Object Oriented System Analyze and Design of Revenue Information System using UML

Sany Ang
Department of Accounting
Petra Christian University, Surabaya, Indonesia
san_angkasa@yahoo.com

and

Prof. Dr. Chaiyong Brahmawong
Chief Operating Officer
College of Internet Distance Education.
Assumption University of Thailand
chaiyong@ksc.au.edu

Abstract- **This paper presents the analysis and design of Revenue Information System using object oriented approach with the notation of Unified Modeling Language (UML). Object oriented depicts the system as object, which is closer to the real world than the structural approach. The UML diagrams shown here include Activity Diagram, Use Case Diagram, Behavioral State Machines and Class Diagram.**

The object of study is the revenue cycle of a pharmaceutical products distributor. The current revenue system relies on the paper works using computers in which the staffs from each division record almost the same data (redundancy) to maintain its accounting data as well as marketing data, shipping data and inventory data. The proposed system will be a computerized accounting information system using database used by all related division in revenue cycle. With the new system customer enquiries can be handled in little time giving competitive advantage for the company.

Keywords- **Object-oriented, system analysis and design, UML**

1. INTRODUCTION

Accounting Information System process financial transactions and nonfinancial transactions that directly affect the financial transaction processing (Hall, 2001). One of the subsystem or ‘cycle’ in Accounting Information System is Revenue cycle. Revenue cycle is a critical cycle of a business, either the revenue generates cash or credit. The company whose sales is in credit, customer data and account receivable data becomes an important asset besides the inventory.

2. OBJECT ORIENTED SYSTEM ANALYZE AND DESIGN (OOSAD)

In information system development there are two approaches to analyze and design an information system, namely traditional approach and object oriented approach. Traditional or structured approach uses three techniques: logical data modeling, data flow modeling and entity behavior modeling. The object oriented approach uses UML. The object oriented approach is chosen for it’s characteristic that it reflects the system as objects which is more understandable.

Unified Modeling Language (UML) is a language which provides a comprehensive notation for communicating the requirements, architecture, implementation, deployment, and states of a system. The objective of UML is to provide a common vocabulary of object oriented terms and diagramming techniques that is rich enough to model any systems development project from analysis through implementation. There are fourteen diagramming techniques in UML 2.0 that can be used to represent the existing and proposed system, but this paper only discusses Activity Diagram, Use Case Diagram, Behavioral State Machines and Class Diagram.

According to Dennis (2005) Object Oriented system Analyze and design (OOSAD) focuses on three aspects namely object, analysis and design. Concepts in object oriented system include classes, objects, methods, encapsulation, inheritance, and dynamic binding. Shoemaker (2004) stated that the concept of object which contains both data and process represents the real world. Object can be a person, place, event, or thing about we want the capture information. With the concept of encapsulation, an object hides it’s data and allows only the object’s method to access it’s data. This makes the system more flexible and easier to maintain if an object’s data structure is changed.

Object oriented analysis (OOA) looks at the functional of the system resulted in conceptual model while the object oriented design (OOD) looks at how the system works covers both functional and nonfunctional. OOA resulted in functional model, structural model and behavioral models.

At the analysis phase, functional model describes business processes and how the system interacts with its environment. The structural or conceptual modeling presents the logical organization of data independent from how the data are stored, created, or manipulated so that analysts can focus on the business without being distracted by technical details. The behavioral model describes what internal logic of the processes is without specifying how the processes are to be implemented.

Later, at the design phase the structural model is updated with how the data will be stored in databases and files. In other words, object oriented system analysis and design uses the same diagramming techniques, the difference is that in design phase we add system environment details and refined problem domain information to increase the likelihood of successfully delivering a system.

Functional model often used diagramming techniques are Activity Diagrams and Uses Cases. Activity diagram and use cases are functional models that can be used to document the current system or the new system, which do not suggest how the business domain’s activities are conducted.

3. CASE STUDY

ABC is a company distributes medical supplies and equipments to various government hospitals, private hospitals and medical clinics. The sales are in credit and cash, but most of the sales are in credit therefore it is important to record and manage the data of the customer receivable. The revenue cycle includes four main activities: sales order, shipment, billing and cash receipt.

The current system works mostly with the paper documents. Each division maintains their data in computer (stand alone computer) in which the staffs from each division record almost the same (redundant) data to maintain its accounting data as well as marketing data, shipping data and inventory data. A change in product price list must be informed to all related officers to reflect the same price in Sales Officer data, and in A/R Officer data.

The company has its credit sales policy which the sales may not exceed a customer’s credit limit and the extension of credit must be approved by the credit officer. Since each officer maintains its own set of data, it takes time to process customer's order and answer customers' enquiry when a Sales Officer needs to check following information:

- the product availability to Warehouse Officer
- the customer balance to Account Receivable (A/R) Officer
- the credit balance to Credit Officer
- the shipping data to Shipping Officer

If the product quantity is less than the order, Sales Officer will inform the Customer whether they agree with partial shipment or order only the available quantity. If the order exceeds certain amount, it needs authorization from Sales and Marketing Manager. From the accounting department, the A/R Officer records the sales and the collection of the receivable and updates the balance of each Customer card.

4. UML REPRESENTATION OF OOSAD

The Activity Diagram of the existing system is shown in Figure 1. The diagram shows the primary activities in revenue system which are: pay bill, prepare Sales Order, ship products, bill Customer and record cash collected. Each activity consists of several actions, for example the activity of 'record cash collected' consists of the actions of A/R Officer to receive remittance advice from Cashier, make Cash Receipt Voucher, record in Cash Receipt Journal, post to Customer Card, deduct the Customer balance and post to General Ledger. Since Customer is not an internal actor in our revenue system, the 'pay bill' activity will not be covered in our design. The 'pay bill' activity is shown here to represent the flow of the activities in revenue system.

The functionality of the revenue information system is presented by Use Case Diagram as shown in Figure 2. Use case is represented by an oval and labeled with verb-noun phrase. Each use case is major process of the system that will benefit an actor(s). The major use cases of the system are:

- Allow Sales Officer to maintain customer data, with the exception for new customer needs the Credit Officer to approve/maintain the credit limit.
- Allow the Sales Officer to prepare Sales Order, in case if the Sales Order amount exceeds 40,000 Baht then the Sales Order needs the approval of Sales and Marketing Manager.
- Allow Sales Officer, Credit Officer, Sales and Marketing Manager, Shipping Officer, and A/R Officer to print report
- Allow Shipping Officer to record shipment
- Allow A/R Officer to prepare invoice, record cash received and post the transaction to General Journal.

The actors who use the system are Sales Officer, Credit Officer, Sales and Marketing Officer, Shipping Officer and A/R Officer. Using inheritance concept, all of them inherit the attributes and methods of Employee object and separated by their role in the revenue system.

Behavioral State Machine diagram depicts the dynamic aspect of a single object, by showing the various states that an object may be in and the transitions between those states. For this case study the object that has several states is Sales Order. The states of Sales Order without indicating the sequence of the states are: Received, Sufficient credit/Non sufficient credit, Need Approval, Approved, Partially Shipped, Fully Shipped, Billed and Paid. The system will evaluate the condition of the Sales Order and indicate the state of the Sales Order. From this Behavioral State Machine we know that the company allows partial shipment, if certain products are not in stock. The Behavioral

State Machine of the proposed system is shown in Figure 3.

The structural model of the proposed revenue information system is represented by Class Diagram as shown in Figure 4. Class Sales_Order has attributes SO_No, date, Customer_No, Sales_Emp_No, Q_Order, Q_Shipped, Price, and a calculated attribute "/SO_Amt". The operation of method of Class Sales_Order are calc SO_Amt (to calculate SO amount), check SO_Amt (to check the SO amount), set SO_Amt (to set the SO amount), get SO_Info (to get SO information), set Approve (to approve SO), and update Q_Shipped (to updated the quantity of SO shipped). The four major activities in revenue system are represented by transaction class Sales_Order, Shipment, Sales_Invoice and Cash_Receipt. The actors are represented by class Customer and Employee. The class Employee consists of subclasses Sales Officer, Shipping Officer, Credit Officer, A/R Officer and Sales and Marketing Manager.

The accounting data represented by class G/L_Transfer which contains data of sales (from Sales_Invoice class) and collection of receivable (from class Cash_Receipt). The data in class G/L_Transfer then will be posted to G/L_Master.

There are two types of relationships in the revenue system, first association relationship between: Sales_Order and Inventory, Sales_Order and Sales Officer, Sales_Order and Shipment, etc. Association relationship has multiplicity which indicates how many objects of a class can be associated to other objects from different classes. For example the relationship between Sales_Order and Customer, on the Sales_Order side it is shown "0..*" means a Customer may initiate many Sales Orders or none (this enables system to capture potential Customer data also). On the Customer side, it shows "1" means a Sales Order is made only to one Customer. Second, generalization/inheritance relationship among superclass Employee and the subclasses Sales Officer, Credit Officer, Shipping Officer, A/R Officer, and Sales and Marketing Officer.

A relational database will be used to store data of the revenue information system, which will be shared by all division. The RDBMS supports referential integrity, providing input check of the uniqueness of primary key. The objects in our design will be converted so they can be stored in a table.

5. CONCLUSION

The computerized revenue information system is designed to minimize the data inputted by the employee therefore minimizing the human error, reduce order processing time. With the use of database it will reduce redundancy therefore increase the integrity/quality of the data/information and generate report to support decision making in little time, giving better customer service with the updated information.

To reduce the possibility of human error, the data should be entered at the point of transaction. For example the sales order data is entered by the Sales Officer when the orders are received from Customer. The application should provide input check to validate the data and minimize keystroke, for example the Officer selects data from list instead of type the data in text box.

REFERENCES

1. Dennis, A., Wixom, Barbara H. and Tegarden, David (2005). Systems Analysis and Design with UML Version 2.0: An Object-Oriented Approach, 2nd edition. New Jersey: John Wiley & Sons, Inc.
2. Hall, James A. (2001). Accounting Information Systems, 3rd edition. Ohio: Thomson South-Western.
3. Jones, Frederick L., and Rama, Dasaratha V. (2003). Accounting Information Systems: A Business Process Approach. Ohio: Thomson South-Western.
4. Romney, Marshal B. and Steinbart, Paul John (2000). Accounting Information Systems, 8th edition. New Jersey: Prentice-Hall, Inc.
5. Shoemaker, Martin L. (2005). UML Applied: A .Net Perspective. California: Apress Publishing

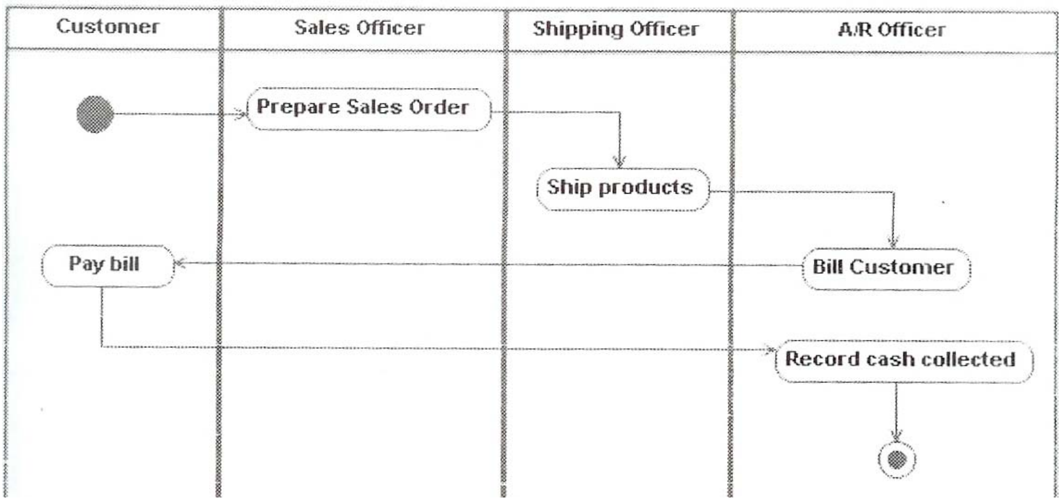

Figure 1. Activity Diagram of Existing System

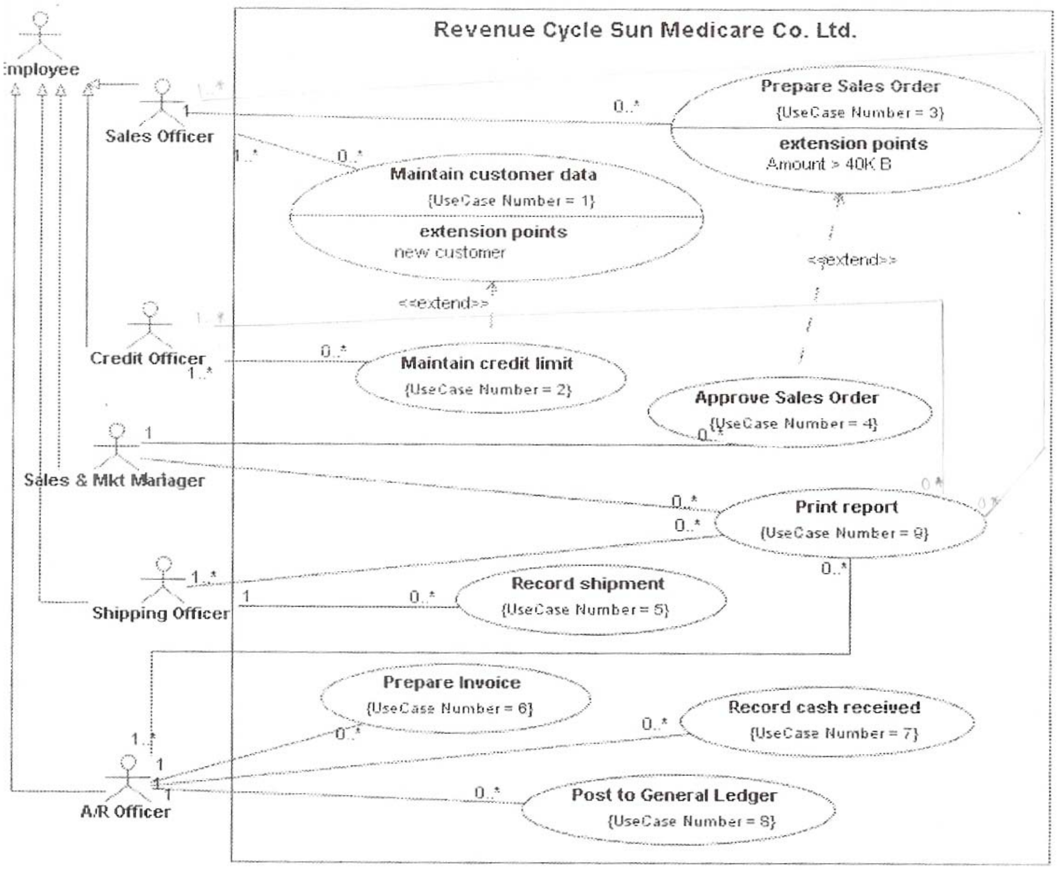

Figure 2. Use Case Diagram of the Proposed System

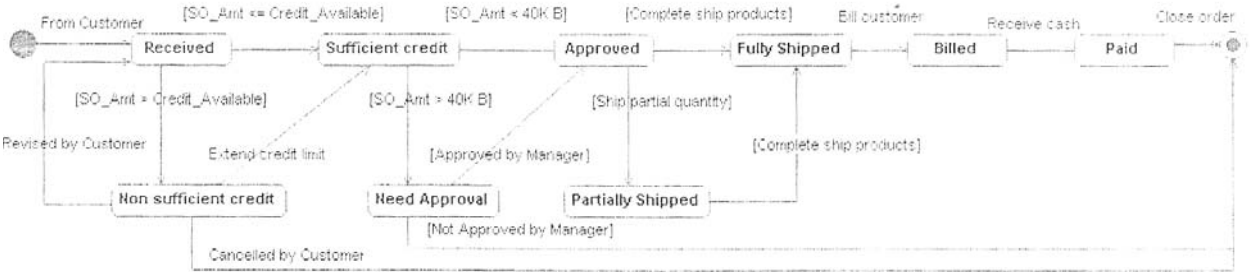

Figure 3. Behavioral State Machine Diagram

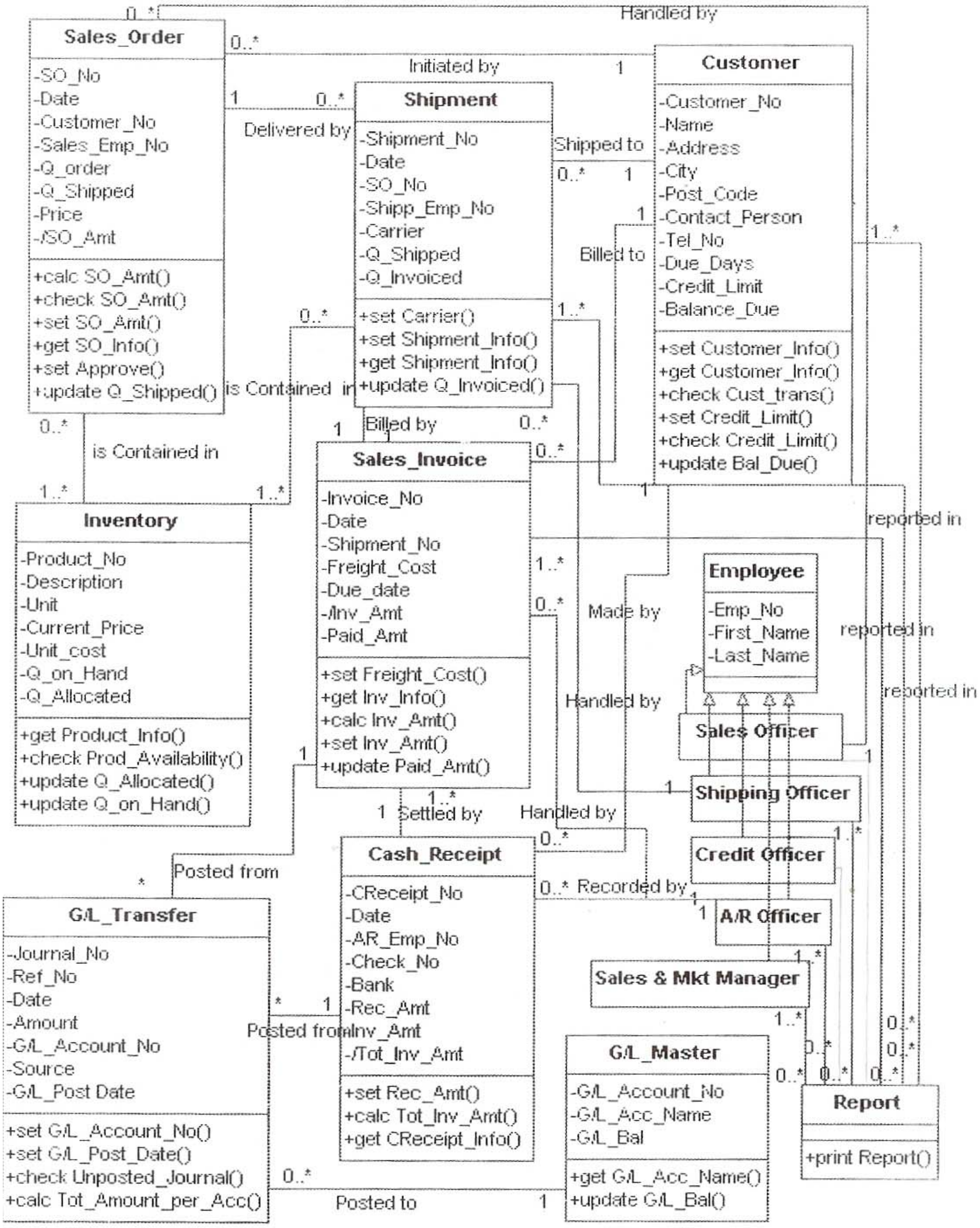

Figure 4. Class Diagram