Report on the Payments Data Warehouse

Ashutosh Pandey, IIIT Delhi
Vaibhav Pande, IIIT Delhi

1. INTRODUCTION

Good Business Intelligence (BI), allows organizations to query data obtained from trusted sources and use the answers gathered by this data to take good decisions and gain a competitive edge in the industry. The first step though for achieving effective BI is a well-designed warehouse. Data warehouse design is the process of building a solution to integrate data from multiple sources that support analytical reporting and data analysis. A poorly designed data warehouse can result in acquiring and using inaccurate source of data that negatively affect the productivity and growth of your organization.

Our problem at hand was to identify a data warehouse for handling 3 decisions using the information. The company at hand was a multinational company which was into manufacturing. It could operate out of 11 countries where each country would have a seprate business unit. Each unit would comprise the business activity, sales, purchase, orders and payments received. The customers of a country will have to place an order to the business unit of the county they are in. Once the product is delivered, an invoice is generated with the quantity of each product purchased and the payment details.

The company though could also be purchasing raw material from suppliers by placing orders on them and receiving an inward invoice.

We targeted the purchase problem of the data warehouse and were tasked upon to handle the following decisions :-

(1) Which customers need to be contacted to make their outstanding payments ?
(2) Refund excess payments made by customers ?
(3) Which business unit has maximum outstanding ?

A whole process of making influence diagrams, making star schema and the ETL process was followed and a data warehouse fragment was made.
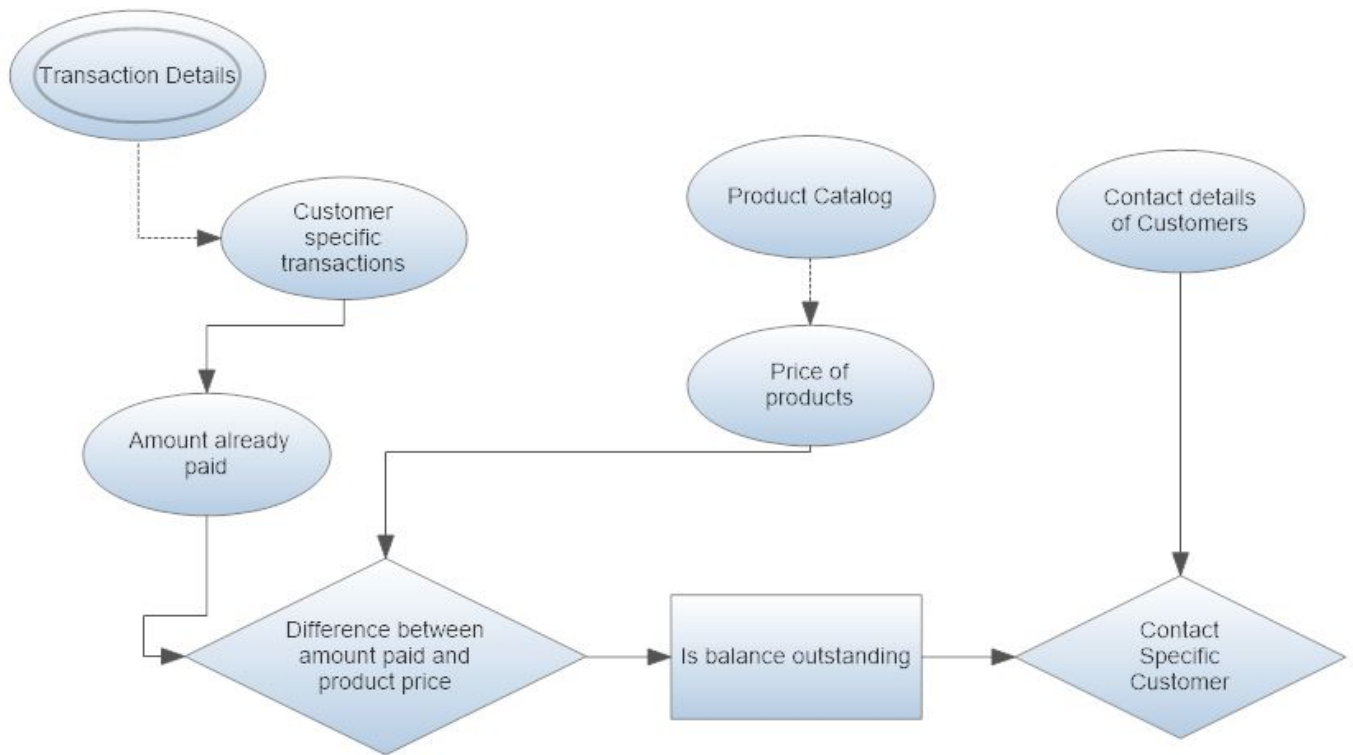
Fig. 1. Influence Diagram for Problem 1

2. INFLUENCE DIAGRAMS

An influence diagram (ID) is a compact graphical and mathematical representation of a decision situation. As has been taught in the class a decision modelling using influence diagrams has been done.

2.1 Problem 1

The influence diagram for problem 1 has been shown in Figure 1. This shows the decision flow to decide the which customers should be contacted to make their outstanding payments.

2.2 Problem 2

The influence diagram for problem 2 has been shown in Figure 2. This shows the decision flow on how the refunding of excess payments will be done.

2.3 Problem 3

The influence diagram for problem 3 has been shown in Figure 3. This shows the decision flow in order to get the business unit which has the maximum outstanding balance.

3. REQUIREMENTS

The requirements per problem have been got in the following way :-

(1) First a Ends means hierarchy was created
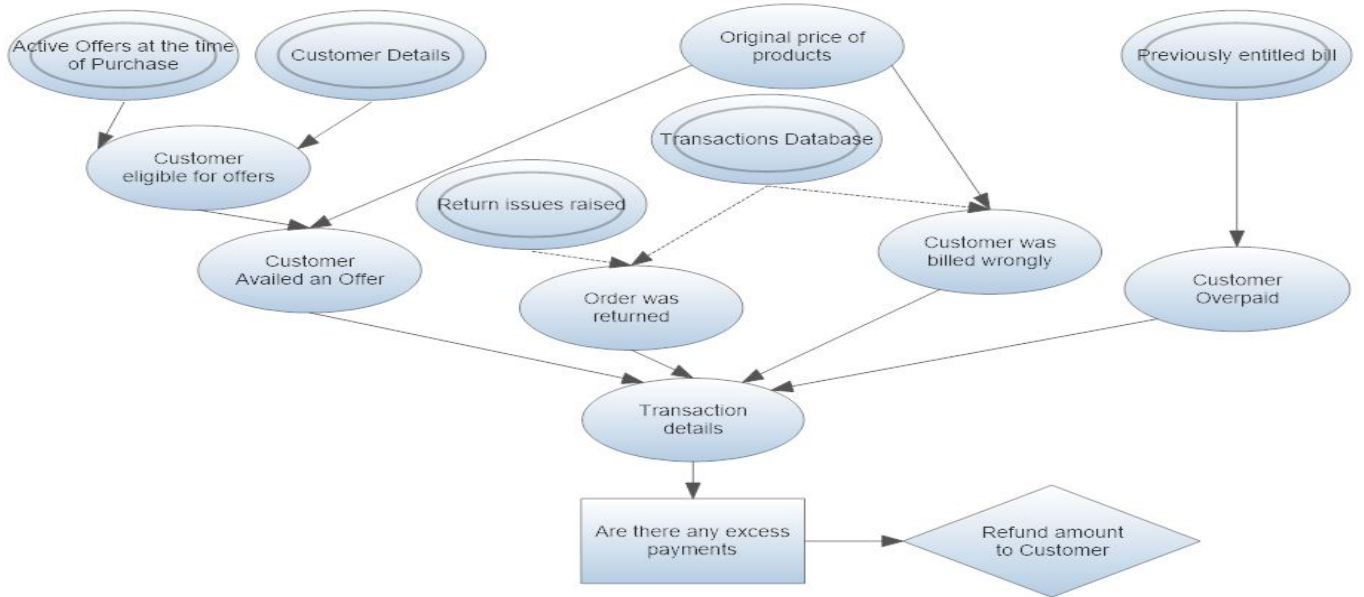(2) Removal of redundant facts that would have lead to the same decision
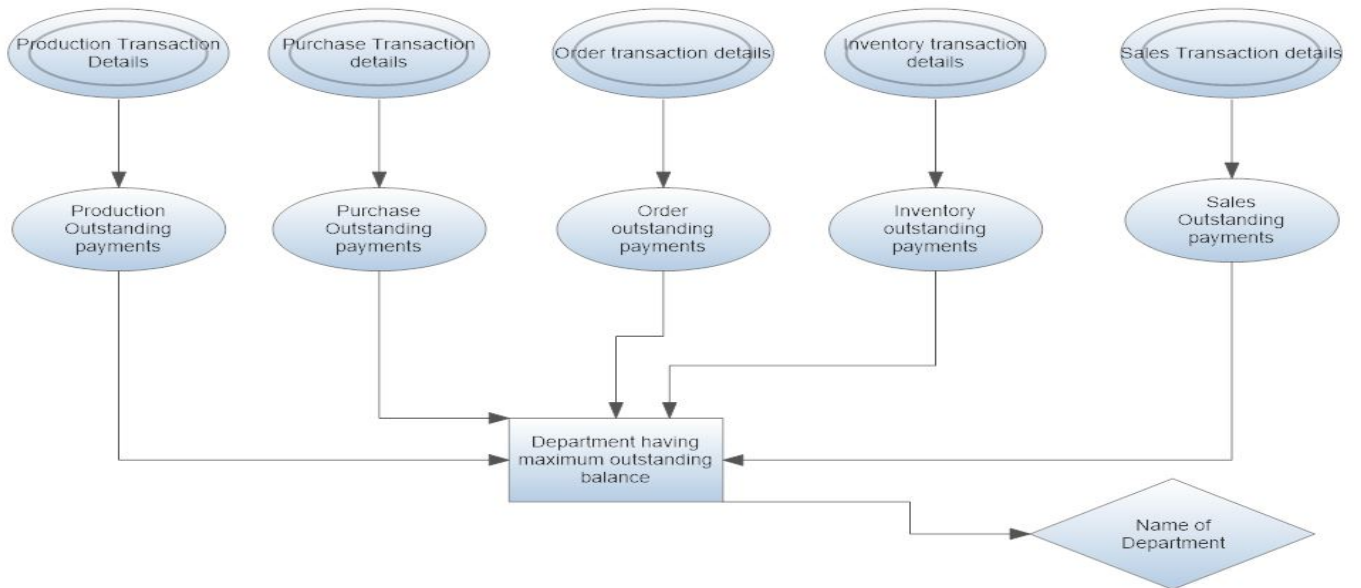
Fig. 2. Influence Diagram for Problem 2

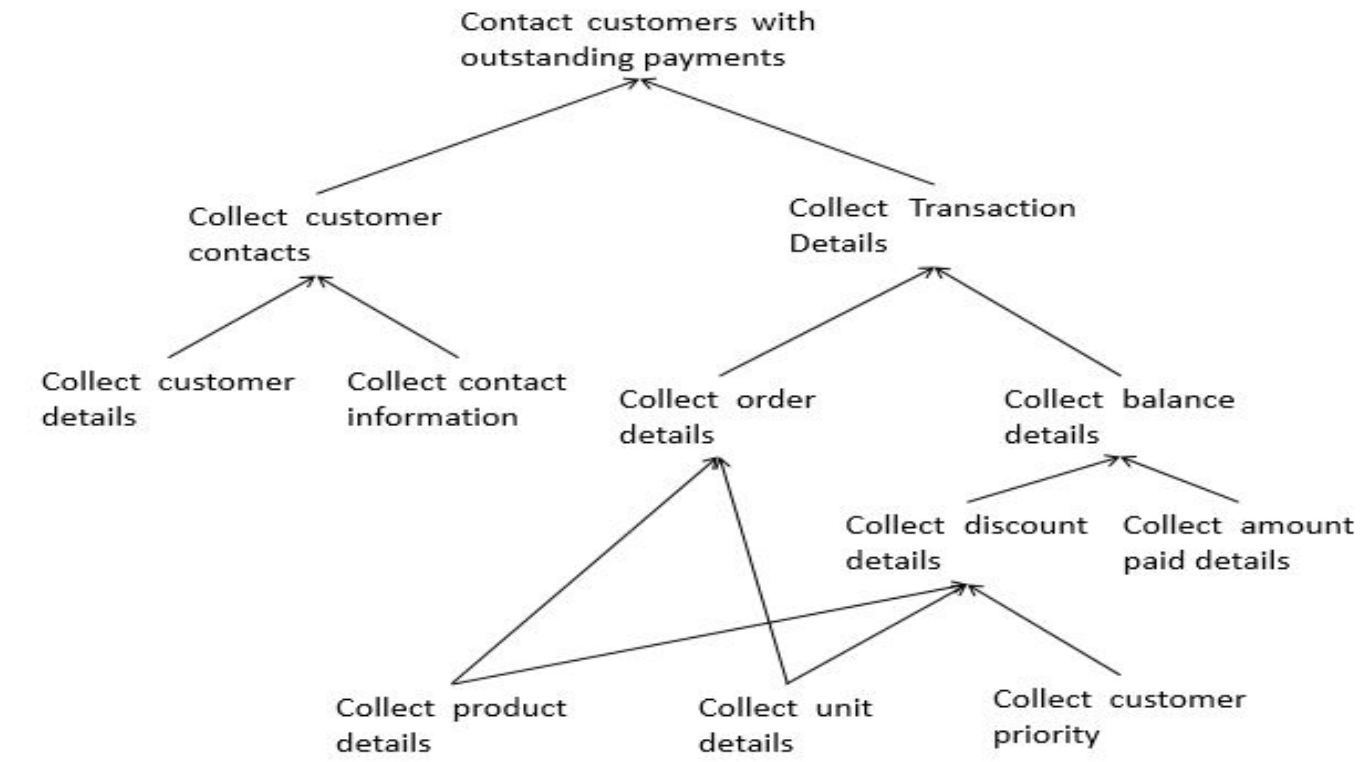Fig. 3. Influence Diagram for Problem 3

Fig. 4. End Means Hierarchy for Problem 1

(3) Based on the above a star schema was created

3.1 Problem 1

Then end means hierarchy for problem 1 has been shown in Figure 4.

The end objective of this decision is to Contact the customers. Now, the 2 ends objective that can eventually satisfy this requirement are-

(1) Getting the contact details
(2) Getting the transaction details

Both of these end objectives are then broken down into the means required, for e.g., for transaction details, we require the discounts active at that time, the order details, customer priority status of the customer for the company etc. (these are stated in the end-means hierarchy).

Similarly the means for the contact details are to extract data from two tables - one containing the basic information and the second containing the contact details of the customers. These two can be related by some customer Id or similar key.

The star schema for the above decision can be seen in Figure 5.

3.2 Problem 2

Then end means hierarchy for problem 2 has been shown in Figure 6.

This is again a similar end objective as decision 1 and thus can be broken down into 2 similar ends objective -
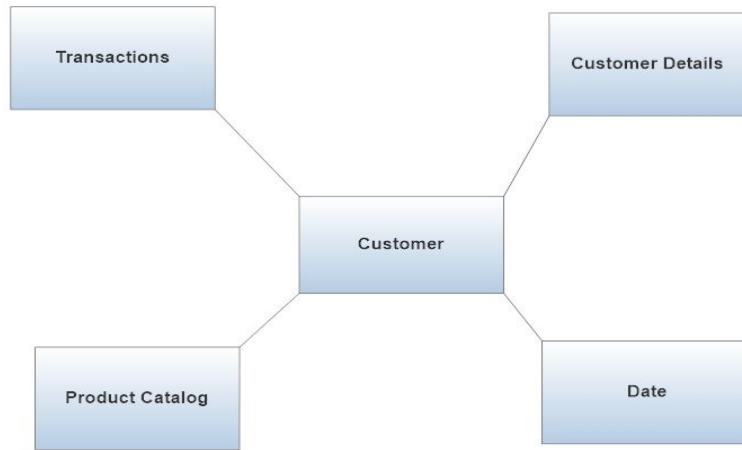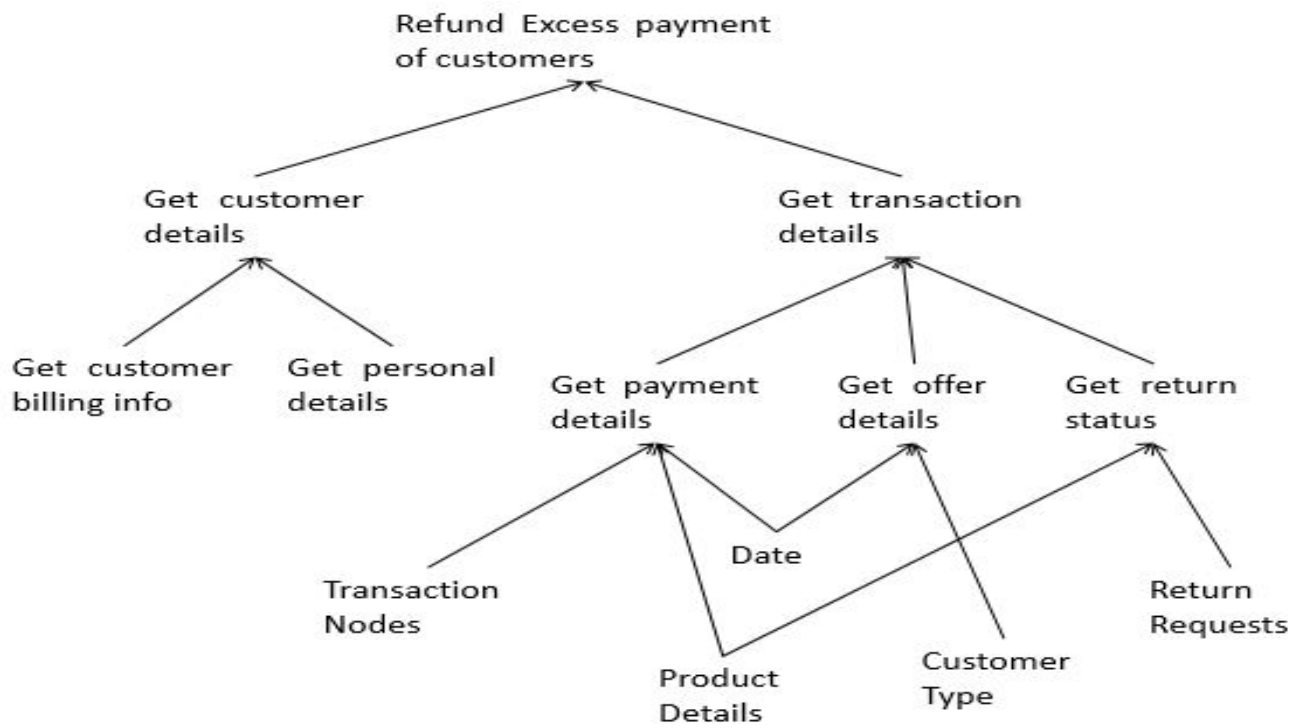
Fig. 5.   Star Schema for Problem 1

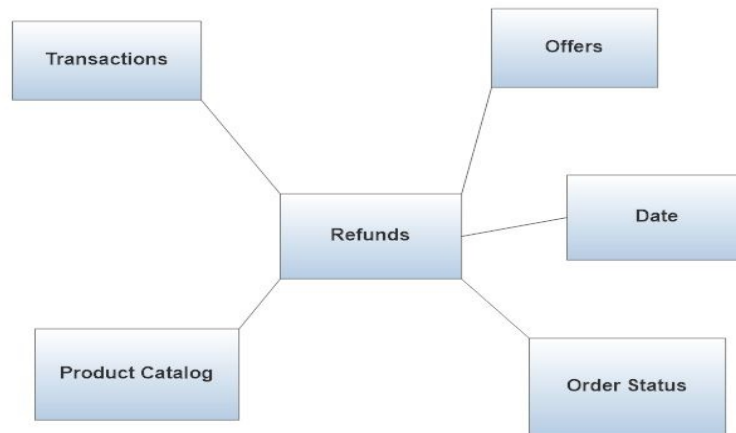Fig. 6.   End Means Hierarchy for Problem 2

Fig. 7. Star Schema for Problem 2

(1) getting the customer details
(2) Getting transaction details

These are the 3 ends objective that are further satisfied by various means objective. This requires some different means than the earlier problem. The transactions here will involve the returned orders, excess payments made by customers etc. These means are further broken down into the product catalog(required to initiate the proper amount of refund), the date of order (as refund should be done within some fixed number of days) etc. The ends means network was then converted to star schema with the required fact and dimensions.

The star schema for the same can be seen in Figure 7.

3.3 Problem 3

The ends means hierarchy for this problem can be seen in Figure 8.

The end objective here was clear - to come up with a ranked list of departments depicting the outstanding payments. This is atomic and does not need to be further broken down into further ends objectives. Thus, only the balance as per the departments are needed as means objective to fulfill the end objective. The same is depicted in the ends-means hierarchy.

The star schema for the problem thus can be seen here in Figure 9.

4. DIMENTION DESIGNING

We now come up with a dimension design for all the problems so as to finally come up with a data warehouse later on.

4.1 Problem 1

The dimensional modelling can be found in the Figure 10 here.

4.2 Problem 2

The dimensional modelling can be found in the Figure 11 here.

4.3 Problem 3

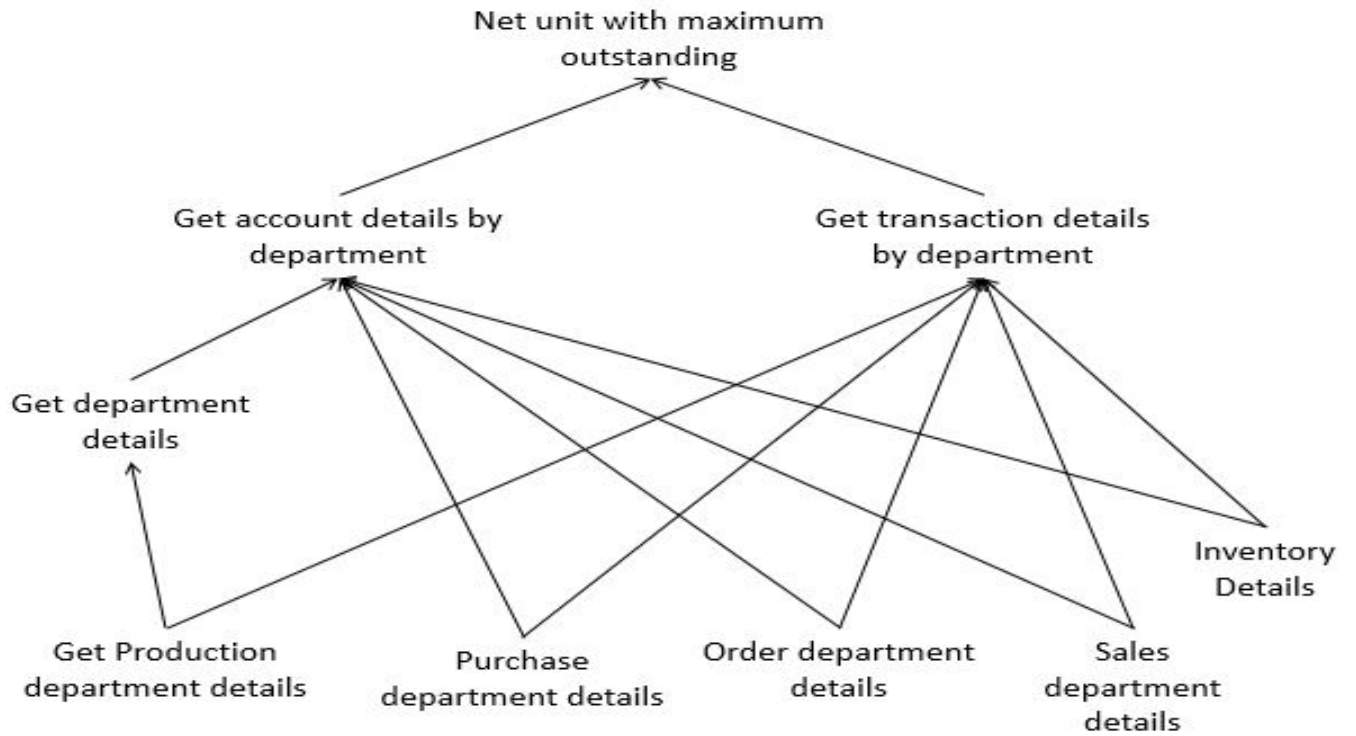The dimensional modelling can be found in the Figure 12 here.

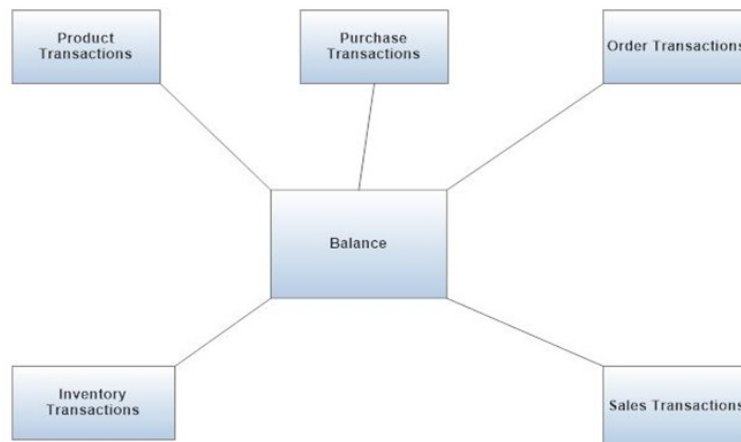Fig. 8.   End Means Hierarchy for Problem 3
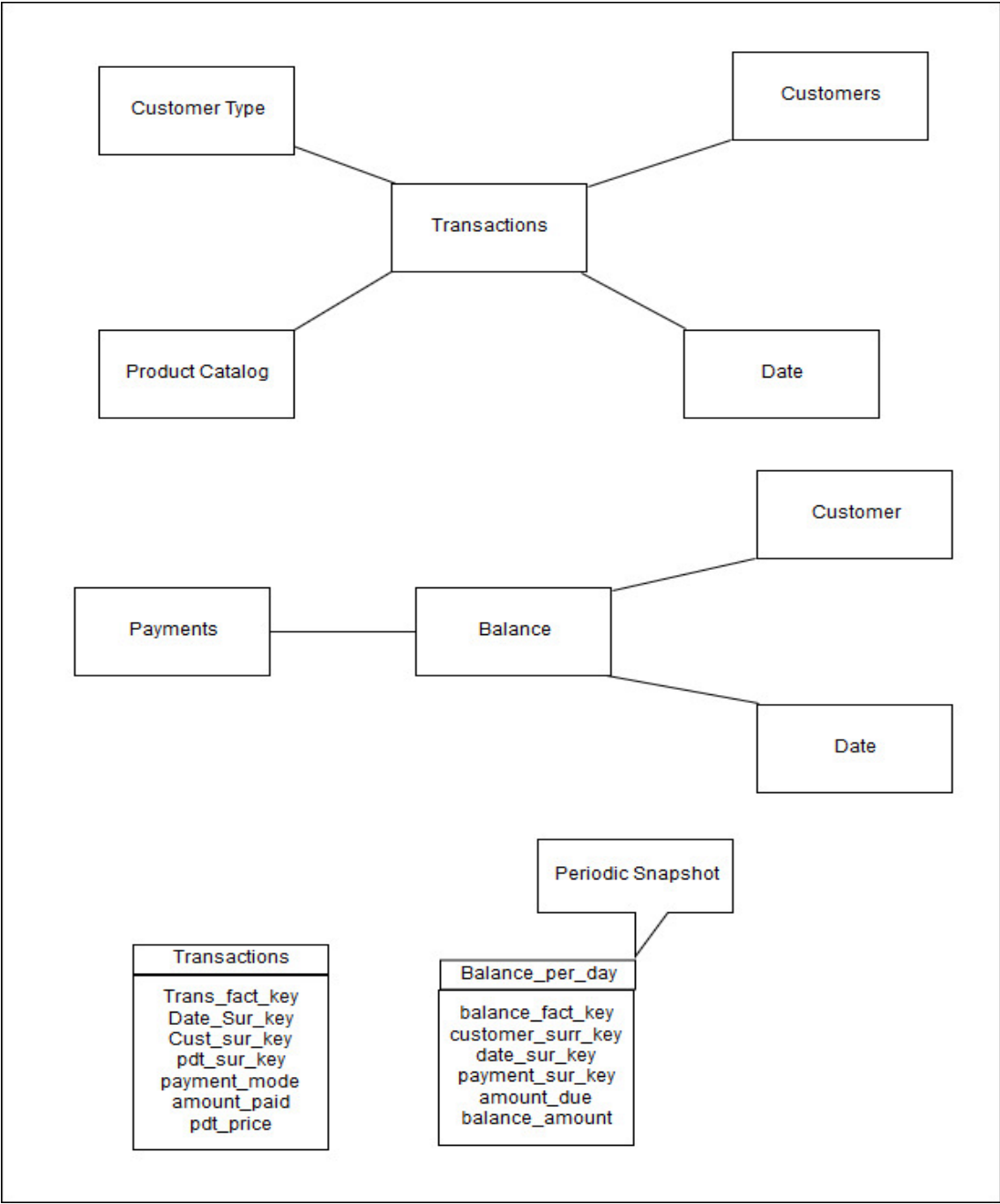

Fig. 9.   Star Schema for Problem 3

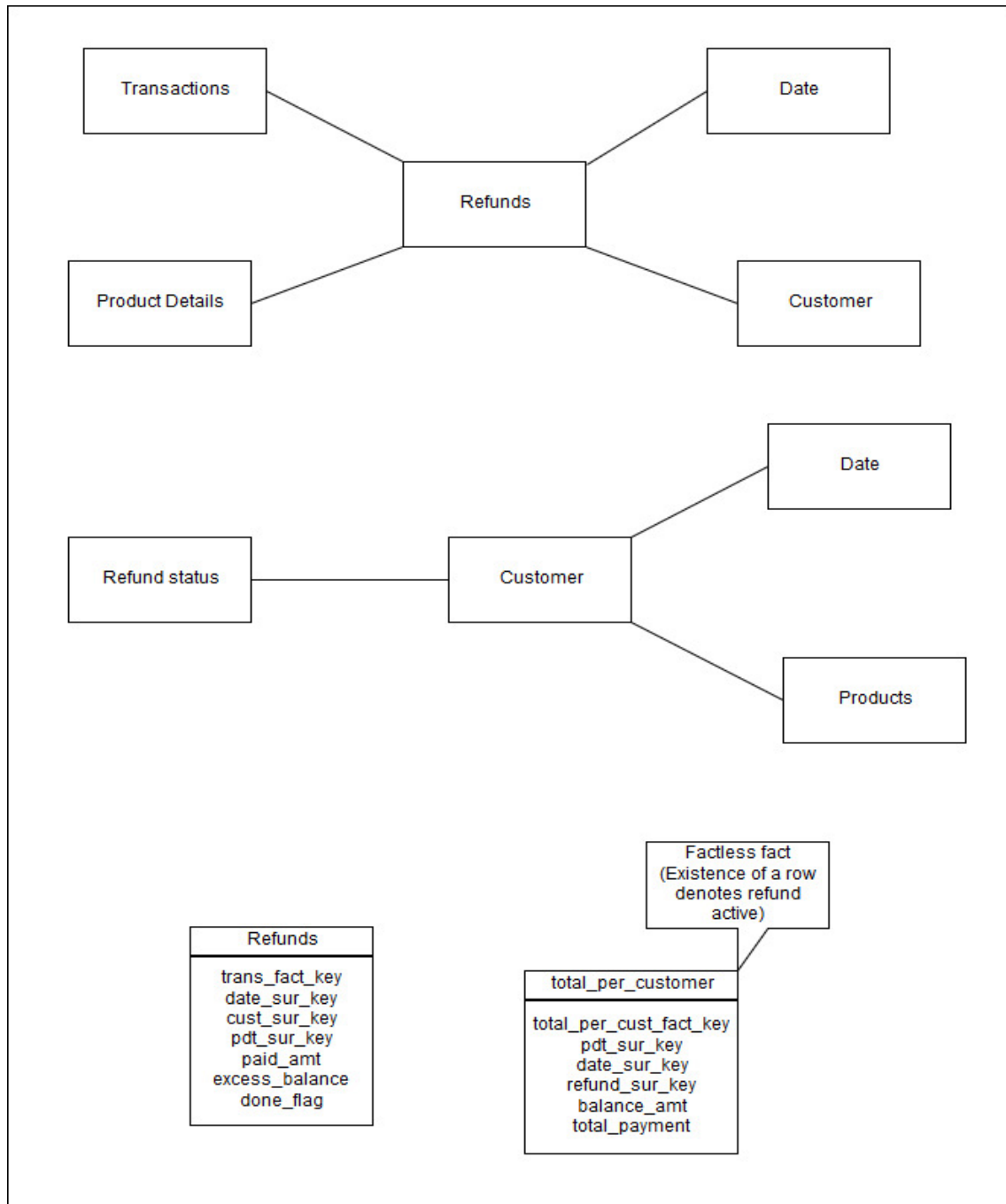Fig. 10.   Dimentional Modelling for Problem 1
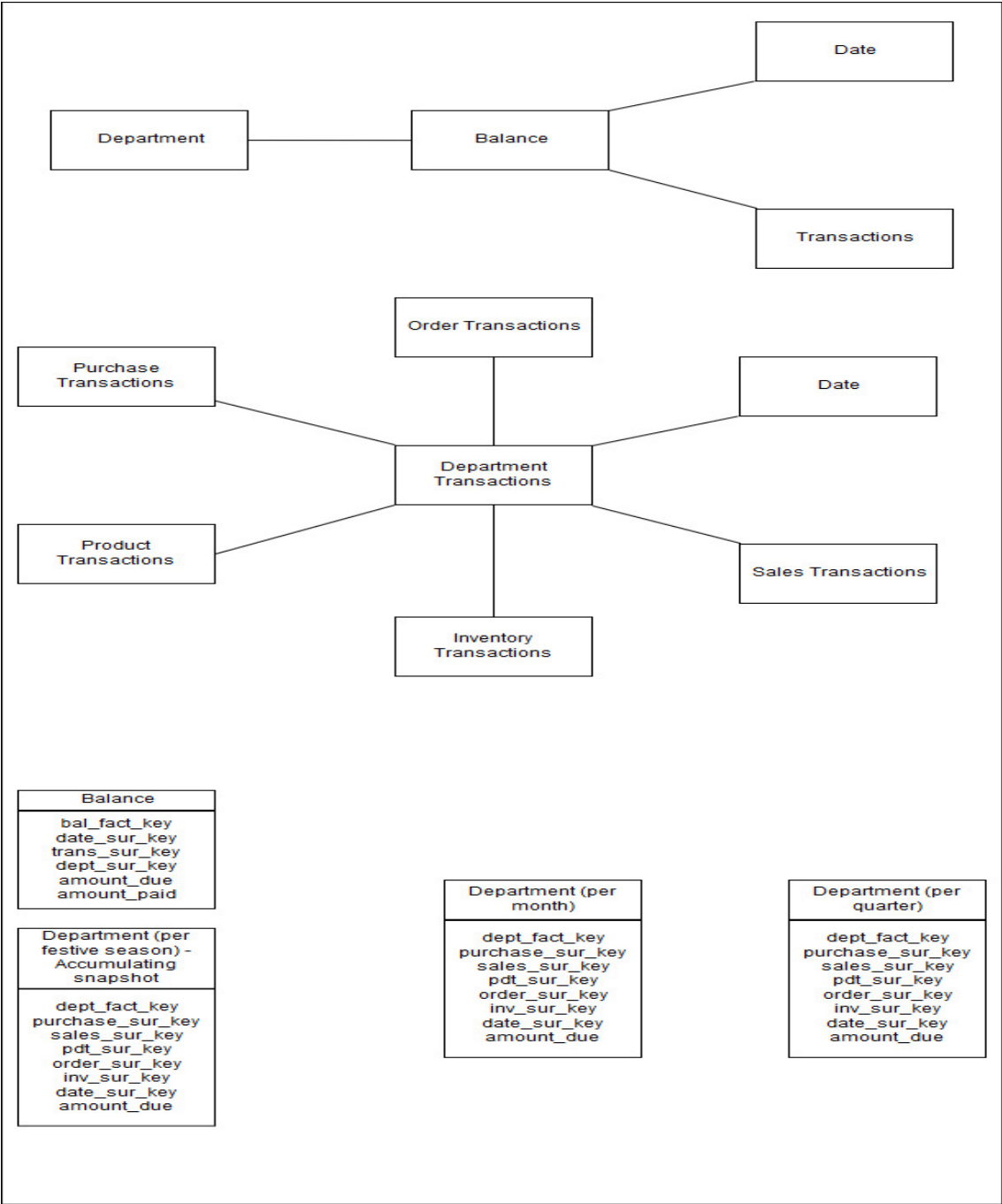
Fig. 11.   Dimentional Modelling for Problem 2

Fig. 12. Dimentional Modelling for Problem 3

5. DISCUSSION

Though the whole process is quiet difficult and as we had no end user with whom we can interact, we, with whatever came to our mind tried to make the best system anyways. The database related code for all this can be found out in the related repositories of this submission and can be run on MSSQLSERVER to see the results. Though, a dashboard could have been made but in the non-availability of real time data like in the case of our application this is really not useful and has thus been avoided.