly detection is implemented as one-class classification- an unsupervised algorithm, because only one class is represented in the training data. An anomaly detection model predicts whether a data point is typical for a given distribution or not. [19] An atypical data point can be either an outlier or an example of a previously unseen class. Normally, a classification model must be trained on data that includes both examples and counter-examples for each class so that the model can learn to distinguish between them. For example, a model that predicts side effects

of a medication should be trained on data that includes a wide range of responses to the medication. A one-class classifier develops a profile that generally describes a typical case in the training data. [20] Deviation from the profile is identified as an anomaly. One-class classifiers are sometimes referred to as positive security models, because they seek to identify "good" behaviors and assume that all other behaviors are bad.

4.6.1 One-Class SVM

Support Vector Machines (SVM) is a powerful, state-of-the-art data mining algorithm with strong theoretical foundations based on the Vapnik-Chervonenkis theory that can address problems not amenable to traditional statistical analysis. [21] SVM has strong regularization properties. Regularization refers to the generalization of the model to new data.

When SVM is used for anomaly detection, it has the classification mining function but no target. One-class SVM models, when applied, produce a prediction and a probability for each case in the scoring data. If the prediction is 1, the case is considered typical. If the prediction is 0, the case is considered anomalous. This behavior reflects the fact that the model is trained with normal data.

One implementation of the one-class SVM is to map the input data into a high-dimensional feature space and then fit all or most of the data into a hypersphere. The idea of this technique is that the volume of the hypersphere is minimized and all of the data samples that do not fall in the hypersphere are considered as anomalies.

Tax and Duin introduced a so-called support vector data description (SVDD) method that fits most of the training data into a hypersphere. The sphere is centered in '*a*' with a radius of $R > 0$. The volume of the sphere is minimized by minimizing R^2 such that most of the training data vectors x_i are fitted into the sphere.

Slightly similar to two-class classification, a function that minimizes the volume of the hypersphere that contains most of the training vectors x_i is introduced:

Formula:

$$F(R,a) = R^2 + C\sum_{i=1}^{n}\varepsilon_i \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (4.1)$$

subject to

$$\left||x_i - a|\right|^2 \leq R^2 + \varepsilon_i \; and \; \varepsilon_i \geq 0 \qquad for\; all\; i = 1, \ldots\ldots, n$$

where variable ε_i is a slack variable that allows some of the training vectors x_i to lie outside of the hypersphere. C is a penalty parameter that controls the miss-classifications just like in two-class classification.

After solving this by introduction Lagrange multipliers αi, a new data point z can be tested to be in or out of class. It is considered in-class when the distance to the center is smaller than or equal to the radius, by using the Gaussian kernel as a distance function over two data points:

Formula:

$$\left||z - x_i|\right|^2 = \sum_{i=1}^{n}\alpha_i \, exp\left(\frac{-\left||z - x_i|\right|^2}{\sigma^2}\right) \geq -\frac{R^2}{2} + C_R \quad \ldots\ldots\ldots\ldots\ldots (4.2)$$

where $\sigma \in R$ is a kernel parameter.

Thus, one-class SVM tries to find a separating hyperplane and maximizes the distance between the two classes while two-class SVM can be solved by constructing a hypersphere that captures most of the training data and minimizes its volume or separates training data from the origin with maximum margin.

4.7 Classification & Regression Tree Algorithm

CART algorithm consists of main three parts:

1. Construction of Maximum tree
2. Choice of the right tree size
3. Classification of new data using constructed tree

4.7.1 Construction of Maximum tree

This part is most time consuming. Building the maximum tree implies splitting the learning sample up to last observation, i.e. when terminal nodes contain observations only of one class. Classification trees are used when for each observation of learning sample we know the class in advance. Classes in learning sample may be provided by user or calculated in accordance with some exogenous rule. [22]

Let t_p be a parent node and t_l, t_r – respectively left and right child nodes of parent node tp. Consider the learning sample with variable matrix X with M number of variables x_j and N observations. Let class vector Y consist of N observations with total amount of K classes.
Classification tree is built in accordance with splitting rule – the rule that performs the splitting of learning sample into smaller parts.

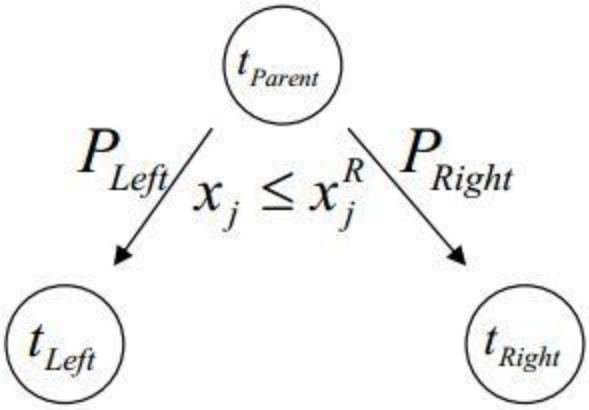

Figure 4.3: Splitting Algorithm of CART

Where t_p, t_l, t_r – parent, left and right nodes; x_j – variable j; x_j^R – best splitting value of variable x_j. Maximum homogeneity of child nodes is defined by so-called impurity function $i(t)$. Since the impurity of parent node t_p is constant for any of the possible splits $x_j \le x_j^R$, $j = 1,...,M$, the maximum homogeneity of left and right child nodes will be equivalent to the maximization of change of impurity function $\Delta i(t)$:

Formula:

$$\Delta i(t) = i(t_p) - E[i(t_c)] \text{ ---------- (4.3)}$$

where t_c – left and right child nodes of the parent node t_p. Assuming that the P_l, P_r – probabilities of right and left nodes, we get:

Formula:

$$\Delta i(t) = i(t_p) - P_l i(t_l) - P_r i(t_r) \text{ ---------- (4.4)}$$

Therefore, at each node CART solves the following maximization problem:

Formula:

$$\underset{x_j \le x_j^R,\ j=1,\ldots,M}{\arg\max} \left[i(t_p) - P_l i(t_l) - P_r i(t_r) \right] \text{ ---------- (4.5)}$$

Above equation implies CART will search through all possible values of all variables in matrix *X* for the best split question $x_j < x_j^R$ which will maximize the change of impurity measure *Δi(t)*.

4.7.1.1 Gini splitting

Gini splitting rule (or Gini index) is mostly broadly used rule. It uses the following impurity function *i(t)*:

Formula:

$$i(t) = \sum_{k \neq l} p(k|t) p(l|t) \text{ ---------- (4.6)}$$

where *k*, *l 1,…,K* – index of the class; *p(k|t)* – conditional probability of class *k* provided we are in node t.

Applying the Gini impurity function to maximization problem we will get the following change of impurity measure *Δi(t)*:

Formula:

$$\Delta i(t) = -\sum_{k=1}^{K} p^2(k|t_p) + P_l \sum_{k=1}^{K} p^2(k|t_l) + P_r \sum_{k=1}^{K} p^2(k|t_r) \text{ --------- (4.7)}$$

Therefore, Gini algorithm will solve the following problem:

Formula:

$$\underset{x_j \le x_j^R,\ j=1,\ldots,M}{\arg max} \left[-\sum_{k=1}^{K} p^2\left(k|t_p\right) + P_l \sum_{k=1}^{K} p^2(k|t_l) + P_r \sum_{k=1}^{K} p^2(k|t_r) \right] \text{-------- (4.8)}$$

Gini algorithm will search in learning sample for the largest class and isolate it from the rest of the data. Gini works well for noisy data.

4.7.2 Choice of the Right Size Tree

Maximum tress may turn out to be of very high complexity and consist of hundreds of levels. Therefore, they have to be optimized before being used for classification of new data. Tree optimization implies choosing the right size of tree – cutting off insignificant nodes and even subtrees. We have used Cross validation method to prune the tree.

4.7.2.1 Cross Validation

The procedure of cross validation is based on optimal proportion between the complexity of the tree and misclassification error. With the increase in size of the tree, misclassification error is decreasing and in case of maximum tree, misclassification error is equal to zero. But on the other hand, complex decision trees poorly perform on independent data. Performance of decision tree on independent data is called true predictive power of the tree. Therefore, the primary task – is to find the optimal proportion between the tree complexity and misclassification error. This task is achieved through cost-complexity function:

Formula:

$$R_\alpha(T) = R(T) + \alpha\left(\tilde{T}\right) \rightarrow \min_T \text{ ..... (4.9)}$$

Where R(T) – misclassification error of the tree T; $\left(\tilde{T}\right)$ – complexity measure which depends on $\tilde{T}$ – total sum of terminal nodes in the tree. α – parameter is found through the sequence of in-sample testing when a part of learning sample is used to build the tree, the other part of the data

is taken as a testing sample. The process repeated several times for randomly selected learning and testing samples.

Although cross-validation does not require adjustment of any parameters, this process is time consuming since the sequence of trees is constructed. Because the testing and learning sample are chosen randomly, the final tree may differ from time to time.

4.7.3 Classification of New Data

As the classification tree is constructed, it can be used for classification of new data. The output of this stage is an assigned class or response value to each of the new data. The output of this stage is an assigned class or response value to each of the new observations. By set of questions in the tree, each of the new observations will get to one of the terminal nodes of the tree. A new observation is assigned belongs to.

Dominating class – is the class that has largest amount of observation in the current node. For example, the node with 5 observations of class 1, two observation of class 2 and 0 observation of class 3, will have class 1 as a dominating class.

5 SYSTEM ANALYSIS

5.1 Requirement Specification

The functional & non-functional requirement for our system are specified below.

5.1.1 Functional Requirements

Functional requirements define what the system or application will do - specifically in the context of an external interaction (with a user, or with another system). Our system is all about storing the banking transaction and customer profile data in warehouse, processing these data, analyzing them and finally produce reports and important patterns, intrusions that are hidden in the data. So the system have following functional requirements.

- The system shall store the transaction data in suitable format in the data warehouse.
- The system shall accept the data from daily transaction or weekly based transaction.
- The system shall produce different types of visual graphs and charts that shows the information about customers and their transactions.
- The system shall show the customer segmentation based on their interest, behavior and transactions.
- The system shall predict when a particular customer is at high risk of ceasing the relationship with bank.
- The system shall detect the fraud in ATM transaction based on location and time when a data with fraud transaction is processed.

5.1.2 Non-functional Requirements

Non-functional requirements are not concerned with the functions of the system. Instead, they look at the criteria to which the software is expected to conform to. Non-functional requirements can include things like response time and reliability. It can also be closely tied to user satisfaction. So our system have following non-functional requirements

- The system shall be easy to use by bank staffs and managers, the interface shall be attractive and user friendly
- The system shall give constant performance irrespective of input transactions
- This system shall runs on operating system platforms : Windows and Linux
- This system shall be highly portable and can be move from one platform to another platform easily.
- The platform must be java enabled.
- The system must produce predictable results.
- Only the authorized person can modify the data.
- The system must adapt to changes in any input record format without the need to recompile any code.
- There can be no unhandled exceptions from incorrect user input.
- A user can install and operate the program without assistance of any kind.

5.2 Feasibility Assessment

5.2.1 Operational Feasibility

As our product is related to data warehousing & data mining, so it relies mainly in the data. So, without data, its existence is impossible. The collection of data is the most challenging part in our application. As for the scope within our Major Project, this application uses raw data available in flat files and unstructured format from some of the banking institutions such as Nepal Investment Bank. The data that we need is mainly about the customer profile, ATM withdrawals and customers' deposits. But, these data are hard to collect and more difficult to convince the banking personnel about the nature of data required and sometimes it becomes so difficult that they reject to share even the shuffled data. But, the good news is that we were able to get the anonymized data from the NIBL & real schema of OLTP database tables from Everest Bank. After the collection of data, valuable information can be mined.

Thus, the project is operationally feasible to implement though it is hard to collect the required source data.

5.2.2 Technical Feasibility

The technical feasibility of the project deals with the availability and its actual implement ability with the existing tools and techniques available in the software market world. The following are the notable points about the technical feasibility of the project:

- For the extraction, transformation and loading of the staging tables, *i.e.,* for the ETL processing of the external data, we can use the UNIX shell scripting tool such as Korn (.ksh) shell or Bash (.bash) shell scripting.
- For maintaining the target tables of the data warehouse, we can use the Oracle database since the warehouse needs to store thousands and millions of data records.
- We can implement the appropriate business and presentation models creating the metadata repositories with the help of the Oracle Administration Tool.
- The Oracle Business Intelligence Enterprise Edition (OBIEE) Tool provides the appropriate environment for implementing the business intelligence goals of our project providing fraud detection in ATM transactions, churn prediction and other analytical and predictive results with different visualization techniques. For the implementation of the data mining algorithms, we can use Python as the programming language.

With all these perspectives taken into consideration, the project is technically feasible to implement.

5.2.3 Economic Feasibility

The economic feasibility of the project includes its economic appropriateness with respect to its presented output. If a project provides results of lower significance but requiring higher budgets then that project can't be economically viable. For the case of our project, we need to have Oracle database along with the BIEE tool and their freely available express editions can be used even for the corporate uses too. Similarly, different packages of python & 'Orange' for data analysis are also free and open source. The major cost of the project covers during the data collection and the product development. This application doesn't have any specific external hardware requirements, so it is not going to be expensive for production. The potential

customers of our application are different banking institutions and it is, thus, economically feasible for development providing useful business insights to the clients.

6 SYSTEM DESIGN

6.1 Use Case Modeling

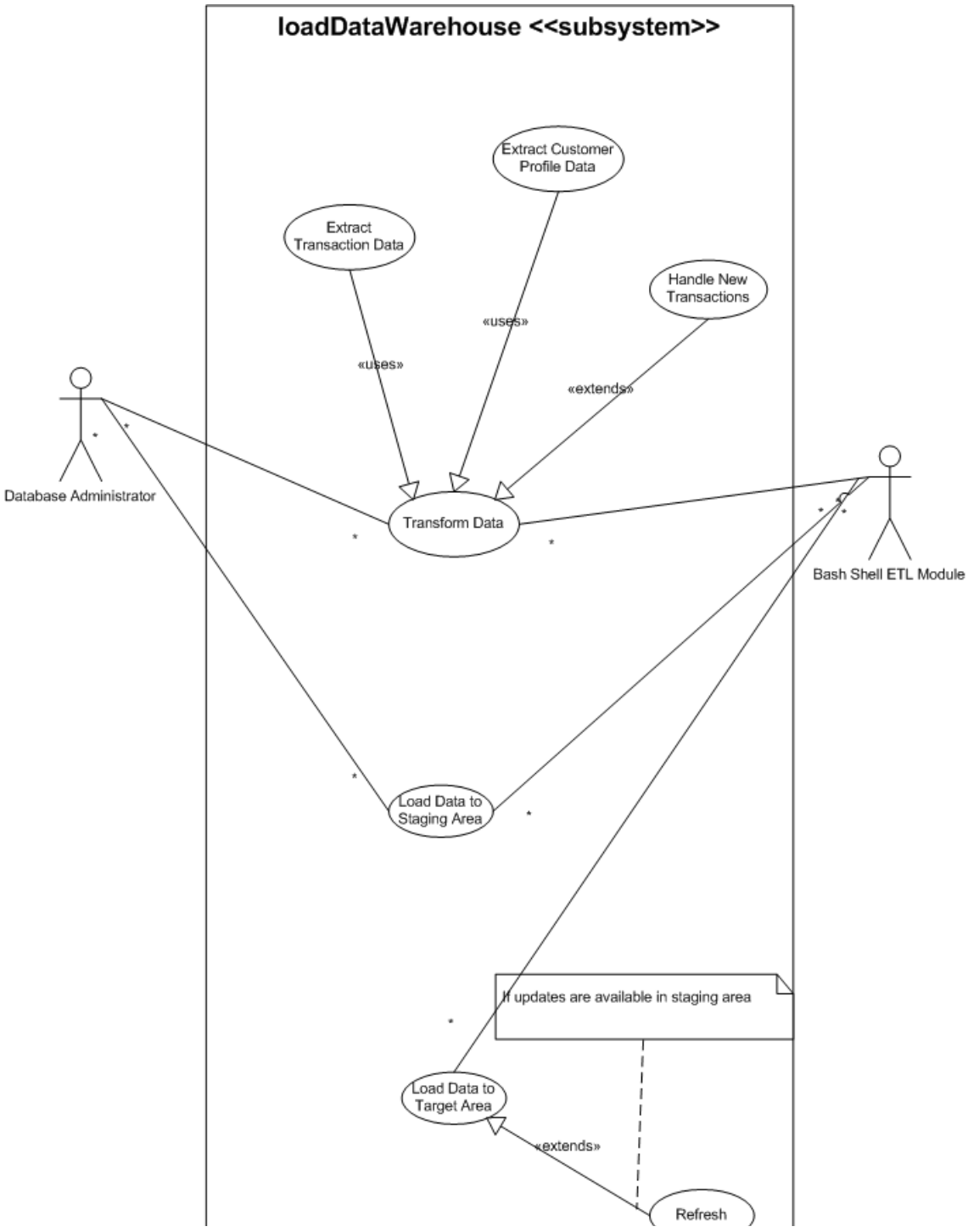

Figure 6.1: Use Case Diagram Modeling the Data Warehouse Loading Process

As shown in diagram above, the loadDataWarehouse subsystem contains two actors one primary actor- Database Administrator and the other supporting actor- Bash shell ETL module. Each of the actors show their one behaviour in context to the system. The relationships between the actors and the use case scenarios have been shown by the association lines and those between use cases have been shown by the dependency lines with appropriate stereotypes shown within guillemets. The specification of the actors with main success scenarios and other alternate scenarios have been shown as follows:

6.1.1 Specification of Actor: Database Administrator

Table 6.1: Specification of Actor: Database Administrator

Element	Details
DESCRIPTION	Database Administrator is the primary actor who is responsible for the extraction, transformation and loading of the data to the staging area. This actor is also responsible for the handling of new transaction data.

Table 6.2: Specification of Actor: Bash Shell ETL Module

Element	Details
DESCRIPTION	Bash Shell ETL Module is the supporting actor who is responsible for actual extraction, transformation and loading of the data to the staging area. This actor is also responsible for the loading of the target area and handling updates, *i.e,* refreshment of the warehouse.

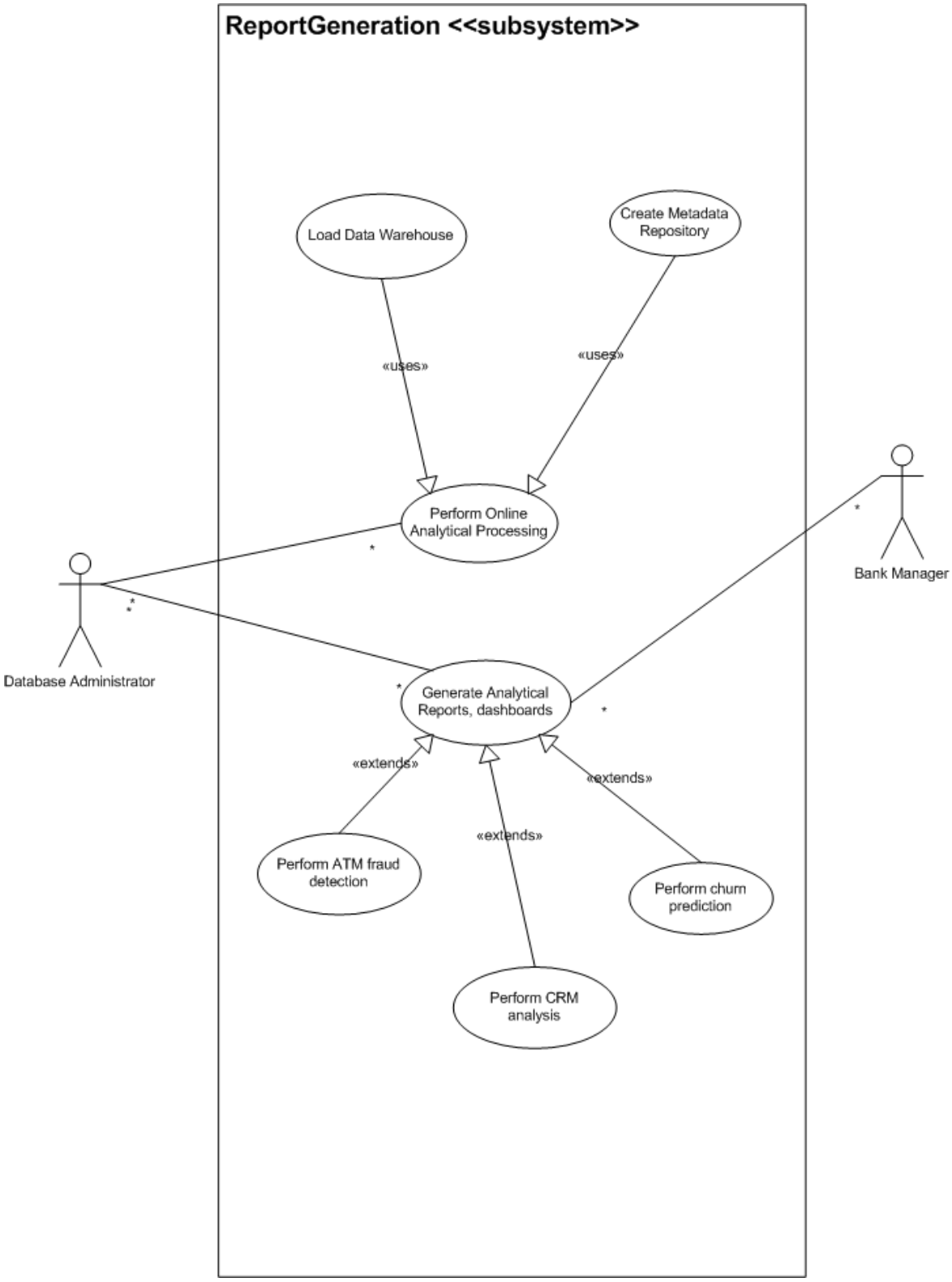

Figure 6.2: Use Case Diagram Modeling the Report Generation Process

As shown in the figure above, the use case diagram modeling the report generation and data mining subsystem consists of two actors- the database administrator and the end users which are the executives belonging to the tactical level of the management in the bank. The relationships between the actors and the use case scenarios have been shown by the association lines and those between use cases have been shown by the dependency lines with appropriate stereotypes shown within guillemets. The specification of the actors with main success scenarios and other alternate scenarios have been shown as follows:

6.1.2 Specification of Actor: Bank Manager

Table 6.3: Specification of Actor: Database Administrator

ELEMENT	DETAILS
DESCRIPTION	Database Administrator is the supporting actor who is responsible for the handling of different multidimensional processing of data. This actor is also responsible for providing the supporting environment for the end users.

Table 6.4: Specification of Actor: Bank Manager

ELEMENT	DETAILS
DESCRIPTION	Bank Manager is the primary actor who is responsible for the generation of different analytical reports, dashboards and also perform the fraud detection and churn prediction including the CRM analysis provided by the subsystem.

6.2 System Architecture

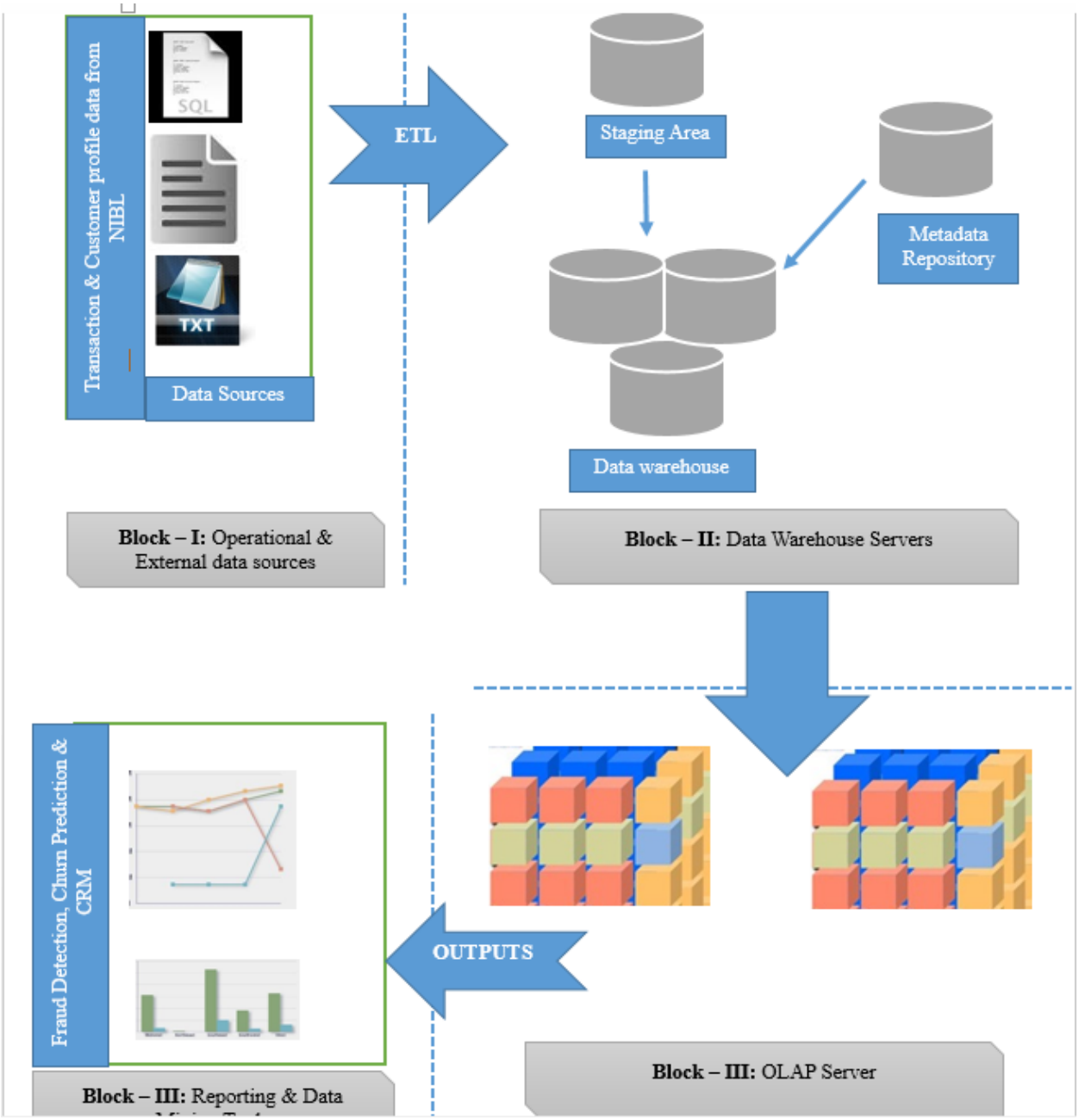

Figure 6.3: The Architecture of the Overall System Showing Different Blocks in WIBTAS

The application system architecture consists of four blocks including the data sources, data warehouse servers, OLAP servers and reporting & data mining block. The functionality of each of the blocks is illustrated below:

6.2.1 Block –I: Operational & External Data Sources

For the implementation of a data warehouse & business intelligence system, the availability of reliable and actual data sources is essential and the most important without which the information reported, mined and forecasted may not be fruitful. For our system, the vendor for the operational & the external data sources is Nepal Investment Bank. The bank provided the bank's customers' profile and transaction databases in various format such as *.txt* and *.sql*. These data sources are flat files and need to be converted in multi-dimensional format for OLAP operations.

6.2.2 Block – II: Data Warehouse Servers

This block contains the staging area, warehouse database servers & metadata repository. There is physical data movement from source database to data warehouse database. Staging area is primarily designed to serve as intermediate resting place for data before it is processed and integrated into the target data warehouse. This staging are serves many purposes above and beyond the primary function:

- The data is most consistent with the source. It is devoid of any transformation or has only minor format changes.
- The staging area in a relation database can be read/ scanned/ queried using SQL without the need of logging into the source system or reading files (text/xml/binary).
- It is a prime location for validating data quality from source or auditing and tracking down data issues.
- Staging area acts as a repository for historical data if not truncated.

The next component is a warehouse database server that is almost always a relational database system. Back-end tools and utilities are used to feed data into the bottom tier from operational

databases or other external sources (such as customer profile information provided by external consultants). These tools and utilities perform data extraction, cleaning, and transformation (e.g., to merge similar data from different sources into a unified format), as well as load and refresh functions to update the data warehouse.

This block also contains a metadata repository, which stores information about the data warehouse and its contents.

6.2.3 Block – III: OLAP Server

The middle block is an OLAP server that is typically implemented using either

(i) a relational OLAP (ROLAP) model, that is, an extended relational DBMS that maps operations on multidimensional data to standard relational operations; or

(ii) a multidimensional OLAP (MOLAP) model, that is, a special-purpose server that directly implements multidimensional data and operations.

6.2.4 Block – IV: Reporting & Data Mining Tools

The front-end client layer in data warehousing is the presentation phase which contains query and reporting tools, analysis tools and data mining tools for trend analysis, fraud detection and customer churn behavior analysis. The reporting tool that we've used for this purpose is Oracle Business Intelligence Enterprise Edition (OBIEE) 11g.

For providing the analytical result, we will be using some of the Online Analytical Processing (OLAP) operations such as slicing & dicing, roll up & roll down and pivoting. The analytical results will be provided in a multi-dimensional view using OLAP Cube Technology projected to assist decision makers such as visualization with comparison to different dimensions e.g. locations, time etc.

For trend & prediction analysis featuring churn analysis and CRM, we will be using some of the data mining algorithms such as CART, C5.0 & Rule based algorithms. Transactions made by fraudsters using counterfeit cards and making cardholder-not-present purchases will be detected through methods which seek changes in transaction patterns, as well as checking for particular patterns which are known to be indicative of counterfeiting.

6.3 Component Diagram

<<Data warehouse subsystem>>

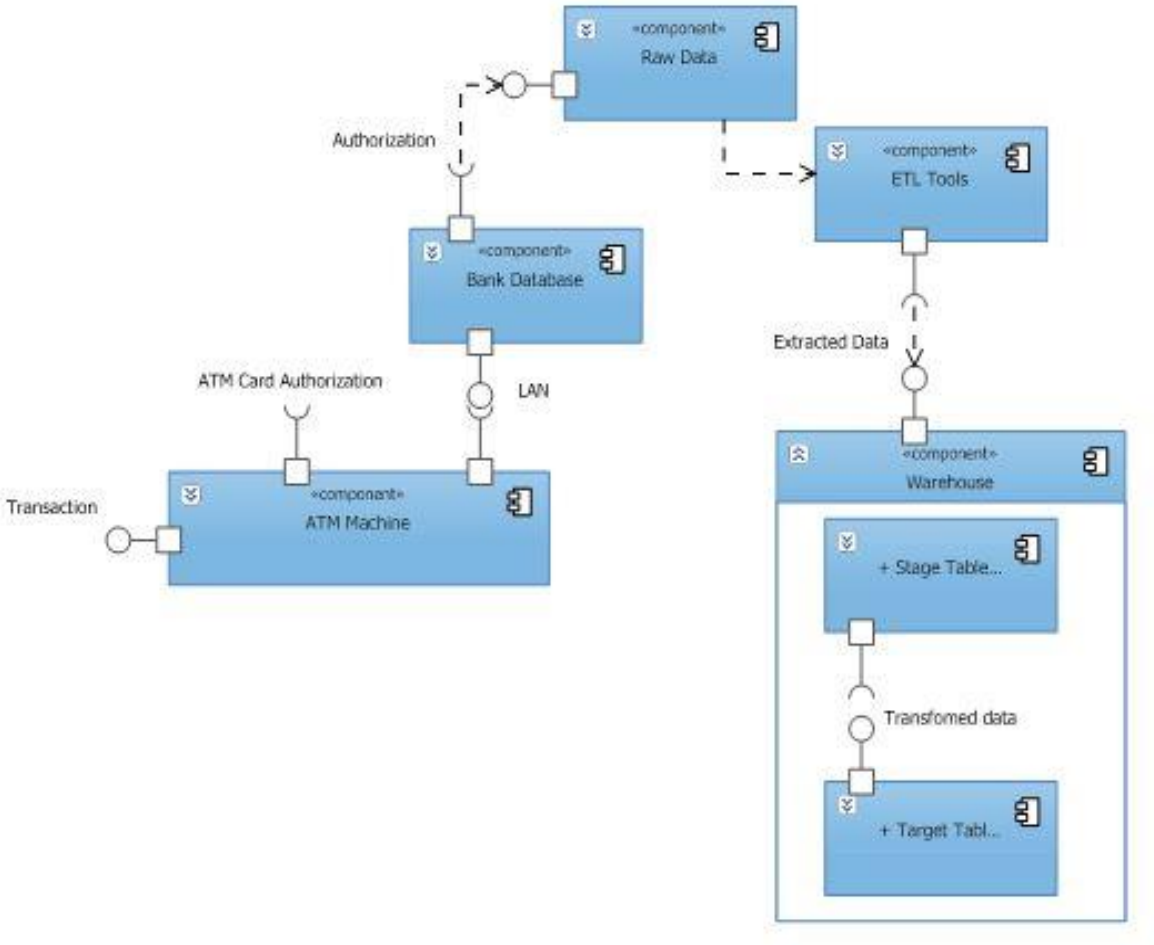

Figure 6.4: Component Diagram showing the Banking Data Warehouse Subsystem

The above component diagram displays the major components involved in this sub system and communication between them. The ATM Machine component denotes the ATM Machine where a transaction takes place, the users must authorize themselves before performing any transaction. The ATM transaction is passed to the Bank Database component via LAN interface. The data stored in Bank Database are accessed by ETL tools component with necessary credentials and authority to perform extraction, transformation and loading of the raw data from Bank Database. Finally, the transformed and refined data are loaded into the data warehouse.

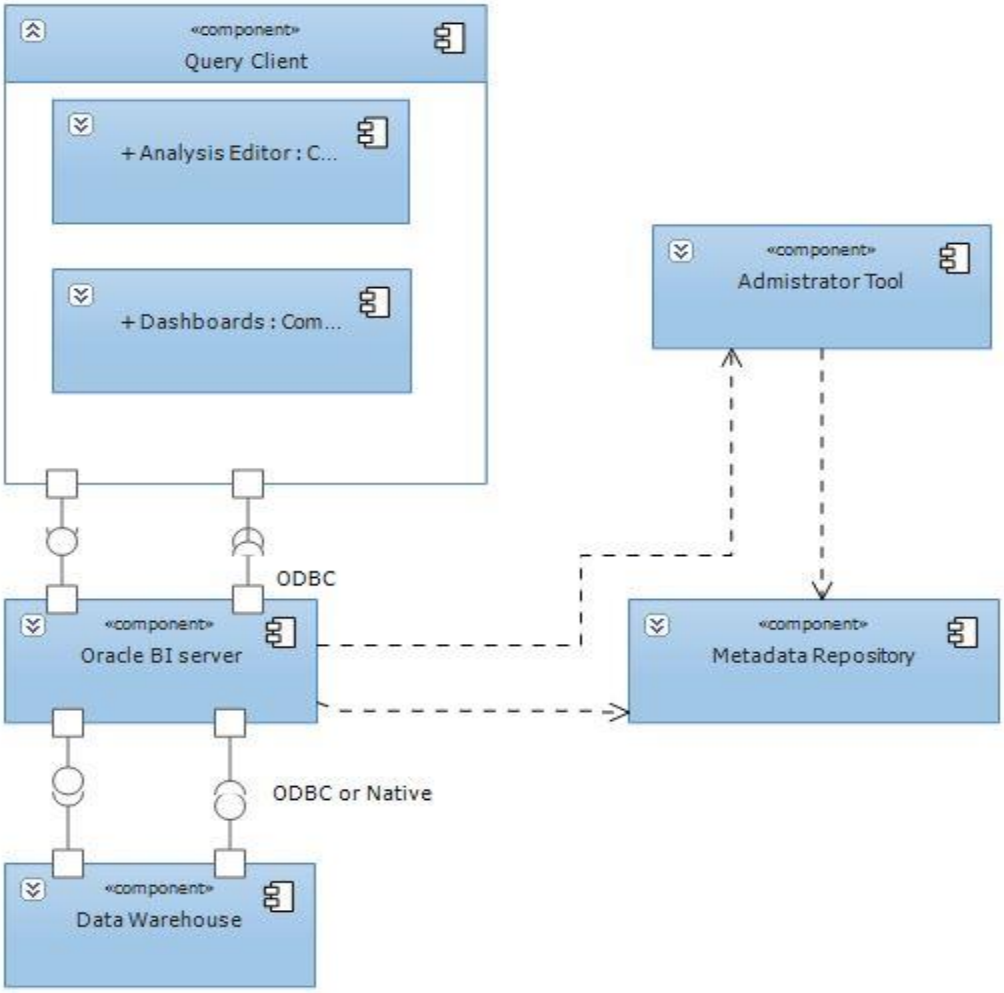

Figure 6.5: Component Diagram Showing the Report Generation Process in OBIEE

Above component diagram displays the components involved in report generation using OBIEE tool. The data from Data Warehouse component are fed into the Oracle BI server. Oracle Bi server provides efficient processing of data and structure information intelligently. It uses metadata to direct processing. The Metadata Repository component stores the metadata used by Oracle BI server. The query client component contains two components, Analysis Editor Component and Dashboards component. Analysis Editor Component contains set of graphical tools that enable users to build, view, and modify analyses that provide analytical information. Dashboards display results the analyses and other items. The Administrator tool component exposes the Oracle BI repository as three separate panes of layers: Physical, Business Model and Mapping and Presentation.

6.4 Class Diagram

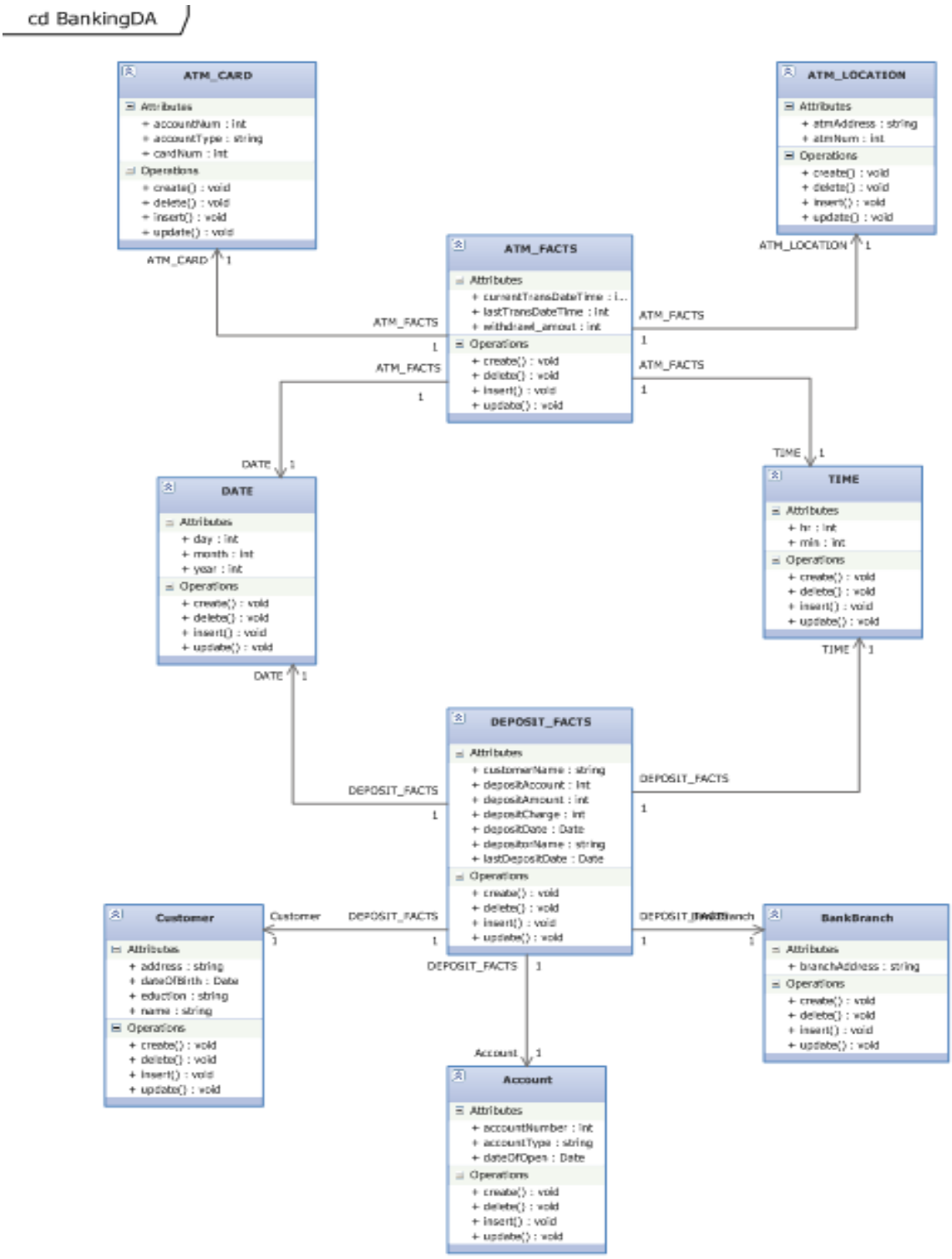

Figure 6.6: Class Diagram Showing the Interaction between Dimensions & Facts

6.5 System Sequence Diagram

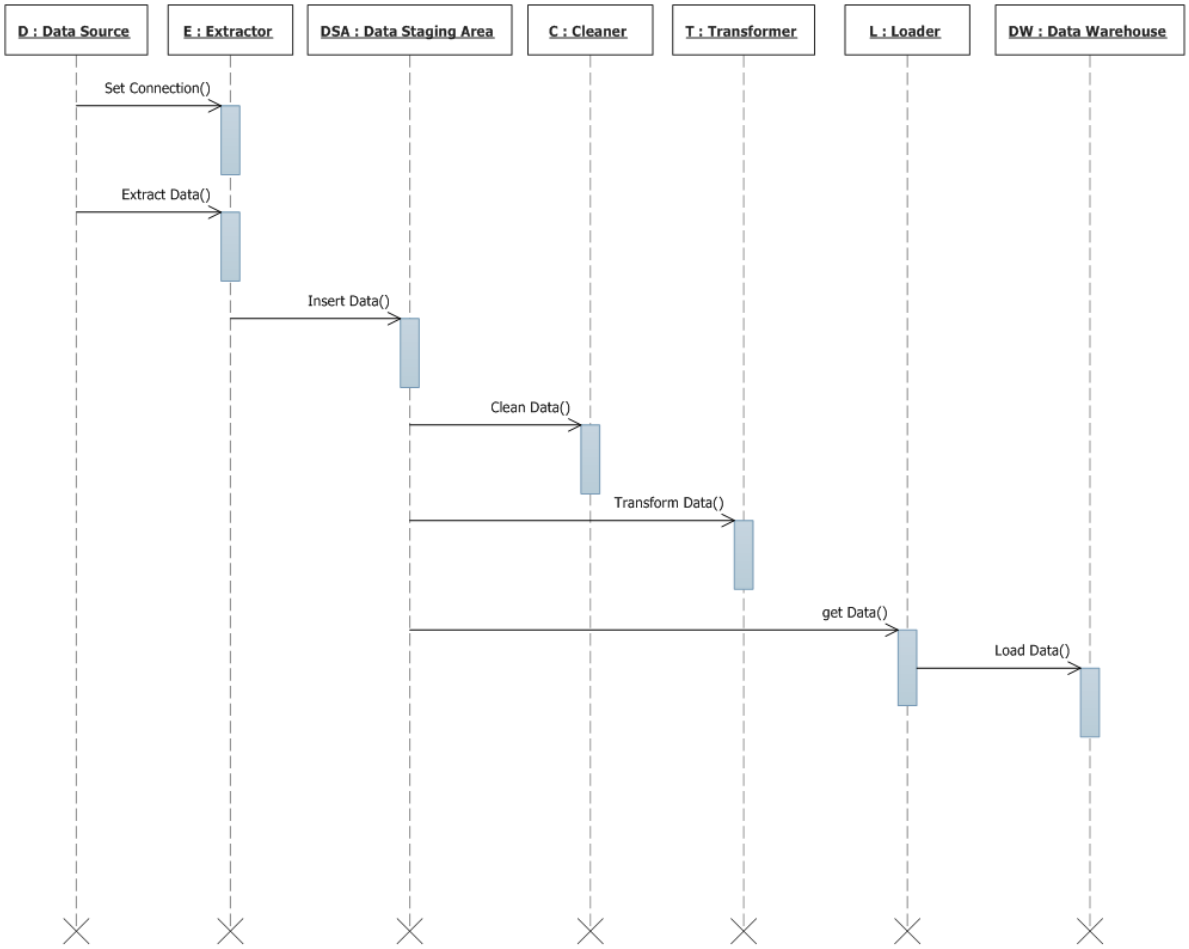

Figure 6.7: System Sequence Diagram Showing the Data Warehouse Loading Process

Data source are heterogeneous may have different storage file. Data is extracted from the distributed heterogeneous data sources and stored in DSA (Data Staging Area) and is used in the next level of ETL process, i.e., cleaning and transformation. The cleaned and transformed data is then loaded into the data warehouse.

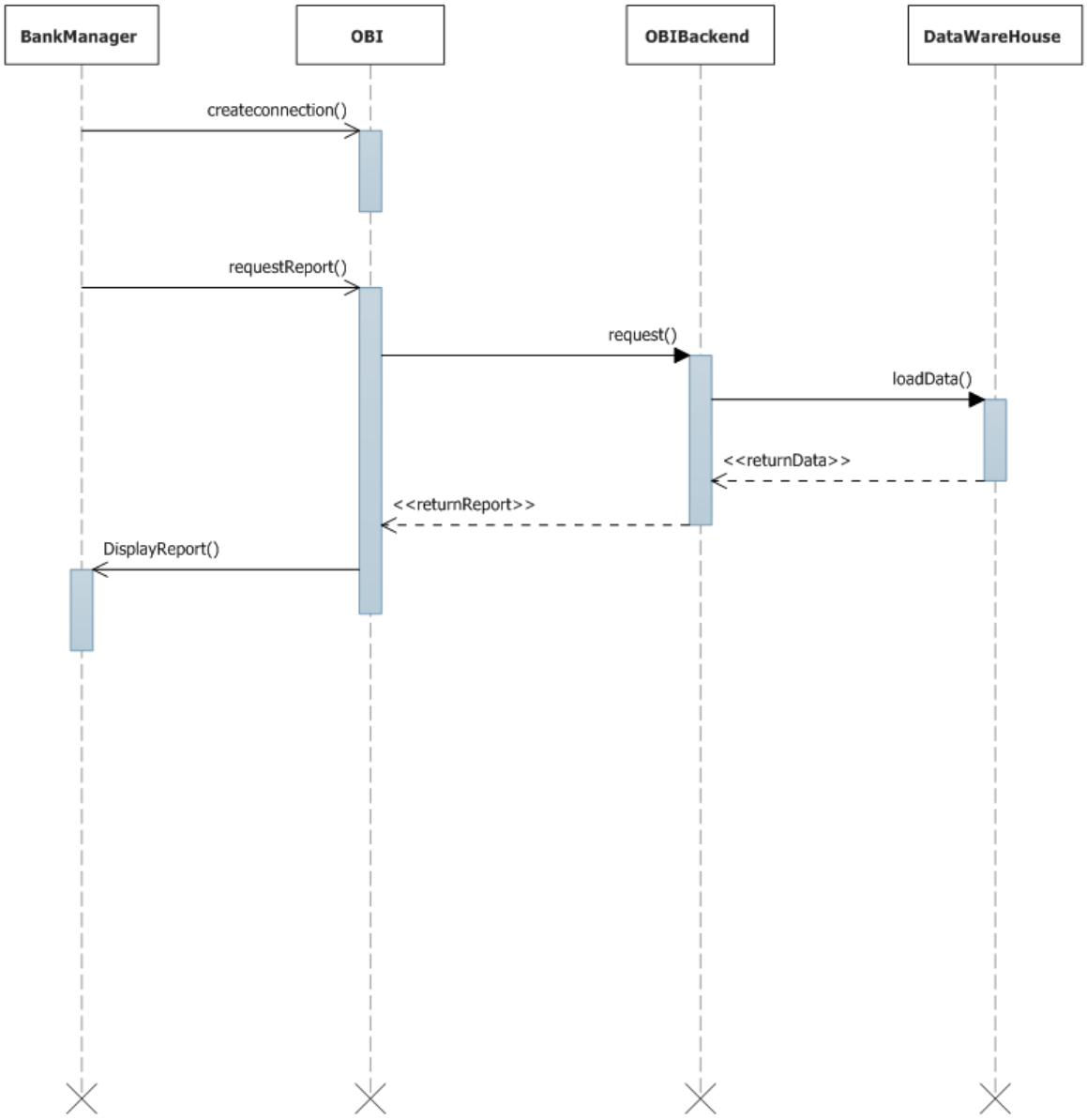

Figure 6.8: System Sequence Diagram Showing the Reporting Process in OBIEE

As in the figure there is three class OBI, OBIBackend, DataWarehouse and BankManger is user. Synchronous message passing is done between classes which is shown by the solid and dotted line. OBI get the request from user and OBI send request to OBIBackend to generated report.

OBIBackend gets necessary data from dataWarehouse class and generated report is send to OBI and report is displayed to the user.

6.6 Data Warehouse Schema Diagram

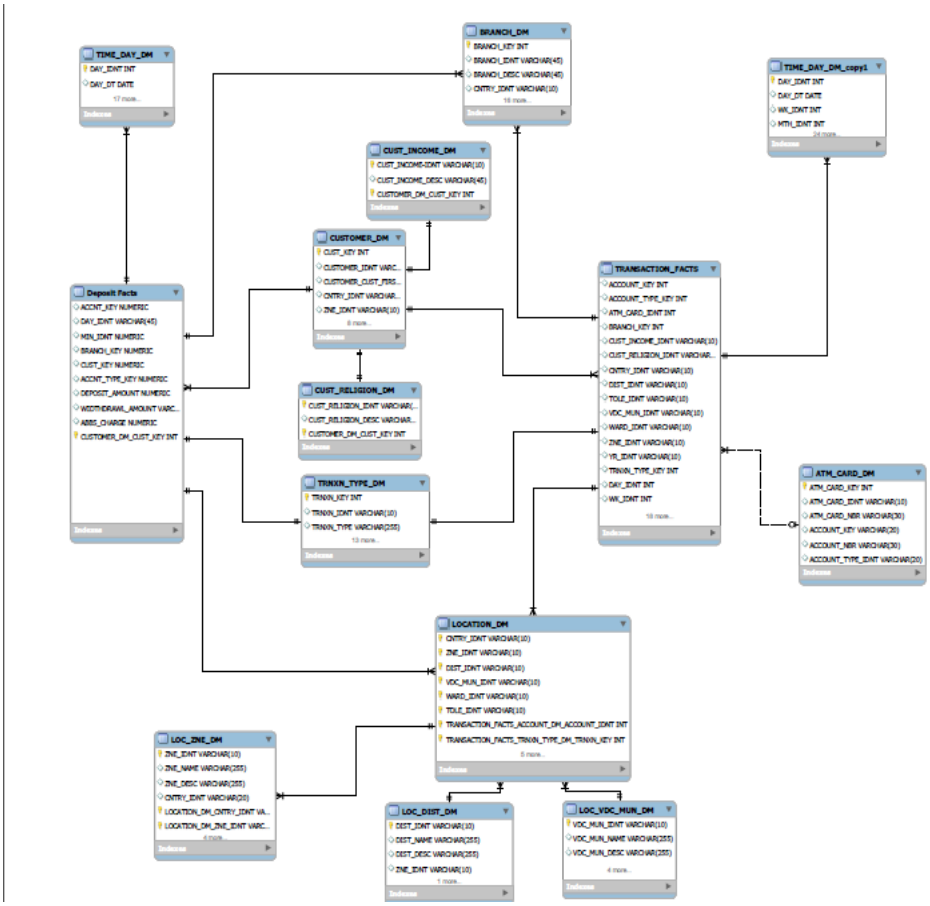

Figure 6.9: Data Warehouse Schema Diagram for the WIBTAS

Figure 6.9 shows different dimensions & facts table used during the development of our system.

6.7 Deployment Diagram

Data warehouse is established and stored in a data server which is accessed through the OBIEE client and is made to access to the user providing better and easy interface.

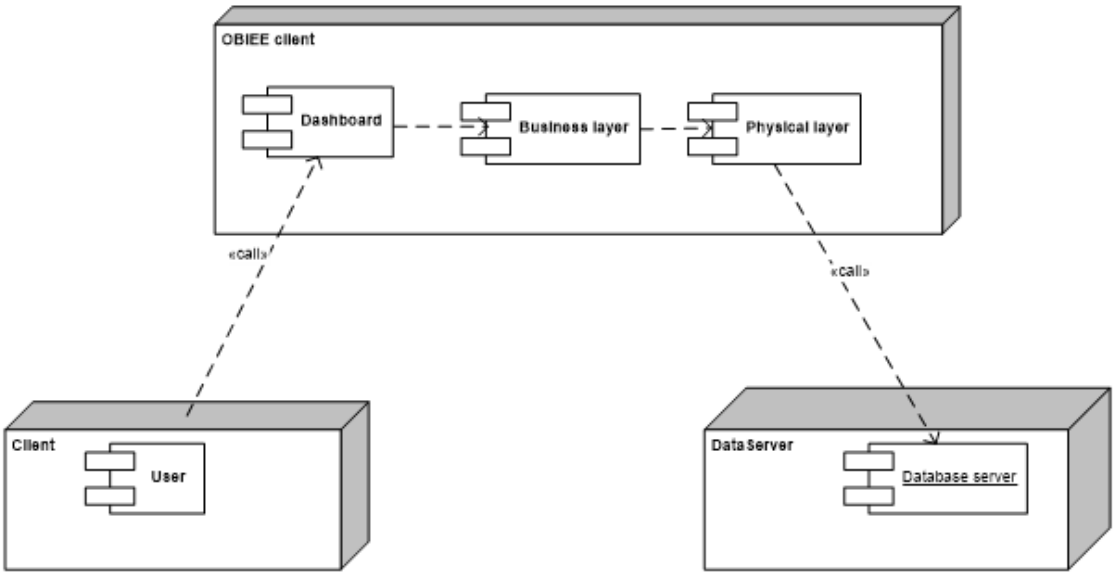

Figure 6.10: Deployment Diagram Showing the Overall System

7 IMPLEMENTATION

7.1 Data Collection

Data Collection is the most challenging part of our project. Data domain needed for our project includes customer profile, ATM and debit transaction which is critical and confidential. We made approach to several Banks' for the collection of data. Finally, with the support of Yomari Inc. Pvt. Ltd, we became able to obtain the data after signing the non-disclosure document. So, the source of data is kept as confidential.

7.2 ETL Process

ETL refers to the extract, transform and load. It is an important part in the data warehouse design.

7.2.1 Extract

The first part of an ETL process involves extracting the data from different source like flat files, comma separated, SQL format and other. The goal of the extraction phase is to convert the data into a single format which is appropriate for transforming processing. It is the most challenging aspect of ETL, as extracting useful information from data to meet project objective. Thus the useful data were extracted from provide data source using Bash scripting.

7.2.2 Transform

In ETL process after extraction, transformation and staging is a key step. The transform step applies a set of rules to transform the data from the source to the target. The transformation step also requires joining data from several sources, generating aggregates, generating surrogate keys, sorting, deriving new calculated values, and applying advanced validation rules. Here, we have identified the dimension and facts related to the project's objectives and the source data were transformed as per the requirements to meet the objectives. Primary key, surrogate key,

validation rules were added to the available data and were made appropriate to be loaded in the target tables.

7.2.3 Loading

Once all the data has been cleaned and transformed into a structure consistent with the data warehouse requirements, data is ready for loading into the data warehouse. As the data were transformed to the target format, it is then loaded into target tables for the presentation.

7.3 Staging Tables & Data Warehouse Design

Design of the data warehouse starts with the identification of the dimension and fact of the objectives.

7.3.1 Dimension Identification

We first identified the dimension of our requirements. As our objective is based on the fraud detection, churn analysis and CRM, we identified the dimensions like branch, time, customer, location, ATM, account etc. and the dimension tables were created. Dimension tables, also known as lookup or reference tables, contains the relatively static data in the warehouse. Dimension tables store the information you normally use to contain queries. Dimension tables are usually textual and descriptive and we can use them as the row headers of the result set

LOCATION_DM

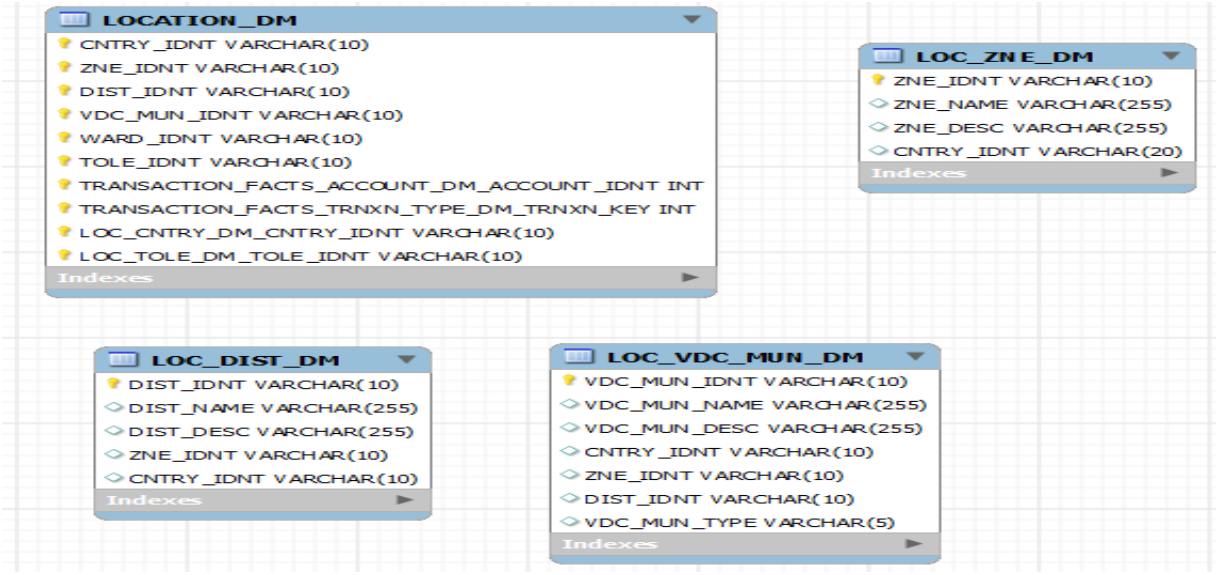

Figure 7.1: Location Dimension Showing Different Fields

TIME_DM

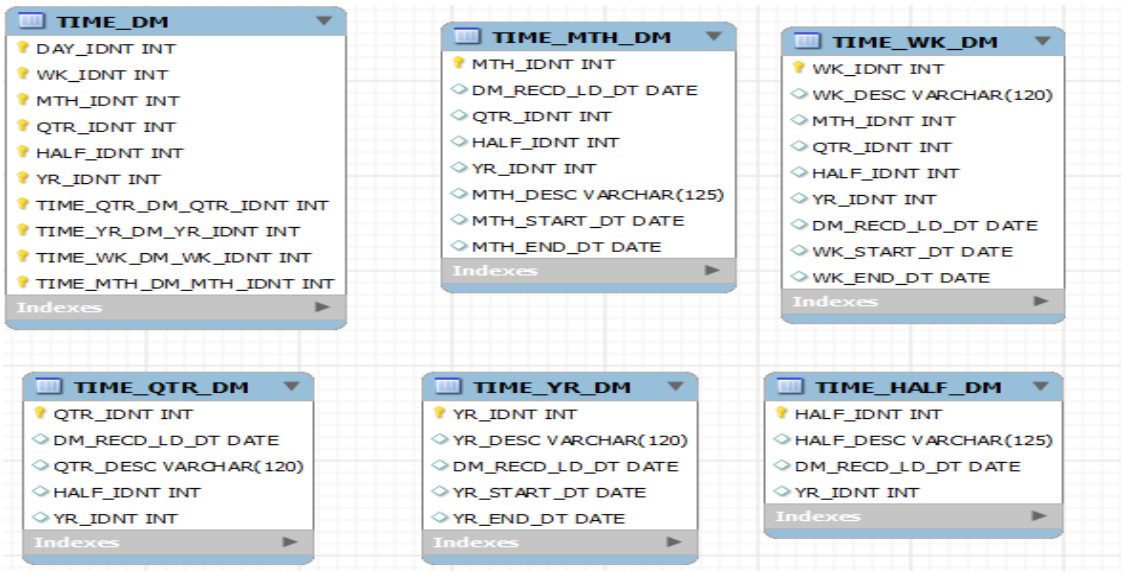

Figure 7.2: Time Dimension Showing Different Fields

CUSTOMER_DM

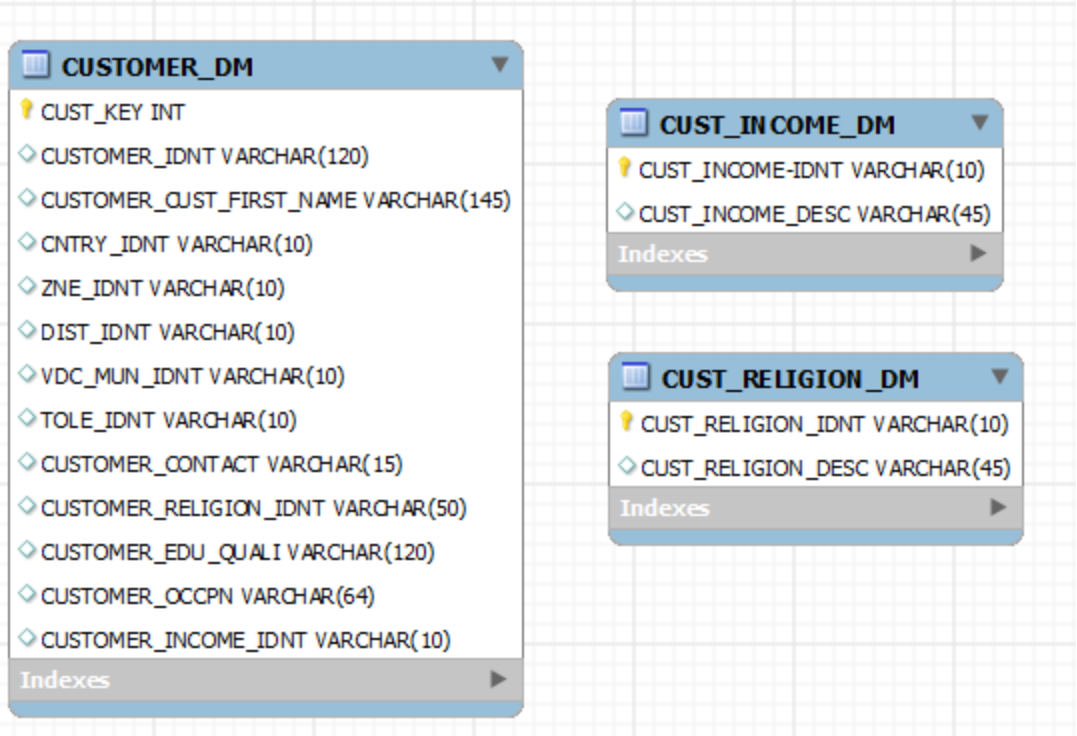

Figure 7.3: Customer Dimension Showing Different Fields

ATM_CARD_DM

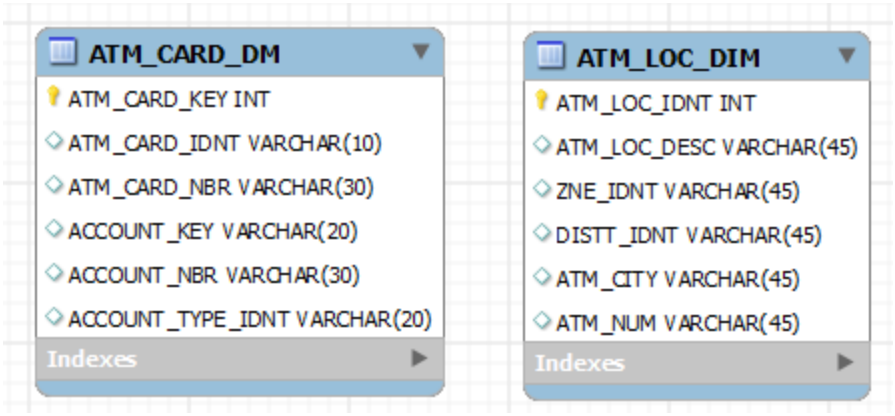

Figure 7.4: ATM Card Dimension Showing Different Fields

7.3.2 Fact Identification

The fact tables are then created which consists of the transactional fact and are looked up in the dimension table to obtain the detail information of the fact. The fact are of no meaning without dimensions. The dimension tables consists of the primary key , surrogate key and other required identification keys through which the particular data is identified in the fact table. Fact tables contains facts and those that are foreign keys to dimension tables. Facts may be composed of measures, degenerated dimensions. The primary key of a fact table is usually a composite key that is made up of all of its foreign keys.

TRANSACTION_FACTS Table

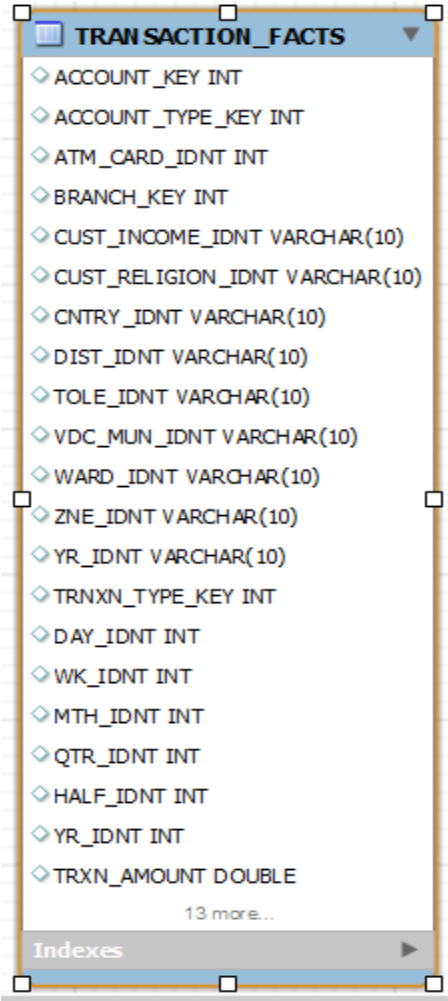

Figure 7.5: Transaction Fact Table Showing Different Fields

Staging table is created in the data warehouse in addition to the target tables. The reason behind it to ensure that if the transformation fails, it is not necessary to restart the extract process again from the early step. In a simpler and easier way, staging tables are for the storing of the source data so that the same data should not extracted again and again in case some error occurs. Staging tables are not accessed by the end user but only by the ETL process.

7.4 Implementing Financial Reporting & Business Intelligence

The implementation of financial reporting and of business intelligence has been visualized through the Oracle BIEE 11g. After the creation of dimension and facts, the next step is the creation of dashboard for visualizing the analysis and report generated. During the process of creation of financial report, it involves mainly two process.

- Repository creation
- Creation of dashboard

The creation of Repository and Dashboard is supported by Oracle Business Intelligence Enterprise Edition. This application aims at providing financial report of a particular financial or banking institution occurring through the country in one place, so financial reports are categorized in to 3 different areas.

- Trend analysis
- Geo-demographic analysis
- Time series predictive analysis

7.4.1 Repository Creation

First of all repository (RPD) files are created with the support of Oracle administrative tool. Steps involved in repository creation are

- metadata are imported from databases and other data sources
- Physical layer are created.
- Logical and business layer are created.
- Presentation layer are created.

After the creation of repository, repository are saved and are uploaded from OBIEE fusion middleware and thus data are analyzed using analysis editor.

7.4.2 Analysis Creation

Before the dashboard are created, Analysis on the facts and dimension are created using Analysis editor. For this purpose Repository created with joins and hierarchy are uploaded from OBIEE fusion middleware and thus data are analyzed using analysis editor.

The Analysis Editor is composed of tabs and panes representing the subject area (columns), available catalog objects, selected columns for the analysis, and filters (which limit the selected data).

A subject area contains folders, measure columns, attribute columns, hierarchical columns, and hierarchy levels that represent information about the areas of an organization's business or about groups of users with an organization. Subject areas usually have names that correspond to the types of information that they contain, such as Time, customer, and so on. The analysis are saved so that thus analysis and the graph can be used in the dashboard.

7.4.3 Dashboard Creation

After the analysis are created and saved. Dashboard are created using thus saved analysis. Firstly the new dashboard are created following new dashboard from the global header then the saved analysis are dragged in the dashboard so as to display it. Selection criteria like filters and selection can be used. Furthermore Prompts are the essential part in dashboard to filter the data as per the user at the front end.

Links can also be used in the dashboard from the available Action link and Action link menu available in OBIEE. OBIEE also has the flexibility of creation of new page and tabs as per the requirement of user.

7.4.4 Implementing Trend Analysis

Implementation of trend analysis includes 3 step of repository creation, analysis creation and dashboard creation respectively.

Appropriate RPD file containing sufficient joins and relations, time and location hierarchy are created in the logical layer of RPD file and reports depicting the trend analysis are created with the help of OBIEE. Included reports in trend analysis are:

7.4.4.1 Duration - wise Report

Date tables forming the hierarchy (day, week, month, half year, quarter year, year) were imported from data warehouse into RPD , joins and relation with tables, aggregates were defined in the logical layer of RPD file. Finally RPD file was uploaded. Analysis and reports were generated in OBIEE and thus displayed in Dashboard. The duration wise reports are implemented with

- Yearly report
- Quarterly report
- Monthly report

7.4.4.2 Time - wise ATM Withdrawal Report

Time table forming the hierarchy (seconds, minutes ,hour) were imported from data warehouse into RPD , joins and relation with fact and dimension table , aggregates were defined in logical layer of RPD file. Finally RPD file was uploaded. Analysis and reports were generated in OBIEE and thus displayed in Dashboard.

7.4.4.3 Location - wise ATM Withdrawal Report

Information of ATM location throughout the country for the particular bank was collected. Location table forming the hierarchy(VDC, municipality, District, Zone) were imported from data warehouse into RPD , joins and relation with fact and dimension table , aggregates were defined in logical layer of RPD file. Finally RPD file was uploaded. Analysis and reports were generated in OBIEE and thus displayed in Dashboard.

7.4.5 Implementing Geo-Demographic Analysis

Implementation of geo-demographic includes 3 step of repository creation, analysis creation and dashboard creation respectively.

Customer profile including their Date of birth, permanent address, profession, marital status education and daily transactions were imported from the warehouse to administration tool. Dimension and facts tables are joined, aggregation are defined in logical layer of RPD file. Further date of birth is converted to age by subtracting date of birth from present date and reports are generated with respect to age, education, marital status, annual income using OBIEE.

Geo-demographic analysis is implemented in two format. First one is customer count on the basis of geographical location, age, education, annual income and marital status. Second is the total amount withdrawal from ATM on the basis of age, education, annual income and marital status.

7.4.6 Implementing Time - series Predictive Analysis

Implementation of time series predictive analysis includes 3 step of repository creation, analysis creation and dashboard creation respectively.

As this report gives the insight comparison with respect to time, date and time hierarchy table are imported to administrative tool and repository are created with sufficient joins and connections. Finally, reports are created with OBIEE.

Steps to create repository in time series data:

- Time dimension and chronological key are identified.
- 1 year ago, 2 year ago etc. Time measures are created in logical layer.
- Measures are dragged to presentation layer.

7.5 Implementing ATM Card Fraud Detection

Frauds are cases that are unusual because they fall outside the distribution that is considered normal for the data. The distance from the center of a normal distribution indicates how typical a given point is with respect to the distribution of the data. Each case can be ranked according to the probability that it is either typical or atypical. The presence of fraudulent transactions can have a deleterious effect on many forms of data mining. The following steps are taken while implementing the ATM card fraud detection.

7.5.1. Feature Extraction: Choosing What Features to Use

Before building the fraud detection model, it is necessary to plug out the features, out of several features available, which might take on unusually large or small values in the event of anomaly. In data warehousing based approach, the fact table consists of the fact data set, in our case the ATM transactions, and it is the dimensionally reduced form of original staged table keeping in

mind that data warehouse does not lose any of the original information. So, for feature extraction, the fact table is our base table from which we can build the feature set for modeling the ATM fraud. In our case, we've tried to implement time wise as well as location wise frauds, so the extracted features must address these requirements. For building the feature set, we've also used some of the non-Gaussian features such as current_balance/withdrawal_amount, average, min & max. So, for the decision of what features to use, we've used the following error analysis approach for detection of fraud:

$$\text{Want} \quad P(x) \text{ large for normal examples } x$$

$$P(x) \text{ small for anomalous examples } x$$

Based on these criteria, the features extracted for building the ATM fraud detection model are given below:

Table 7.1: Extracted Feature set for ATM Fraud Detection

Transaction ID
Current balance
Withdrawal amount
Days since previous withdrawal
Previous withdrawal amount
AVG (withdrawal amount)
MIN (withdrawal amount)
MAX (withdrawal amount)
Proximity from previous withdrawal location
Current_balance – Withdrawal amount
Current balance – AVG (withdrawal amount)

7.5.2. Data Transformation

Data transformation is another important step while performing data mining. In data transformation, the data are transformed or consolidated into forms appropriate for mining. Data transformation can involve the following:

- Smoothing
- Aggregation
- Generalization
- Normalization
- Feature Construction

For our case, we've used the normalization technique for the data transformation. In this method, an attribute is normalized by scaling its values so that they fall within a small specified range, such as 0.0 to 1.0. Normalization is particularly useful for classification algorithms, distance measurements such as nearest-neighbor classification and clustering. [23] Normalization helps prevent attributes with initially large ranges (e.g., balance) from out-weighing attributes with initially smaller range (e.g. days left from last withdrawal). Among several normalization methods, we've used z-score normalization or zero - mean normalization. In this method, the values for an attribute, X, are normalized based on the mean and standard deviation of X. A value, v, of X is normalized to v' by computing

Formula:

$$v' = \frac{v - avg(X)}{\sigma_X} \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots (7.1)$$

This method of normalization is useful for our case as the actual minimum and maximum of attribute X are unknown, or as there may be outliers that dominate the min-max normalization.

7.5.3. Building Fraud Detection Model

For building the fraud detection model, we've used to two methods multi-variate Gaussian distribution & support vector machines. The former one is used for the initial approach of building & testing the fraud detection. The later one will be used for the final fraud detection model.

7.5.3.1. Multi-variate Gaussian Distribution

As a brief introduction to this algorithm, we first of all build a matrix of column-wise means & a covariance matrix. For our case the univariate distribution is of no use as we need to deal with

several features, however it can be considered as the extension of univariate one for dealing with multiple variables. Mathematically, a vector of n Gaussian random variables $x = [x_1, \ldots, x_n]^T$ is completely defined by its mean vector $\mu = [m_1, \ldots, m_n]^T$ and it's n×n covariance matrix Σ. The multivariate pdf of x is given by:

Formula:

$$p(x|\mu,\Sigma) = \frac{1}{(2\pi)^{\frac{n}{2}}|\Sigma|^{\frac{1}{2}}}\exp\left\{-\frac{1}{2} * (x-\mu)^T\Sigma^{-1}(x-\mu)\right\} \ldots\ldots\ldots\ldots\ldots (7.2)$$

The parameters for this algorithm are fitted for the given training examples $\{x^{(1)}, x^{(2)}, \ldots, x^{(m)}\}$ as:

Formula:

$$\mu = \frac{1}{m}\sum_{i=1}^{m} x^{(i)} \text{ and } \Sigma = \frac{1}{m}\sum_{i=1}^{m}\left(x^{(i)}-\mu\right)\left(x^{(i)}-\mu\right)^T \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (7.3)$$

Fraud Detection with the Multi-variate Gaussian

1. Fit model p(x) by

   Formula:

   $$\mu = \frac{1}{m}\sum_{i=1}^{m} x^{(i)} \text{ and } \Sigma = \frac{1}{m}\sum_{i=1}^{m}\left(x^{(i)}-\mu\right)\left(x^{(i)}-\mu\right)^T \ldots\ldots\ldots\ldots\ldots (7.4)$$

2. Given a new example x, compute

   Formula:

   $$p(x|\mu,\Sigma) = \frac{1}{(2\pi)^{\frac{n}{2}}|\Sigma|^{\frac{1}{2}}}\exp\left\{-\frac{1}{2} * (x-\mu)^T\Sigma^{-1}(x-\mu)\right\} \ldots\ldots\ldots\ldots\ldots\ldots (7.5)$$

   Flag an anomaly if $p(x) < \varepsilon$.

In order to perform the comparative study of the data mining algorithms, we first of all implemented the multi-variate Gaussian distribution on the 1000 training examples of our ATM

transactions. Based on this algorithm, some of the test examples were flagged as the fraudulent transactions. In this algorithm, one of the most important step is the selection of the threshold value (ε). We used the F_1-score calculation criteria for the selection of threshold.

Formula:

$$\text{precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \dots\dots\dots\dots\dots\dots\dots\dots\dots (7.6)$$

$$\text{recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots (7.7)$$

$$\text{F}_1 - \text{score} = \frac{2 * \text{precision} * \text{recall}}{\text{precision} + \text{recall}} \dots\dots\dots (7.8)$$

Based on the F_1-score, we calculated the best value of threshold using the p(x) values of cross-validation set.

7.5.3.2. One-Class Support Vector Machines

As explained in Section 4.4.1, the one-class support vector machine have been used as the base algorithm for detecting the fraudulent transactions. The OCSVM based model is implemented in R language. The algorithm is first of all trained using the training set of about 1,200,000 training examples and then it is validated using the cross-validation set of about 800,0000 examples.

The model built by using OCSVM was first fed by only normal examples as suggested in the algorithm so that it can distinguish the atypical examples labeled 0. The model returns the percentage of the fraudulent transactions for each example and only those transactions which have 70% above anomaly percentage are extracted as the fraudulent ones. The fraudulent transactions are then inserted into the Oracle database & through which are published in the Oracle Business Intelligence Enterprise Edition via meta-data repository.

The implementation of the OCSVM based model to detect ATM frauds and the publishing of the fraud results in the OBIEE have been shown in the following figure:

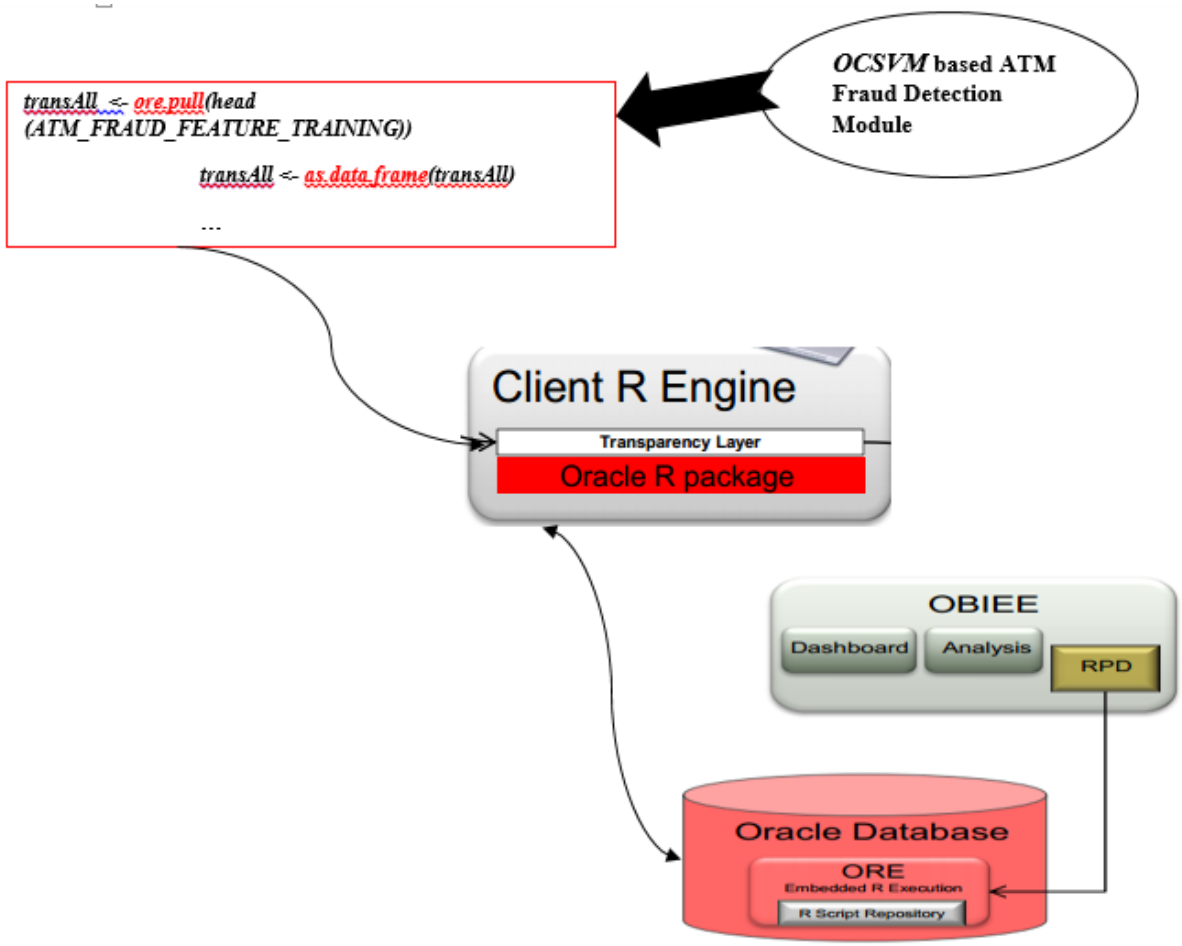

Figure 7.6: Architecture of ATM Fraud Detection Model Interrelating Client R Engine, Oracle Database & OBIEE

As shown above, the mined fraud results are pushed into the Oracle database and then published into the OBIEE with different analyses such as time-wise, ATM location-wise & higher percentage-wise dashboard.

7.5.4. Fraud Detection Model Evaluation using Holdout Method

The constructed models using Multi-variate Gaussian distribution & one-class support vector machines were evaluated using holdout method. Using this method, we randomly divided the whole data set into training set (60%), cross-validation set (20%) & test set (20%). In order to

fix the high bias condition, we've increased the number of features from 7 to 11 as explained in the Section 7.5.1. The training set of about 1200000 examples have been used to build the model and the cross-validation set & test set of about 400000 examples each have been used to test the model's accuracy. The accuracy obtained with two types of models is shown in the following table:

Table 7.2: ATM Fraud Detection Model Evaluation

Detector	Accuracy
Multi-variate Gaussian Distribution	84.3%
One-class Support Vector Machines	93.2%

Thus, after this evaluation procedure, we finally used the one-class SVM as the algorithm to build the fraud detection model which also showed the reasonable scalability with different number of data sets.

7.6 Implementing Customer Churn Prediction

For forecasting the future churn, a very vigorous model should be in hand and an active model can only be built if we have a vigorous dataset in hand. Hence, data preparation is a vital step in churn prediction and it takes almost 60-70 percent of total time. [3] Constructing a model for churn prediction means that we are trying to model the customer's behavior churning out. For this to be successful, the customer transaction activities should be analyzed in a specific period of time. Hence, taking a data would never be enough for the requirement. On the other hand, considering the transaction activities in a fixed time period would not satisfy the requirement. The reason can be explained by an example. Say, for example, a model is built using data of 1000 customers of which 700 are active and 300 are known to be churned out and their 1 year activities are analyzed (say, Feb 2008 to Feb 2009). Here the time period is fixed and the activities done in this time period of all the 1000 customers are only analyzed. Now, out of 300 churn customers, say 50 per cent of them have churned away in February. This means, the model

will not be fully trained with the behavior of churn customers before churning as only one month's activity the timeline before hand as shown in figure 7.7 (a).

In this project, we consider a dynamic time period, which differs for each customer. This concept would be better explained by continuing the above example. If a customer has churned away in Feb. 2006, from that point of time, the past 1 year activity is considered i.e., transaction activities done in Feb 2005 to Feb 2006 are considered. [24] And if another customer churns away in march, transaction activity of March 2005 to March 2006 should be considered. This can be seen in figure 7.7(b).

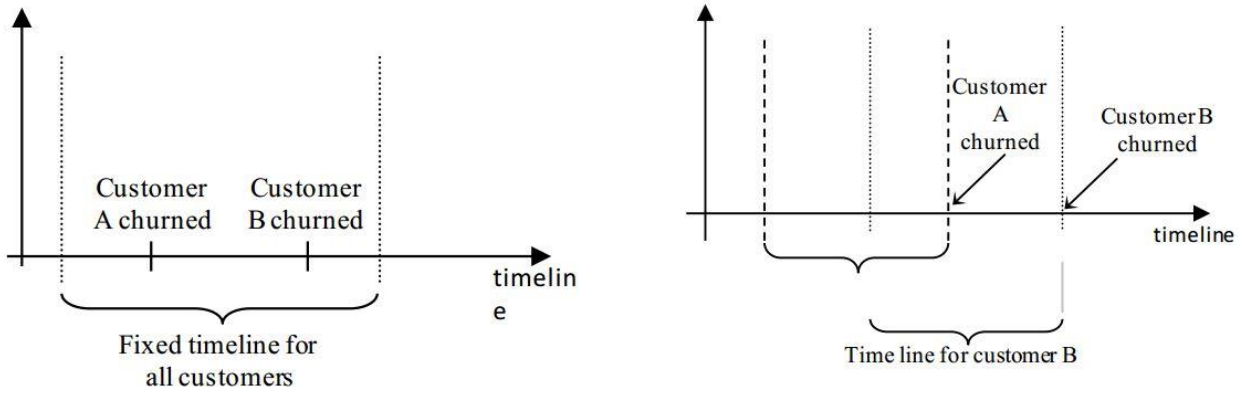

Figure 7.7: Churn Prediction Model (a) Static (b) Dynamic

7.6.1 Data Set Preparation

We acquired the customer data from a Nationalized Nepalese Bank. The particulars of the data acquired are shown in Table below.

Table 7.3: Feature Set for Churn Prediction Model

Table Name	Attributes	No of Records
Customer	Custno, Name1, Name2, Address, Status, DoB (Date of Birth), Qualification	10,136

General Ledger	Custno, AcNo, Descr, DOP (Date of a/c opening)	10, 136
Dormant	Acno, Descr, Dormant	4,778
Master	Acno, Balance, Dormant flag	3,152
Txn	Acno, Trntype, Date, Amount	19,00,000

Customer table details like customer number, name, address, date of birth and status. Completely there are 10,136 customers. The general ledger table holds the account numbers, account type, date of opening and description of accounts. Here, we have 10,136 accounts for the above defined informed as dormant since a while. The master table holds all the account numbers and their latest balances. The Txn table holds the last 10 years transactional details of all accounts.

7.6.2 Data Pre-processing

To avoid the possibility of poor modeling, we performed some data cleaning operation that removes observations that may result into poor modeling. There are some accounts whose duration is not so much i.e., number of months transacted with the bank was very less. These accounts gave a notion that these accounts are opened for a particular purpose and closed as soon as that purpose completed. Considering these accounts behavior may provide a poor dataset and consequently despicable predicting model. So we are ignoring the records whose duration is less than 6 months. There are some set of accounts whose duration is more than 6 months but transaction activities not much went in these account (but have very less transaction activities). This gave a view that these customers have just opened the savings accounts and rarely done transactions through them. Such type of data also effect in poor modeling. After operating all these filtering steps, target customer base reduced to 7,341 account. Out of 7,341 account 6,979 are active and 362 are churn records.

7.6.3 Feature Extraction

Before we train the data with specified algorithm, we need to first extract important features that can effectively provide the good training model. We identified some features first, trained them and observed the result. We did this several times until we get the accuracy of the model above the threshold. We then used that model to predict the customer record to identify whether they are predicted to be churned or not. The final feature set are given below. Feature no 1 – 5 are predictor variables and no 6 is target variable.

1. Average withdrawal amount (AGV_WITHDRAWL)
2. Average deposit amount (AGV_DEPOSIT)
3. Number of Transaction (NO_OF_TRAXN)
4. Duration (DUR)
5. Account Type (ACCNT_TYPE)
6. Status (STATUS)

7.6.4 Model Design

Constructing good churn prediction model is most important part in the classification steps. As mentioned earlier in section 4.5, we used Classification and Regression Tree (CART) algorithm to build the model. To reduce the risk of underfitting and overfitting, we divided the whole data set into three distinct parts.

Part 1: Training Data Set (60%)

Part 2: Cross Validation Data Set (20%)

Part 3: Testing Data Set (20%)

First we trained the classifier using training data set then we checked the result using cross validation data set. If there is large, error we repeated the process from feature extraction and then again train the classifier. In this way we repeatedly performed the above steps until we got the minimum cross validation error. Using the Complexity Parameter (CP) which give the minimum cross validation error, we pruned back the tree to get the optimized classifier.

After building the modeling, we tested the model using test data. We got the overall test set accuracy of 94.7%.

Churn prediction is 'Class Imbalance' problem. It means the number of positive example (churned customers) is negligible in compared to the negative examples (active customers). In our case we have 5% negative examples. We will gate 95% training accuracy if we predict all examples to be negative examples. We remove the risk of such prediction by observing the Confusion Matrix. From confusion matrix we calculated the accuracy of the algorithm by using the parameters what is called "Recall" and "Precision". The confusion matrix we obtained is shown in below:

	Active (Actual)	Churn (Actual)
Active (Predicted)	6979 Records	13 Records
Churn (Predicted)	0 Records	349 Records

Figure 7.8: Confusion Matrix for Churn Prediction

From above table we calculated the precision equals to 0.9981, recall equals to 1, accuracy = 0.9982 (99.82%) and F1 Score = 0.9904. This is very satisfactory result and we can say that our model is good prediction model.

7.7 Development Tools

The tools that we've used for the completion of the above tasks are as follows:

- Bash Shell Scripting

- Oracle SQL Developer
- OBIEE 11g
- R-Studio
- Oracle R Enterprise
- R Programming Language
- MySQL Workbench
- Microsoft Visio 2010
- SubVersion – version controlling tool
- Redmine – project management tool

8 TESTING

The system has been tested because without testing data warehouse could produce incorrect answer and quickly loose the faith of the business intelligence users. Testing of the data warehouse at every point throughout the ETL (extract, transform and load) process in becoming increasingly important as more data is being collected and used for strategic decision making.

8.1 Requirements Testing

The main aim for doing Requirement testing is to check stated requirements for completeness. In a Data warehouse, the requirements are mostly around reporting. We verified whether these reporting requirement can be catered using the data available. We design the high level of the data model using defined requirements. We tested the requirements on following factors.

1. Are the requirements Complete?
2. Are the requirements Developable?
3. Are the requirements Testable?

8.2 Unit Testing

Unit testing is done by the developers during the development process. We check the ETL procedures/mappings/jobs and the reports developed. We also perform the unit test of each element implementing data mining algorithm (CART and SVM).

Following things are involve during Unit testing

1. Whether ETLs are accessing and picking up right data form right source.
2. All the data transformations are correct according to the business rules and data warehouse is correctly populated whit the transformed data.
3. Testing the rejected records that don't fulfill transformation rules.
4. Testing the module implementing data mining algorithm whether the output is correct or not.

8.3 Integration Testing

After unit testing is complete, it should form the basis of staring integration testing. Integration testing means testing the system operation from beginning to the end, focusing on how data flows through the system. It is sometimes called the “system testing” or “end-to-end” testing.

Integration testing will involve following

1. Sequence of ETLs jobs
2. Initial loading of records on data warehouse.
3. Incremental loading of records at a later data to verify the newly inserted or updated data.
4. Testing the rejected records that don’t fulfill transformation rules.
5. Error log generation.

8.4 Black-box Testing

Black box testing treats the software as a "black box" without any knowledge of internal implementation and look the output of application where expected or not. The output pages visualize the result and there value is check for the validation and necessary result is corrected.

8.9 Alpha Testing

System is tested by the programmer in group or individually to find out the error. We tested the module of data mining algorithm output given is valid or not.

8.10 Performance Testing

Test is designed and executed to show how well the system performs to heavy loads of data. Performance testing is performs on the following steps:

1. **Extract Performance Test:** Performance of the system is tested while extracting a large amount of data. We concluded that the extract process takes a little bit longer time.
2. **Transform and Load Performance Test:** We tested the performance of the system while transforming and loading a large amount of data. Testing with a high volume is sometimes called a "stress test". We concluded that this process also takes a little bit longer time due to huge amount of data.
3. **Analytics Performance Test**: Performance of the system is tested manipulating the data through calculations. Implementing the data mining algorithm. After the test we concluded that the output is fast. Performance speed of our algorithm for data mining is found fast and quick responsive.

8.11 Documentation Testing

We well documented the every phase of system development. It is very verified by the project supervisor for their consistency. Each team member also reviewed the entire document to confirm the validity of its parts.

8.12 Problems Faced

- Understanding the banking terminology, rules and regulations.
- There are lots of algorithms available for ATM fraud detection and customer churn prediction. We took a lot of time searching the best algorithms that best address our problem domain.
- OBIEE generates the report based on the data source residing in the oracle database. We faced a big problem in generating the report for ATM fraud detection and Churn prediction because we have to push back our result to the oracle database for this. This is the big problem faced in integrating the project.

- We carried out the project in Remote server in the distributed system. Since the distributed system uses the LAN for carrying the data. We faced the problem of LAN speed and network reliability carrying out the project.

9 RESULTS & CONCLUSIONS

9.1 Financial Reporting & Business Intelligence

Financial reporting & BI is categorized into following sub-headings:

9.1.1 Trend Analysis

9.1.1.1. Yearly Report

Figure 9.1: Bar Graph Showing Yearly Report Generation for ATM Withdrawals

This report shows the ATM withdrawal amount with respect to time in year. User can compare the withdrawal amount through the selection prompts available at the top of the page.

9.1.1.2. Quarterly Report

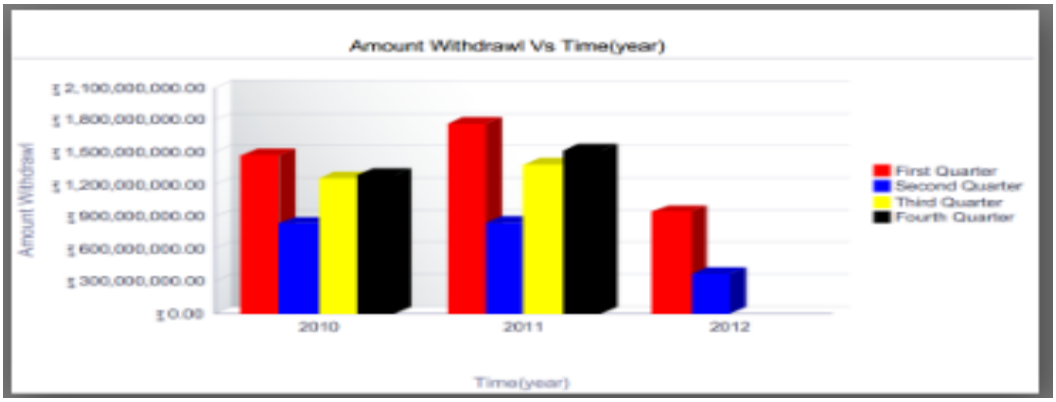

Figure 9.2: Bar Graph Showing Quarterly Report Generation for ATM Withdrawals

This report shows the ATM withdrawal with respect to time in quarter year interval. Each bar shows the duration of 3 months each. User can compare quarterly report between any desired year selected from the prompt available above.

9.1.1.3. Monthly Report

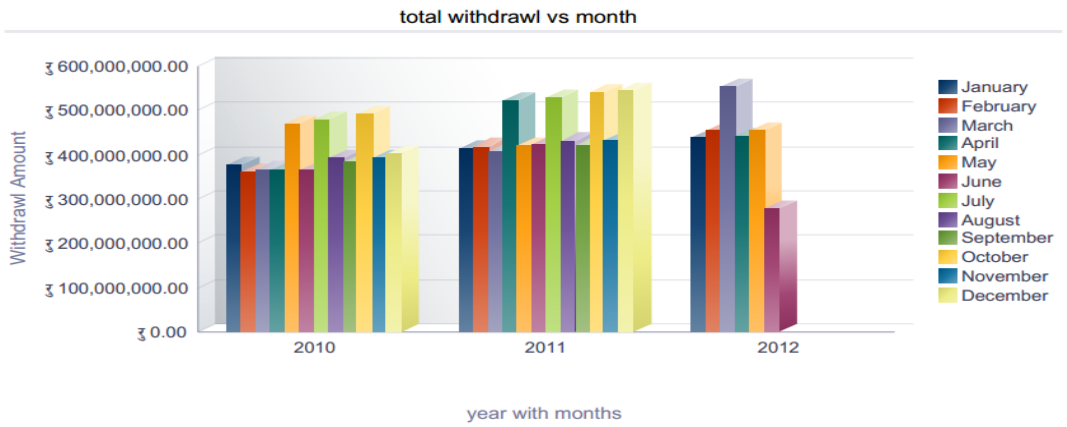

Figure 9.3: Bar Graph Showing Monthly Report Generation for ATM Withdrawals

This report shows the ATM withdrawal amount with respect to time in month. User has the flexibility of comparing withdrawal in any months of a year through the available prompts.

9.1.1.4. ATM Withdrawal Time - wise Report

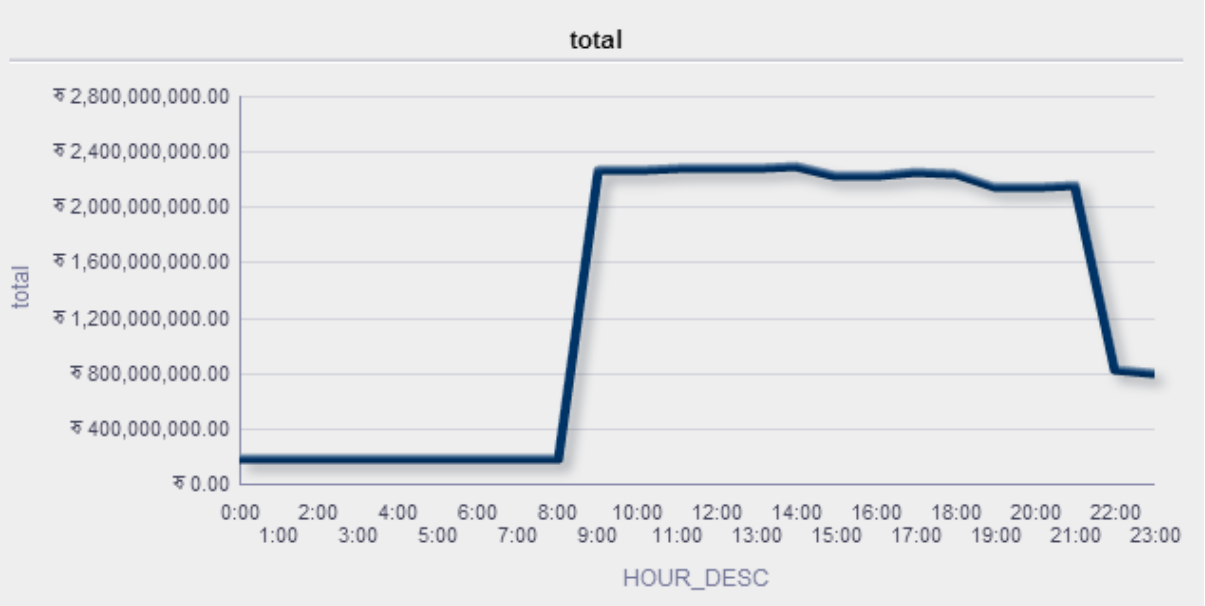

Figure 9.4: ATM Withdrawals at each hour of day for years from 2000 to 2011

This report overall ATM withdrawal amount with respect to hour in a day. Above graph shows that withdrawal of amount is maximum from 8 am in the morning to 9 pm in the evening. User has the flexibility to select the particular time from the prompt as per the requirement.

9.1.1.5. ATM Withdrawal Location - wise Report

Figure 9.5: Location wise ATM withdrawal report

This graph shows the ATM withdrawal amount in different time and in different ATM location place. User has the flexibility of getting insight of each and every ATM places and their withdrawal detail through the use of prompts available at the top.

9.1.2 Geo-demographic Report

Geo-demographic Reports provides the insight of number of customer and the withdrawal amount in different geographical location and also as per demographic factors like age, education, marital status and annual income.

9.1.2.1. Geographical segmentation

Figure 9.6: Geo-Demographic Customer Segmentation

This graph shows the number of total customer in every geographical location. The right part of graph shows total customer number in different years. User can go insight in to every location and count the number of customer in different year as per the selection made in the prompt.

9.1.2.2. Demographic Segmentation

9.1.2.2.1. Customer Count Age and Education - wise

Figure 9.7: Age - wise & Qualification - wise Customer Segmentation

This graph shows the total number of customer on the basis of education wise and also at different location. User can make selection to go insight of each and every location and find the detail.

9.1.2.2.2. Customer Count Annual income wise

Figure 9.8: Annual Income - wise Customer Segmentation

This graph shows the total number of customer on the basis of their annual income. User can go insight into each and every geographical zone and get the information of customer number as per their selection in the prompt.

9.1.2.2.3. Customer Count Marital Status - wise

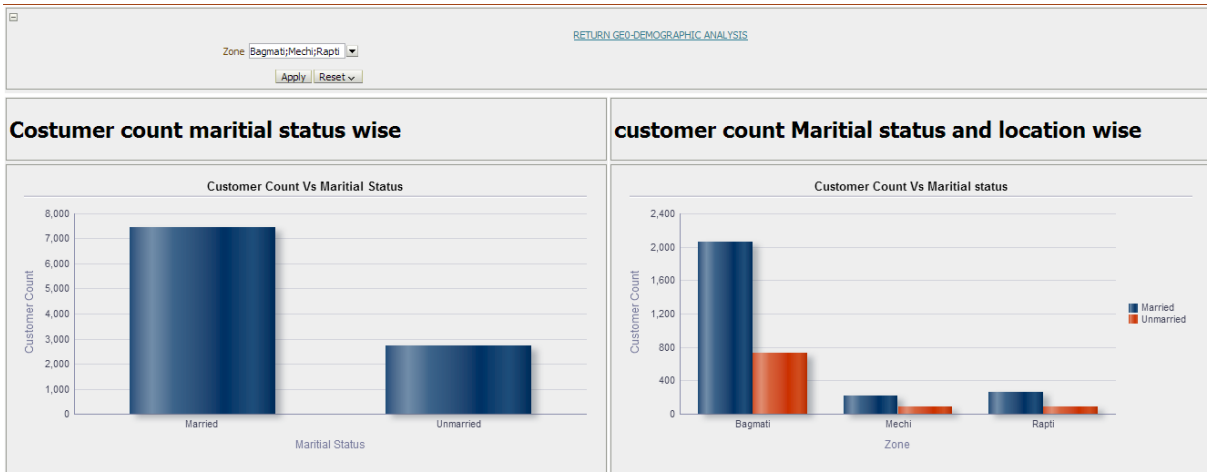

Figure 9.9: Marital Status - wise Customer Segmentation

This graph shows the total number of customer on the basis of status of their marriage at the time of account opening. The right part is insight view at different location as selected by user.

9.1.2.2.4. ATM Withdrawal Age - wise

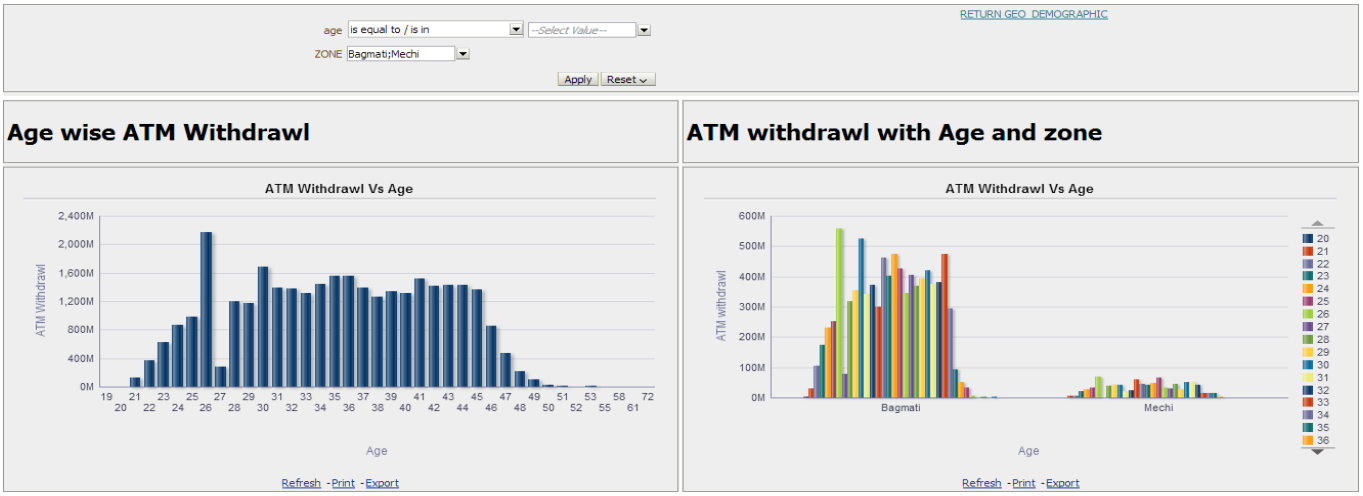

Figure 9.10: Zone & Customer Age - wise ATM Withdrawals

This graph provides the information of ATM withdrawal amount on the basis of Age. User can go insight of information as per location as well from the selection made.

9.1.2.2.5. ATM Withdrawal Education - wise

Figure 9.11: Qualification Level wise ATM Withdrawals

This graph shows the information of ATM withdrawal amount on the basis of education. User can go insight of information as per location as well from the selection made.

9.1.2.2.6. ATM Withdrawal Marital Status - wise

Figure 9.12: Marital Status - wise ATM Withdrawals

This graph shows the information of ATM withdrawal amount on the basis of marital status. User can go insight of information as per location as well from the selection made.

9.1.3 Time - series Predictive Analysis

9.1.3.1. Time – series a Year and a Month Ago Report

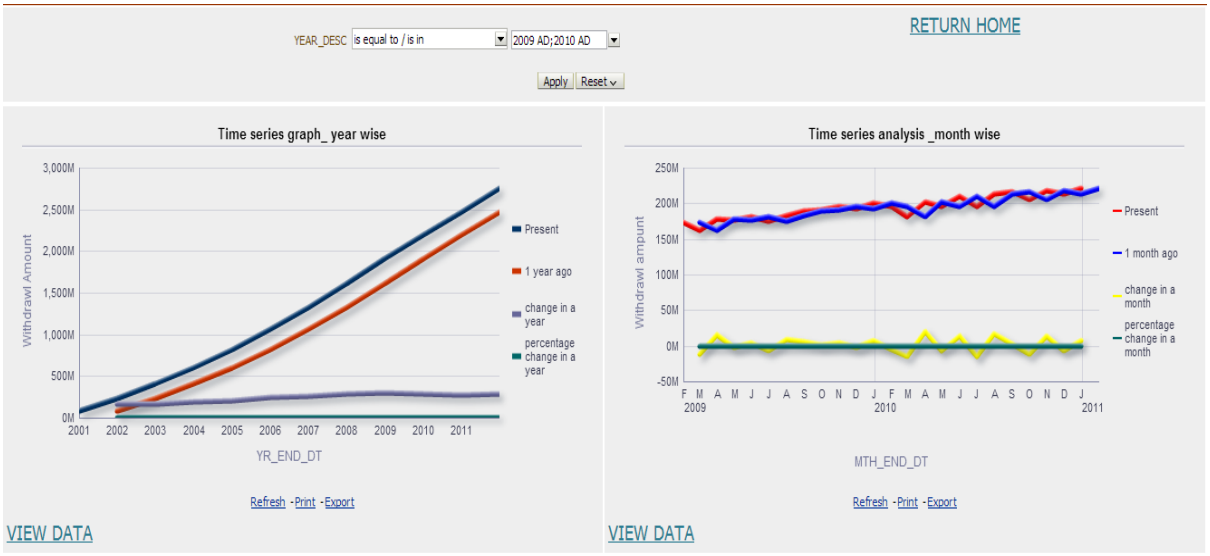

Figure 9.13: Time - series a Year and a Month ago Report

This graph gives the information of amount withdrawal with respect to year, month etc. and also provides the information of amount withdrawal a year ago. It also provides information of withdrawal amount change with respect to previous year and also percentage change in a year.

9.1.3.2. Time – series 1 Year Ago, 2 Years Ago & 3 Years Ago Report

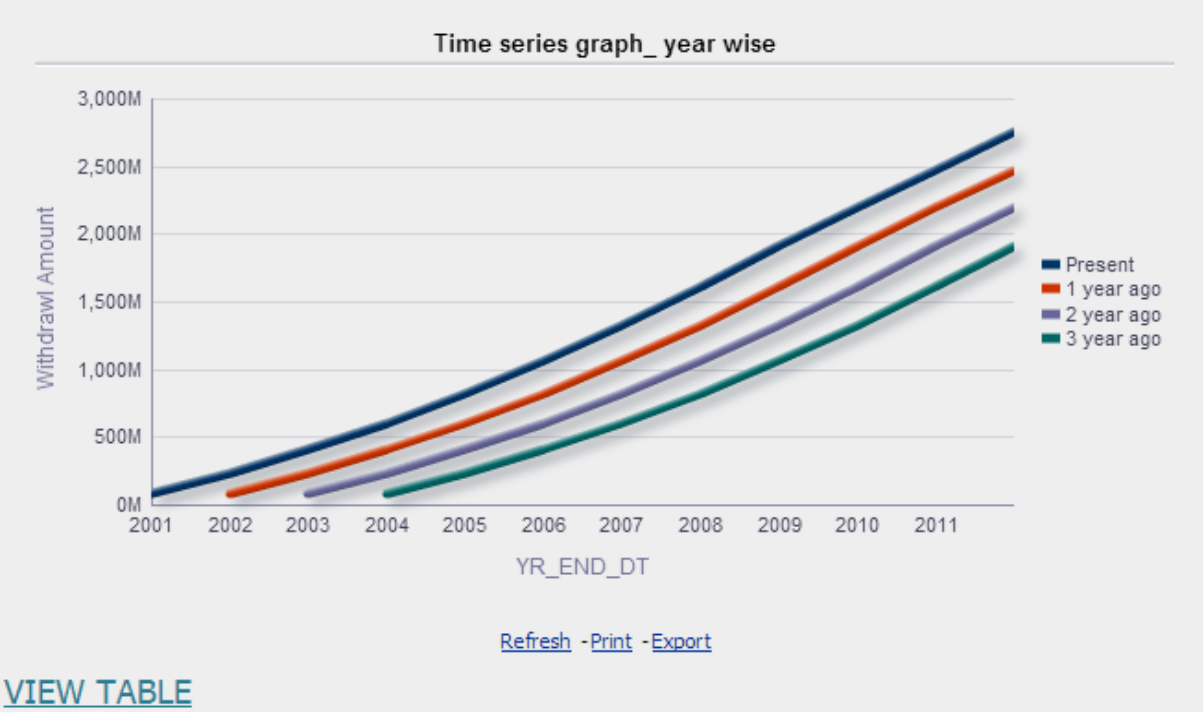

Figure 9.14: Times - series 1, 2 & 3 year(s) ago ATM Withdrawals Report

This graph gives the view of present ATM withdrawal amount and compares to 3 years back withdrawal amount.

9.2 ATM Card Fraud Detection

The following is the table obtained from our system for finding the fraudulent ATM transactions.

Table 9.1: Fraudulent ATM Card Transactions Sorted According to Withdrawal Timestamp

ATM Card Number	ATM Place	ATM Number	Withdrawal Timestamp	Withdrawal Amount	Fraud Percentage
4003142846561480	Putalisadak	III	6/4/2012 7:40:32 PM	500.00	85.10%
4003144845073750	Boudha-Pipalbot	-	6/4/2012 10:23:37 AM	10,500.00	88.51%
4543005356584500	Banepa	-	6/4/2012 10:02:10 AM	500.00	82.37%
4543806376277380	Bhaktapur	-	6/1/2012 10:22:28 AM	31,500.00	72.16%
4003148831051340	Thamel	-	5/29/2012 11:02:25 AM	500.00	76.31%
4003145617767840	Mangal Bazar	-	5/25/2012 11:54:30 AM	1,000.00	89.76%
4003142815053280	Gongabu	II	5/19/2012 10:34:48 PM	2,500.00	71.48%
4003147545838280	New Baneshwor	I	5/15/2012 4:48:36 PM	1,000.00	75.99%
4003142415171530	Butwal	-	5/14/2012 6:44:35 PM	1,500.00	83.65%
4003148867424650	Lalbandhi	-	5/12/2012 1:23:25 PM	2,500.00	90.02%
4003145412304180	Lakeside Pokhara	-	5/7/2012 5:30:59 PM	1,000.00	83.91%
4003145412304180	Lakeside Pokhara	-	5/7/2012 12:45:24 PM	3,000.00	89.30%
4003148782024320	Dhangadi	-	5/6/2012 6:38:23 PM	2,000.00	82.36%
4003144862614750	Dhangadi	-	5/5/2012 10:09:30 AM	1,500.00	76.62%
4003148304431430	Maharajgunj	-	4/28/2012 11:02:25 AM	2,000.00	83.83%
4003143606408370	Lagankhel	-	4/26/2012 4:42:47 PM	500.00	71.13%
4543275784036230	Gongabu	II	4/26/2012 12:21:04 PM	6,500.00	92.15%
4003140021481060	Pulchowk	I	4/24/2012 1:32:03 AM	40,000.00	85.04%

Table 1 shows that the possible fraudulent transactions can be viewed by the user and appropriate action can be taken through the Fraud Prevention model of the bank. As shown from the table, most of the fraudulent transactions are from the ATM locations in Kathmandu valley. If the fraud percentage, the percentage upto which the given transaction can be fraud. If the fraud percentage is above 90%, then it is shown with red color indicating that it is noticeable transaction.

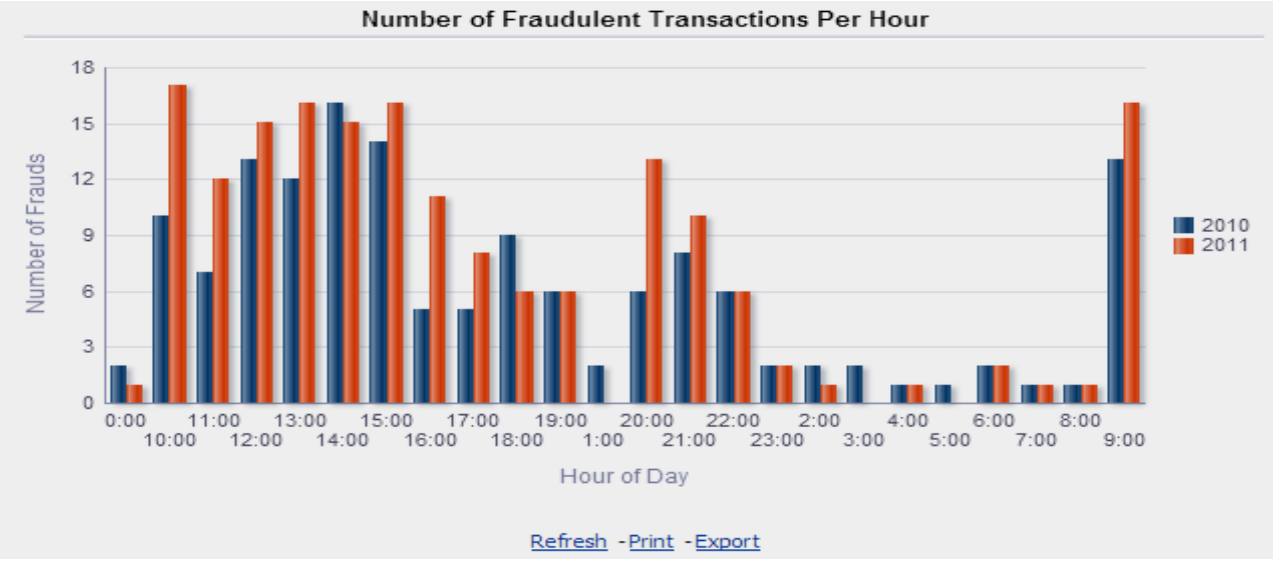

Figure 9.15: Number of Fraudulent Transactions in each Hour of day for year 2010 & 2011

The Figure 2 shows the number of fraudulent transactions in each our of 24-hour day in the year of 2010 & 2011. The figure shows that most of the fraudulent transactions occur during 10:00 AM, 9:00 AM & 3:00 PM in the year of 2011. Moreover, in 2010, most of the fraudulent transactions occur during 2:00 PM, 3:00 PM & 9:00 AM of the day. It also showed that there were no fraudulent transactions during 1:00 AM in the year 2011.

9.3 Customer Churn Prediction

After running the customers' transactions data into the churn prediction model, we got an interesting results of customer churned behavior. We analyzed the churned behavior in following dimensions: (a) Location wise (b) Account Type wise and (c) Income wise.(d) Qualification wise

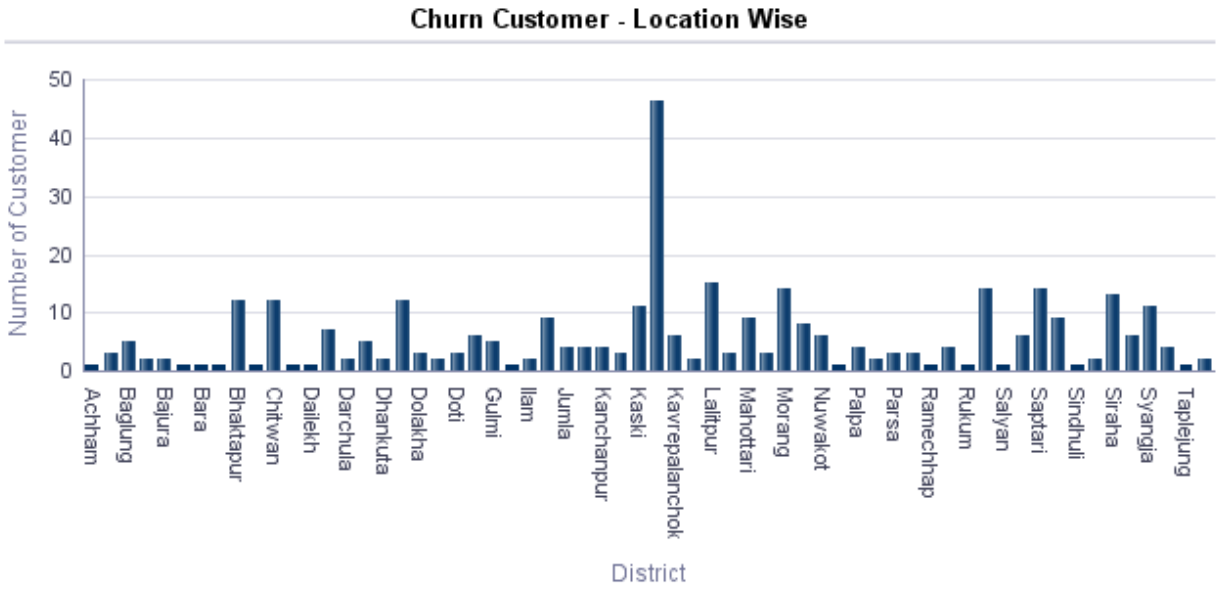

Figure 9.16: Location wise customer churn prediction

Figure 9.16 shows that districts which have high bank's customers also predicted to have high churned customer. Kathmandu district is predicted to have highest number of churned customers

and also high percentage of churned customer. This shows that the people from Kathmandu switch their bank account frequently from one bank to another. High competition between banks and more choices among banks may resulted this churned behavior.

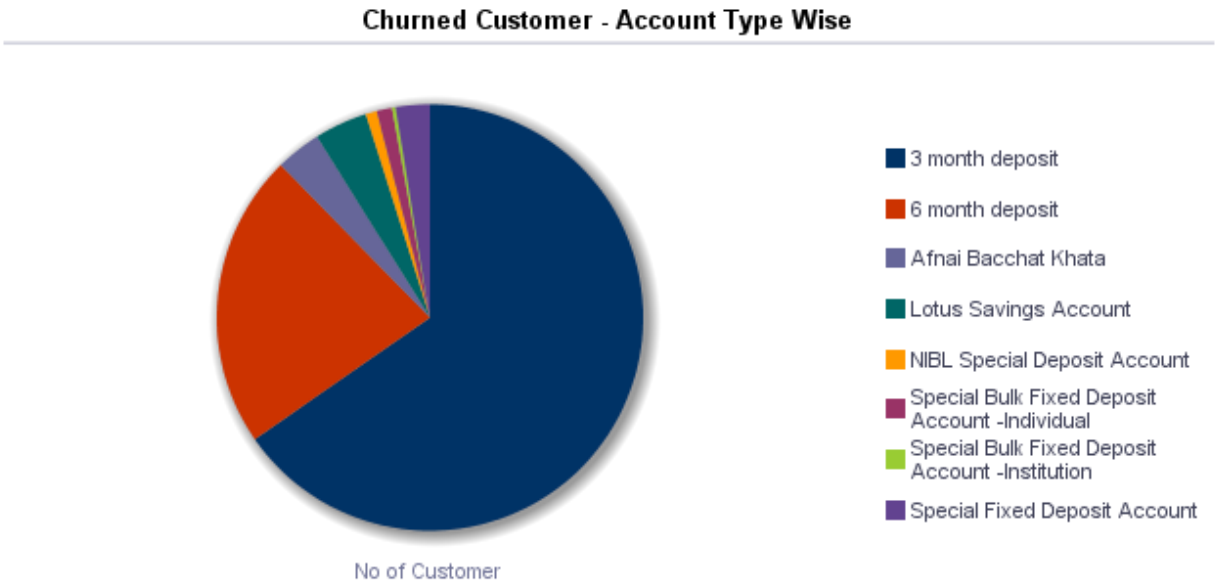

Figure 9.17: Account type wise churn prediction

Figure 9.17 shows that the account type which has high churned customer is 3 months deposit. People are not interested toward 3 months deposit account type and if they open their account in this account type they are more likely to close that account in the future than other account type. The most likely region for this may be interest rate, services etc.

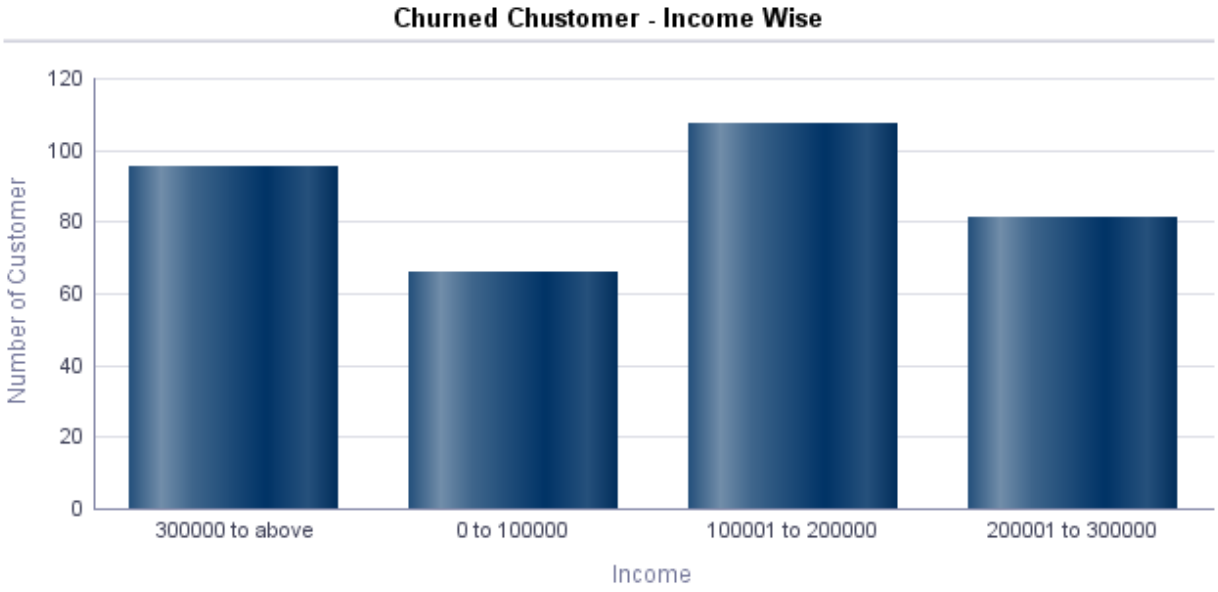

Figure 9.18: Income wise churn customer

Figure 9.18 shows the number of customer that are predicted to be churned income wise. People having annual income between Rs. 100000 – 20000 have high percentage churn rate. The last dimension analyzed is qualification wise.

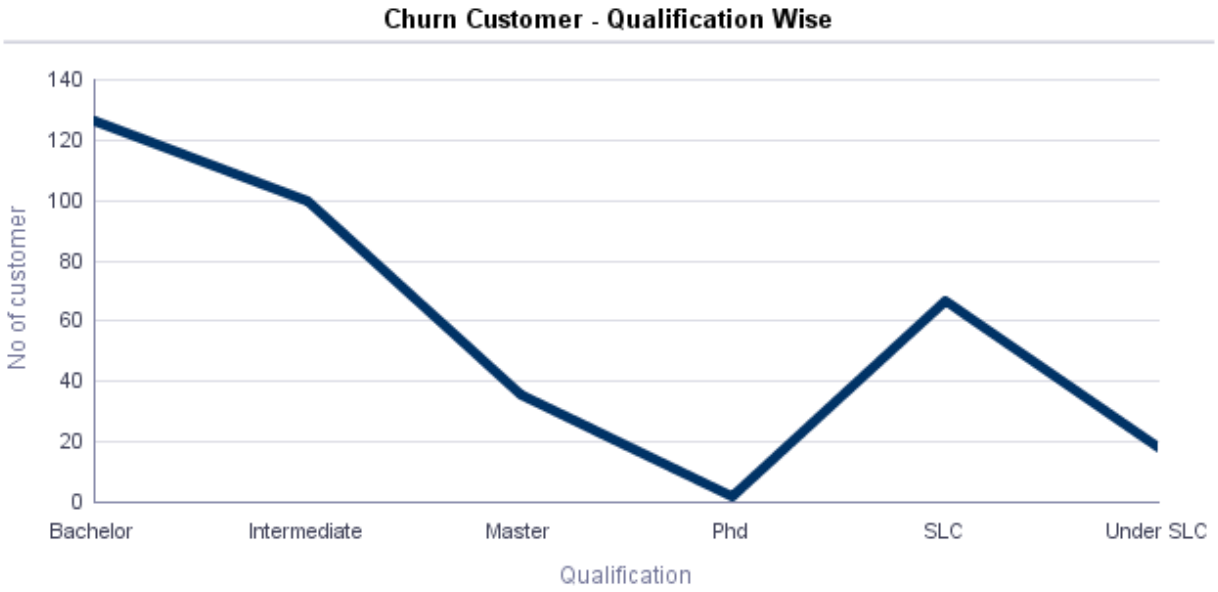

Figure 9.19: Qualification wise churn prediction

Figure 9.19 shows the predicted no of customer based on their academic qualification.

9.4 Limitations & Further Enhancements

- Fraud detection is a challenging field of research, development and creativity. There are no universal fraud patterns and these patterns change dynamically. So processing of the millions of transactions per second in real time is challenging and some of the self-learning algorithm fraud pattern detection can be further enhancement for the project.
- Due to the high confidentiality of the banking data, we didn't get enough field of data to make rich features set of customer transaction to predict the customer churn behavior. It can be further enhanced by adding more features while designing the model.
- Trend analysis, customer segmentation and time series analysis are built analyzing the data without using any data mining algorithms. This can be made more intelligent using specific data mining tools and techniques.
- The project doesn't work on real time due to data-warehousing. We have used Online Analytical Processing. This can be further enhanced by using Online Transaction Processing to make it to handle real time processing.

9.5 Conclusions

Analysis on world of Bank domain is the great part of the project. Although, validating the data, extracting the useful information, building the high level data model and implementing the efficient algorithm for ATM fraud detection & customer churn prediction is most challenging part of our project we became successful to build the application which visualizes the information for decision support due to our extensive effort and time.

As a whole, complete project development lifecycle has become the good learning experience for us. We learn the important of systematic development process to carry out the application smoothly. At the same time, we learn how to work on team as professionally and guide us to developed enterprise level application in near future.

REFERENCES & Bibliography

[1] K. Chitra and B. Subashini, Customer Retention in Banking Sector using Predictive Data Mining Technique, India: Thiagarajar School Of Management, 2011.

[2] Republica Daily, "ATM fraud on rise, banks keep mum fearing embarrassment," in *Nepal Republic Media*, Kathmandu, Nepal, 2013.

[3] V. Bhambri, "Data Mining as a Tool to Predict Churn Behaviour of Customers," *International Journal of Computer & Organization Trends,* vol. II, no. 3, 2008.

[4] D. Bhattarai and S. D.R., Banking and Financial Statistics, Kathmandu, Nepal: Nepal Rastra Bank, 2012.

[5] S. Dibb, "Market Segmentation: Strategies for Success," in *Marketing Intelligence & Planning*, USA, 1998.

[6] C. K and S. B., "Customer Retention in Banking Sector using Predictive Data Mining Technique," *ICIT The 5th International Conference on Information Technology,* 2011.

[7] O. Alis, E. Karakurt and P. Melli, "Data Mining for Database Marketing at Garanti Bank," in *Data Mining 2000, WIT Publications*, USA, 2000.

[8] H.-Y. Liu, "Development of a Framework for Customer Relationship Management (CRM) in the Banking Industry," *International Journal of Management,* 2007.

[9] D. SINGH, "Frauds related to bank cards on rise lately," International Media Network Nepal Pvt. Ltd., 11 January 2013. [Online]. Available: http://www.thehimalayantimes.com/fullTodays.php?headline=Frauds+related+to+bank+cards+on+rise+lately+&NewsID=361777. [Accessed 23 April 2013].

[10] P. Yakuel, "Churn Prediction Prevention," Optimove Learning Center, 2012. [Online]. Available: http://www.optimove.com/churn-prediction-prevention.aspx. [Accessed 24 January 2013].

[11] D. U. D. Prasad and S. Madhavi, "Prediction of Churn Behavior of Bank Customers using Data Mining Tools," *Business Intelligence Journal,* 2012.

[12] P. Lane, V. Schupmann and I. Stuar, Oracle Database Data Warehousing Guide, 11g, Oracle Inc., 2007.

[13] E. Melnick, P. Nayyar, M. Pinedo and S. Seshadri, "Creating Valuein Financial Services: Strategies, Operations and Technologies," in *Kluwer Academic Publisher*, USA, 2000.

[14] S. Sun, "An Analysis on the Conditions and Methods of Marketing Segmentation," *International Journal of Business and Management,* vol. III, no. 4, pp. 233-242, 2009.

[15] B. Ubiparipovic and E. Durkovic, Application of Business Intelligence in the Banking Industry, USA: Management Information System, 2011.

[16] J. Sarokin, Oracle BI 11g R1: Build Repositories volume I, USA: Oracle Inc., 2011.

[17] J. Sarokin, Oracle BI 11g R1: Build Repositories volume II, USA: Oracle Inc., 2011.

[18] T. Fawcett and F. Provost, "Adaptive fraud detection," *Data Mining and Knowledge Discovery,* 2009.

[19] W. Kowalczyk, "Detecting Fraud with Data Mining: Three Success Stories," in *Fraud Detection Expertise Center, Leiden University*, Rapenburg 70, 2311 EZ Leiden, Netherlands, 2012.

[20] F. Ogwueleka, "Data Mining Application in Credit Card Fraud Detection System," *Journal of Engineering Science and Technology,* vol. VI, 2011.

[21] M. M. Campos and B. L.Milenova, "Creation and Deployment of Data Mining-Based Intrusion Detection System in Oracle Database 10g," in *Oracle Data Mining Technologies*, USA, 2012.

[22] L. Pekelis, "Classification And Regression Trees: A Practical Guide for Describing a Dataset," *Bicostal Datafest,* 2013.

[23] F. Khaloof and R. Razouk, "Using of Data Mining Techniques for Fraud Detection in Banking System," *Damascus University Journal,* vol. 25, 2009.

[24] W. Au, C. C. Chan and X. Yao, "A Novel Evolutionary Data Mining Algorithm with Applications to Churn Prediction," *IEEE Transactions on Evolutionary Computation,* vol. VII, no. 6, pp. 532-545, 2003.

[25] R. Bolton and D. J. Hand, "Statistical Fraud Detection: A Review," *Statistical Science,* vol. XVII, 2002.

[26] F. Ogwueleka, "Data Mining Application in Credit Card Fraud Detection," *Journal of Engineering Science & Technology,* vol. VI, no. 3, pp. 311-322, 2011.

[27] K. Tsiptsis and A. Chorianopoulos, Data Mining Techniques in CRM: Inside Customer Segmentation, United Kingdom: A John Wiley and Sons, Ltd., 2009.

[28] M. Hornick, Learning Oracle R-series, USA: Oracle Inc., 2012.

[29] J. Hadden and A. Tiwari, "Churn Prediction using Compliants Data," *World Academy of Science,* 2006.

[30] S. Sivaprakasam, Churn Prediction: Approach to retain Profitable Customer, India: Infosys, 2010.

[31] M. Cahill, D. Lambert, P. C. and D. Sun, "Detecting Fraud in the Real World," 2000.

[32] P. Chan K, w. Fan, A. Prodromidis and S. Stolfo, "Distributed Data Mining in Credit Card Fraud Detection," in *Florida Institute of Technology*, USA, 2011.

[33] S. Urman, R. Hardman and M. Mclaughin, Oracle Database 10g PL/SQL Programming, Osborne: MCGraw-Hill, 2010.

[34] L.Milenova, Boriana, J. S.Yarmus and M. M.Capmos, "SVM in Oracle Database 10g: Removing the Barrier to Widespread Adoption of Support Vector Machines," in *Oracle Data Mining Technologies*, USA, 2010.

[35] J. Peppard, "Customer Relationship Management (CRM) in Financial Services," *European Management Journal,* vol. XVIII, 2000.

[36] R. T. F. Trevor Hastie, The Elements of Statistical Learning - Data Mining, Inference and Prediction, California, USA: Springer, 2008.

[37] J. N. Hannu Hautakangas, "Anomaly detection using one-class SVM with wavelet Packet Decomposition," UNIVERSITY OF JYVÄSKYLÄ, Jyväskylä, 2011.

[38] S. E, H. Y, A. K and S. M., "A Proposed Churn Prediction Model," *International Journal of Engineering Research and Application (IJERA),* vol. II, no. 4, pp. 693-697, 2012.

APPENDIX – A: APPLICATION SNAPSHOTS

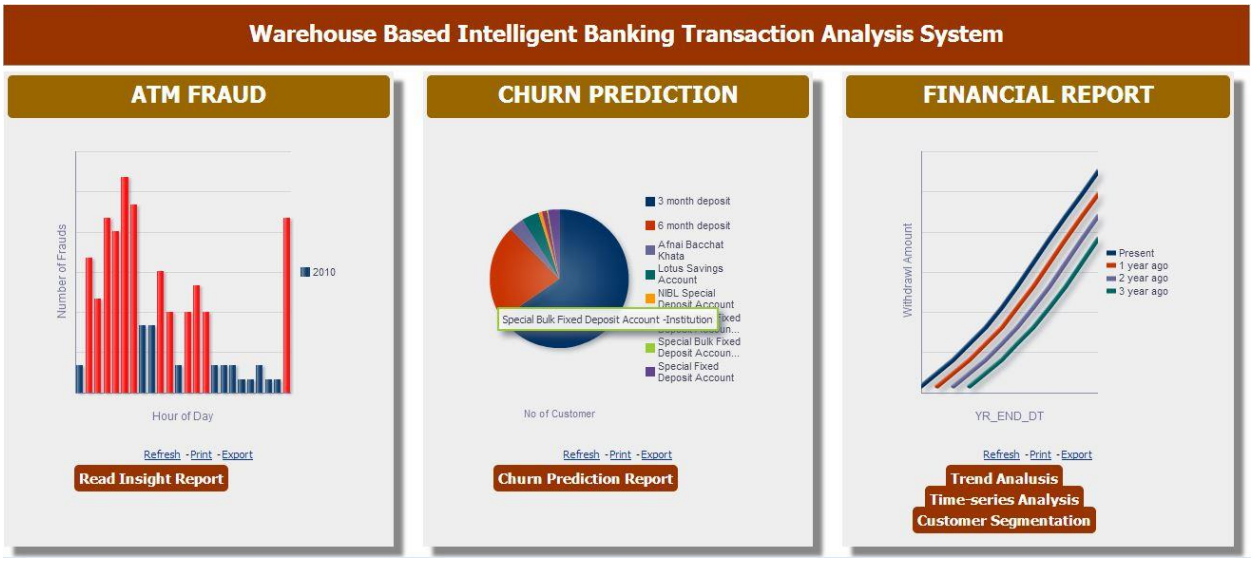

Figure A.0.1: Home Page of the Application showing the Three Features

Figure above shows the home page of Warehouse Based Intelligent Banking Transaction Analysis System. User views this home page in the first look. It consists of 3 parts: ATM Fraud, Churn Prediction and Financial report. This is the menu page which consists of navigation link through which is navigated to particular page as per the user choice. Financial Reports has 3 navigation links associated to it: Trend Analysis, Time series Analysis and Geo-demographic segmentation.

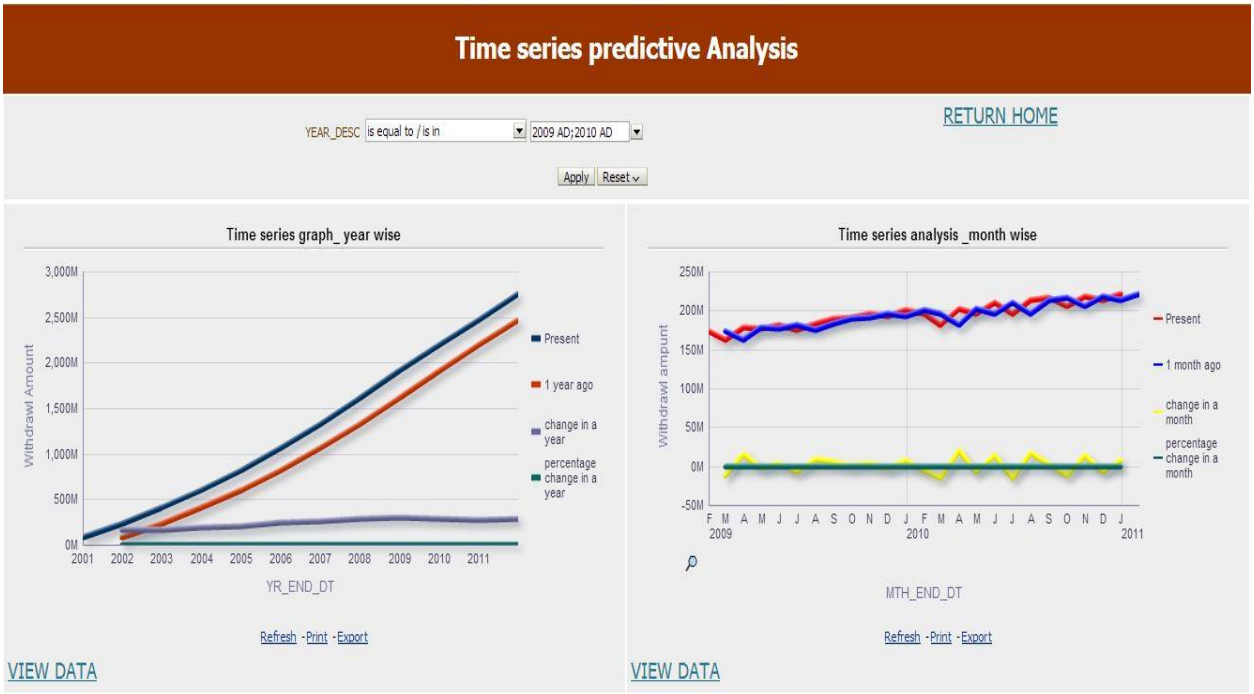

Figure A.2: Time series Predictive Analysis

Figure above shows the Time Series Analysis Page which is navigated from the home page. This page consists of graphs and also navigation links like Return Home. This link is for returning back to home page. User can also view data through the view data link.

APPENDIX – B: BANKING DATA SCHEMA USED AS DATA SOURCE

```
CREATE TABLE [Master] (
        [MainCode] [u_MainCodeLen] NOT NULL ,
        [BranchCode] [char] (3) COLLATE Latin1_General_BIN NOT NULL ,
        [AcType] [char] (2) COLLATE Latin1_General_BIN NOT NULL ,
        [CyCode] [char] (2) COLLATE Latin1_General_BIN NOT NULL ,
        [Name] [varchar] (35) COLLATE Latin1_General_BIN NOT NULL ,
        [ClientCode] [char] (5) COLLATE Latin1_General_BIN NOT NULL ,
        [Obligor] [char] (5) COLLATE Latin1_General_BIN NOT NULL ,
        [StmntFrq] [char] (1) COLLATE Latin1_General_BIN NOT NULL ,
        [IntCalcFrq] [char] (1) COLLATE Latin1_General_BIN NOT NULL ,
        [IntPostFrqCr] [char] (1) COLLATE Latin1_General_BIN NOT NULL ,
        [IntPostFrqDr] [char] (1) COLLATE Latin1_General_BIN NOT NULL ,
        [TaxPostFrq] [char] (1) COLLATE Latin1_General_BIN NOT NULL ,
        [BalnXfrFrq] [char] (1) COLLATE Latin1_General_BIN NOT NULL ,
        [IntCalcTypeCr] [char] (1) COLLATE Latin1_General_BIN NOT NULL ,
        [IntCalcTypeDr] [char] (1) COLLATE Latin1_General_BIN NOT NULL ,
        [ValueDateTrans] [char] (1) COLLATE Latin1_General_BIN NOT NULL ,
        [MoveType] [char] (1) COLLATE Latin1_General_BIN NOT NULL ,
        [LimitType] [char] (1) COLLATE Latin1_General_BIN NOT NULL ,
        [CommisionCode] [char] (1) COLLATE Latin1_General_BIN NULL ,
        [StmntDelivery] [char] (1) COLLATE Latin1_General_BIN NULL ,
```

[Balance] [money] NULL ,

[GoodBaln] [money] NULL ,

[Limit] [money] NULL ,

[MiscBaln] [money] NULL ,

[HeldAmt] [money] NULL ,

[LastDayBaln] [money] NULL ,

[HighBaln] [money] NULL ,

[LowBaln] [money] NULL ,

[CumCrBaln] [money] NULL ,

[CumDrBaln] [money] NULL ,

[CumCrAmt] [money] NOT NULL ,

[CumDrAmt] [money] NOT NULL ,

[IntCrAmt] [money] NULL ,

[IntDrAmt] [money] NULL ,

[IntPaidDr] [money] NULL ,

[IntPaidCr] [money] NULL ,

[TotalTax] [money] NULL ,

[TotWithDraw] [money] NULL ,

[NoStandInstr] [smallint] NULL ,

[NoStopChq] [smallint] NULL ,

[NoTransCr] [smallint] NULL ,

[NoTransDr] [smallint] NULL ,

[NoOfWDNotice] [smallint] NULL ,

[NoOfSign] [smallint] NULL ,

[NoOfBlobs] [smallint] NULL ,

[CanTranCrossCy] [char] (1) COLLATE Latin1_General_BIN NULL ,

[IsDormant] [char] (1) COLLATE Latin1_General_BIN NULL ,

[IsBlocked] [char] (1) COLLATE Latin1_General_BIN NULL ,

[IsSpecial] [char] (1) COLLATE Latin1_General_BIN NULL ,

[CanTranDeno] [char] (1) COLLATE Latin1_General_BIN NULL ,

[HasStmntValue] [char] (1) COLLATE Latin1_General_BIN NULL ,

[ReconAuto] [char] (1) COLLATE Latin1_General_BIN NULL ,

[IsNormalDr] [char] (1) COLLATE Latin1_General_BIN NULL ,

[HasNomForDrInt] [char] (1) COLLATE Latin1_General_BIN NULL ,

[HasNomForCrInt] [char] (1) COLLATE Latin1_General_BIN NULL ,

[HasNomForTax] [char] (1) COLLATE Latin1_General_BIN NULL ,

[DoSummTran] [char] (1) COLLATE Latin1_General_BIN NULL ,

[IntRateScheme] [char] (1) COLLATE Latin1_General_BIN NULL ,

[HasGroupLimit] [char] (1) COLLATE Latin1_General_BIN NULL ,

[AcOpenDate] [datetime] NULL ,

[LastTranDate] [datetime] NULL ,

[LimitExpiryDate] [datetime] NULL ,

[StmntStartDate] [datetime] NULL ,

[StmntShowDate] [datetime] NULL ,

[BalnChangedDate] [datetime] NULL ,

[AcOfficer] [varchar] (6) COLLATE Latin1_General_BIN NULL ,

[Remarks] [varchar] (20) COLLATE Latin1_General_BIN NULL ,

[PassCode] [varchar] (5) COLLATE Latin1_General_BIN NULL ,

[TaxAccrued] [money] NULL ,

[DoSummBaln] [char] (1) COLLATE Latin1_General_BIN NULL ,

[IntCrRate] [float] NULL ,

[IntDrRate] [float] NULL ,

[MarginRate] [float] NULL ,

[EffectAfterDays] [smallint] NULL ,

[CanNotBeNominated] [varchar] (1) COLLATE Latin1_General_BIN NULL ,

[ProductSchemes] [varchar] (7) COLLATE Latin1_General_BIN NULL ,

[IncludeMinBaln] [tinyint] NULL ,

[HasWarning] [varchar] (1) COLLATE Latin1_General_BIN NULL ,

[WarningMesg] [varchar] (50) COLLATE Latin1_General_BIN NULL ,

[StmntCalendar] [varchar] (1) COLLATE Latin1_General_BIN NULL ,

[IntroducedBy] [varchar] (20) COLLATE Latin1_General_BIN NULL ,

[Beneficiary] [varchar] (50) COLLATE Latin1_General_BIN NULL ,

[IsIntDrRateFloat] [varchar] (1) COLLATE Latin1_General_BIN NULL ,

[IsIntCrRateFloat] [varchar] (1) COLLATE Latin1_General_BIN NULL ,

[TrackDP] [varchar] (1) COLLATE Latin1_General_BIN NULL ,

[DPExpiryDate] [datetime] NULL ,

[DrawingPower] [money] NULL ,

[LastChqNo] [int] NULL ,

[UseLastCheque] [varchar] (1) COLLATE Latin1_General_BIN NULL ,

[LastStmntPageNo] [int] NULL ,

[DPPenalRate] [float] NULL ,

[ParentRef] [varchar] (12) COLLATE Latin1_General_BIN NULL ,

[PBLineNo] [tinyint] NULL ,

[PBPageNo] [tinyint] NULL ,

[AtmChargeAmt] [money] NULL ,

[TaxOnInt] [float] NULL ,

[TaxPercentOnInt] [float] NULL ,

[ServChgCode] [varchar] (1) COLLATE Latin1_General_BIN NULL ,

[OldAcNum] [u_MainCodeLen] NULL ,

[AcType1] [char] (2) COLLATE Latin1_General_BIN NULL ,

[CyCode1] [char] (2) COLLATE Latin1_General_BIN NULL ,

[BonusCode] [varchar] (1) COLLATE Latin1_General_BIN NULL ,

[MinAmtForInt] [money] NULL ,

[AcTypeType] [char] (1) COLLATE Latin1_General_BIN NOT NULL ,

[SendForRecon] [char] (1) COLLATE Latin1_General_BIN NULL ,

[LimitReusableCount] [int] NULL ,

[UsedLimit] [money] NULL ,

[MinBalnReqd] [money] NULL ,

```
	[IntRateOnPastInt] [float] NULL
) ON [PRIMARY]
GO

CREATE TABLE [TransDaily] (
	[BranchCode] [char] (3) COLLATE Latin1_General_BIN NOT NULL ,
	[AcType] [char] (2) COLLATE Latin1_General_BIN NOT NULL ,
	[CyCode] [char] (2) COLLATE Latin1_General_BIN NOT NULL ,
	[MainCode] [u_MainCodeLen] NOT NULL ,
	[TranDate] [datetime] NOT NULL ,
	[Desc1] [varchar] (35) COLLATE Latin1_General_BIN NULL ,
	[Desc2] [varchar] (20) COLLATE Latin1_General_BIN NULL ,
	[Desc3] [varchar] (20) COLLATE Latin1_General_BIN NULL ,
	[ValueDate] [datetime] NOT NULL ,
	[TranCyCode] [char] (2) COLLATE Latin1_General_BIN NOT NULL ,
	[Amount] [money] NULL ,
	[LCYAmount] [money] NULL ,
	[TranId] [char] (9) COLLATE Latin1_General_BIN NOT NULL ,
	[TranCode] [char] (3) COLLATE Latin1_General_BIN NULL ,
	[ApprovedBy] [char] (6) COLLATE Latin1_General_BIN NULL ,
	[EnteredBy] [char] (6) COLLATE Latin1_General_BIN NOT NULL ,
	[Status] [char] (1) COLLATE Latin1_General_BIN NULL ,
```

```
	[ABBSCode] [char] (1) COLLATE Latin1_General_BIN  NULL ,
	[HasProcessed] [char] (1) COLLATE Latin1_General_BIN  NULL ,
	[Notice] [char] (1) COLLATE Latin1_General_BIN  NULL ,
	[IsGenerated] [char] (1) COLLATE Latin1_General_BIN  NULL ,
	[RateCode] [char] (1) COLLATE Latin1_General_BIN  NULL ,
	[ReferenceNo] [varchar] (20) COLLATE Latin1_General_BIN  NULL ,
	[BankCode] [char] (3) COLLATE Latin1_General_BIN  NULL ,
	[ChequeNo] [u_ChqStrLen]  NULL ,
	[AcTypeType] [varchar] (1) COLLATE Latin1_General_BIN  NULL ,
	[SeqNo] [int] IDENTITY (1, 1) NOT FOR REPLICATION  NOT NULL ,
	[NostroBranch] [varchar] (3) COLLATE Latin1_General_BIN  NULL ,
	[CityCode] [varchar] (3) COLLATE Latin1_General_BIN  NULL ,
	[IsPBPrinted] [varchar] (1) COLLATE Latin1_General_BIN  NULL ,
	[TimeStamp] [datetime]  NULL ,
	[InterBank] [char] (1) COLLATE Latin1_General_BIN  NULL ,
	[ReconType] [varchar] (2) COLLATE Latin1_General_BIN  NULL ,
	[NostroCode] [varchar] (3) COLLATE Latin1_General_BIN  NULL
) ON [PRIMARY]
GO

CREATE TABLE [dbo].[ClientTable] (
	[ClientCode] [char] (5) COLLATE Latin1_General_BIN  NOT NULL ,
```

[Name] [varchar] (35) COLLATE Latin1_General_BIN NOT NULL ,

[Address1] [varchar] (35) COLLATE Latin1_General_BIN NOT NULL ,

[Address2] [varchar] (35) COLLATE Latin1_General_BIN NULL ,

[Address3] [varchar] (35) COLLATE Latin1_General_BIN NULL ,

[City] [varchar] (25) COLLATE Latin1_General_BIN NULL ,

[Phone] [varchar] (25) COLLATE Latin1_General_BIN NULL ,

[CountryCode] [char] (2) COLLATE Latin1_General_BIN NOT NULL ,

[Obligor] [char] (5) COLLATE Latin1_General_BIN NULL ,

[KeepMIS] [char] (1) COLLATE Latin1_General_BIN NULL ,

[ClientTag1] [varchar] (6) COLLATE Latin1_General_BIN NULL ,

[ClientTag2] [varchar] (6) COLLATE Latin1_General_BIN NULL ,

[ClientTag3] [varchar] (6) COLLATE Latin1_General_BIN NULL ,

[ClientId] [varchar] (20) COLLATE Latin1_General_BIN NULL ,

[Gender] [varchar] (1) COLLATE Latin1_General_BIN NULL ,

[DateOfBirth] [datetime] NULL ,

[Beneficiary] [varchar] (50) COLLATE Latin1_General_BIN NULL ,

[AcOfficer] [varchar] (6) COLLATE Latin1_General_BIN NULL ,

[FaxNo] [varchar] (15) COLLATE Latin1_General_BIN NULL ,

[eMail] [varchar] (50) COLLATE Latin1_General_BIN NULL ,

[ContactAdd1] [varchar] (35) COLLATE Latin1_General_BIN NULL ,

[ContactAdd2] [varchar] (35) COLLATE Latin1_General_BIN NULL ,

[ContactAdd3] [varchar] (35) COLLATE Latin1_General_BIN NULL ,

[IntroducedBy] [varchar] (20) COLLATE Latin1_General_BIN NULL ,

[ClientStatus] [varchar] (1) COLLATE Latin1_General_BIN NULL ,

[OffPhone] [varchar] (25) COLLATE Latin1_General_BIN NULL ,

[CompanyName] [varchar] (50) COLLATE Latin1_General_BIN NULL ,

[MobileNo] [varchar] (12) COLLATE Latin1_General_BIN NULL ,

[PagerNo] [varchar] (12) COLLATE Latin1_General_BIN NULL ,

[NoOfDependents] [tinyint] NULL ,

[SpouseName] [varchar] (35) COLLATE Latin1_General_BIN NULL ,

[FathersName] [varchar] (35) COLLATE Latin1_General_BIN NULL ,

[GFathersName] [varchar] (35) COLLATE Latin1_General_BIN NULL ,

[Designation] [varchar] (50) COLLATE Latin1_General_BIN NULL ,

[InsuranceInfo] [varchar] (255) COLLATE Latin1_General_BIN NULL ,

[MaritalStatus] [varchar] (1) COLLATE Latin1_General_BIN NULL ,

[Salutation] [varchar] (5) COLLATE Latin1_General_BIN NULL ,

[MailStmnt] [varchar] (1) COLLATE Latin1_General_BIN NULL ,

[BranchCodeOfAcOp] [varchar] (3) COLLATE Latin1_General_BIN NULL ,

[CIFCode] [varchar] (10) COLLATE Latin1_General_BIN NULL ,

[PinnedMesg] [varchar] (255) COLLATE Latin1_General_BIN NULL ,

[CitizenshipNo] [varchar] (15) COLLATE Latin1_General_BIN NULL ,

[CitizenDistrict] [varchar] (25) COLLATE Latin1_General_BIN NULL ,

[PassportNo] [varchar] (15) COLLATE Latin1_General_BIN NULL ,

[PassportExpiryDate] [datetime] NULL ,

```
        [PassportCountry] [varchar] (25) COLLATE Latin1_General_BIN NULL ,
        [WebAddress] [varchar] (100) COLLATE Latin1_General_BIN NULL ,
        [AlternateAdd1] [varchar] (35) COLLATE Latin1_General_BIN NULL ,
        [AlternateAdd2] [varchar] (35) COLLATE Latin1_General_BIN NULL ,
        [AlternateAdd3] [varchar] (35) COLLATE Latin1_General_BIN NULL ,
        [Limit] [money] NULL ,
        [NepaliName] [varchar] (50) COLLATE Latin1_General_BIN NULL ,
        [IdType] [varchar] (5) COLLATE Latin1_General_BIN NULL ,
        [UsedLimit] [money] NULL ,
        [UsedLimitOB] [money] NULL ,
        [LimitOB] [money] NULL
) ON [PRIMARY]
GO
```