Information Retrieval System of Hospital using UML and Data Cube

Thesis submitted in partial fulfillment of the requirement for

The award of the degree of

Masters of Science

In

Mathematics and Computing

Submitted by

Rajni

Roll No.- 30703016

Under

The guidance of

Dr. Rajesh Kumar

School of Mathematics and Computer Applications

Thapar University

Patiala-147004 (PUNJAB)

INDIA

JULY 2009

CERTIFICATE

I hereby certify that the work which is being presented in the thesis entitled "Information Retrieval System of Hospital using UML and Data Cube" in partial fulfillment of the requirements for the award of degree of Master of Science, School of Mathematics and Computer Applications, Thapar University, Patiala is an authentic record of my own work carried out under the supervision of Dr. Rajesh Kumar.

The matter presented in this thesis has not been submitted for the award of any other degree of this or any other university.

Rajni
(Rajni)

This is to certify that the above statement made by the candidate is correct and true to the best of my knowledge.

09/07/09

(Dr. Rajesh Kumar)
Supervisor
SMCA, Thapar University
Patiala

Countersigned by:

9.7.09

Dr. S.S. Bhatia
(Professor & Head)
School of Mathematics & Computer Applications
Thapar University, Patiala.

21/7/09

Dr. R.K. Sharma
Dean of Academic Affairs
Thapar University
Patiala.

ACKNOWLEDGEMENT

The key elements concentration, dedication, hard work and application are not the only essential factors for achieving the desired goals but also guidance, assistance and co-operation of people is necessary.

I would like to express my deep and sincere gratitude to my supervisor Dr. Rajesh Kumar, Associate Professor, School of Mathematics and Computer applications (SMCA). His wide knowledge and logical way of thinking have been of great value for me. His understanding and personal guidance have provided a good basis for the present thesis. Especially the strict and extensive comments and many discussions and the interactions with Dr. Rajesh Kumar had a direct impact on the final form and quality of this thesis. I could never imagine to have better mentor than him.

I owe my sincere thanks to (Dr.) S.S. Bhatia, Professor & Head of School of Mathematics and Computer Applications (SMCA), Thapar University, Patiala for providing facilities. I wish to express my sincere thanks to Mr. Avadhesh Kumar, Assistant Professor, Amity University, Noida, for the possible help extended especially, whenever needed. I also thank to Ms. Neelam Gulati, Lecturer, Mr. Sanjeev Manchanda, Lecturer of School of Mathematics and Computer Applications, Mr. Harcharan Jit Singh, Systems Analyst Cum Programmer Computer Center, Thapar University, Patiala, for their co-operation and support throughout the project work.

My thesis work would have incomplete without thanking my friends, who were always there in the hour of need. I also extend my thanks to Vineet Khera, student of M.E. (Software Engineering) for his help.

I was very fortunate to have unconditional support from my family. I thank my parents, who gave me the courage to get my education, supported me in all achievements throughout my life. Without their encouragement and suggestion, this work would indeed have been very difficult for me to tackle. Last but not least, thanks to God for giving me inner peace and strength. May your name be exalted, honored and glorified.

Rajni

Rajni

(30703016)

CONTENTS

ABSTRACT

The technique of making a model of our ideas or the world is a use of Abstraction. For example, a map is a model of the world — it is not the world in miniature. It's a conventional abstraction that takes a bit of training or practice to recognize how it tracks reality, but we can use this abstraction easily. Similarly, each UML diagram depicts every relationship between model and reality. And the UML abstractions were developed as conventions to be learned and used easily. Data Cube used for under different abstraction levels in a dimension hierarchy. This thesis describes how information can be retrieve through UML and Data Cube. It also contains all introductory topics to required to understand the content of this work. Also this explains how UML diagrams can be made for specific problem. UML activity diagram, sequence diagram and class diagram have been designed to retrieve the information efficiently. It also proposes the framework of the hospital system. This thesis contains how can we use Data Cube for fast retrieval of information and several queries can be performed. This thesis advocates a wider use of Data Cube for solving problem.

Chapter 1

Introduction

Modeling is an activity that has been carried out over the years in software development. When writing applications by using the simplest languages to the most powerful and complex languages, we need to model. Modeling can be as straight forward as drawing a flowchart listing the steps carried out by an application. Defining a model makes it easier to break up a complex application or a huge system into simple, discrete pieces that can be individually studied. We can focus more easily on the smaller parts of a system and then understand the "big picture". Hence, the reason behind modeling can be summed up in two words: Readability and Reusability. We will concern ourselves with modeling object-oriented systems and applications using the Unified Modeling Language. Information retrieval system of Hospital is principally based on the Object Oriented Techniques. In this we use Unified Modeling Language and Data Cube for making Hospital System. In different chapters, various activities have been discussed by Unified Modeling Language and Data Cube.

1.1 Unified Modeling Language (UML)

Conceptual models are formal descriptions of application domains that are used in early stages of system development to support requirements analysis. The Unified Modeling Language was formed by integrating several diagramming techniques for the purpose of software specification, design, construction and maintenance.

The Unified Modeling Language (UML) has quickly become the de-facto standard for building Object-Oriented software [1]. UML is a graphical notation for expressing object-oriented designs. It is called a modeling language and not a design notation. UML allows representing various aspects of the system, not just design that has to be implemented. Due to the ability of UML to create different models, it becomes an aid for

understanding the system, designing the system, as well as a notation for representing design.

The design activity begins when the requirements document for the software to be developed is available and the architecture has been designed.

Object-Oriented (OO) system offers us many advantages. In Object Oriented Design (OOD), we are identifying the classes and their relationships that should exist in the software. OOD models the solution domain. It is designed in late of analysis phase an elaborated in early phase.

1.2 System Development Life Cycle: SDLC

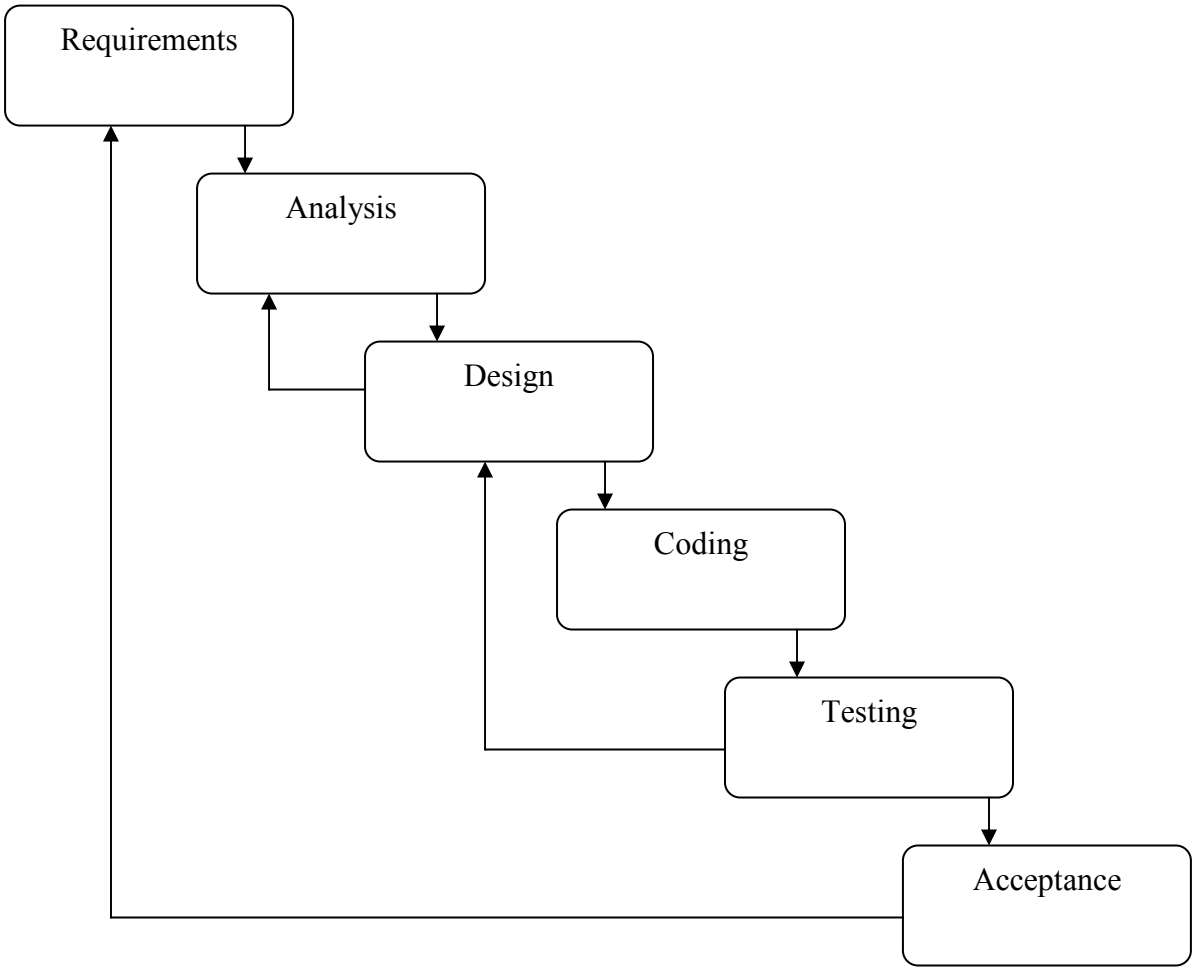

Figure 1

To make any system, we have to follow the steps of any model. We have taken the System Development Life Cycle (SDLC) Model. Firstly we have to collect the requirements from various resources. Figure 1 shows the stages of the system development life cycle and we follow these steps as shown in Figure 1. In this stage, we should give emphasis on identifying what is needed from the system, not how the system will achieve its goals. After that stage we will switch to analysis stage. In this stage, we have to understand the problem domain. We should do analysis to understand the problem, which has to solve. After completing the analysis, we will move to design stage. The purpose of the design phase is to plan a solution of the problem specified by the requirement document. After making the design, coding will be done. The goal of the coding phase is to translate the design of the system into code in a given programming language. After the coding, we should do testing of the document. Testing phase is an important phase. The goal of testing is to uncover requirement, design and coding errors in the programs. If there is an error in the testing phase, it will go back to reconsider design phase. If there is no error in design phase then we will switch to analysis phase. If there is no error in the testing stage then system will be ready to use and it comes in acceptance phase. If system is not accepted it means there are some problem in the system and it is not according to user's needs. Then we will switch back to requirement phase and steps should be repeated as mentioned above.

1.3 UML Diagrams

It would be advantageous to use the same modeling method throughout the development process of an information system. We have use UML for our system over conceptual modeling. This would require assigning well-defined, real-world meaning to UML constructs. UML modeling constructs one aspect of the system. There are three classifications of UML diagrams.

Behavior Diagram: This type of diagrams depicts behavioral features of a system or business process. This includes activity, state machine, and use case diagrams as well as the four interaction diagrams.

Interaction Diagrams: A subset of behavior diagrams, which emphasize object interactions. This includes communication, interaction overview, sequence, and timing diagrams.

Structure Diagrams: A type of diagram that depicts the elements of a specification that is irrespective of time. This includes class, composite structure, component, deployment, object, and package diagrams.

1.3.1 Use Case Diagram

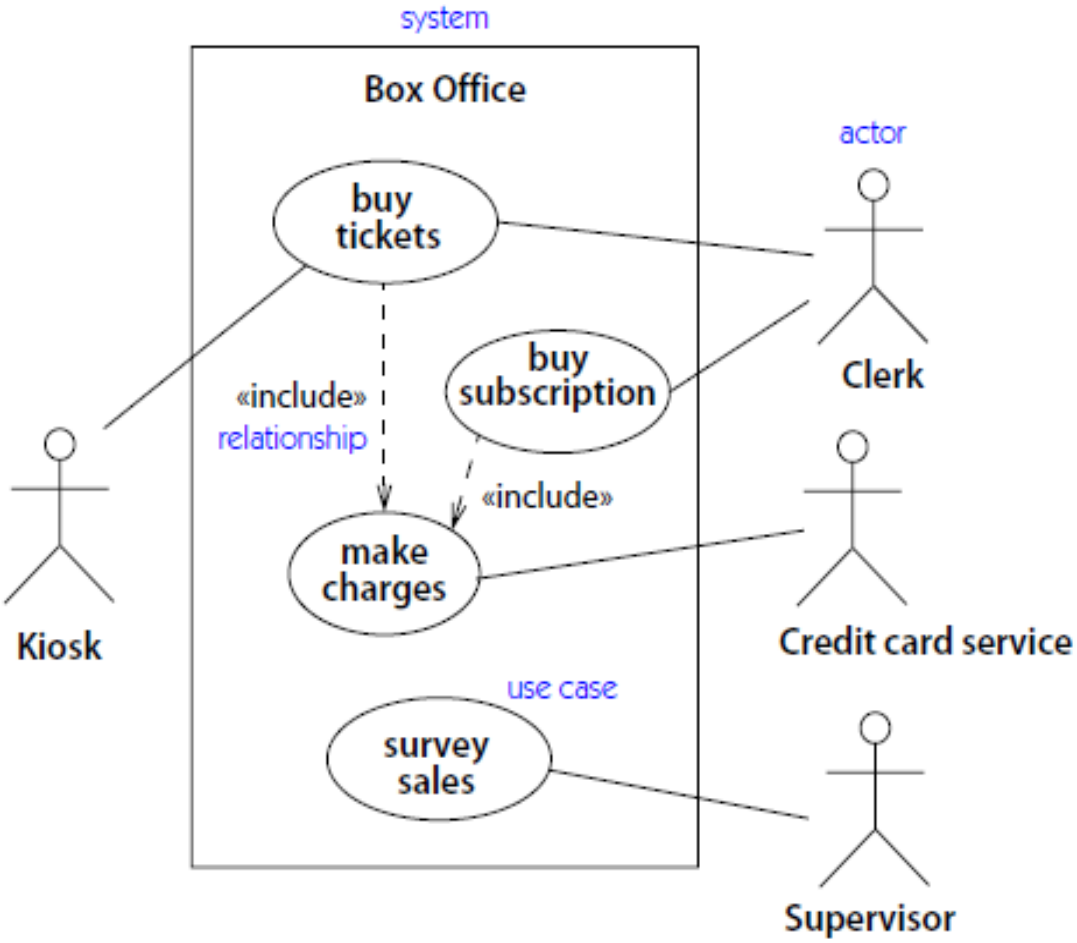

Figure 2

Use case shows the functionality of the system from the prospective of each user of the system. A use case is a coherent unit of functionality expressed as a transaction among actors and the system. Figure 2 shows a set of actors, use cases and their relationships. Buy tickets, buy subscription, make charges and survey sales are use cases. Clerk, kiosk,

credit card services and supervisor are actors. Here we have taken the example of ticket reservation. If we have to combine two use cases then we use include and merge in single use case.

1.3.2 **Sequence Diagram**

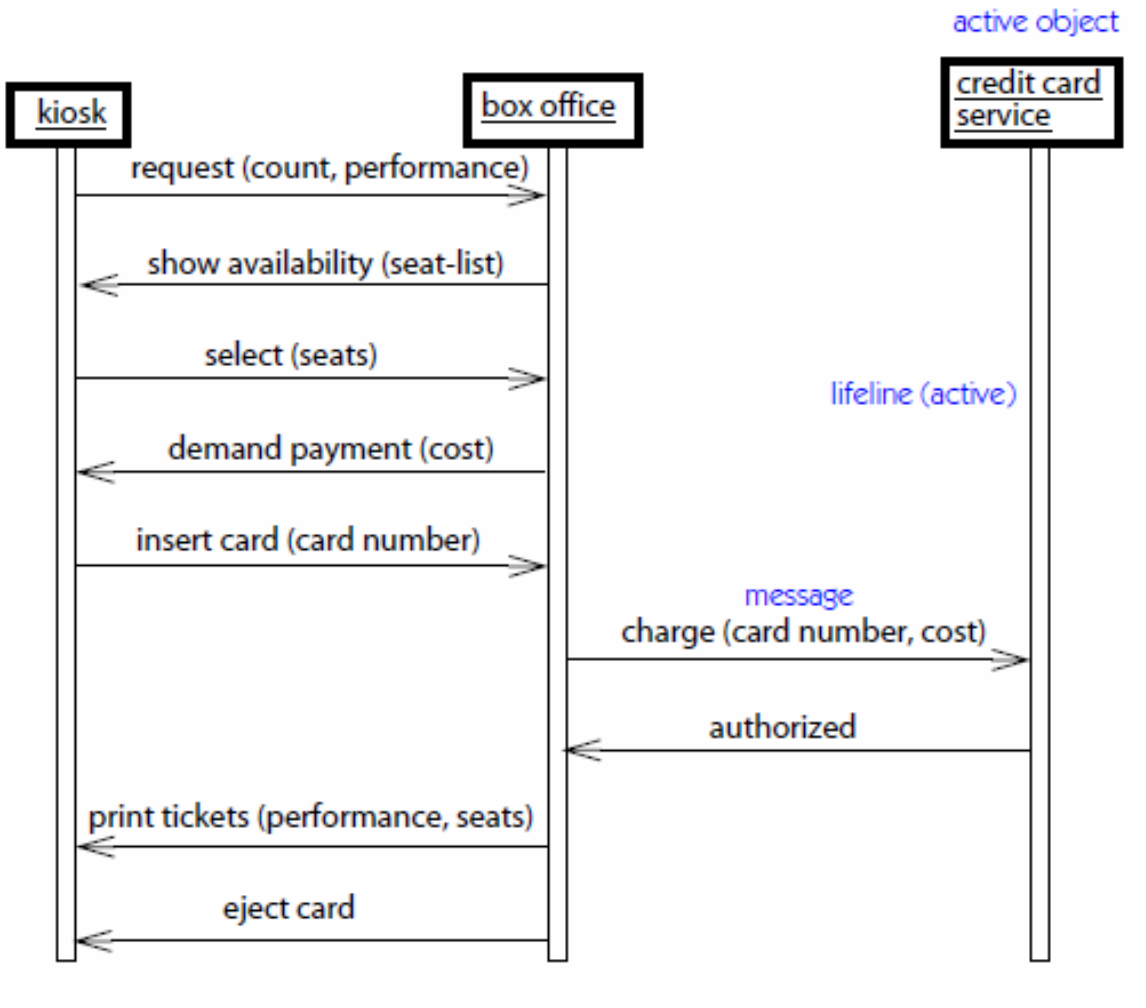

Figure 3

A sequence diagram is a graphical view of a scenario that show object. A sequence diagram is used to show the behavior sequence of a use case. Above diagram shows a sequence diagram for the buy tickets use case. A sequence diagram shows a set of messages arranged in time sequence. In the sequence diagram, we have shown the active object those are active in processing and lifeline shows object instance that participate in

the sequence being modeled. UML sequence diagrams are useful design tools because they provide a dynamic view of the system behavior, which can be difficult to extract from static diagrams or specifications. Figure 3 shows a sequence diagram of ticket reservation.

1.3.3 Class Diagram

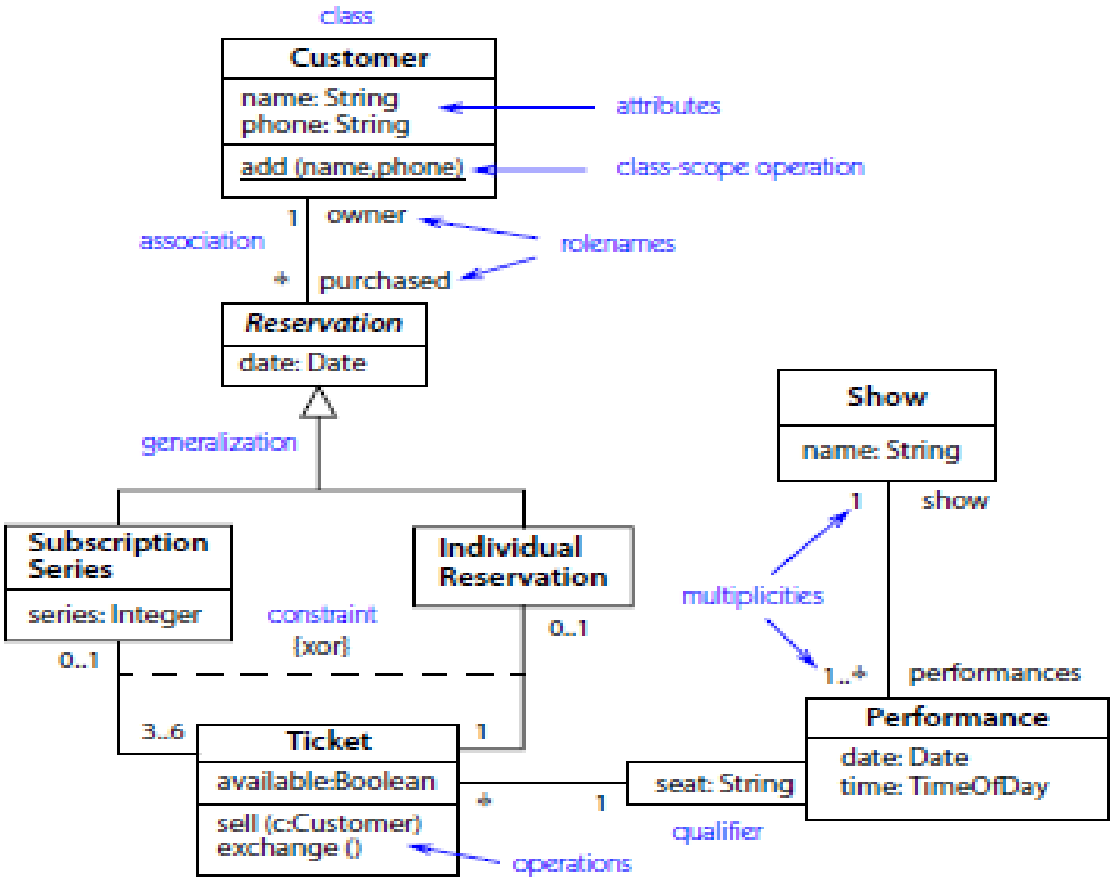

Figure 4

A class is represented as a three-part box.

- Class Name
- Attributes
- Operations

A class diagram shows the existence of classes and their relationships in the logical view of the system. A class diagram shows the graphical representation of the static view. Figure 4 shows a class diagram of the ticket reservation. In above class diagram Customer, Reservation, Ticket and Performance are class. Class diagrams are used in nearly all Object Oriented software designs. Use them to describe the Classes of the system and their relationships to each other. The association relationship is the most common relationship in a class diagram. The association shows the relationship between instances of classes. Another common relationship in class diagrams is a generalization. A generalization is used when two classes are similar, but have some differences.

1.3.4 Advantages of UML

UML takes some time to get used to, it is an intuitive way to design a piece of functionality. The visual modeling of systems provides the following reasons.

- Users from different backgrounds can understand how the system works and it shows how different pieces of the system interact with each other. It illustrates the system from different viewpoints, so that a complete design can be created.
- Software system are professionally designed and documented before they are coded so that all stakeholders know exactly what they are getting in advance.
- Since system design comes first, UML enables re-usable code to be easily defined and coded with the highest efficiency, thus reducing software development costs.
- UML enables logic 'holes' to be spotted in design drawing so that software will behave as expected.
- The overall design describe in UML will dictate the way the software is developed, so that the right decisions are made early on in process. Again, this reduces software development costs by eliminating re-work in all areas of the life cycle.
- UML provides a vehicle of communication with both internal and external stakeholders as it documents the system much more efficiently.
- UML diagrams assist in providing efficient training to new members of the development team member.
- UML enable ease of the maintenance by providing more effective visual representations of the system. Consequently maintenance costs are reduced.
- UML provides an enterprise level view of the system and, as a result, more memory and processor efficient systems can be designed.
- UML easily conveys and communicates system behaviors to non-technical staff.
- UML provides a high-level of abstraction, allowing architects and developers to see the "big picture".
- UML masks details so that engineers can identify opportunities for additional optimization and reuse.

1.4 Data Cube

Data Cube computation is an essential task in data warehouse implementation. The precomputation of all or part of a Data Cube can greatly reduce the response time and enhance the performance of on-line analytical processing.

In many database and data warehouse applications, users tend to be drowned in data and even in knowledge discovered from data, use more dimensions (variables) than necessary to explain a set of phenomena and check predefined hypotheses, and analyze a generally heterogeneous population of observations/individuals [11]. In order to reduce their memory overload, we are using Data Cube representation technique. Many topics have attracted researchers in the area of data warehousing: data warehouse design and multidimensional modeling, efficient cube computation, query optimization, discovery-driven exploration of cubes, cube reorganization. Multidimensional structure is called the Data Cube. In short we have described it as a data abstraction that allows one to view aggregated data from a number of perspectives [21]. Conceptually, the cube consists of a core or base cuboid, surrounded by a collection of sub-cubes/cuboids that represent the aggregation of the base cuboid along one or more dimensions [7]. We refer to the dimension to be aggregated as the measure attribute, while the remaining dimensions are known as the feature attributes [14,19]. When we try to extract information from a stack of data, we need tools to help us find what is relevant and what is important and to explore different scenarios. A report, whether printed on paper or viewed on-screen, is at best a two-dimensional representation of data, a table using columns and rows. That is sufficient when we have only two factors to consider, but in the real world we need more powerful tools. Data Cubes are multidimensional extensions of 2-D tables, just as in geometry a cube is a three-dimensional extension of a square. But Data Cubes are not restricted to just three dimensions.

1.4.1 Benefits of Data Cube representation

Using a cube to analyze data offers following advantages.

- View more data than with traditional two-dimensional tables.
- Drill down in two directions.
- Visualize data.
- Data Cube provides reliable vision for robotics, medical imaging, surveillance, inspection, teleconferencing, animation etc.
- Data Cube provides a new perspective for analyzing information in a data warehouse.
- A number of operations may be applied to Data Cubes.

    i. Roll-up
    ii. Drill-down
    iii. Slice and dice
    iv. Pivot

- Data Cube use to represent N-dimensional using 2-D relations [21].
- Data Cube computes the aggregate information within the database.
- Data Cube creates language extensions to SQL that will support the require operations.
- Data Cube allow us to look at workforce information in an easy way.
- Data Cube allow us to look at complex data in a simple format.
- Data Cube allow us to analyze specific workforce data.

1.5 UML and Data Cube

The Unified Modeling Language is a visual language designed for specifying, constructing, and documenting the artifacts of the systems. We can use UML for all processes, throughout the development lifecycle, and across different types of implementation technologies. Modeling has become an essential part of engineering, art and construction. Complex software designs that would be difficult for us to describe textually can readily be conveyed through diagrams. Modeling provides three key benefits: visualization, complexity management and clear communication. UML is a language for specifying and not a method or procedure. UML is used to define a software system. To glean the useful information early in information retrieval system of hospital, we focus mainly on three diagrams viz. activity diagram, class diagram and sequence diagram. Activity diagram is the object oriented concept of flow charts and data flow diagrams and used to explore the logic of complex operation, software processes etc. Although activity diagrams can potentially model the internal logic of a complex operations. Only through activity diagram we cannot estimate.

UML sequence diagrams are used to represent or model the flow of messages, events and actions between the objects or components of a system, this diagram also show the interactions between objects in the sequential order that those interactions occur. Analysis of class diagram provides information to efforts required to build the software. The class diagram can be used by analysts, business modelers, developers, researchers and testers throughout the software development life cycle. Class diagrams are used in nearly all Object Oriented software designs. We use class diagram to describe the Classes of the system and their relationships to each other. Data Cube queries compute aggregates over database relations at a variety of granularities and they constitute an important class for decision support queries [15,16]. The database may represent business data, medical data or scientific data. We shall have a brief lookup on these approaches and few others in later sections.

1.6 Thesis Organization

This work encompasses all terms described in earlier section and correlate then into one context. Our focus is to provide a framework for UML information retrieval system of UML and Data Cube. This thesis, hence, is organized in following manner.

Chapter 1 (Introduction): contains all introductory topics that are required to understand the content of this work. This chapter describes introduction of UML, SDLC Model, and UML diagrams, Data Cube.

Chapter 2 (Literature Survey): contains the literature survey for existing and related research with respect to this work.

Chapter 3 (Problem Statement and Proposed Framework): describes specific problem, statement of problem and brief analysis of the problem, gives solution of the problem and design of the problem in graphical way.

Chapter 4 (Implementation): contains the description of Data Cube and respective result sheet.

Chapter 5 (Conclusion): concludes the final result of the case studies and future related work.

Chapter 2

Literature Survey

UML is the standard language for object-oriented modeling of software applications [2]. The case study considers only form-based user interface. A library case study is used to identify user interface modeling problems [9]. The Library System in the case study could be considered too simple to catch real problems faced during the modeling of user interfaces. As per as our knowledge, no researcher has applied such approaches like Data Cube and UML diagrams on information retrieval system of hospital.

When using the existing features of the commonly used OO languages, the insertion of the additional data can be arranged either through multiple inheritances or using a code generator. Since Java and C# do not have multiple inheritance and no associations in object oriented languages [20]. The UML specification supports multiple inheritance in a natural way. There are UML diagram for modeling static class relationships, dynamic temporal interactions between objects, the usages of objects, the particulars of an implementation, and the state transitions of systems.

An activity diagram enables us in the process to understand and monitor the development of system functionality. Activity diagrams can be used for many purposes: diagramming, use-case flows, modeling complex business operations depicting data and information flows, and even computing algorithms. Today's software systems get increasingly complex. Complexity is observed in all software application domains: in business information systems, technical and administrative systems, as well as in embedded systems such as in avionics, automotive and telecommunications. Often, major sources of the complexity are interactions between system components [4]. We in particular use interaction-centric description techniques such as UML sequence diagrams or message sequence. Sequence diagrams can be used to explore the logic of a complex operation, function, or procedure.

The UML diagrams like activity diagram, sequence diagram can only give the sequence flow of the application but class diagram is a bit different. So it is the most popular UML diagram in the coder community.

So the purpose of the class diagram can be summarized as:

- Analysis and design of the static view of an application.
- Describe responsibilities of a system.
- Base for component and deployment diagrams.
- Forward and reverse engineering.

Class diagrams are the most popular UML diagrams used for construction of software applications. So it is very important to learn the drawing procedure of class diagram. Class diagrams have a lot of properties to consider while drawing but here the diagram will be considered from a top-level view. In the current style of the object-oriented programming, classes are highly visible but their relations (associations, data structures, design patterns) are buried inside class definitions and not easy to find. UML gives us one unified view where classes and their relations have the same importance [20]. For example, to analyze relationships between hospital and patients. The query would require several table scans and multi-way table joins which can degrade performance significantly [21]. It requires at least four inner joins across five tables. A real-life database will have many tables and the time taken to process the joins will be unacceptable. OLAP [14] uses multidimensional tables called Data Cubes. We will implement UML class diagram through the design of Data Cube. Data Cube is basically the part of the online analytical processing. Data Cubes have been designed as logical models. For improving response time, fast retrieval of information Data Cubes are designed to hold the measure values together with summarized information [14,19]. Specifically, Data Cube quickly provides summaries of the data at different meaningful levels of detail, rather than arbitrary summarizations such as aggregating every two records [19].

The purpose of creating cubes is to provide an easy to use mechanism for data access and quick and uniform response time for queries [3]. Consequently Jim Gray et al. [3]

proposed the Data Cube operator as a means of simplifying the process of Data Cube construction.

There are two popular Data Cube technologies: Relational OLAP (ROLAP) and Multidimensional OLAP (MOLAP).

In ROLAP, we use a relational DBMS and a cell of the cube is represented as a record and part of attributes identifies the location of the record in the multidimensional hole of the cube and other attributes holds the data value contained in cell [5].

In MOLAP, elements of the array hold numeric values only and the position in the array represents the connection of numeric values to the dimensions. MOLAP use multi dimensional DBMS rather than a data warehouse to store and access data [5,16]. In this problem of hospital system, we use ROLAP technique to implement Data Cube. Data Cube divides information in two classes: dimensions and measures. For example, states of U.S are dimensions, while the population of each state is a measure.

Star schema: is a simple database design in which dimensional (describing how data are commonly aggregated) are separated from fact or event data.

A star schema consists of two types of tables: fact table and dimension tables.

In this study, we build a cube from a star schema [21]. A star schema is a data source that contains tables in a database in which a single fact table is connected to multiple dimension tables [23]. With a cube based on a star schema, we identify the fact table, the dimension tables, and the keys that map the tables together. Figure 5 shows the star schema.

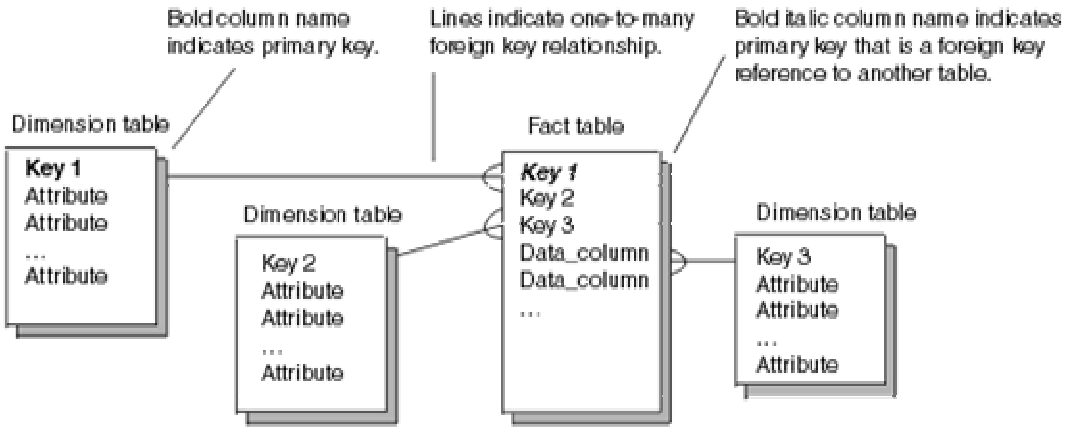

Figure 5. Simple Star Schema

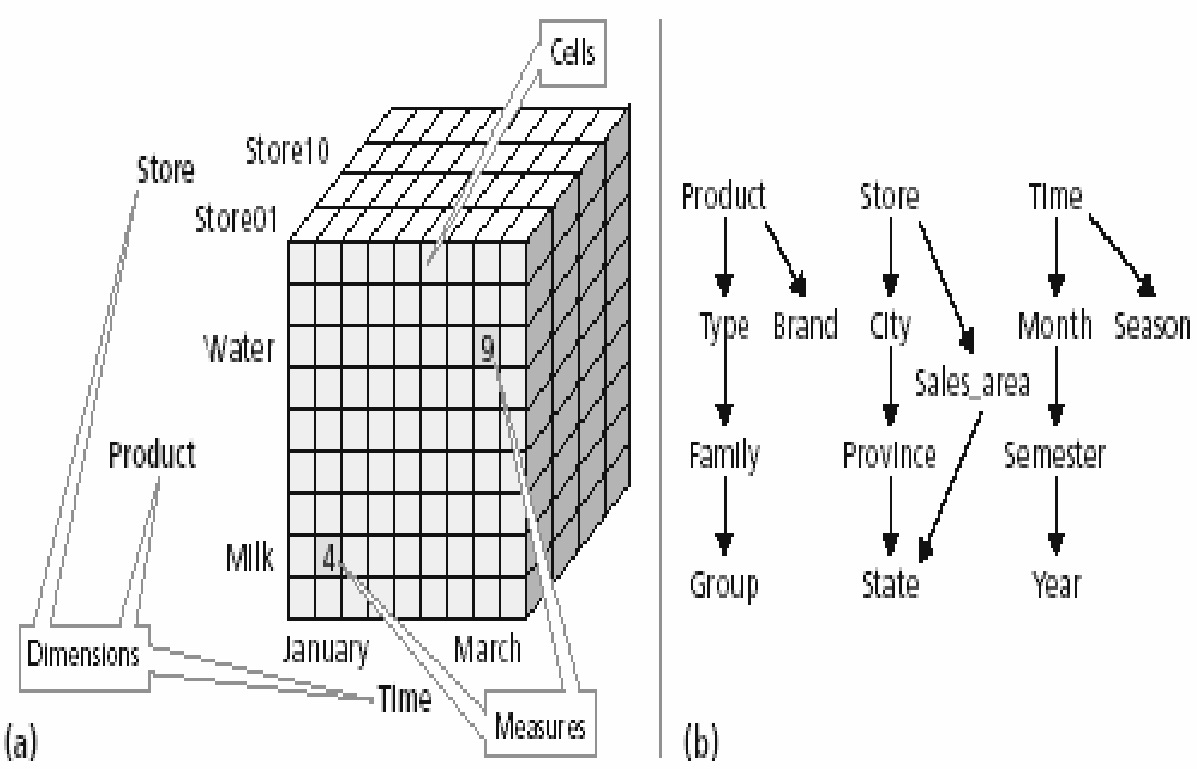

Figure 6

The Figure 6 [7] below depicts a small, practical Data Cube example; consider a hypothetical database of sales information maintained by a company. This particular Data Cube has three feature attributes - store, product, and time - and a single measure attribute. In later section we will make Data Cube for information retrieval system of hospital.

Chapter 3

Problem Statement and Proposed Framework

So far we have discussed the UML diagrams and Data Cube representation. In following section, we will discuss the information retrieval system of hospital using UML and Data Cube.

3.1 Statement of Problem

In this study, requirements for modeling retrieval system of hospital have been deduced. In this we will propose the UML model and Data Cube representation for system. A real case study of Rajindra government hospital, Patiala, India is considered. The hospital is itself a system, in which human beings and machines carry out specific actions by following some rules. As per the information gathered, the whole system comprises the paper based patients information, doctor's information, different type of department's information. It is too slow and cannot provide updated lists. New techniques for information retrieval have been tried using UML and Data Cube.

In later section, we will show how patient gets registered in hospital. In activity diagram and class diagram, we will show how system works in hospital. The system will be used to allocate beds to patient; staff that is in direct contact with the patients will use the system to keep track of available beds, the patients in the different wards, and the types of medication required for each patient within a reasonable timeframe. Doctors must make rounds to pick up patients' treatment cards in order to know whether they have cases to treat or not. The intentions of the system are to reduce over-time pay and will keep full detail of patients coming in OPD and Nursing Home. The doctor would be at ease in accessing full history of the patients, maintained in electronic database. The system would decreases money spent on manpower and would help in increasing the number of patients that can be treated accurately.

3.2 Analysis of Problem Statement

In order to solve the problem, we will use UML diagrams notation to express information retrieval system of hospital. For faster information retrieval, Data Cube has been used to represent the data. We are representing the large set of data in Data Cube. Data Cubes facilitate the answering of data mining queries as they allow the computation of aggregates data at multiple levels of granularity. Data Cube provides us various operations like slicing, roll up, dicing and cube. Data Cube is commonly used for abstracting and summarizing relational databases.

3.3 Proposed Framework

We have already identified the few aspects of information retrieval system of hospital. We had already discussed how to solve the problem by making UML diagrams and representation in Data Cube. In this chapter we will propose the framework for information retrieval system of hospital and we will show the UML diagrams.

Admissions

This module helps in registering information about patients and handles patients' queries. A unique id is generated for each patient after registration. This helps in maintaining medical history of the patient.

Doctor appointments

This module deals with when the id is generated the patient receives the appointment time & number from the receptionist and accordingly visit the doctor.

Test appointments

This module deals when the id is generated the patient receives the appointment time & number from the receptionist and accordingly undergoes the tests.

Bed allotment

This module handles with allotment of bed to various patients by checking their id.

Staff login

This module checks whether the person is a doctor/staff and handles the various activities such as draw salary and give salary.

Draw salary

This module checks whether the person is a doctor/staff and draws salary based on the available information.

Prescribe tests

This module handles various activities such as diagnoses of the patient by doctor, gives treatment, gives suggestions to the patients & prescribes laboratory tests and medicines.

Ward wise bed status

This modules takes care of medical equipment, doctor visit, vitals recording, patient case sheet, diet ordering, blood requisition, transfer intimation and discharge intimation etc., it also deals with the maintenance of the wards.

Admission/discharge reports

This module helps in generating patient discharge summary, which include patients' health at the time of discharge, medical history, various diagnosis and drug prescriptions, history of present ill ness and course in hospital.

Patient information

This module helps in generating the patient information, which is provided by doctor.

3.4 Activity Diagram

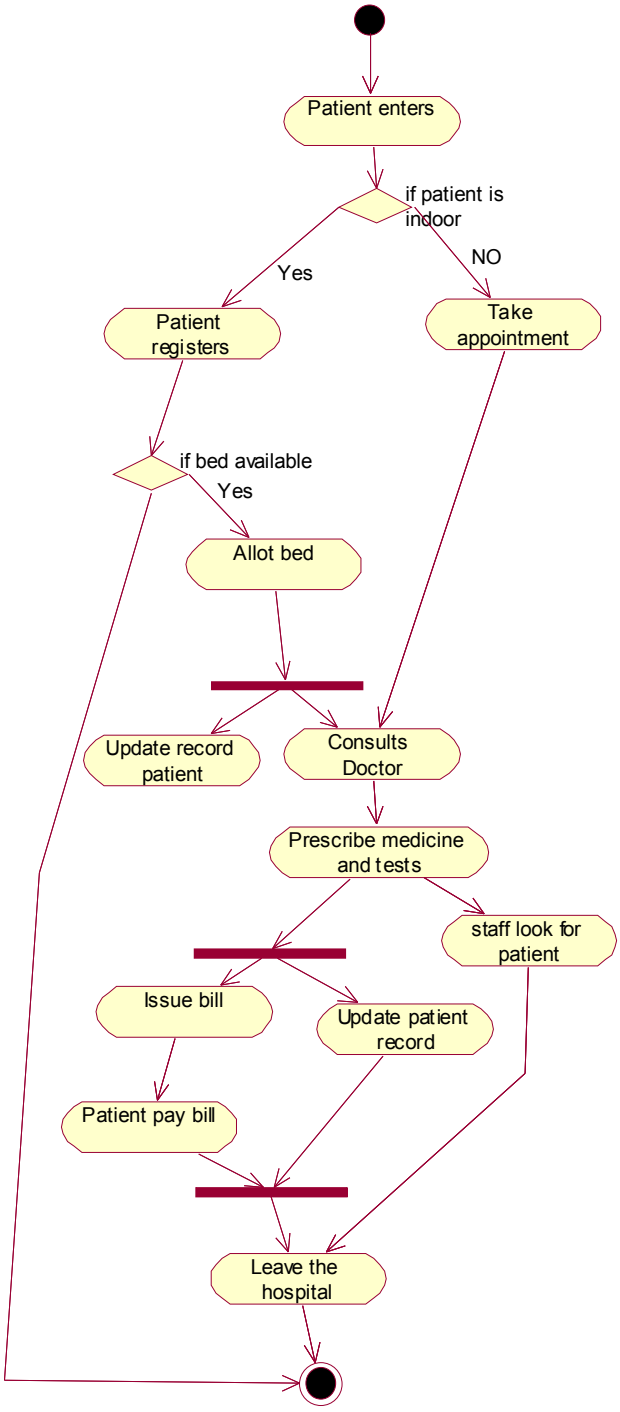

Figure 7

Figure 7 shows the activity diagram of information retrieval system of hospital. In the hospital system, we have taken two types of patient, indoor patient and outdoor patient. In case of indoor patient, OPD registration center will check availability of bed. If bed is available then patient will be registered. As shown in Figure 7, we will use fork to create concurrent activities. Activity update record patient will activate after the allotment of bed. Patient will consult with doctor. Doctor will prescribe the medicines and tests. After that treatment will start. Staff looks after patient. As soon as the patient is fine, patient will pay all the bills. Patient will leave the hospital. Activity diagram will be terminated. If outdoor patient arrives for registrations, patient will take the appointment from the doctor. After taking the appointment, patient will consult with doctor. Doctor will prescribe the medicines and tests. Staff looks after the patient. Update record patient activity will update the record of the patient. Outdoor patient will pay the fee of doctor and medicine bill. We will use join to merge two activities like patient bill and update patient record. Patient will leave the hospital and activity diagram will be terminated.

3.5 Sequence Diagram

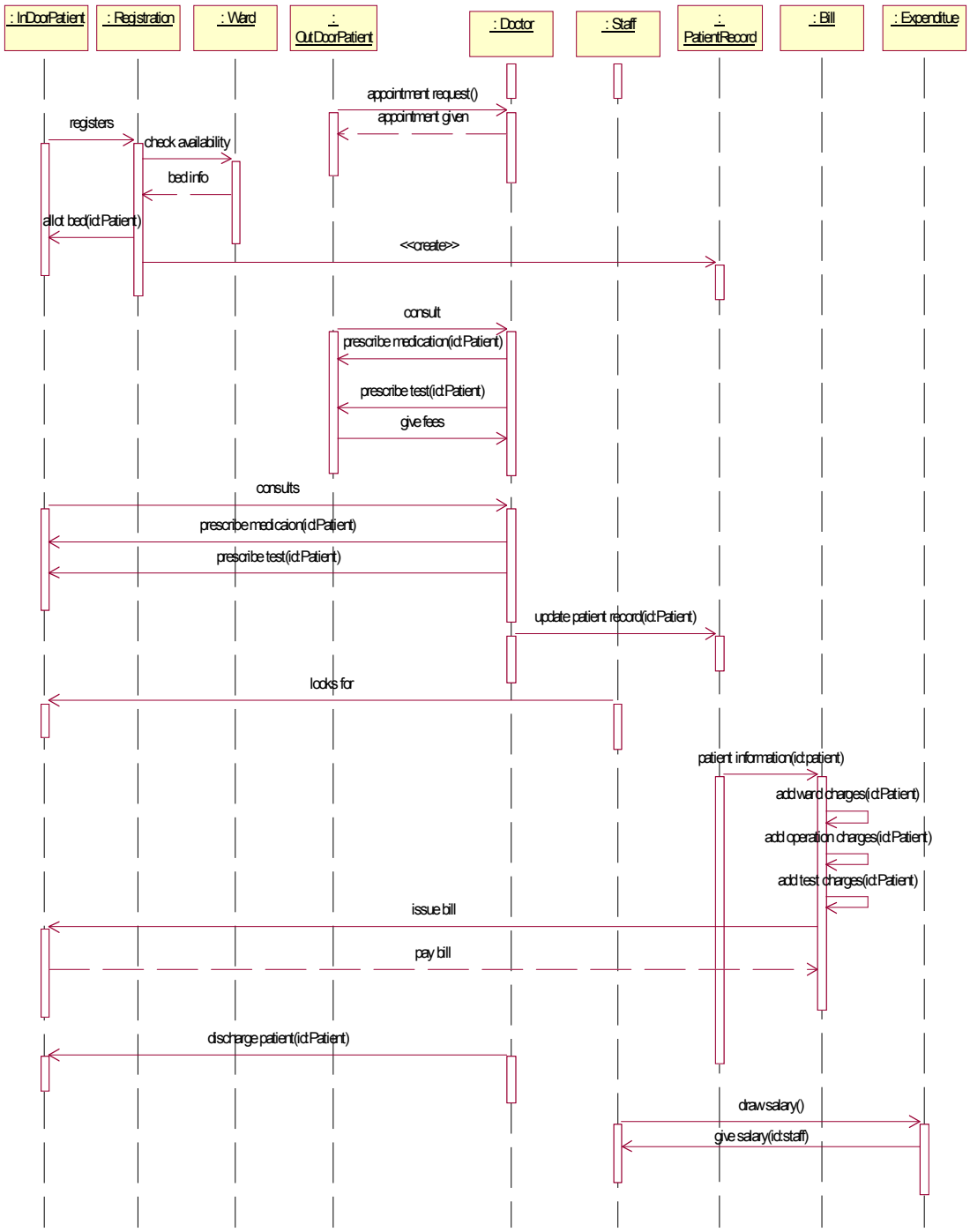

Figure 8

The sequence diagram shows the sequence of transfer of the patient from one object to another object. The sequence diagram is shown in Figure 8. There are nine objects in the sequence diagram i.e. Indoor patient, Registration, Ward, Outdoor patient, Doctor, Patient record, Staff, Bill and Expenditure. The transportation between objects represented by an arrow and message on that arrow. The vertical line shows the lifeline of the objects. Lifeline represents the role over time through the entire interaction. Messages are shown as arrow between lifelines.

Firstly, indoor patient registers and registration department will check the availability of bed in the ward. After getting the information from ward section, registration department will allot the bed to patient. Registration department will provide the bed information and patient information to patient record section.

Outdoor patient will take the appointment from the doctor. After getting the appointment from doctor, patient will consult with doctor. Doctor prescribes the medicine. Patient will pay the fee. Outdoor patient will leave the hospital.

Indoor patient will consult with doctor. Doctor prescribes the test and medicine to indoor patient. Doctor will give the information to patient record section. Staff looks for the patient. Patient record section will give all information of patient to bill section. Bill section will check all the details regarding patient bill, charges etc. Bill section will issue the bill. Indoor patient will pay all the bills. As soon as patient is fine, doctor will discharge the patient. Staff asks for salary from expenditure department and that department will pay the salary.

3.6 Class Diagram

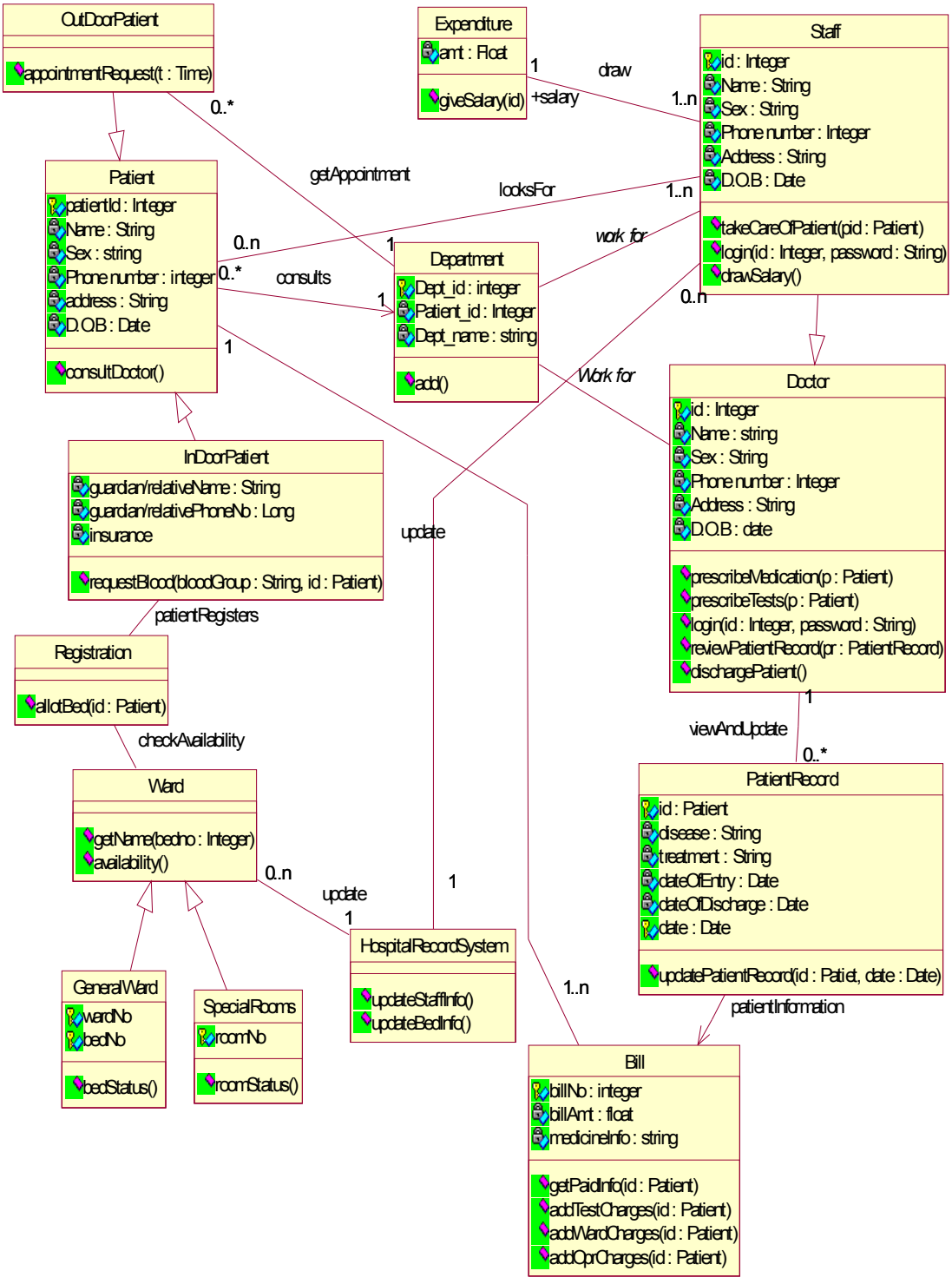

Figure 9

In Figure 9 We have modeled our system with the help of UML class diagram. The system contains fourteen major classes Outdoor patient, Indoor patient, Expenditure, Staff, Patient, Department, Doctor, Registration, Ward, General ward, Special ward, Hospital record system, Bill and Patient record with attributes and operations. Indoor patient class and outdoor patient class inherit the feature of the patient class. In patient class; patient_id, name, sex, phone-number, addresses, D.O.B are attributes and consult doctor () is an operation.

Outdoor patient has same attributes like patient class and appointment request (t: time) is an operation. Outdoor patient take the appointment from department to consult the doctor. Outdoor patient and department have relationship. Department has one relation with outdoor patient and outdoor patient has multiple relationships with department because department has many outdoor patient at time but outdoor patient will take appointment from one department at a time.

Guardian/relative name, guardian/relative phone number are attributes of indoor patient class. Request blood (blood group: string, id: patient) is an operation of indoor patient class. Indoor patient comes and registers. Indoor patient will pay the bill, after payment of the bill, patient will be given room according to availability and choice of patient. Staff takes care of the patient. Doctor will see the report of indoor patient, kept the record own self. As soon as patient is fine, the doctor will discharge the patient. All types of relationships, attribute and operation have shown in the class diagram given in figure 8.

Allot bed (id: patient) is the operation of registration class. Registration class will check the availability of ward.

Get name (bed no: integer), availability () are operations of ward class. General ward class and special rooms class inherit the feature from ward class. Ward no. and bed no. are attributes of general ward class, bed status () is an operation. Room no is the attribute of special rooms class and room status () is an operation.

Id, name, sex, phone-number, address and D.O.B are attributes of doctor class. Prescribe medicine (p: patient), prescribe tests (p: patient), login (id: integer, password: string), review patient record (pr: patient record) and discharge patient () are operations of doctor class. Doctor has one relationship for patient record class but patient record has many relations for doctor class. Doctor updates the patient record.

Staff class generalizes from doctor class. Id, name, sex, phone-number, address, D.O.B are attributes of staff class. Take care of patient (Pid: Patient), login (id: integer, password: string), draw salary () are operations of staff class. Staff has one to many relationships for patient class and patient has many relations to staff class because one patient looks after by one staff member or more than one but one staff member looks after zero patient, one patient or more at a time.

Dept_id, patient_id, dept_name are attributes of department class and add () is operation. Each department works for doctor. Department class depends on the patient class. Patient will be alloted one department at a time but one department has many patients at a time.

In Patient record class; id, disease, treatment, date entry, date of discharge, date are attributes Update patient record (id: patient, date: date) is an operation of patient record class. Patient record class will give information to bill class. Patient record class will view and update the patient record files those are sent by doctor. Bill no, billamt and medicine info are attributes of bill class. Get paid info (id: patient), add test charges (id: patient), add ward charges (id: patient) and add opercharges (id: patient) are operations of bill class. Bill has one to many relationships to patient class but patient class has one relation to bill class because bill class will send bill to one patient at a time but patient will pay the many bills at a time.

Hospital record system has operations update staff info (), update bed info (). Hospital record system has many association to ward class and staff class but ward class, staff class have one association to hospital record system.

Expenditure class has amt attributes and gives salary (id) is an operation. Expenditure class has one relation to staff class but staff class has one to many relationships to expenditure class. Staff class will withdraw salary from expenditure class.

Hospital consists of two live subsystems, OPD and Nursing Home. Under hospital management first we analyze the working of both subsystems and then automate the system. OPD Patients come in hospital by taking appointment through phone or directly. Sometimes emergency case can also be there. Each patient in OPD is given time for 5 to 10 minutes on an average. Medicines are suggested to the patients, which can be collected from any medical shop outside the hospital. In some cases medicine/ injection can be given to patient in the hospital as well. Some test can be suggested to patients for

which laboratory is referred. Patients pay fees to bill section. (Fees that include doctor fees plus any charges related to injection, indoor medicine etc). Doctor himself keeps record of outdoor patients in OPD register. Patients in OPD are told to take the OPD slips, given to them and should bring the same when come to see the doctor next time.

There are general ward and special rooms in the nursing home. Whenever any patient enters in the nursing home, details are entered in register (date of admit, name, age disease, village) and when he/she discharges, the date of discharge is written in register. If a person discharges under normal conditions then a tick mark is put after discharge, if patient is not under normal conditions and doctor feel he cannot treat the patient then patient is referred to another doctor and name of referred doctor is written after date of discharge in place of tick. If patient dies then 'dead' is written in place of tick. On discharging the patient his / her file is referred in Patient record class, date of admit and date of discharge are included along with bill information and any treatment suggested after discharge is also written.

Chapter 4
Evaluation and Validation

4.1 Data Cube representation:

The UML class model can be implemented through the design of Data Cube by the three dimensional representation of Data Cube of information retrieval system of hospital. We will consider the attributes mainly department_id, doctor_id, patient_id, patient_disease and doctor_specialization. But in Data Cube we will take three attributes Department_id, Patient_id, Doctor_id.

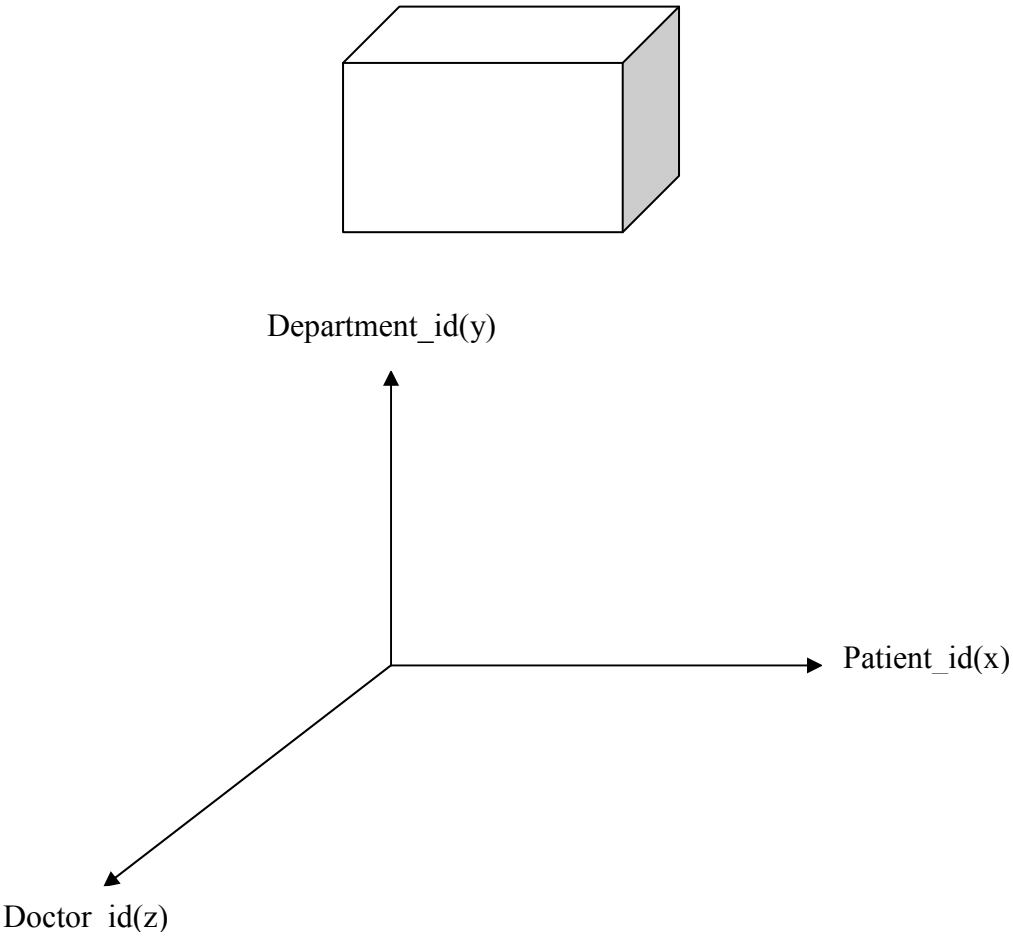

Figure 10

Department_table

Department_id	Department_name
1010	Medicine
2020	Surgery
3030	Orthopedics
4040	Pediatrics
5050	Skin& V.D
6060	Psychiatry
7070	Neurology
8080	Urology
9090	Cardiology
1011	Gynecology
1012	E.N.T
1013	Plastic Surgery

Doctor_table

Doctor_id	Doctor_name	Doctor_specialization	Sex	Phone_Number
1111	Dalveer Singh	E.N.T	Male	9634785368
2111	B.S.Sohal	E.N.T	Male	9838533785
3111	Harbhag Singh	Neurology	Male	9463447553
4111	Varinder SIngh	Skin& V.D	Male	9867325687
5111	Dimple chopra	Skin&V.D	Female	9643853845
6111	Avinash Goyal	Urology	Male	9025746587
7111	Amarjeet Sharma	Urology	Male	9023564566
8111	Surinder Singla	Surgery	Male	9046558687
9111	M.M.Gupta	Medicine	Male	9645784658
1112	Harveer Kaur Rao	Medicine	Female	9467654566
2112	Surjeet Kaur	Gynecology	Female	9568457854
3112	Khuspreet Valia	Gynecology	Female	9789266699
4112	Dhram pal	Pediatric	Male	9628753898
5112	K.K.Lochan	Pediatrics	Male	9683587786
6112	A.S.Sindhu	Orthopedics	Male	9638754853
7112	J.P.Walia	Orthopedics	Male	9678623876
8112	Harcharn Singh	Cardiology	Male	9739646663
9112	Sanjay Single	Cardiology	Male	9635485871

1113	Kuldeep Sharma	Psychiatry	Male	9654545782
2113	\Avtar Singh	Plastic Surgery	Male	9683548542

Patient_table

Patient_id	Patient_name	Phone number	D.O.B	Sex	Patient_disease	Date
3001	Sanket	9634873872	22-sep-1980	Male	Head ace	3-mar-09
3002	Amita	9749367832	2-mar-1987	Female	Bone tumar	22-mar-09
3003	Deepak	9657265763	2-dec-1987	Male	Malaria	04-may-09
3004	Rohan	9758465475	22-dec-1965	Male	Heart Problem	01-jan-09
3005	Nitika	9784565668	15-jan-1982	Female	Preterm Labour	03-mar-09
3006	Komal	9645764856	19-April-1990	Female	Ring worm	07-jun-09
3007	Isha	9645747877	19-feb-1986	Female	Pheumonia	07-may-09
3008	Tisha	9784658569	19-oct-1991	Female	Dangue fever	04-feb-09
3009	Garima	9855344558	22-dec-1995	Female	Tonsillitis	23-jan-09
3010	Rajni	9456485686	11-jun-1986	Female	Infection	22-feb-09
3011	Perteek	9875947571	17-may-1970	Male	Fever	2-jan-09
3012	Sahil	9026548756	18-sep-1966	Male	Cancer	1-jan-09
3013	Shellja	9026558585	18-jun-1996	Female	Smallpox	08-jun-09
3014	Sonu	9024648448	12-mar-1983	Male	Angina Pectoris	28-mar-09
3015	Reema	9025707777	18-nov-1970	Female	Gestantional Diabets	23-jun-09
3016	Parul	9034557565	17-nov-1990	Male	Hernia	22-jan-09
3017	Hari	9815874656	22-dec-1955	Male	B.P	4-april-09
3018	Lusy	9812332449	25-may-1994	Female	Qfever	2-mar-09
3019	Harsh	9814354555	9-may-1984	Male	Tuberculosis	1-jan-09
3020	Monika	9823454356	3-mar-1987	Female	Stomach pblm	10-jun-09
3021	Sunita	9584676776	8-aug-1967	Female	Congestive heart failure	17-feb-09
3022	Anju	9785765767	9-jun-1969	Female	Kidney pblm	02-may-09
3023	Bobby	9469679576	19-dec-2002	Male	Fever	16-jun-09
3024	Rita	9468566596	13-jan-2005	Male	Malignanat neoplasm's of lips	18-feb-09
3025	Tony	9888454356	27-july-1988	Male	Plastic Surgery	3-jun-09
3026	Sonam	9888545466	8-may-1978	Female	Kidney Stone	8-jun-09

HIS_INFO

Department_id	Doctor_id	Patient_id	Doctor_specialization	Patient_disease
1010	1112	3007	Medicine	Pheumonia
8080	6111	3026	Urology	Kidney Stone
1012	2111	3024	E.N.T	Malignant neoplasm's of lips
1010	9111	3019	Medicine	Tuberculosis
7070	3111	3017	Neurology	Malaria
2020	8111	3012	Surgery	Cancer
3030	6112	3002	Orthopedics	Bone tumor
9090	8112	3004	Cardiology	Heart Problem
2020	8111	3016	Surgery	Hernia
9090	8112	3014	Cardiology	Angina Pectoris
5050	5111	3006	Skin& V.D	Ring worm
4040	5112	3023	Pediatric	Fever
1011	2112	3005	Gynecology	Preterm labour
1011	3112	3015	Gynecology	Gestational Diabetes

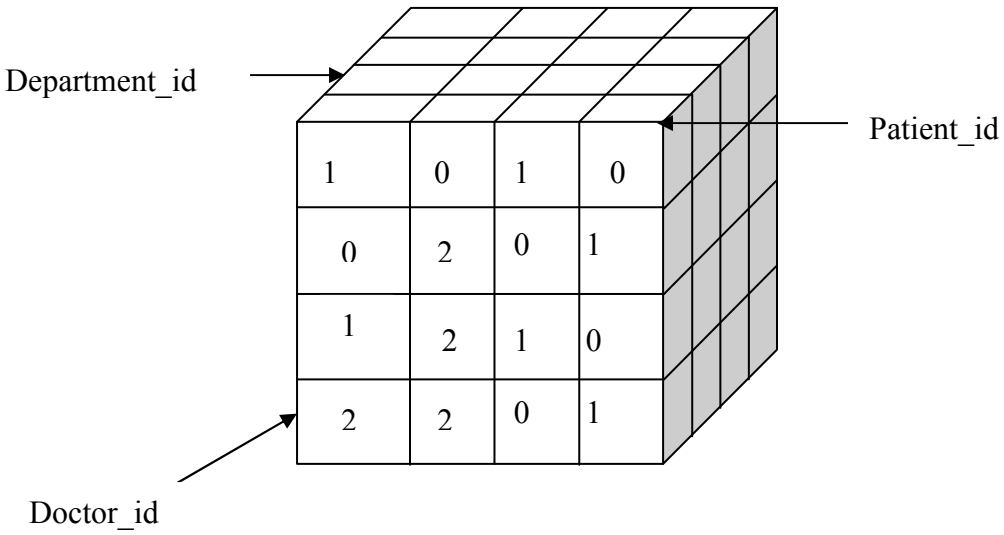

Figure 11 A front view of Data Cube

The Figure 11 depicts a small, practical Data Cube example of the Hospital. This particular Data Cube has three feature attributes --- Department_id, Doctor_id,

Patient_id. We have described the Data Cube as a conceptual model. This is certainly true. However, in the case of a MOLAP server, it is also the physical model, because MOLAP stores the cube structure directly as a multi-dimensional array. Conversely, ROLAP server must map this representation to a relational design. The content of the each cell is the count of the number of the times that specific value comes collectively in the database. The main goal to design Data Cube is to extract the information in faster way.

Query 1.

Select doctor_id, dept_id from his_info group by cube (doctor_id, dept_id);

The output of the above query shows possible combinations of dept_id and doctor_id.

```
 DOCTOR_ID    DEPT_ID
---------- ----------

      1112
      2112
      2113
      3111
      3112
      5111
      5112
      6111
      6112
      8111

 DOCTOR_ID    DEPT_ID
---------- ----------
      8112
      9111
                1010
```

```
      1112       1010
      9111       1010
                 1011
      2112       1011
      3112       1011
                 1012
      2113       1012
                 2020

 DOCTOR_ID    DEPT_ID
---------- ----------
      8111       2020
                 3030
      6112       3030
                 4040
      5112       4040
                5050
      5111       5050
                 7070
      3111       7070
                8080
      6111       8080

 DOCTOR_ID    DEPT_ID
---------- ----------
                 9090
      8112       9090

35 rows selected.
```

Query 2.

Select patient_id, dept_id from his_info group by cube (patient_id, dept_id);

PATIENT_ID DEPT_ID

---------- ----------

1010

1011

1012

2020

3030

4040

5050

7070

8080

9090

PATIENT_ID DEPT_ID

---------- ----------

3002

3002 3030

3004

3004 9090

3005

3005 1011

3006

3006 5050

3007

3007 1010

3012

PATIENT_ID DEPT_ID

---------- ----------

3012 2020

3014

3014 9090

3015

3015 1011

3016

3016 2020

3017

3017 7070

3019

3019 1010

PATIENT_ID DEPT_ID

---------- ----------

3023

3023 4040

3024

3024 1012

3026

3026 8080

39 rows selected.

Query 3.

Select patient_id, dept_id, count (*) from his_info group by cube (patient_id, dept_id);

---------- ---------- ----------

14

1010 2

1011 2

1012 1

2020	2
3030	1
4040	1
5050	1
7070	1
8080	1
9090	2

PATIENT_ID	DEPT_ID	COUNT(*)
3002		1
3002	3030	1
3004		1
3004	9090	1
3005		1
3005	1011	1

Query 4.

Select doctor_id, dept_id, count (*) from his_info group by cube (doctor_id, dept_id) having doctor

_id='8112';

DOCTOR_ID	DEPT_ID	COUNT (*)
8112	9090	2
8112		2

Chapter 5

Conclusion

The system will be used in to retrieve information system of hospital. The doctor, nurses and staff will be the main users. Given the condition that not all the users are computer-literate, some users may have to be trained to use the system. The system is designed to be user-friendly. It uses a Graphical User Interface (GUI).

Here, we concluded that UML modelling is a very efficient modelling language to represent such system. By UML activity diagram, sequence diagram and class diagram, we are able to model the whole system in an efficient way. By implementing data in Data Cube, we can get any information so quickly. We can perform number of queries so fast. We get the desired result. We get any record of patient and doctor by using less number of join. We can retrieve information only through one table. We can handle query for instant decision of bed allocation for patients, and request for the bed transfer easily. Data Cube hides the information at backhand level. For nonexpert UML or database users, the cube class's graphical notation facilitates the definition of initial user requirements.

In future, using the cluster analysis technique can extend this work. By cluster technique we can make the process of grouping a set of physical or abstract objects of information retrieval system of hospital into classes of similar into one.

References

[1] Dzidek W. J., Arisholm E. and Briand L. C. "A Realistic Empirical Evaluation of the Costs and Benefits of UML in Software Maintenance", IEEE Trans. Softw. Eng. 34, 3. pp: 407-432. May 2008.

[2] Da Sliva P. P., Paton N. W. "User Interface Modelling with UML", proceedings of the 10th European-Japanease Conference on Information Modelling and knowledge representation, 2000.

[3] Gray J., Chaudhuri S., Bosworth A., Layman A, Reichart D., Venkatrao M., Pellow F. and Pirahesh H. "Data Cube: A Relational Aggregation Operator Generalizing Group-By, Cross-Tab and Sub-Totals" Data Mining and Knowledge Discovery. 1:1, pp: 29-53, 1997.

[4] Grønmo R., Sørensen F., Møller-Pedersen B. and Krogdahl, S. 2008. "Semantics-Based Weaving of UML Sequence Diagrams", Proceedings of the 1st international Conference on theory and Practice of Model Transformations. A. Vallecillo, J. Gray, and A. Pierantonio, Eds. Lecture Notes In Computer Science, vol. 5063. Springer-Verlag, Berlin, Heidelberg, pp:122-136, July 01 - 02, 2008.

[5] Gupta G.K. "Introduction to Data Mining with Case Studies" published by prentice hall of India private limited, first edition.

[6] Han J. and Kamber M. "Data mining concepts and techniques", an imprint of Elsevier morgan kuafmann publishers, second edition.

[7] Ivanova A., Rachev B., "Multidimensional models – Constructing DATA CUBE", Proceedings of CompSysTech'04**,** Rousse (Bulgaria), 2004.

[8] Jalote P. “An integratedApproach to Software Engineering”, Published by N. K. Mehra for Narosa Publishing House Pvt,Ltd., third edition.

[9] Kuhn K. A., Lenz R. and Blaser R. “Building a hospital information system: design considerations based on result from a Europe-wide vendor select process”, Proceeding of proce AMIR Symp., pp:834-838, 1991.

[10] Kamp V., Sitzmann L., and Wietek F. “A Spatial Data Cube Concept for Supporting Data Analysis in Environmental Epidemiology”, Proceedings of the Ninth international Conference on Scientific and Statistical Database Management, Y. E. Ioannidis and D. M. Hansen, Eds. SSDBM. IEEE Computer Society, Washington, DC, pp:100-103, August 11 - 13, 1997.

[11] Missaoui R., Goutte C., Choupo A. K., and Boujenoui A. “A probabilistic model for Data Cube compression and query approximation” In Proceedings of the ACM Tenth international Workshop on Data Warehousing and OLAP, DOLAP '07. ACM, New York, NY, pp:33-40, November 09 - 09, 2007.

[12] Micol R., Guilbaud M., and Valleron AJ. “Towards a Hospital Geographical Information System Dedicated to Nosocomial Infection Control” Proceedings of the AMIA Annual Symposium, pp:1107, 2002

[13] Mishelevich D. J. and Slyke D. V. “Application development system: The software architecture of the IBM Health Care Support/DL/I-Patient Care System”, IBM Journal, vol 19,number 4, pp: 478, 1980.

[14] Rozeva A. “Data Cubes and aggregates query processing”, In proceeding of international conference on computer systems and technologies-CompSys Tech’ 2005.

[15] Ross K. A., Zaman K.A. “Optimizing Selection over Data Cubes” , In proceeding of the IEEE international conference on scienti and statistical database management, November-24-1998.

[16] Ross K. A. and Srivastava D. “Fast Computation of Sparse Datacubes”. In Proceedings of the 23rd international Conference on Very Large Data Bases. M. Jarke, M. J. Carey, K. R. Dittrich, F. H. Lochovsky, P. Loucopoulos, and M. A. Jeusfeld, Eds. Very Large Data Bases. Morgan Kaufmann Publishers, San Francisco, CA, pp:116-125, August 25 - 29, 1997.

[17] Rumbaugh J., Jacobson I., booch G. “The United Modeling Language Reference Manual” published by pearson Education Asia Pte.Ltd. (Addison-Wesley Object Technology Series, second edition.

[18] Rani U., Jyoti S., Rama Sree R. J. “Data warehousing & Data Mining”, publisher IKON book, first edition.

[19] Stolte C., Tang D. and Hanrahan P. “Multiscale Visualization Using Data Cubes”, In published of the IEEE Transactions on Visualization and Computer Graphics 9, 2. pp:176-187, April,2003.

[20] Soukup M. and Soukup J. “The popularity cycle of graphical tools, UML, and libraries of associations”, In Companion To the 22nd ACM SIGPLAN Conference on Object-Oriented Programming Systems and Applications Companion (Montreal, Quebec, Canada, OOPSLA '07. ACM, New York, NY, pp:753-756, October 21 - 25, 2007.

[21] S. Palaniappan, Chua S.L. “Clinical Decision Support using OLAP with data mining” in IJCSNS Internatinal journal of computer science and network security, VOL.8, pp:9, September 2008.

[22] Stevens P. and Pooley R. “Using UML: Software engineering with objects and components”, published by Harlow [etc.] : Addison-Wesley, second edition, 2006.

[23]http://publib.boulder.ibm.com/infocenter/rbhelp/v6r3/index.jsp?topic=/com.ibm.redbrick.doc6.3/wag/wag32.htm

Abbreviations

DBMS	Database Management System
DEPT	Department
D.O.B	Date of Birth
HIS	Hospital Information System
ID	Identity
INFO	Information
MOLAP	Multidimensional On-Line Analytical Processing
OLAP	On-Line Analytical Processing
OO	Object Oriented
OOD	Object Oriented Design
OOL	Object Oriented Language
OPD	Department of Outdoor Patient
ROLAP	Relational On-Line Analytical Processing
SDLC	System Development Life Cycle
UML	Unified Modeling Language